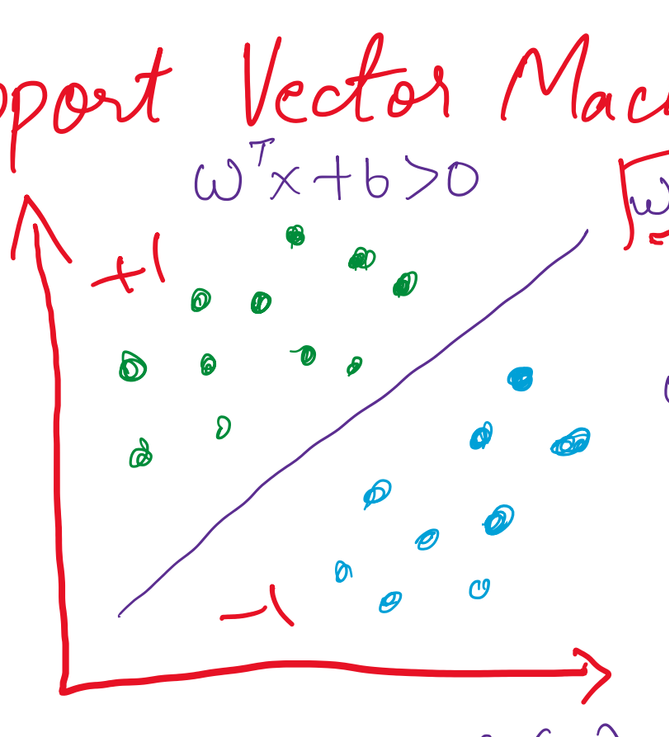
SVM, Decision Trees and Ensemble Methods

Saturday, 15 March 2025 7:27 PM

Support Vector Machines



← Model Equation

w is hyperplane

X ← datapoint vector in R^d

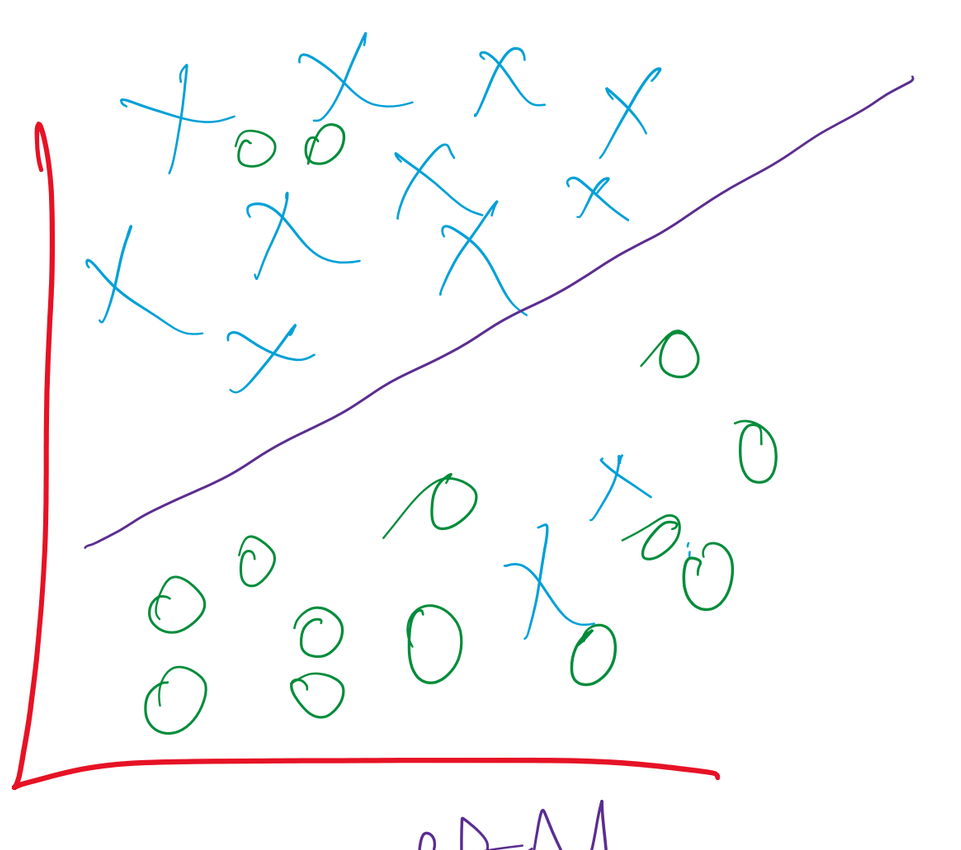
y = +1, -1 (car, truck)

$f(x) = \text{sign}(w^Tx+b)$

+1, -1

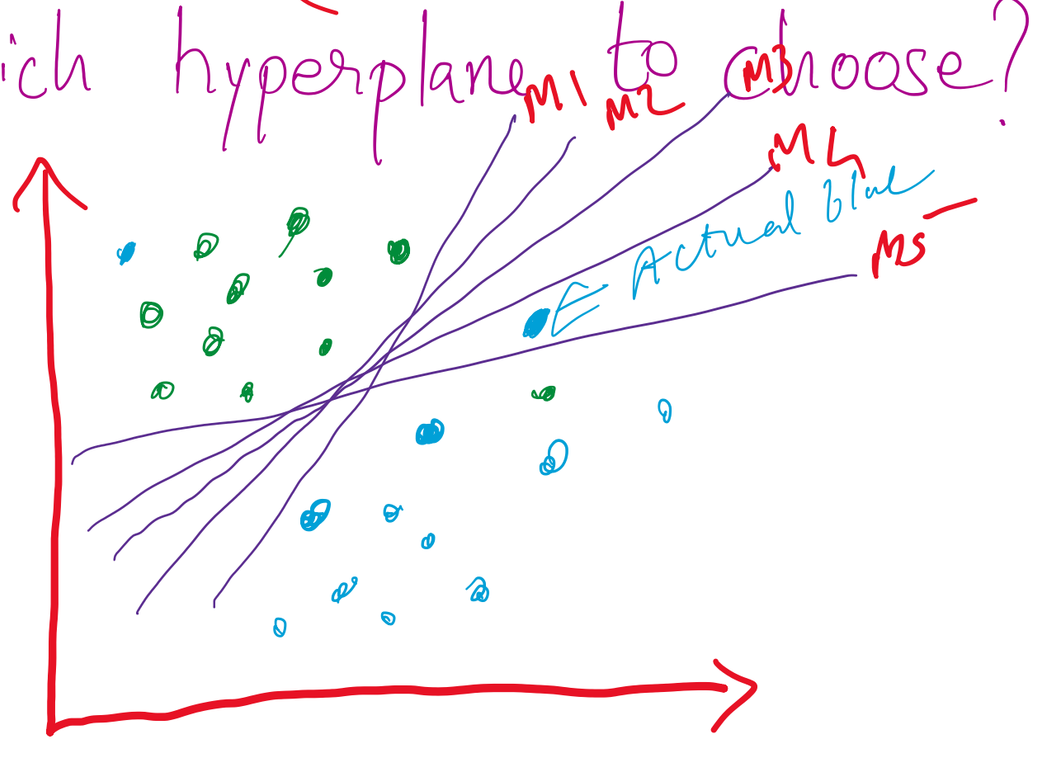
(w^Tx+b) w, b

Learn a vector w, b



Soft M

Which hyperplane to choose?



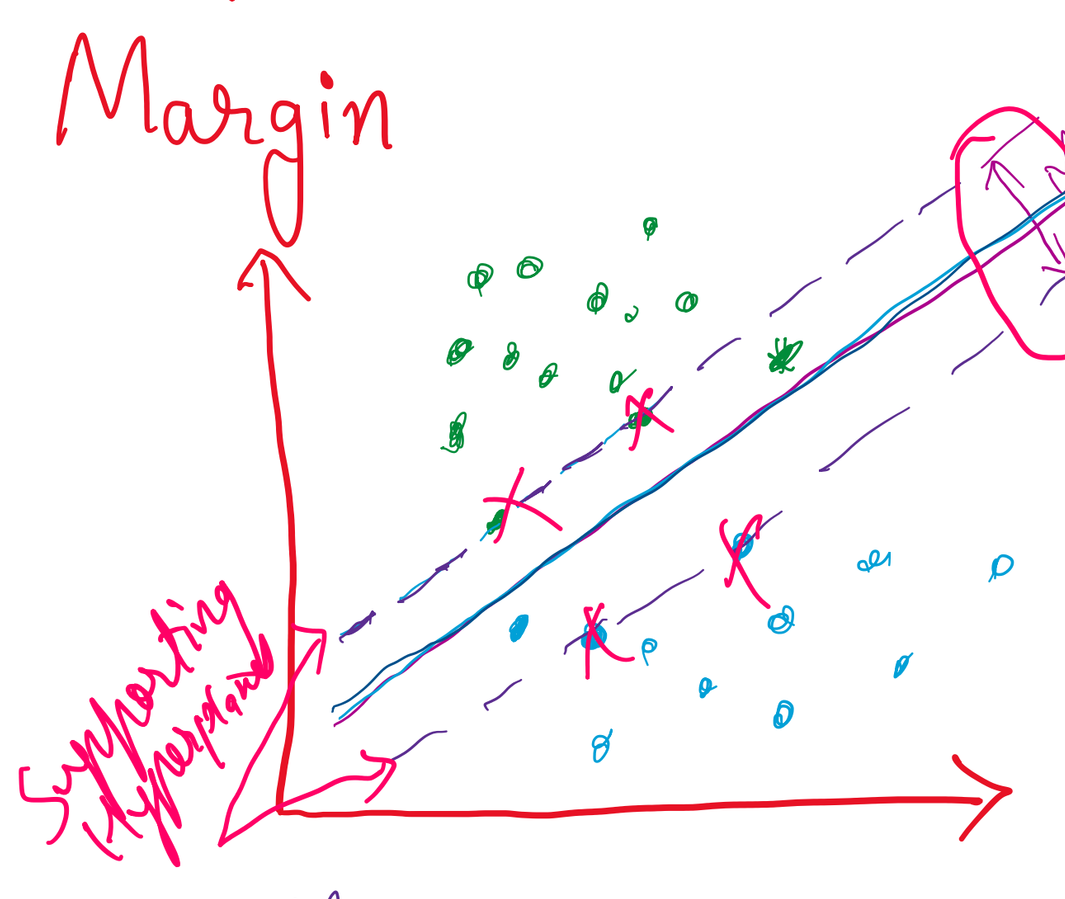
Model w, b

$w \in R^d$, $b \in R$, x is datapoint $x \in R^d$

$\text{sign}(w^Tx+b) > 0$, < 0

$\text{sign}(w^Ta+b) > 0 \Rightarrow +1$, $\Rightarrow -1$

Margin



Maximize our Margin

Maximum Margin Classifier

Support Vectors

SVM also has a convex optimization

Primal — Dual

Gradient

-> SVM is a maximum margin Classifier

-> Issues with SVM
-> Linear Separability
-> Hard v/s Soft Margin SVM

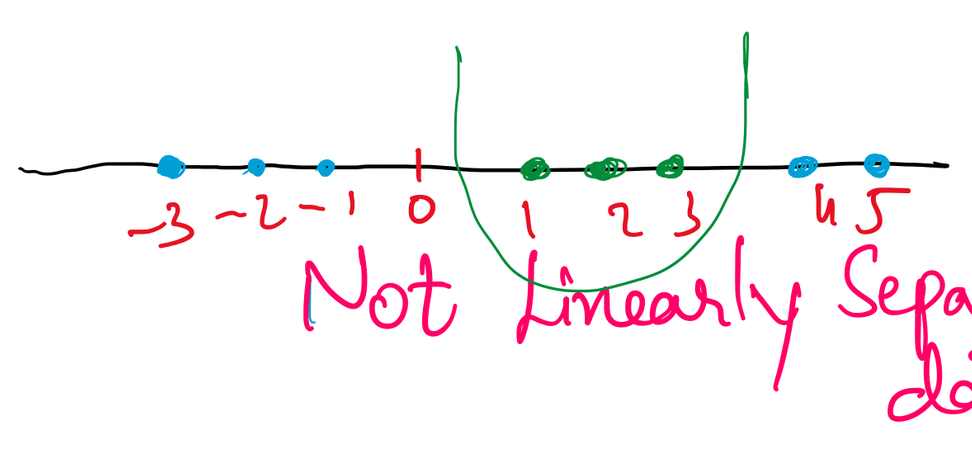
use in practice

Training data

Everything must exactly

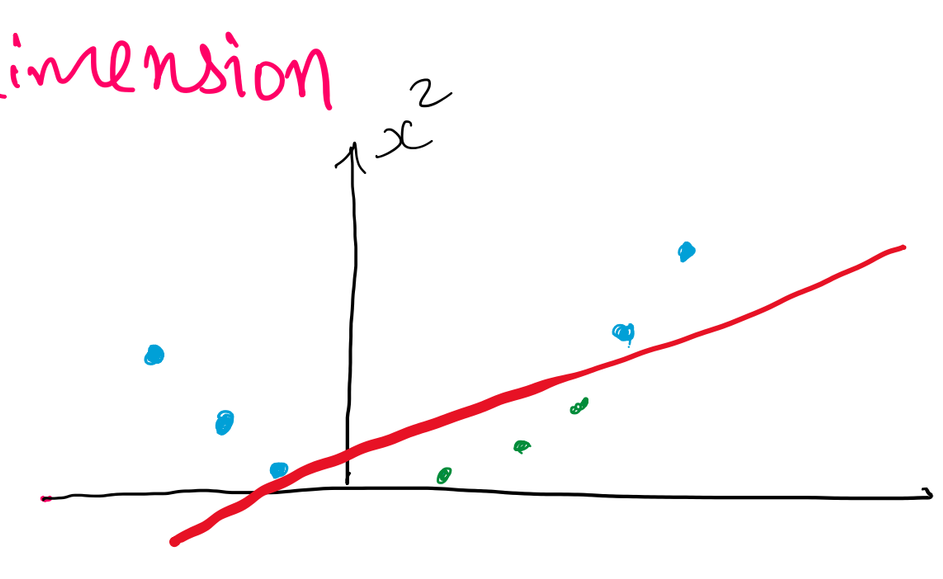
slight change in the optimization function for these

-> The Kernel trick



Not Linearly Separable data

-> However take it to a higher dimension



$X = \begin{bmatrix} x \\ x^2 \end{bmatrix}$

$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} -1 \\ 1 \end{bmatrix}$ $\begin{bmatrix} -2 \\ 4 \end{bmatrix}$ $\begin{bmatrix} -3 \\ 9 \end{bmatrix}$

$\begin{bmatrix} 2 \\ 4 \end{bmatrix}$ $\begin{bmatrix} 3 \\ 9 \end{bmatrix}$ $\begin{bmatrix} 4 \\ 16 \end{bmatrix}$ $\begin{bmatrix} 5 \\ 25 \end{bmatrix}$

High Level Idea:

Original Feature Space Can Always be Mapped to some higher dimensional space where training data is separable

Kernel Trick Ensures you do not have to explicitly take data to higher dimensions.

-> SVM with kernel

-> You do not need to explicitly take your data to a higher dimension

-> SVM specify the kernel

model = SVM("kernel")

Kernels

-> Do not use kernel

Identity — Linear kernel

-> If you use a kernel

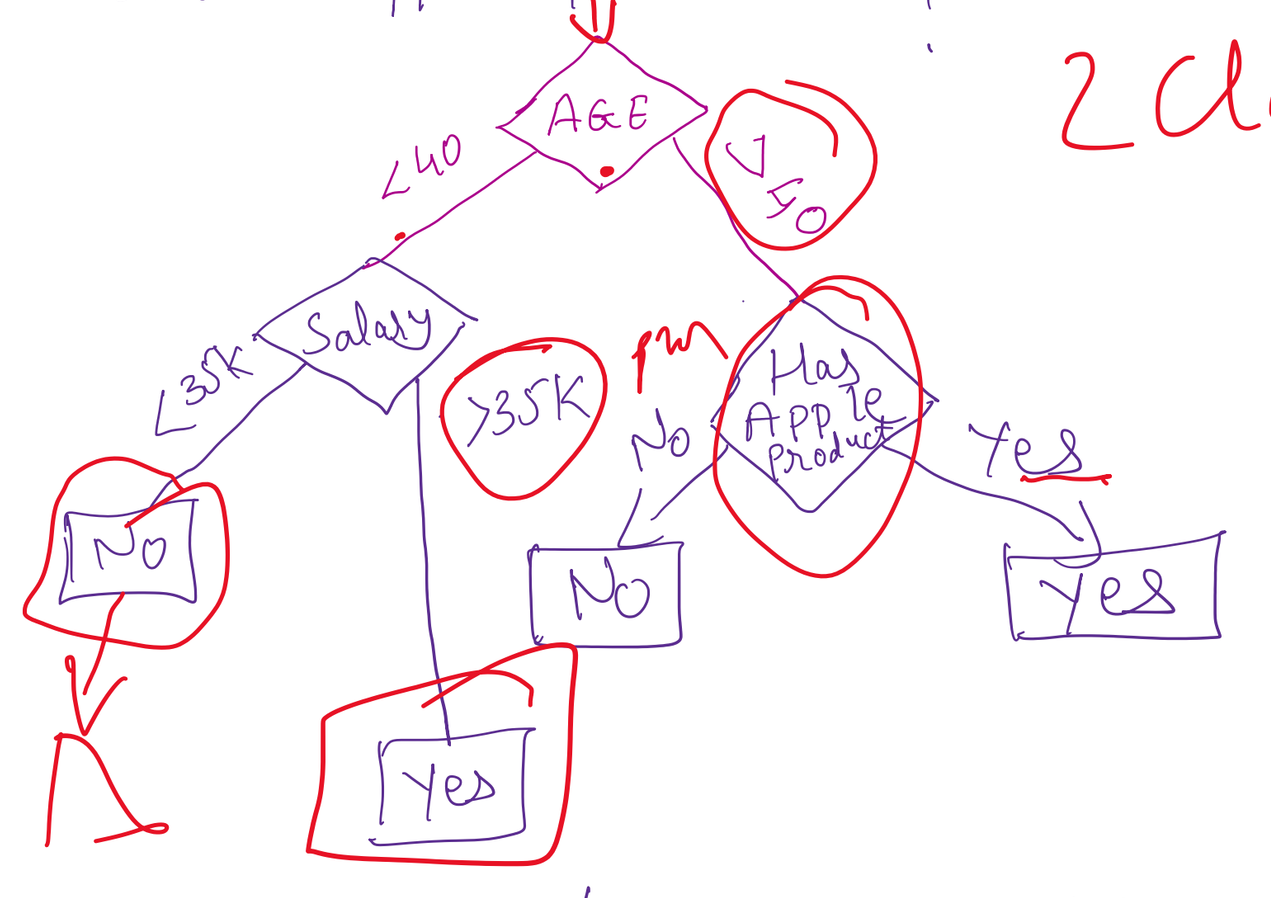
-> RBF ← datapoints to different higher dimensions

-> Polynomial kernel

->

Decision Trees

Whether a customer will buy an apple product?



2 Class

Yes 1

NO 0

->Decision trees are very intuitive classifiers

-> They are explainable & can work with categorical Variables/features too

Points to consider

Which feature to be used for split?

-> Gini Index
-> Entropy
} Performance more or less similar

Capture the amount of information present in a particular feature column

How much should be the depth of the tree

-> Stop Early

-> Prune

At the leaf nodes of the trees you will get your classes

Advantages:-

-> Explainable

-> Categorical Variables

Gender — Male — 0, Female — 1

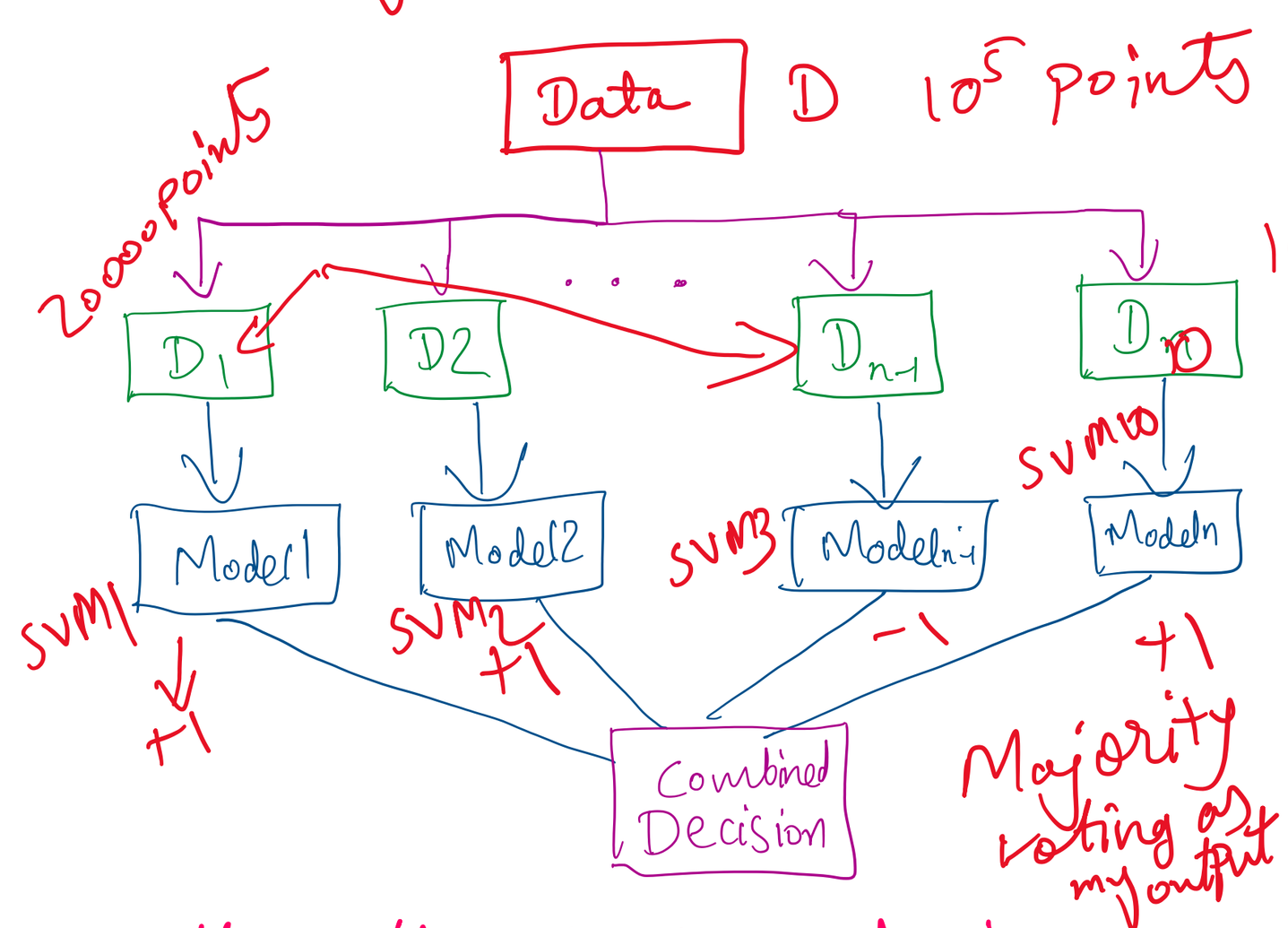
City — Ahme — 1, Mum — 2, Pune — 3

Ensemble Methods:-

-> Single Classifier may not be good

-> Use Multiple Classifiers like a committee of experts

Bagging: Bootstrap Aggregation



When the Model used is a decision tree, it is called a Random Forest Classifier.

100 trees

56 Apple

44 Not buy

$\Sigma (a_i^T x - b_i)^2$

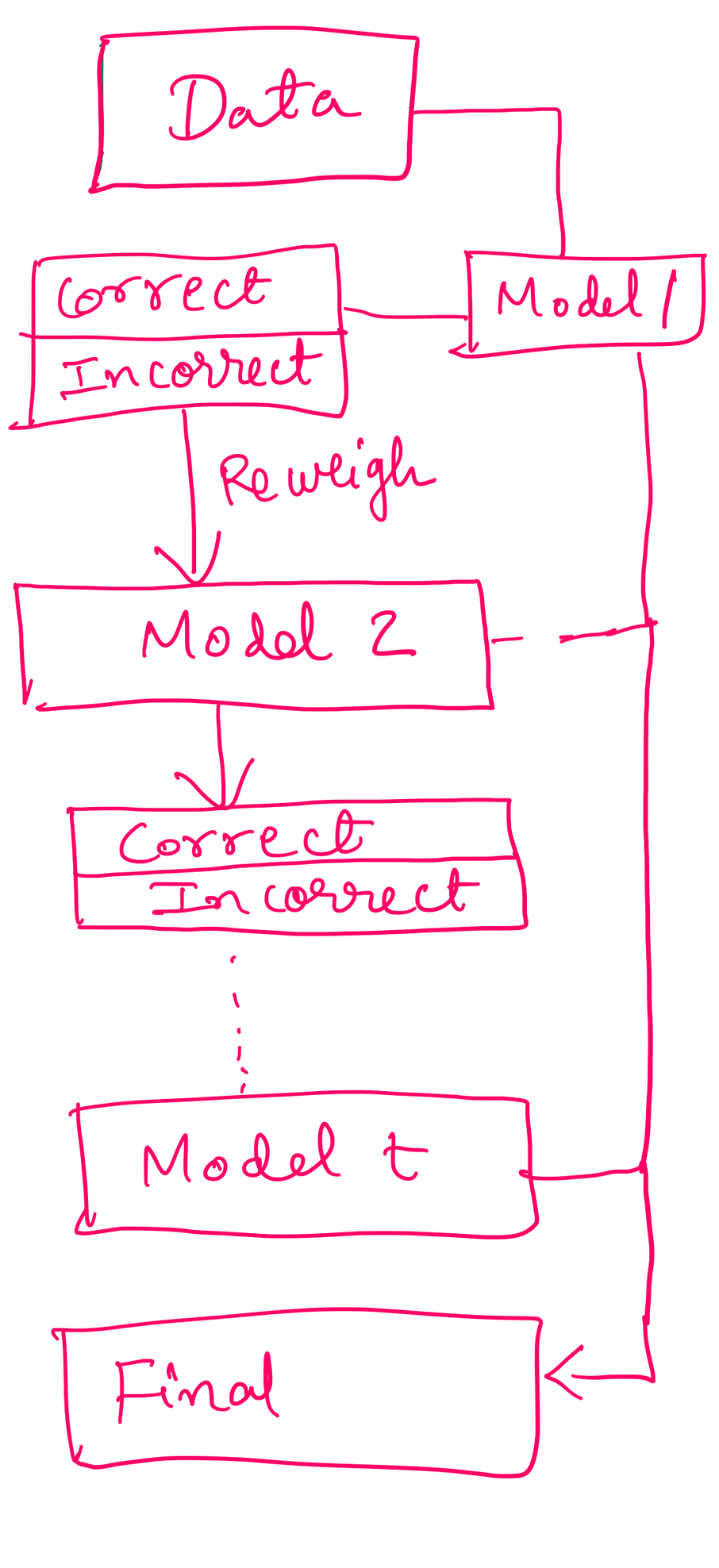
100 a_i's, b_i's

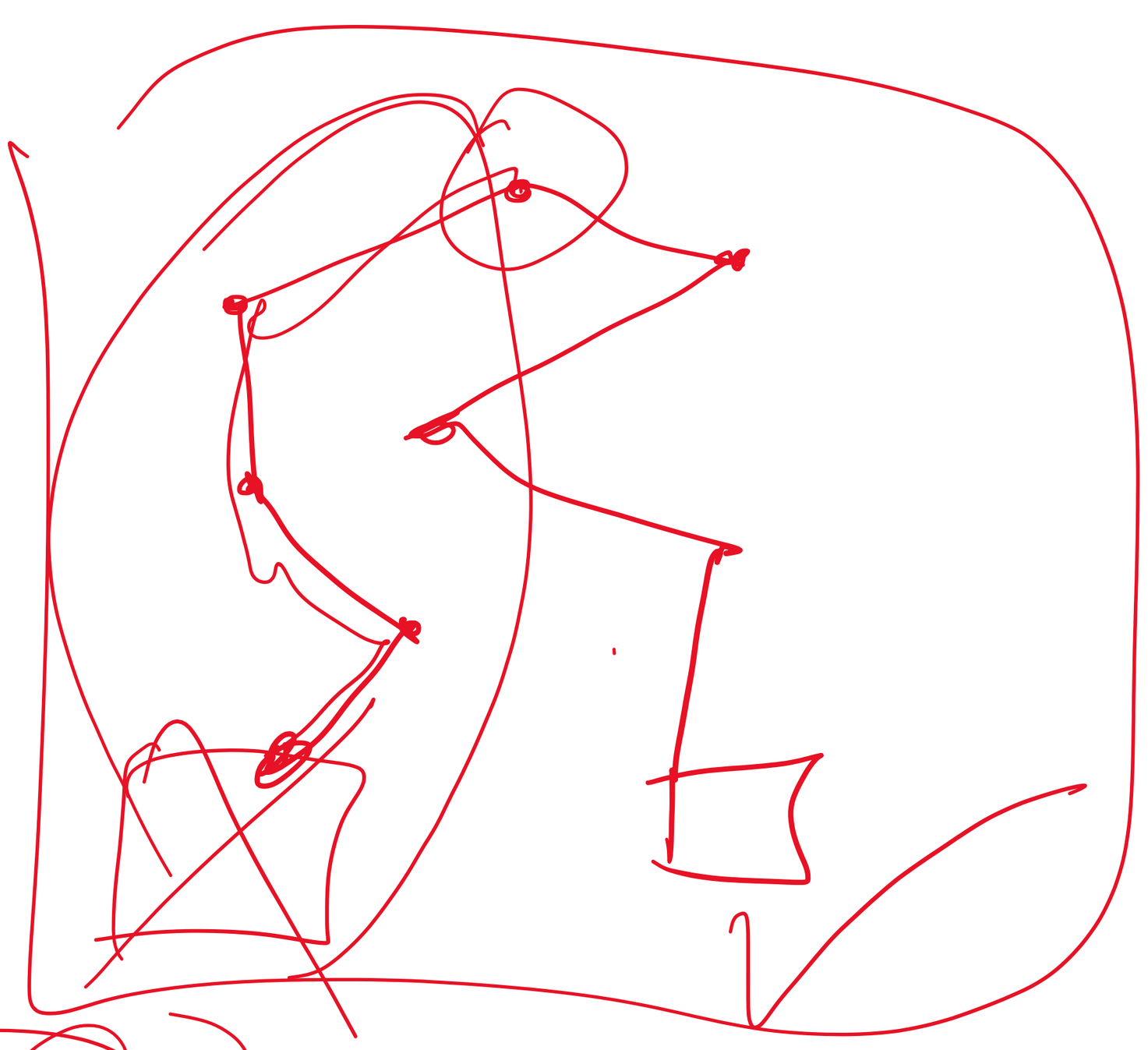
x

Boosting:-

-> Combine "Weak" Learners to form a strong Learner

-> Popular Boosting Techniques AdaBoost & Gradboost





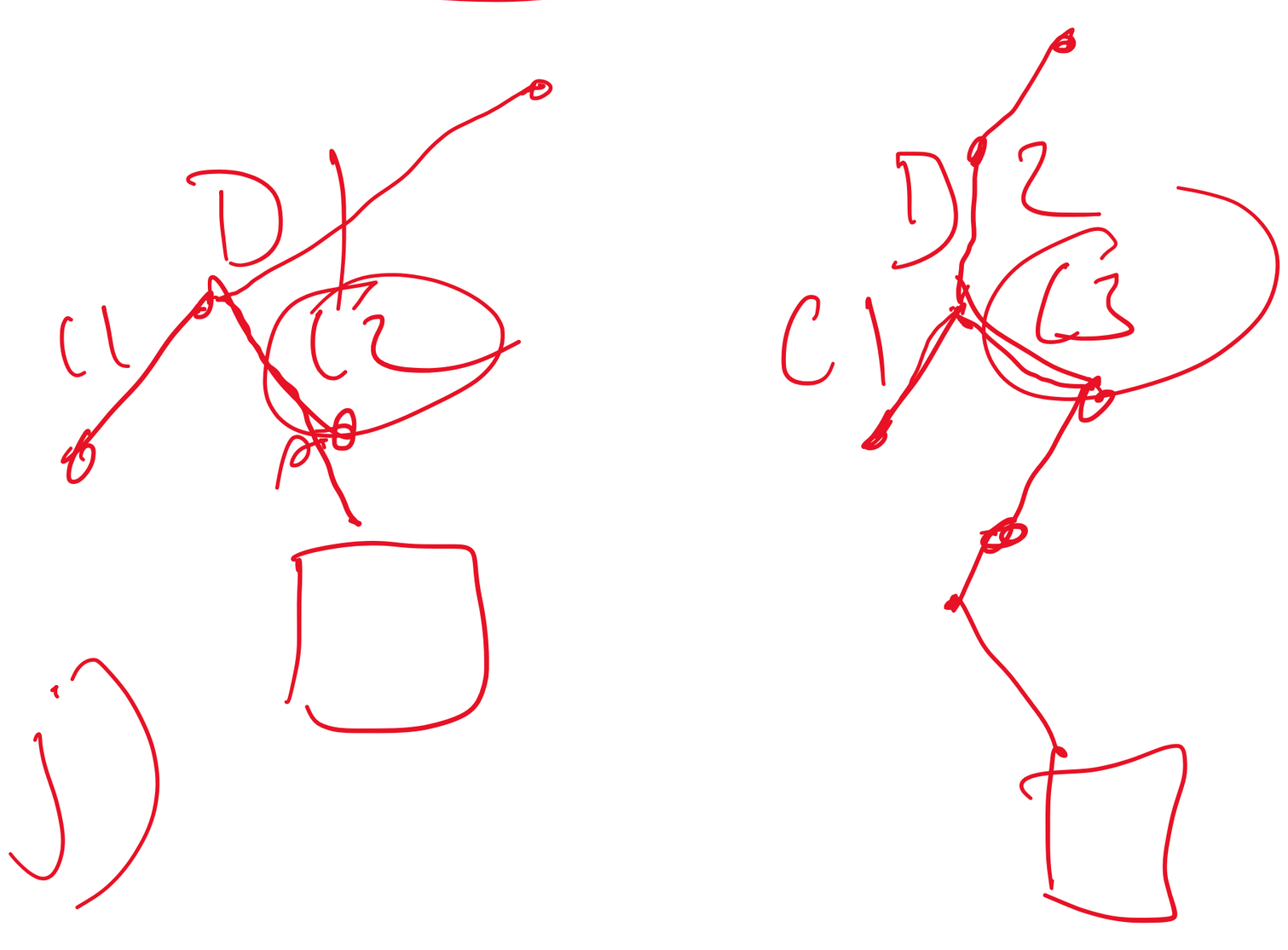
Data1 -> SVM1

Data2 -> SVM2

$a_i^T a_j$

low dimens

$K(i,j)$



testpoint

SVM1 ... SVM 10

+1 +1 -1 +1

8 +1

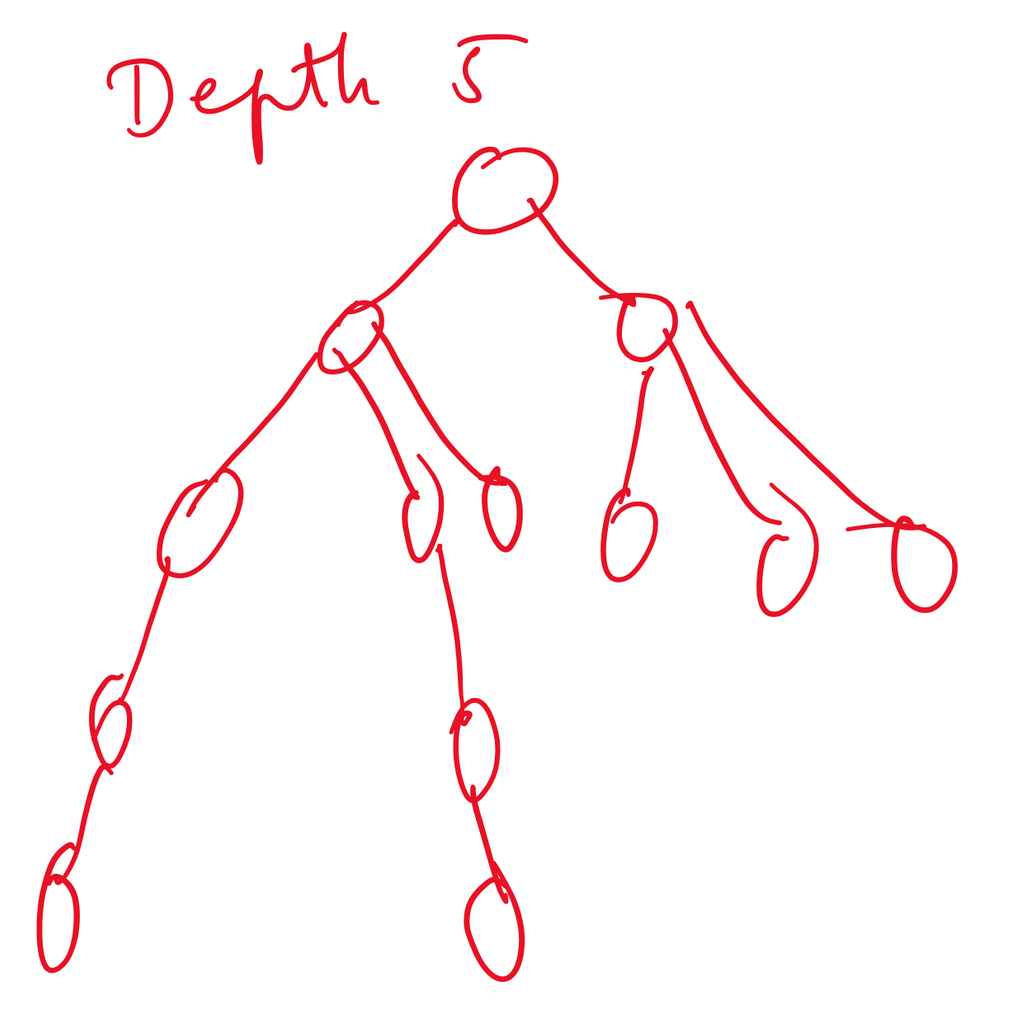
2 = -1

+1

K(i, j)

n×n

Depth 5



Pruning