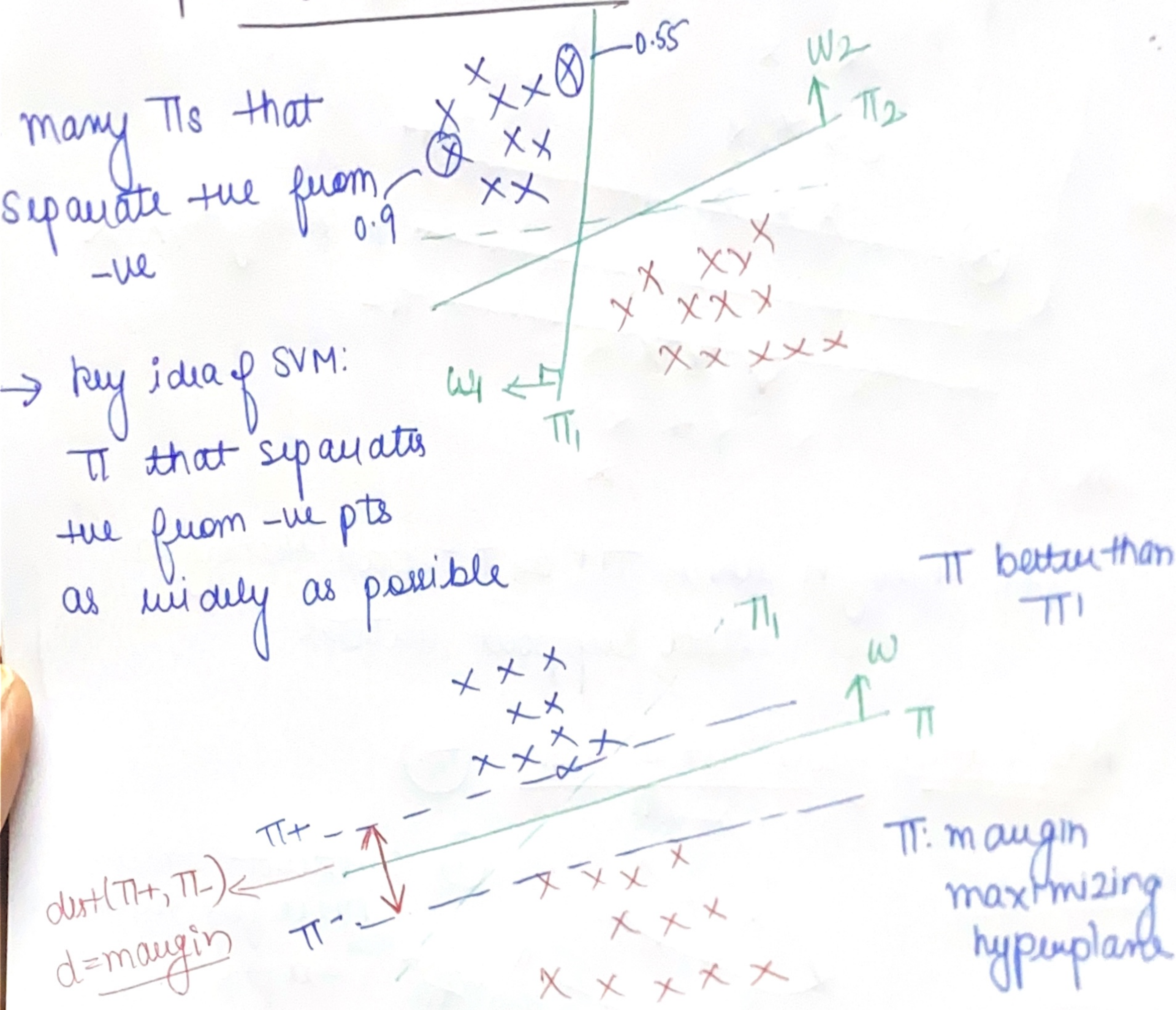
Support Vector Machine → classfn
→ regression

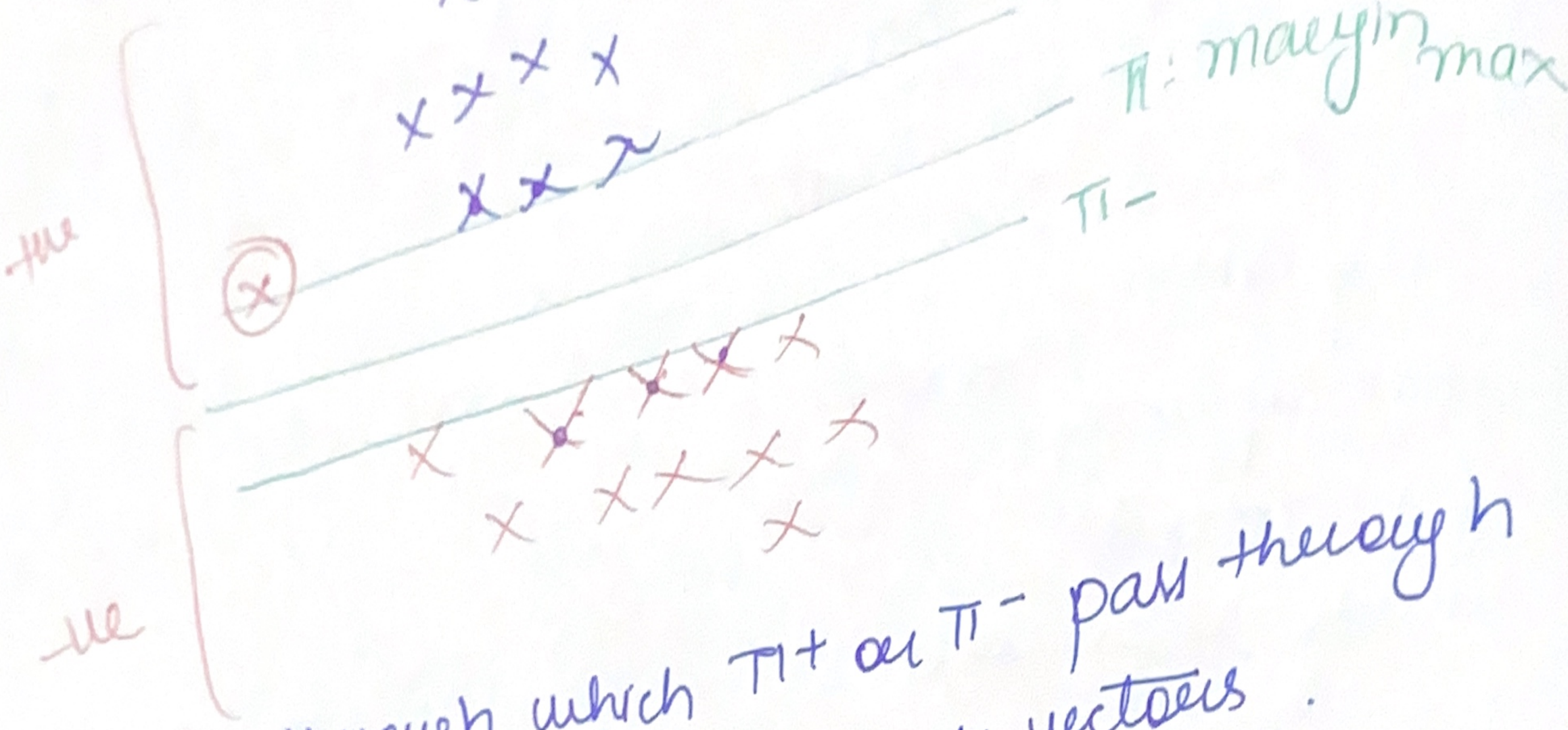
Geometric Institution



SVM: Try to find hyperplane which maximises margin
= dist(π+, π-)

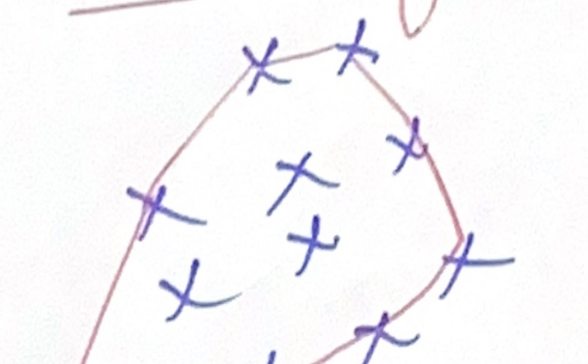
margin ↑ generalization acc ↑

Support Vector

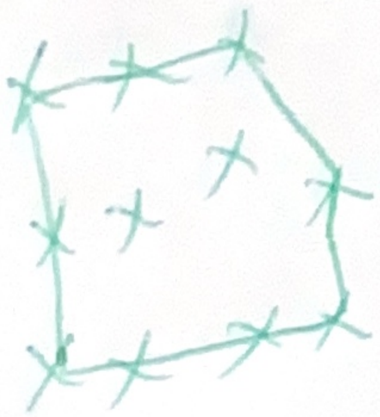


pts through which H+ or H- pass through are called support vectors.

Attendtive geom inst of SVM

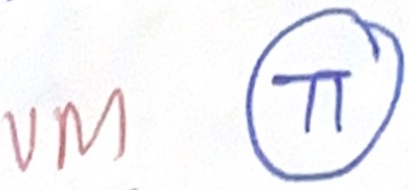


Convex-hull



Convex polygon

↳ non convex polygon



① construct convex hull for two pts sets

② find shortest line connecting hulls

③ Bisect line

Mathematic formulation of SVM

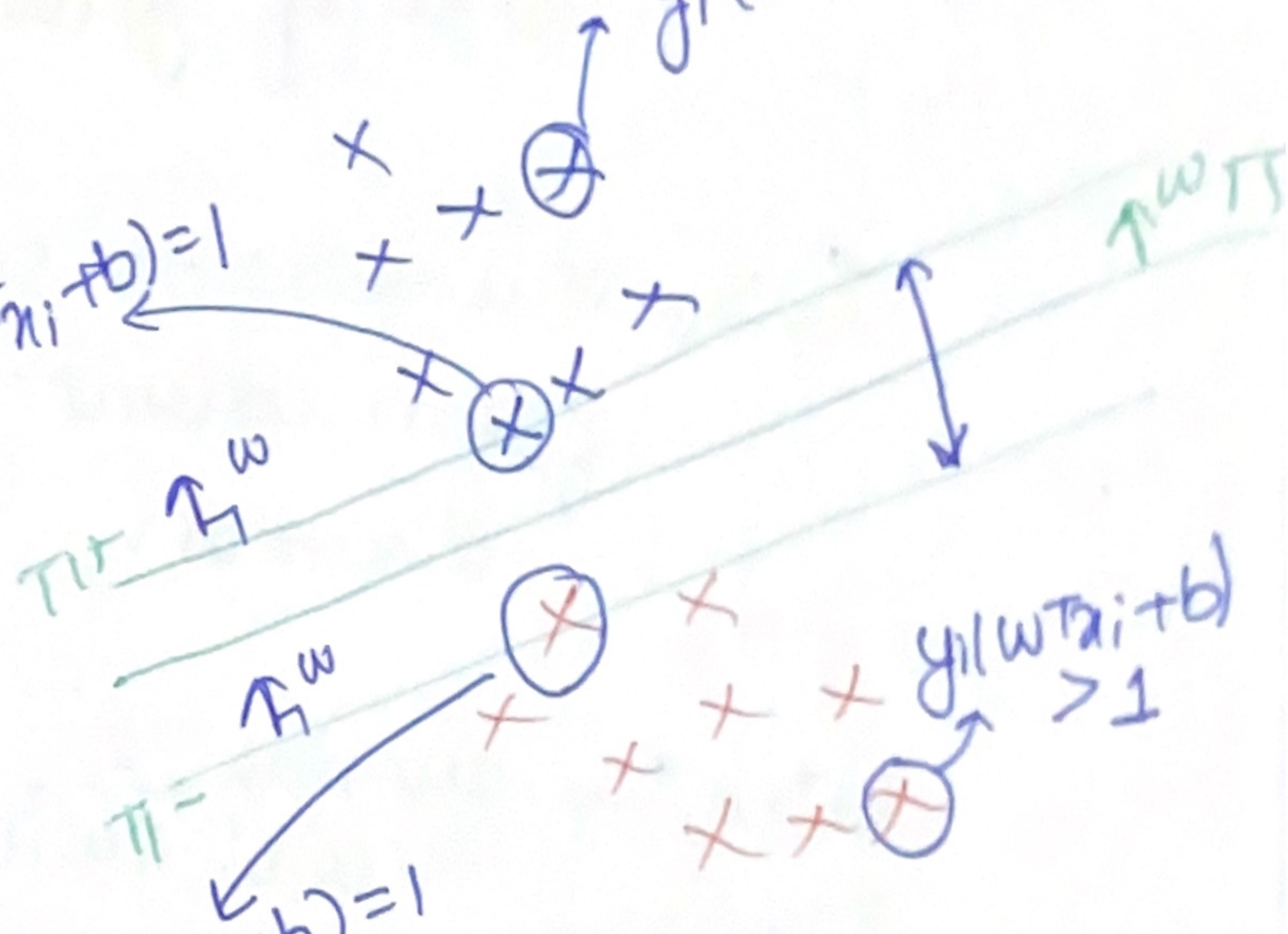
π: margin maximization

$\pi: w^T x + b = 0$

$\pi^+: w^T x + b = 1$

$\pi^-: w^T x + b = -1$

if



Margin: $d = \frac{2}{\|w\|}$

Such that

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

x_i

$$w^*, b^* = \arg\max_{w,b} \frac{2}{\|w\|} =$$

S.t

margin.

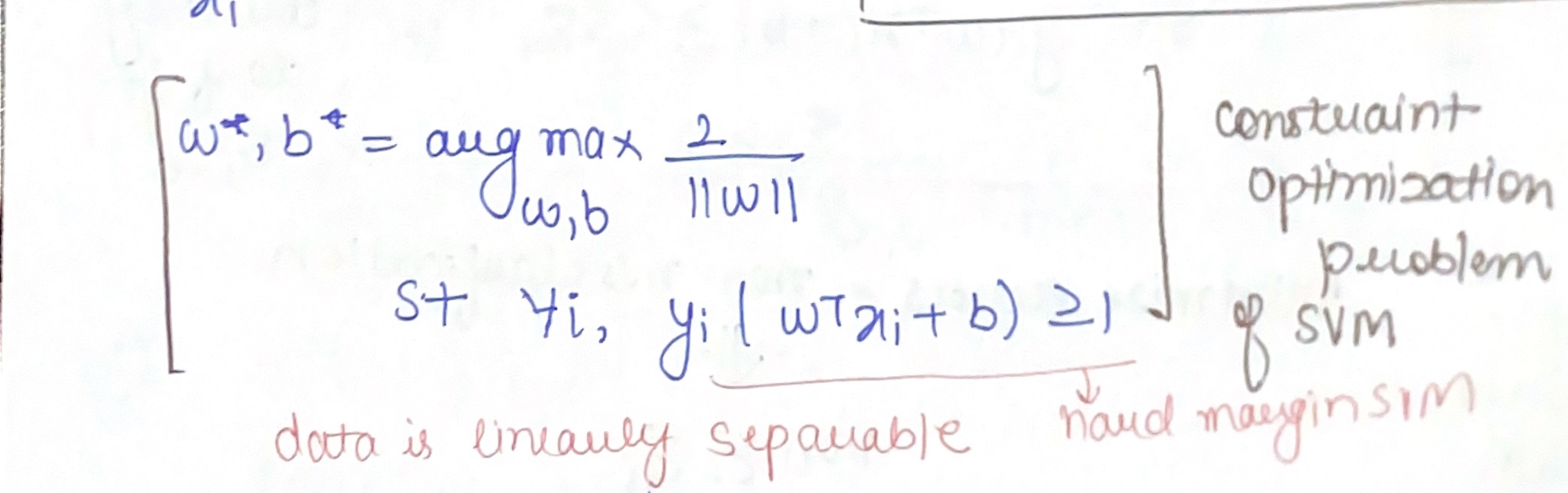
S.t

$$\left[w^*, b^* = \arg\max_{w,b} \frac{2}{\|w\|} \right.$$

$$\left. \text{S.t } \forall i,\ y_i (w^T x_i + b) \geq 1 \right]$$

constraint optimization problem of SVM

data is linearly separable ← hard margin SVM



← not linearly separable almost sep

$\xi_i \uparrow$: pt is farther away from correct π

$x_i \longrightarrow \xi_i$

$\xi_i = 0$ if $y_i(w^T x_i + b) \geq 1$

↳ correctly class (a)

$\xi_i > 0$ 8 is equal to some unit of dist away from correct hyperplane in incorrect direction

$$w^*, b^* = \arg\max_{w,b} \frac{2}{\|w\|} = \arg\min_{w,b} \frac{\|w\|}{2}$$

hyperpara - $\max f(x) = \min 1/f(x)$

$$w^*, b^* = \arg\min_{w,b} \frac{\|w\|}{2} + C \cdot \underbrace{\frac{1}{n}\sum_{i=1}^{n} \xi_i}$$ → avg dist of mis classified pt from π

↑ margin

s.t $y_i(w^T x_i + b) \geq 1 - \xi_i \quad \forall i$] correctly class pt $\xi_i = 0$

$\xi_i \geq 0$

↓ Soft margin SVM

minimize errors = min misclassification

⇓

$\min \sum \xi_i$

$$\min_w \left(\text{logistic loss} \right) + r(\text{reg})$$

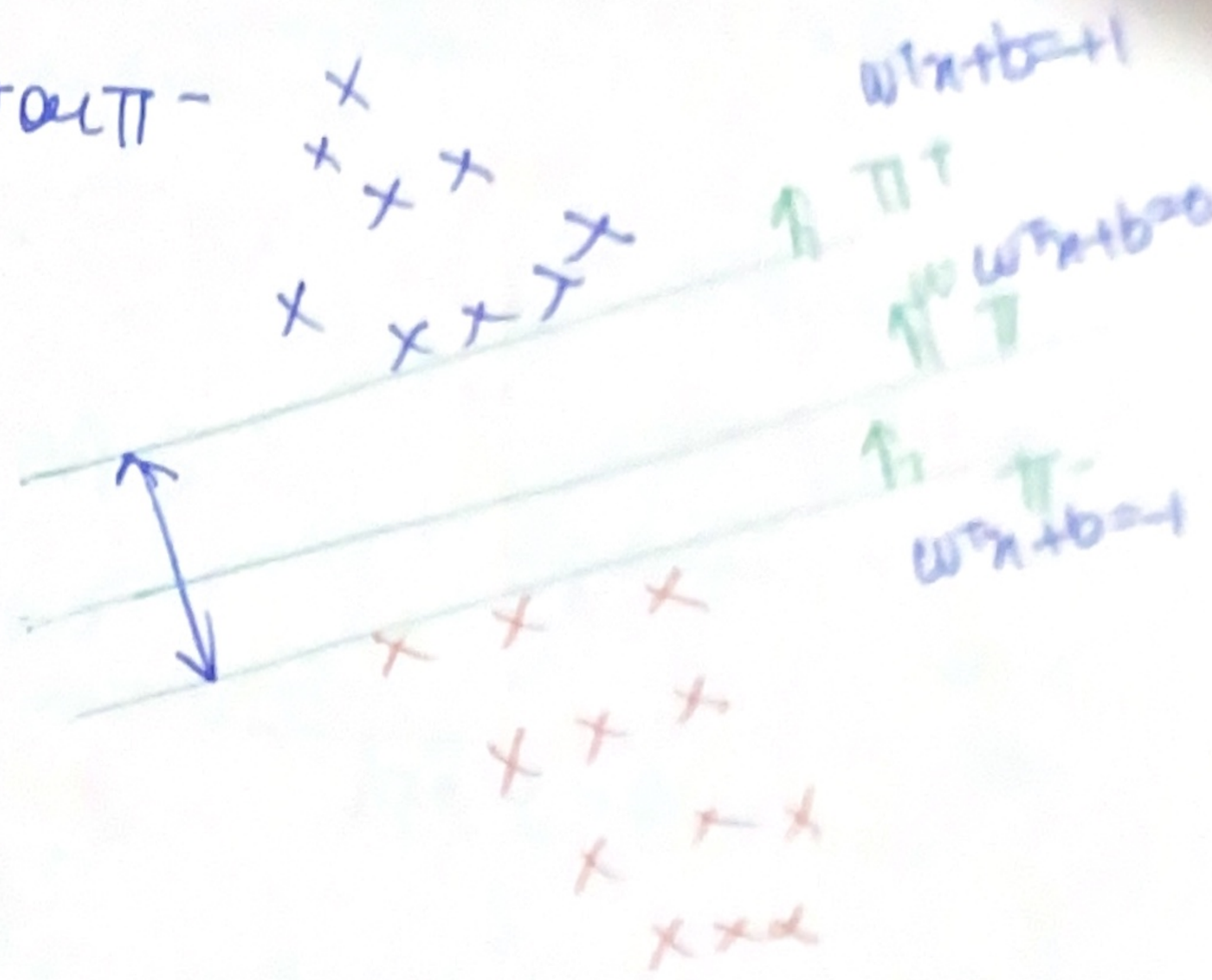
C(true) C ↑; tendency to make mistakes ↓ on D_T

⇒ overfit → high var

C ↓; underfit ⇒ high bias model

SVM:

(Q) why +1 & -1 on RHS of π^+ & π^-



margin: $\frac{2}{\|w\|}$

$w^*, b^* = \underset{b}{\arg\max} \frac{2}{\|w\|}$

$\|w\| \neq 1$ (any vector) need not be unit vector

① $\pi^+ : w^T x + b = k$

$\pi^- : w^T x + b = -k$ $\quad k > 0$

margin: $\frac{2k}{\|w\|}$ $\quad k = 4$

$\underset{w,b}{\arg\max} \frac{2}{\|w\|} = \underset{w,b}{\arg\max} \frac{2k}{\|w\|} = \frac{8}{\|w\|}$

② $w^T x + b = k$

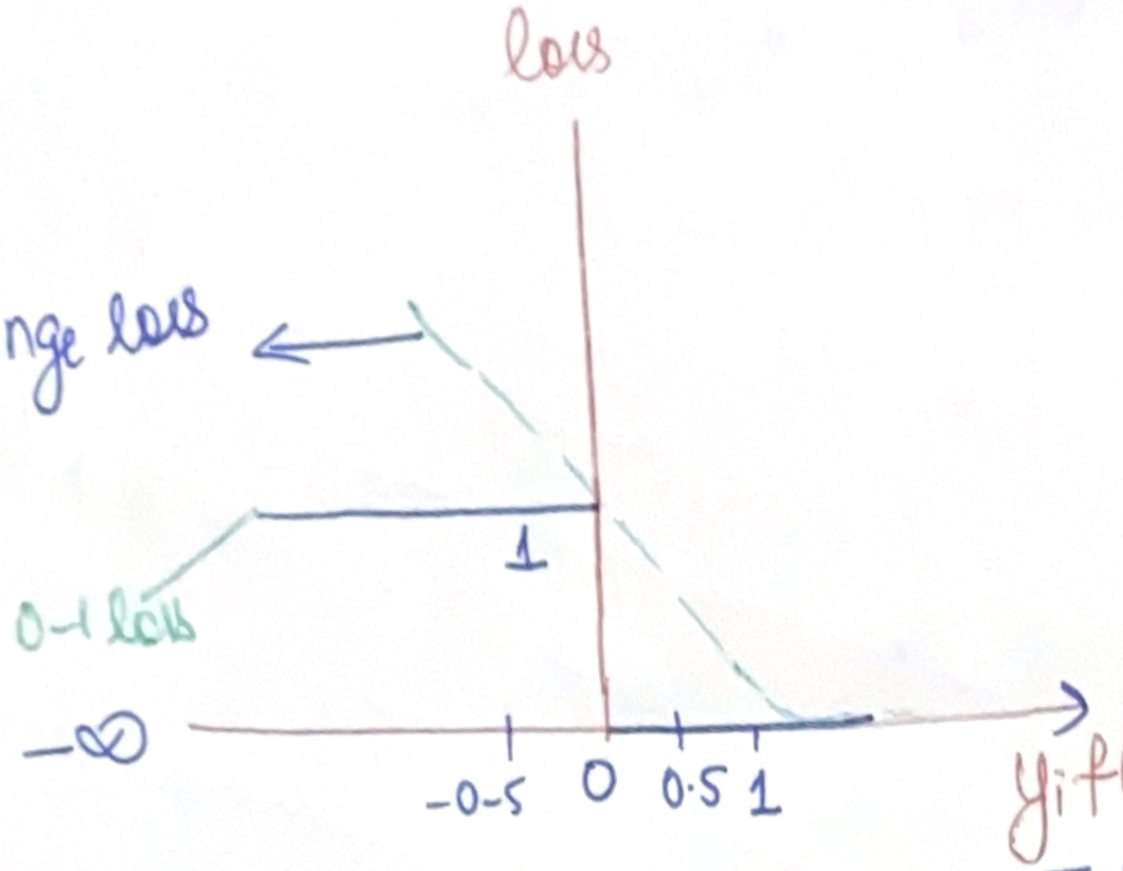
$\left(\frac{w}{k}\right)^T x + \left(\frac{b}{k}\right) = 1$ $\quad w \perp \pi$

$\|w\| \neq 1$

$(w')^T x + b' = 1$

Reason +1 or -1 ← Simplify math

Loss minimization : Hinge-loss



log reg :- logis loss + reg
lr-reg: lr reg + reg
SVM: hinge loss + reg

$= y_i(w^T x_i + b) = z_i$

$\begin{cases} z_i > 0: & x_i \text{ is correctly classf} \\ z_i < 0: & x_i \text{ is incorrectly classf} \end{cases}$

Hinge loss : $z_i \geq 1$; hinge loss $= 0$

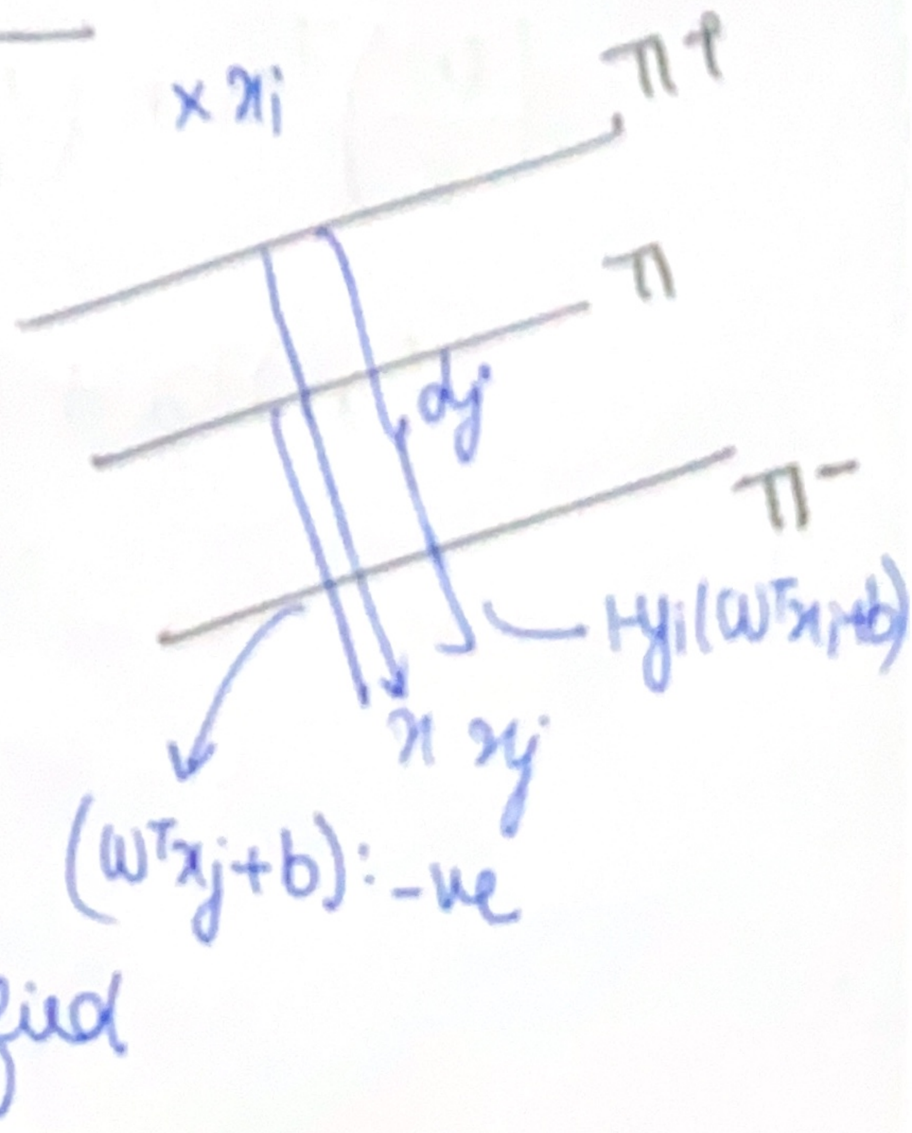
$z_i < 1$; hinge loss $= 1 - z_i$

$\hookrightarrow \max(0, 1 - z_i)$

Case 1: $z_i \geq 1$; $1 - z_i$ is -ve value $\Rightarrow \max(0, 1-z_i) = 0$

Case 2 $z_i < 1$; $1 - z_i > 0 \Rightarrow \max(0, 1-z_i) = 1 - z_i$

Geometric - formulation & loss min



$\xi_i = 0 \leftarrow x_i$

$d_j = 1 - y_i(w^T x_i + b) = 1 - z_j$

ξ_j = dist of x_j to the π^+ = $d_j = 1 - z_j$

$\xi_j = 1 - z_j$ when x_j is misclassified

$$\max(0, 1-x_i) = \xi_i$$

Soft SVM

$$\min_{w,b} \frac{\|w\|}{2} + C\sum_{i=1}^{n}\xi_i$$

$$s.t\ \ 1-y_i(w^T x_i + b) \le \xi_i$$

$$\xi_i \ge 0$$

$C\uparrow \Rightarrow$ overfit

$C\downarrow \Rightarrow$ underfit

~~hard SVM~~

low min

$$\min_{w,b} \sum_{i=1}^{n} \max(0, 1-y_i(w^T x_i + b)) + \lambda\|w\|^2$$

$\lambda\uparrow \Rightarrow$ underfit $\quad \lambda\downarrow \Rightarrow$ overfit

$\|w\|>0 \Rightarrow \min\|w\|$ is same as $\|w\|^2$

Dual form of SVM

Soft margin SVM

$$\min_{w,b} \frac{1}{2}\|w\| + C\sum_{i=1}^{n}\xi_i$$

$$s.t\ \ y_i(w^T x_i + b) \ge 1-\xi_i \quad \forall i$$

$$\xi_i \ge 0$$

Equivalent → primal of SVM

Dual form

$$\max_{\alpha_i} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j$$

$$s.t\ \ \alpha_i \ge 0$$

$$\sum_{i=1}^{n}\alpha_i y_i = 0$$

① $x_i \to \alpha_i$

② x_i's only occur in form of $x_i^T x_j$

③ $f(x_q) =$

③ $f(x_q) = \sum_{i=1}^{n} \alpha_i y_i x_i^T x_q + b$

SV_s: $\alpha_i > 0$

non SV_s: $\alpha_i = 0$

$f(x_q)$: only pts that matter are SV_s

$$\max_{\alpha_i} \sum_{j=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j \longrightarrow \text{Sim}(x_i,$$

s.t $\alpha_i \geq 0 \longrightarrow \alpha_i = 0$ for SV_s

$\sum_{i=1}^{n} \alpha_i y_i = 0$ $\quad \alpha_i > 0$ for non SV_s

$x_i^T x_j = x_i . x_j = \text{Cosine sim}(x_i, x_j)$

if $\|x_i\| = 1$; $\|x_j\| = 1$

$$f(x_q) = \sum_{i=1}^{n} \alpha_i y_i x_i^T x_q + b$$

$\longrightarrow k(x_i, x_q)$

Kernal Trick

$$\max_{\alpha_i} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \underbrace{x_i^T x_j}_{\rightarrow K(x_i, x_j)} \rightarrow \text{Sim}(x_i, x_j)$$

$$\text{s.t} \sum_{i=1}^{n} \alpha_i y_i = 0 \; ; \; \alpha_i \geq 0$$

↓
Kernal func

$$f(x_q) = \sum_{i=1}^{n} x_i y_i K(x_i, x_q) + b$$

→ The most impo idea in SVM is Kernal Trick

→ Soft-SVM - hyperplanes ≒ log reg

↓

margin-max

In SVM; $x_i^T x_j$ $\qquad K(x_i, x_j) = x_i^T x_j$

Kernal SVM; $K(x_i, x_j)$

In SVM; margin max hyperplane (x_i's)

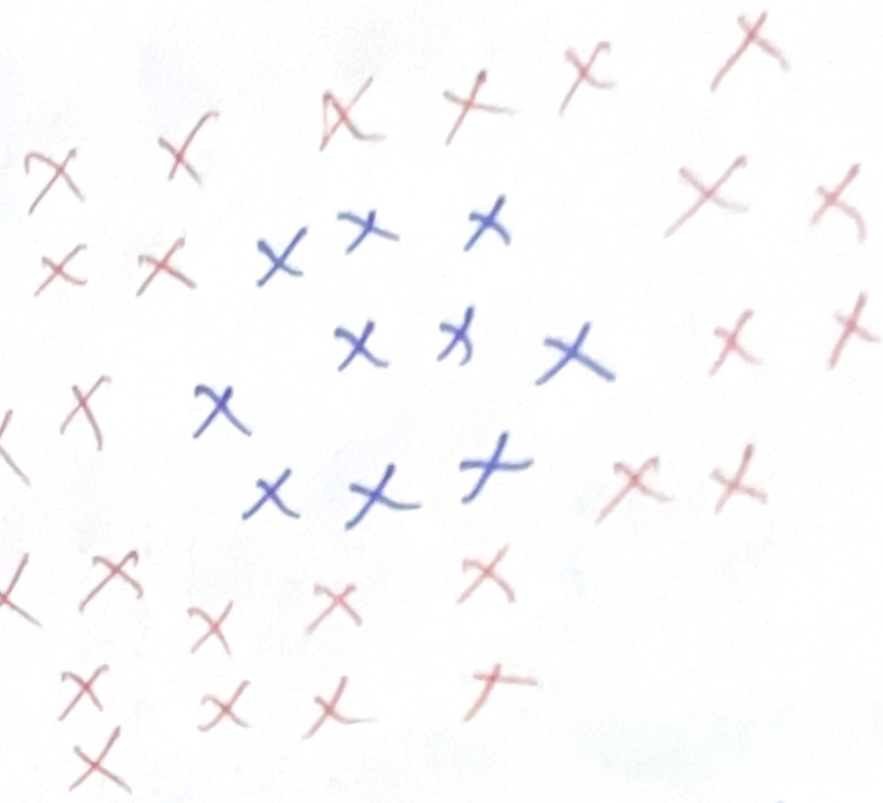
log reg; min logistic loss (x_i's)

Would changing idea; Kernalization → 1990's

Polynomial Kernel

→ Kernalization

$$K(x_1, x_2) = (x_1^T x_2 + c)^d$$



eg. $K(x_1, x_2) = \underbrace{(1 + x_1^T x_2)^2}$

Quadratic Kernel

$(\beta_1, \beta_2) \rightarrow (\beta_1^2, \beta_2^2)$

↓

log reg

$$K(x_1, x_2) = (1 + x_1^T x_2)^2$$

$$= (1 + x_{11}x_{21} + x_{12} + x_{22})^2 \quad x_1 = \langle x_{11}, x_{12} \rangle \quad x_2 = \langle x_{21}, x_{22} \rangle$$

$$= 1 + x_{11}^2 x_{21}^2 + x_{12}^2 + x_{22}^2 + 2x_{11}x_{21} + 2x_{12}x_{22} + 2 x_{11}x_{21}x_{12}x_{22}$$

let $= [1, x_{11}^2, x_{12}^2, \sqrt{2}x_{11}, \sqrt{2}x_{12} + \sqrt{2}x_{11}x_{12}] : x_1'$

$= [1, x_{21}^2, x_{22}^2, \sqrt{2}x_{21}, \sqrt{2}x_{22}, \sqrt{2}x_{21}x_{22}) : x_2'$

$= (x_1')^T (x_2')$

$x_1 \rightarrow x_1'$ $\quad$ $x_2 \rightarrow x_2'$ $\quad$ $x_1^T x_2 \rightarrow x_1'^T x_2'$

Kernalization: d —feature Terms internally→ d' (d'>d)

Mercers Thm

Kernal trick

↓

(2d) d → d' (6d)

↙ not=kernsep

↘ kernsep

d'>d

K(x1, x2)

↙ ⟶

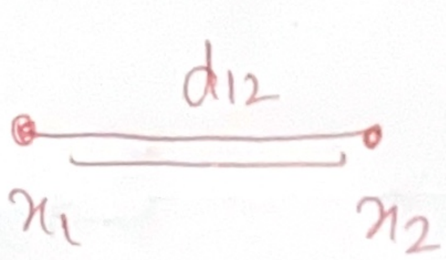
2D (x1, x2)

6D (x1', x2')

Radial Basis Function (RBF) Kernel

SVM: most popular / general purpose: RBF

(x_1, x_2) $K_{RBF}(x_1, x_2) = \exp\left(\frac{-\|x_1 - x_2\|^2}{2\sigma^2}\right)$ ← hyperpara-



$\|x_1 - x_2\|^2 = d_{12}^2$

Soft margin Kernel SVM

((RBF)

→ C: hyperparameter

$$K(x_1, x_2) = \exp\left(-\frac{d_{12}^2}{2\sigma^2}\right) \qquad d_{12}^2 \| (x_1 - x_2)$$

① $d_{12}\uparrow$: $k(x_1, x_2)\downarrow$

$d\uparrow$, $d^2\uparrow$

$ed^2\uparrow$

$\frac{1}{ed^2}\downarrow$

$k(x_1, x_2) > k(x_1, x_3)$

② σ

$\sigma = 1$

$\sigma = 0.1$

$\sigma = 10$

① $d = 0$

$k = 1$

② $d\uparrow$ $k\downarrow$

$d(x_1, x_2) = 4$

$k = 0$

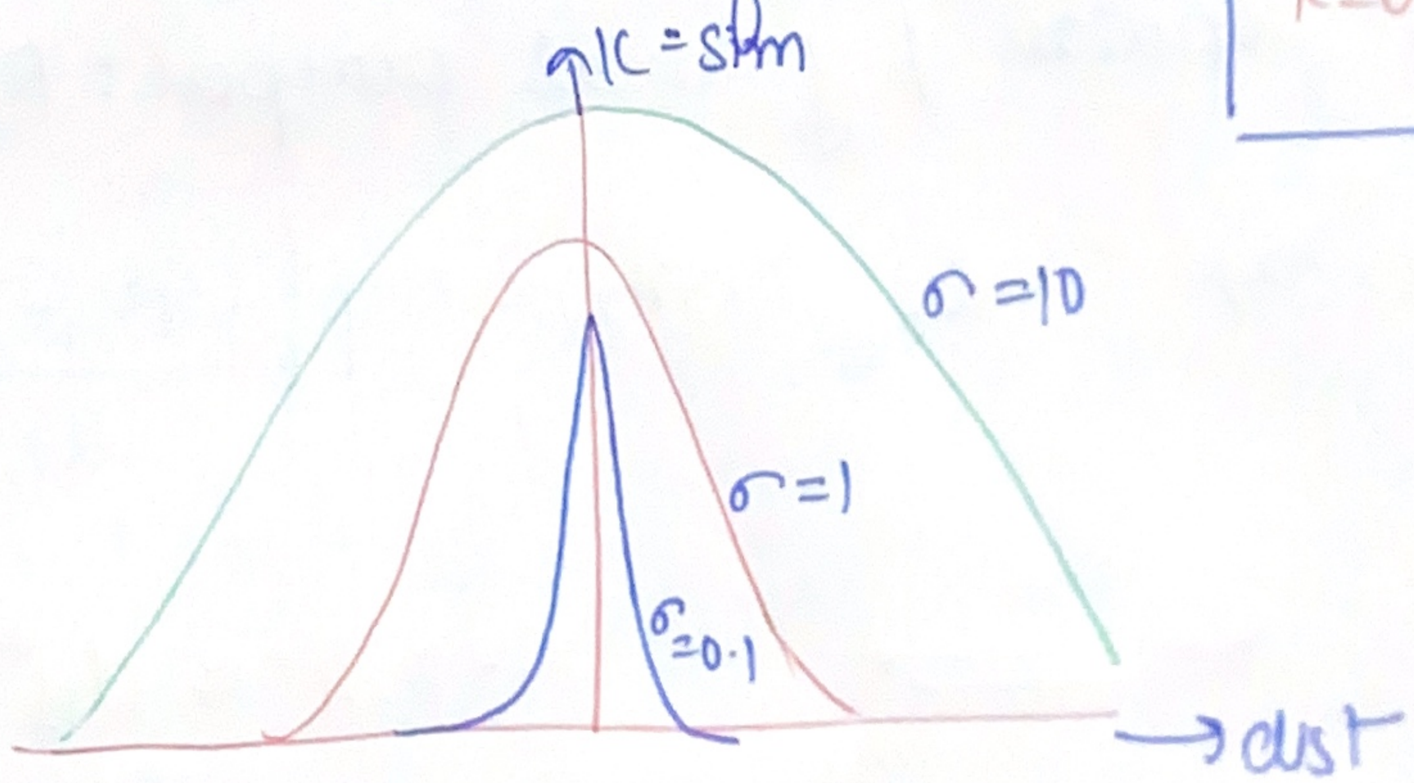
$d(x_1, x_2) = 0.5$

kernel → sim func

distance → dissimi func

$k = 0 \rightarrow sim = 0$

$\Rightarrow dist = \infty$



$\sigma\uparrow$;

[dissim ≒ dist
sim ≒ k]

K-NN & RBF Kernel

RBF-SVM ≈ K-NN

$\sigma \uparrow \Rightarrow K \uparrow$ in KNN (in RBF)

don't know the best kernel

↓

Simply use RBF SVM

↙ ↘

Soft margin ← : C σ : RBF

C, σ ← grid / Random Search

Domain Specific kernels :

Polynomial; RBF → general purpose kernel

↳ KNN

Kernel Trick ≈ (F+) → Domain specific

→ art

given problem

↳ String Kernels → Text classfn

FT: hand part of ML

↳ partially replaced by finding right kernel

Train and Run Time complexity of SVM

Train → SGD

↳ specialized alg (dual) → Sequential minimal optimization (SMO)

LibSVM: best library for training SVMs ↑

↳ Sklearn

Training Time: ∩ $O(n^2)$ for kernel SVMs

∩ (2007): $O(nd^2)$ if $d<n$

if n is large → $O(n^2)$ ↑↑

↑ Typically do not use SVM when n is large

internet

Run Time $f(x_q) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x_q) + b$

↑

$\alpha_i = 0$ for non SVs

SVs = k

$O(kd)$ if # SVs are small

$O(k_d)$ is small

nu - SVM

C - SVM → original formulation $C \geq 0$

↓

parameter

alternate formulation of SVM

nu - SVM $\quad 0 \leq nu \leq 1$

hyperparameter $\quad nu \geq$ fraction of errors

$nu \leq$ fraction of SVs

SVM Drawn

10% errors

↳ $nu = 0.1$

1% errors

$nu = 0.01$

$nu = 0.01 \Rightarrow$ % age of errors $\leq 1\%$

$\#SVs \geq 1\%$ of n

Run Time complx : fewer SVs

But

$n = 0.01 \Rightarrow$ errors $\leq 1\%$

but $SVs \geq 1\%$ of n

$n = 100,000$

$\#SVs = k \geq 1000$

Support Vector regression (SVR) $\quad y_i \in \mathbb{R}$

SVM classfn $\quad$ SVC $\rightarrow y_i \in \{+1, -1\}$

Math $\quad \min_{w,b} \frac{1}{2}\|w\|^2$

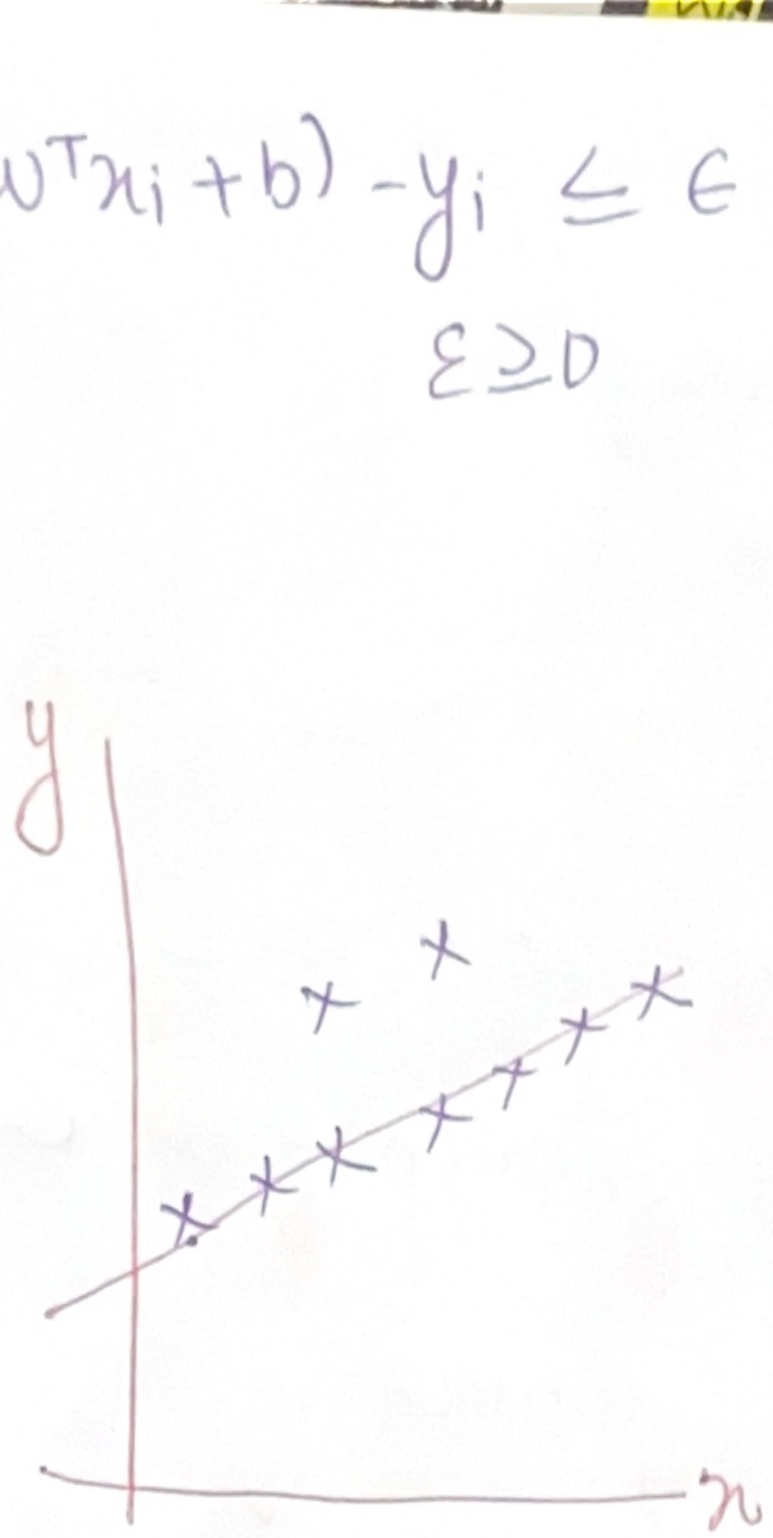
Sot $\quad y_i - (w^T x_i + b) \leq \epsilon \rightarrow$ hyperp

$\hat{y_i}$

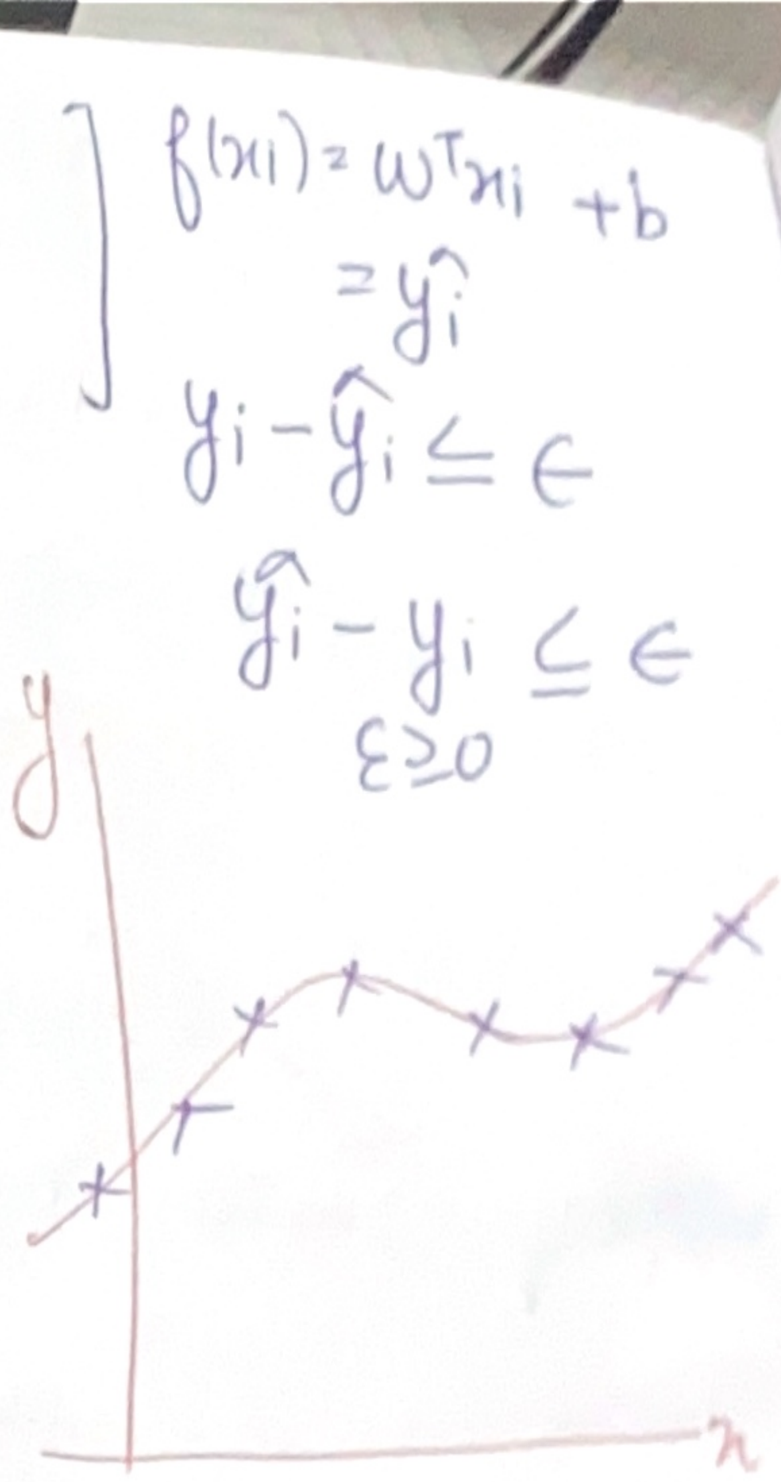
linear form SVR

$$|(w^T x_i + b) - y_i| \leq \epsilon \quad \forall i$$
$$\epsilon \geq 0$$

$$f(x_i) = w^T x_i + b = \hat{y}_i$$
$$y_i - \hat{y}_i \leq \epsilon$$
$$\hat{y}_i - y_i \leq \epsilon$$
$$\epsilon \geq 0$$



LiSVR



Kernel SVR

$\epsilon \downarrow \Rightarrow$ errors are low on Training data
$\Rightarrow$ overfitting ↑

$\epsilon \uparrow \Rightarrow$ errors on $D_{train} \uparrow \Rightarrow$ underfitting ↑

RBF SVR $\longrightarrow$ KNN-reg

Cases (SVM)

→ feature Engg & FT: Kernel design, finding the right Kernel

→ decision Surface Li·SVM: hyperplane
Kernel SVM: (d) $x_i \rightarrow$ non linear Surface
kernel trick ↓
$d' > d$ (d') $x_i' \longrightarrow$ linear Surface

→ Similarity / dist func
  ↳ kernel : $K(x_i, x_j)$

→ Interpretability & feature importance → Kernel SVM
  ↳ FI for various features
  ↳ forward feature selection

→ Outliers → v. little impact
  ↳ SVs that matter
  ↳ RBF with small σ ; → KNN with small K

→ Bias-Var :- C ↑ ⇒ overfit ⇒ high-var
  C ↓ ⇒ underfit ⇒ high bias

RBF-SVM : C, σ ↳ RBF

→ large d → v. good for SVM ⓓ <→ ⓓ'
  ↳ kernel (RBF)

→ Best Case : (right kernel)

Worst Case (n is large) → Train time is high
(K) is large ⇒ low latency is not possible
  ↳ # SVs

Logistic regression