



Lecture 4

CS 6375

# Support Vector Machines

Rishabh Iyer

University of Texas at Dallas



# CS 6375

# Support Vector Machines

Rishabh Iyer

University of Texas at Dallas



# CS 6375

## Support Vector Machines

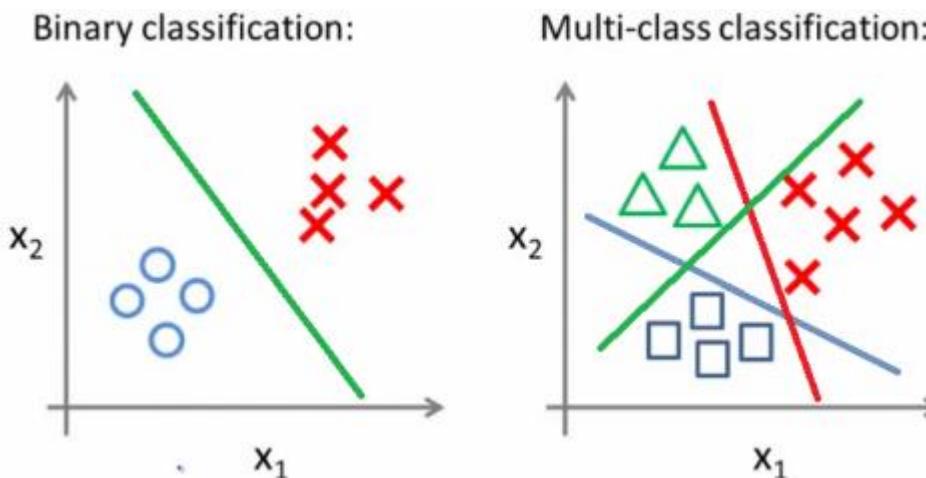
Rishabh Iyer

University of Texas at Dallas

# Recap: Classification

## Classification vs Regression

- Input: pairs of points  $\underbrace{(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})}_{x^{(m)} \in \mathbb{R}^n}$  with
- $y^{(m)} \in [0, k - 1]$
- If  $k = 2$ , we get Binary classification



# Recap: Hypothesis Space

- Hypothesis space: set of allowable functions  $f: \underline{X} \rightarrow \underline{Y}$
- Goal: find the “best” element of the hypothesis space
  - How do we measure the quality of  $f$ ?

$$f(x, w, b) = \text{sign}(w^T x + b) \rightarrow \text{Class}^n$$

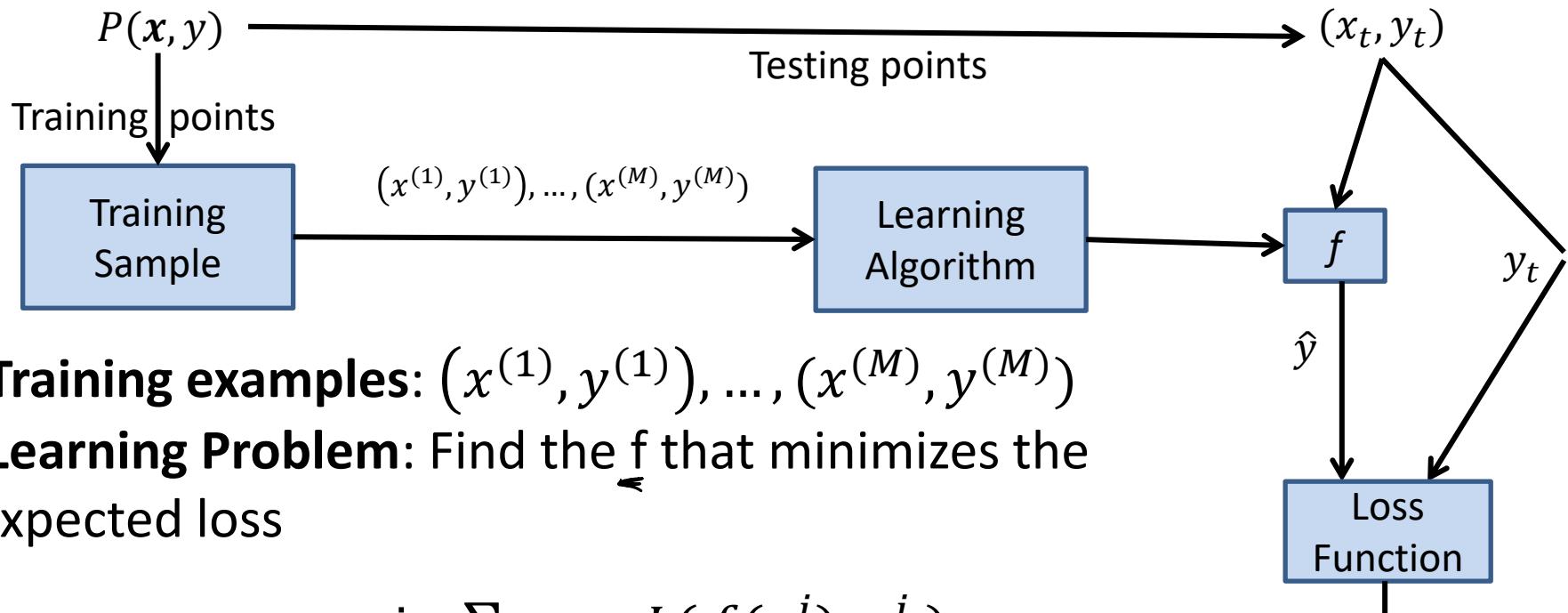
$w^T x + b \rightarrow \text{Regression}$

Goal:  $\forall i, f(x^{(i)}, w, b) \approx y^{(i)}$

Loss:  $L(w, b) = [y^{(i)} - f(x^{(i)}, w, b)]^2$ , (Regression)

$L(w, b)$  =  $\max_3 (0, -y^{(i)} f(x^{(i)}, w, b))$  (Perception)

# Recap: Supervised Learning Workflow



- **Training examples:**  $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$
- **Learning Problem:** Find the  $f$  that minimizes the expected loss

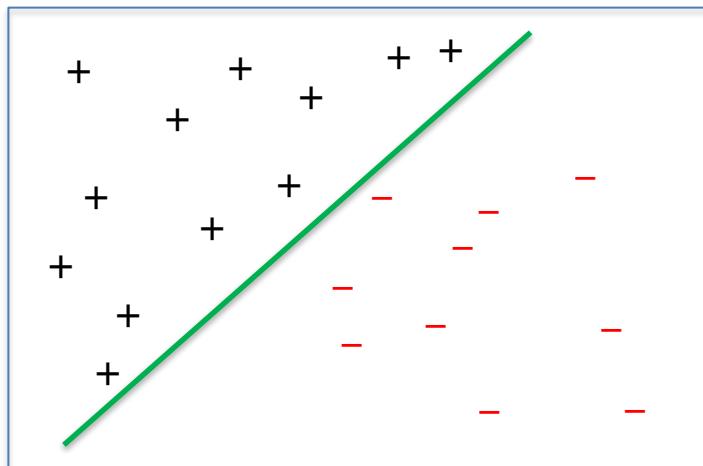
$$\text{Opt} : \min_f \sum_{i=1:m} L(f(x^i), y^i)$$

$$\min_{w, b} \sum_{i=1}^m L(f(x^i, w, b), y^i)$$

- **Testing:** Given a new point  $(x_t, y_t)$  drawn from  $P$ , the classifier is given  $x$  and predicts  $\hat{y}_t = f(x_t)$
- **Evaluation:** Measure the error  $Err(\hat{y}_t, y_t)$  – often same as  $L$

# Recap: Binary Classification

- Input  $(x^{(1)}, y^{(1)}), \dots, (x^{(M)}, y^{(M)})$  with  $x^{(m)} \in \mathbb{R}^n$  and  $y^{(m)} \in \{-1, +1\}$
- We can think of the observations as points in  $\mathbb{R}^n$  with an associated sign (either +/- corresponding to 0/1)
- An example with  $n = 2$



In this case, we say  
that the  
observations are  
linearly separable

# 0/1 Loss Vs Perceptron Loss

- Zero/One Loss which counts the number of mis-classifications:

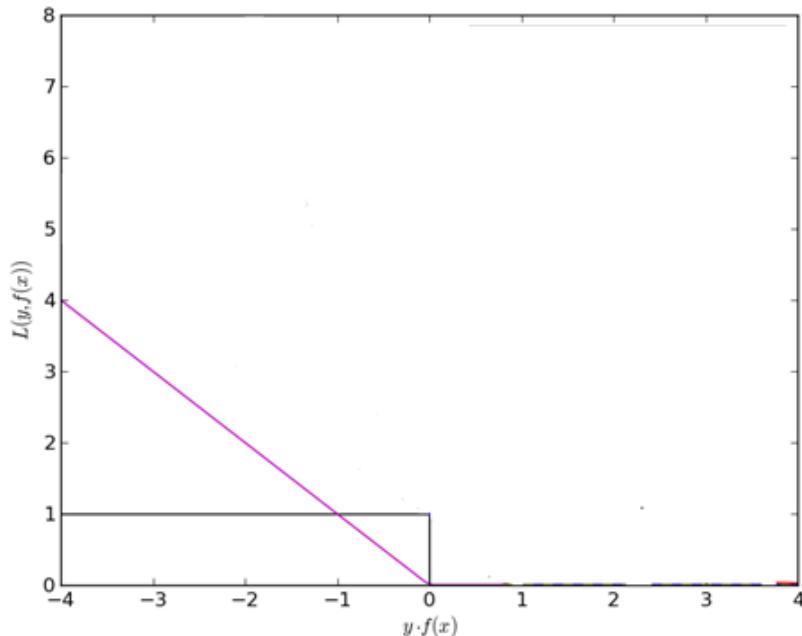
$$\text{zero/one loss} = \frac{1}{2} \sum_m |y^{(m)} - \text{sign}(w^T x^{(m)} + b)|$$

Not Cont  
 Grad techniques  
 won't work

- Perceptron Loss:

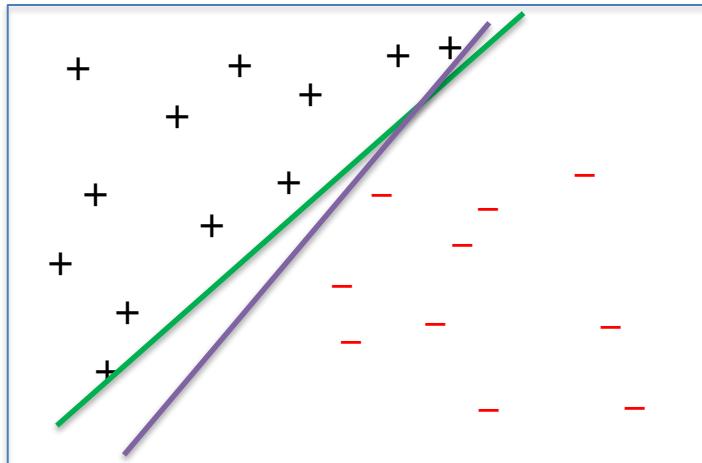
$$\text{perceptron loss} = \sum_m \max\{0, -y^{(m)}(w^T x^{(m)} + b)\}$$

NP hard



# Perceptron Drawbacks

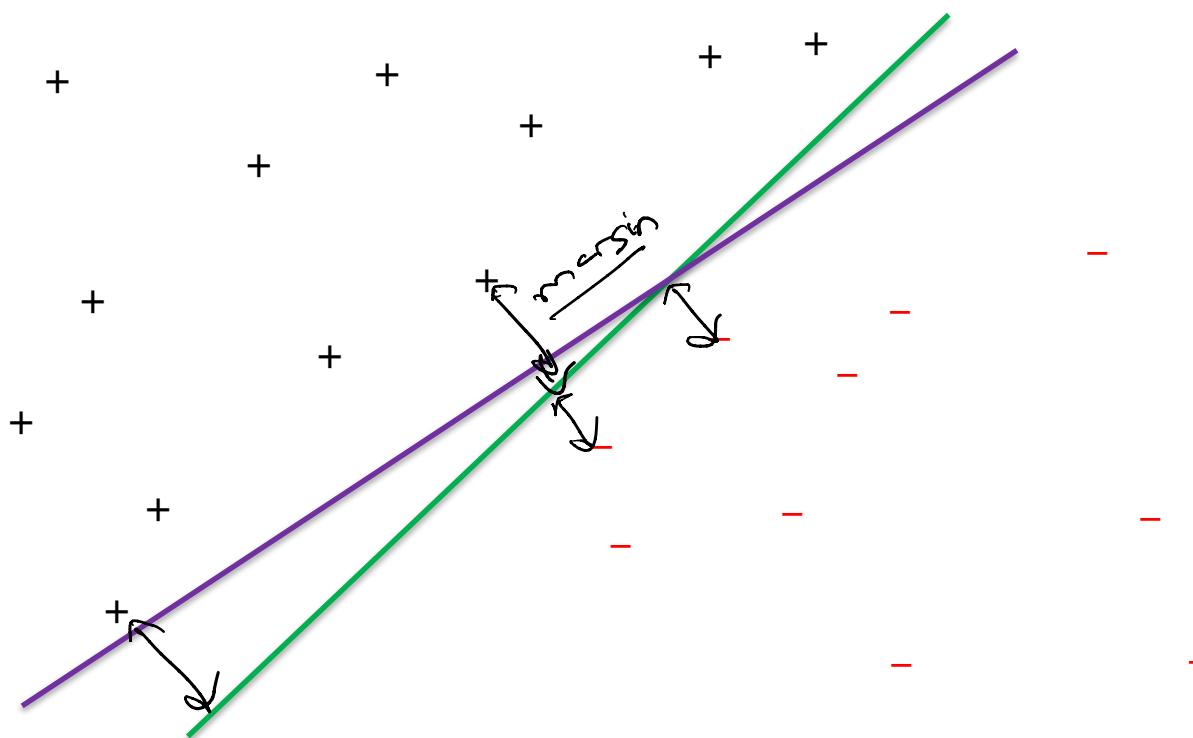
- No convergence guarantees if the observations are not linearly separable
- Can overfit
  - There can be a number of perfect classifiers, but the perceptron algorithm doesn't have any mechanism for choosing between them



# Support Vector Machines



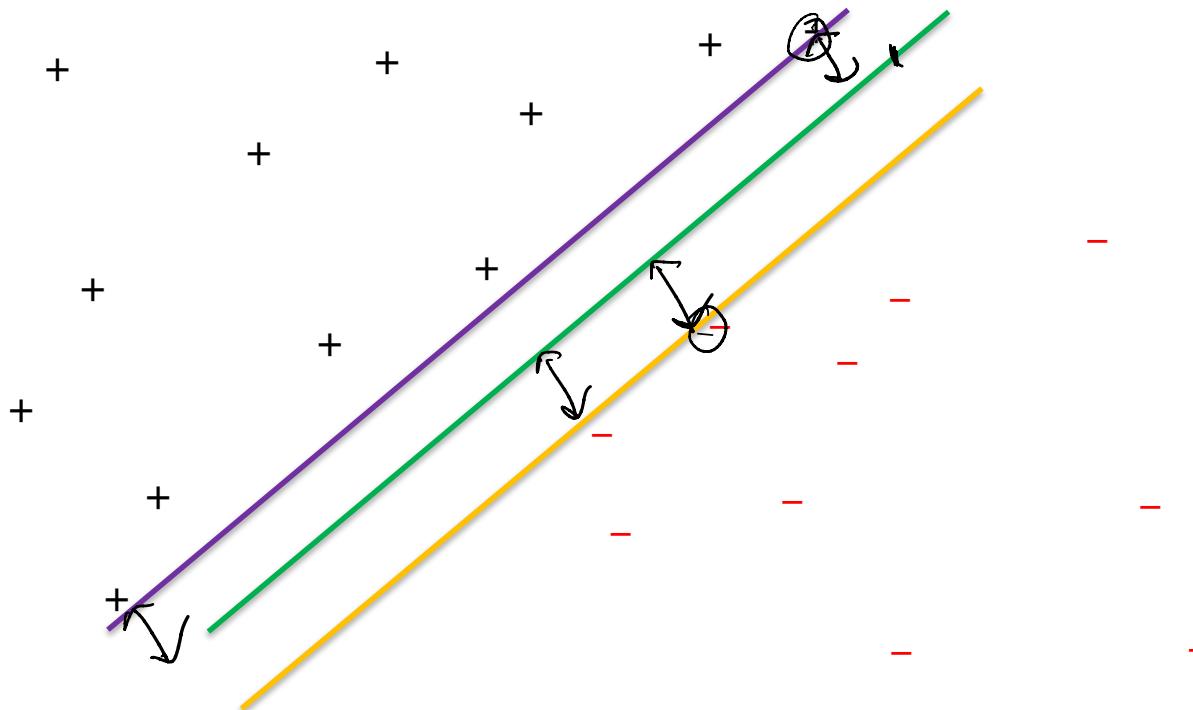
- How can we decide between perfect classifiers?



# Support Vector Machines



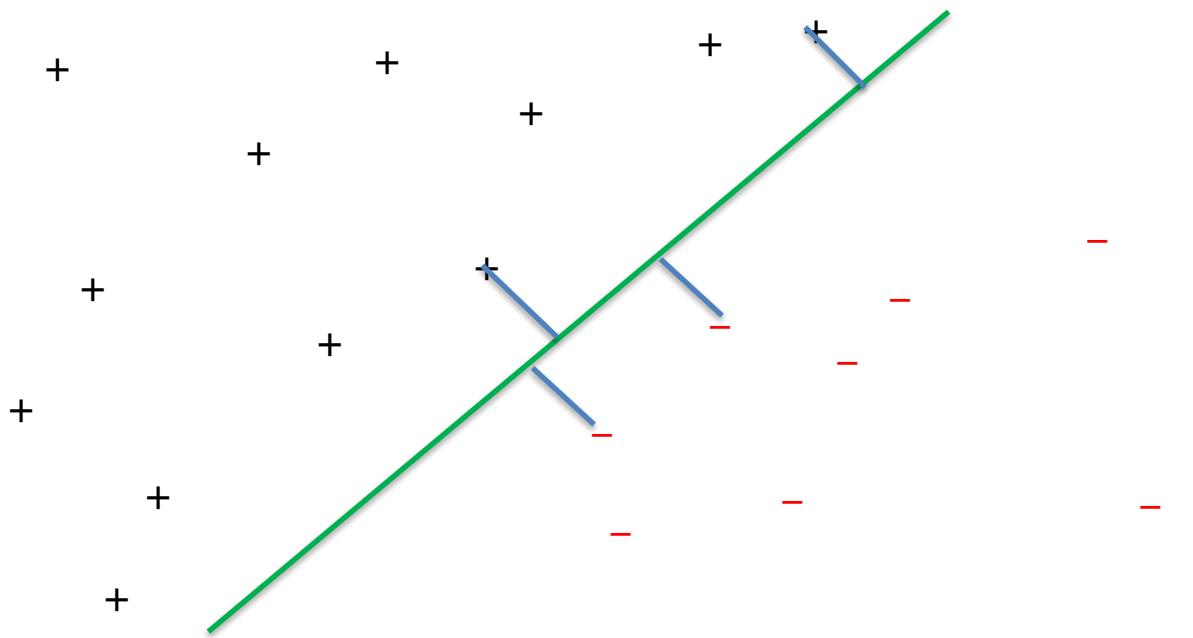
- How can we decide between perfect classifiers?



# Support Vector Machines



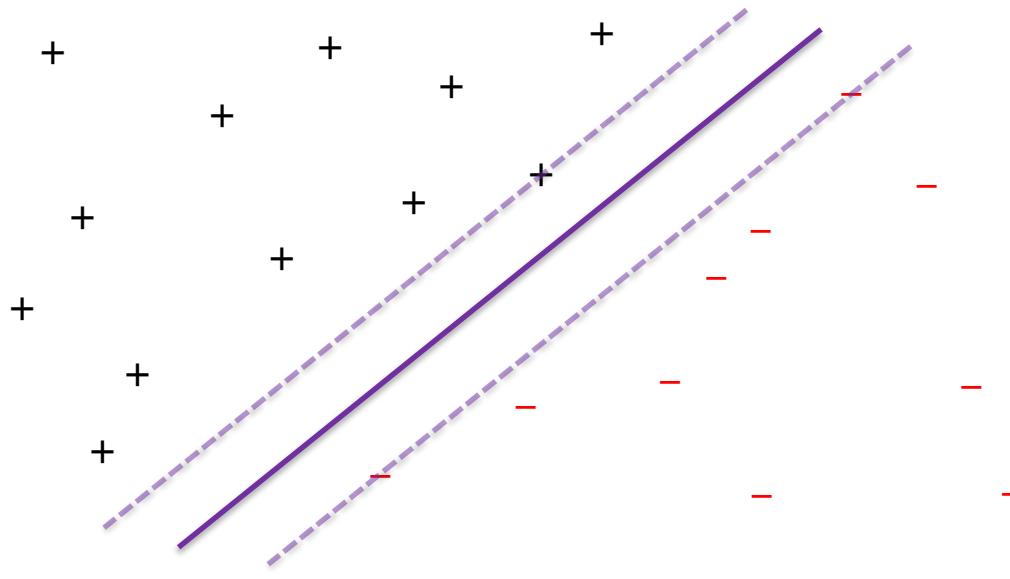
- Define the margin to be the distance of the closest data point to the classifier



# Support Vector Machines



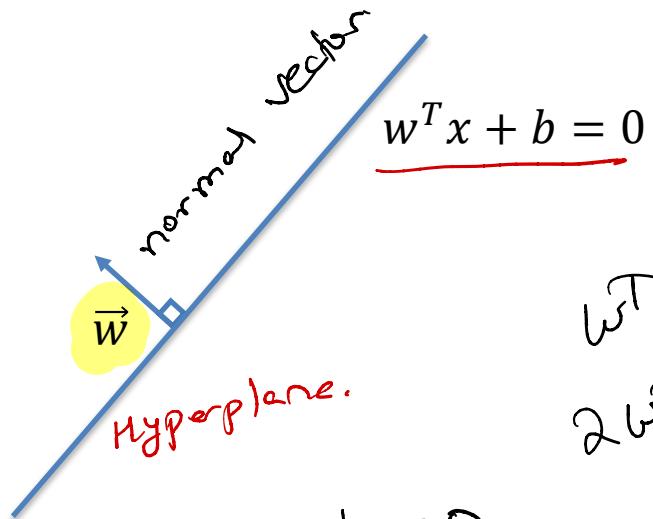
- Support vector machines (SVMs)



- Choose the classifier with the largest margin
  - Has good practical and theoretical performance

"Test Perf"  
Generalization

# Some Geometry



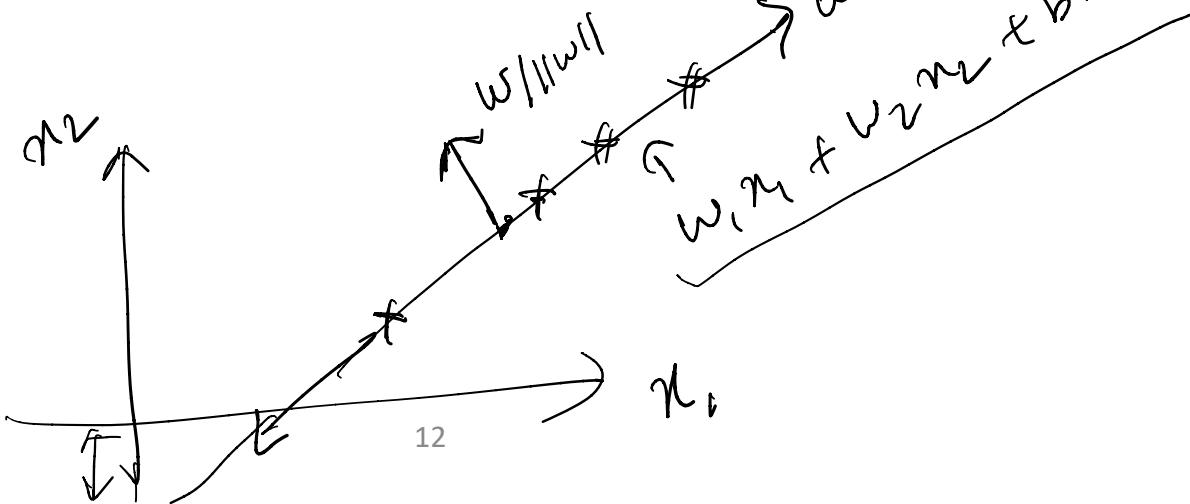
- ① Eq. of hyperplane :
- ② scale invariant

$$\underline{w^T n + b = 0}$$

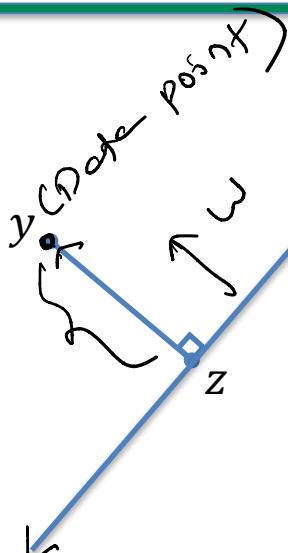
$$2w^T n + 2b = 0$$

$$w^T n + b = 0$$

$$w_1 n + w_2 n + \dots + w_m n + b = 0$$

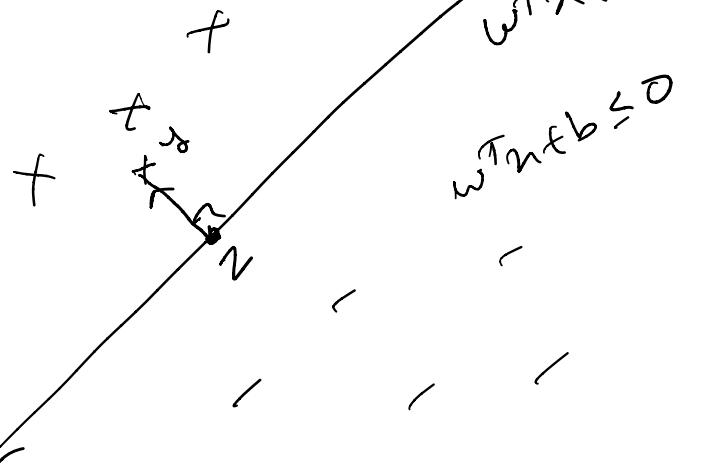


# Some Geometry



$$w^T x + b = 0$$

$$w^T x + b \geq 0$$



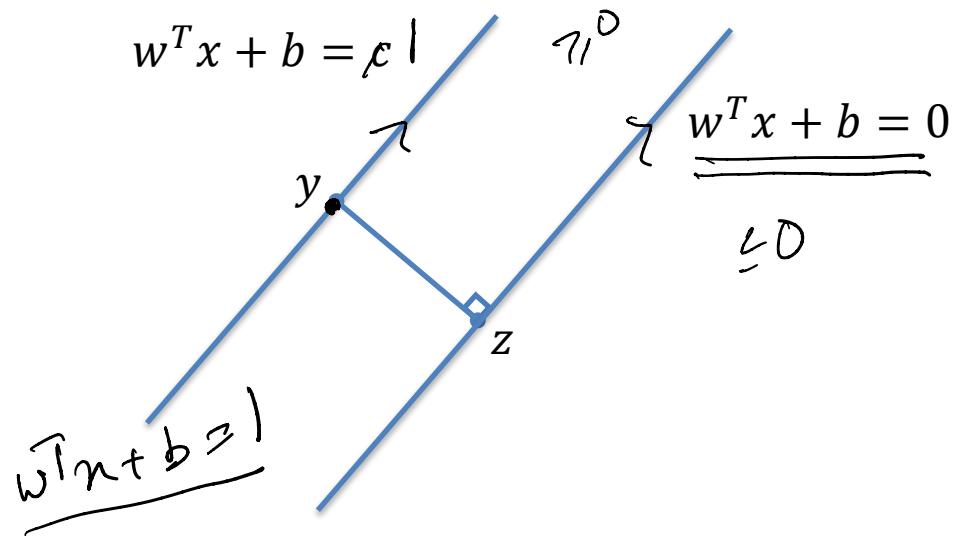
Claim:

$$\|y - z\| = \|y - z\| \frac{w}{\|w\|}$$

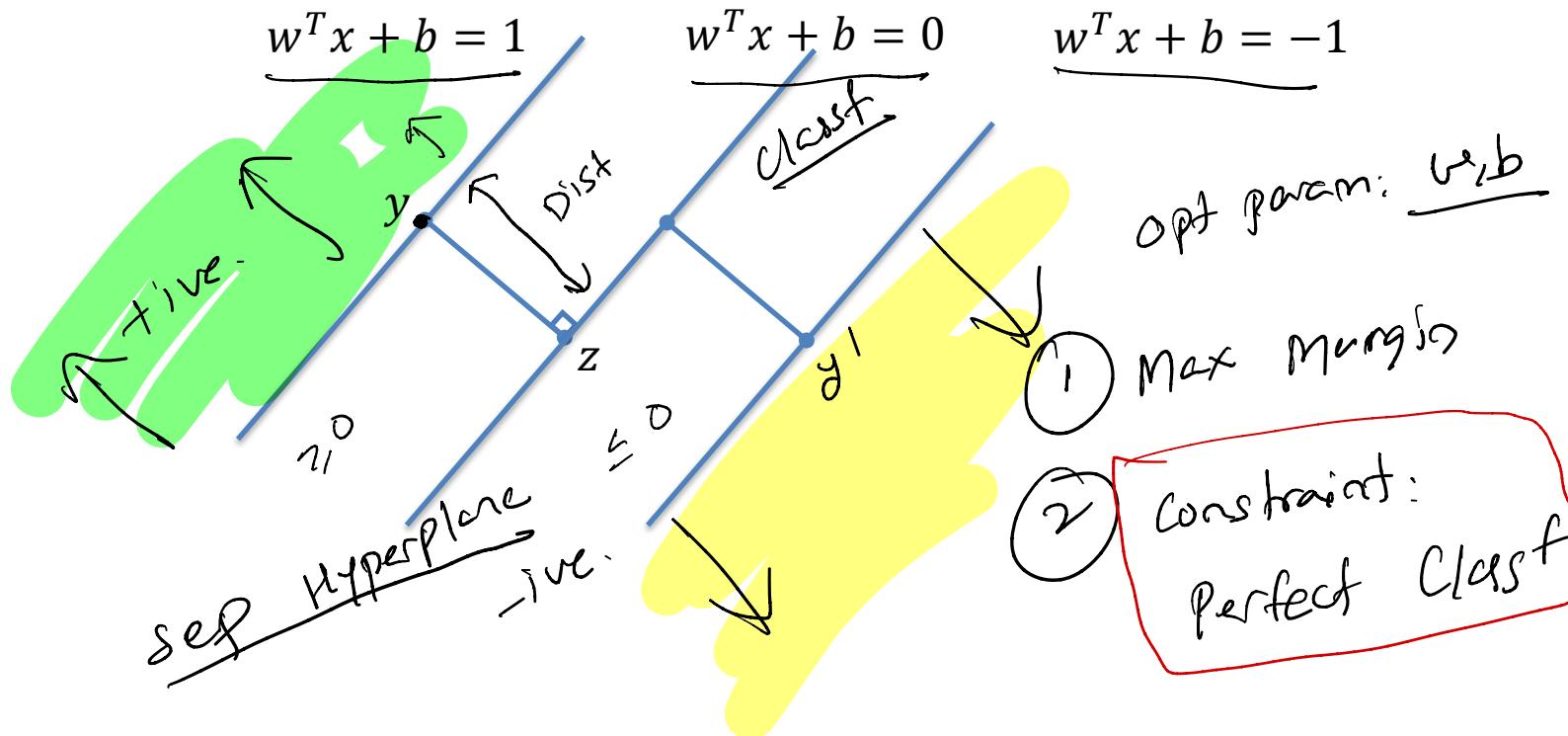
$$\Rightarrow \frac{\|y - z\|}{\|y - z\|} = \frac{w}{\|w\|}$$

$$\frac{\|y - z\|}{\|y - z\|}$$

# Scale Invariance



# Constraints

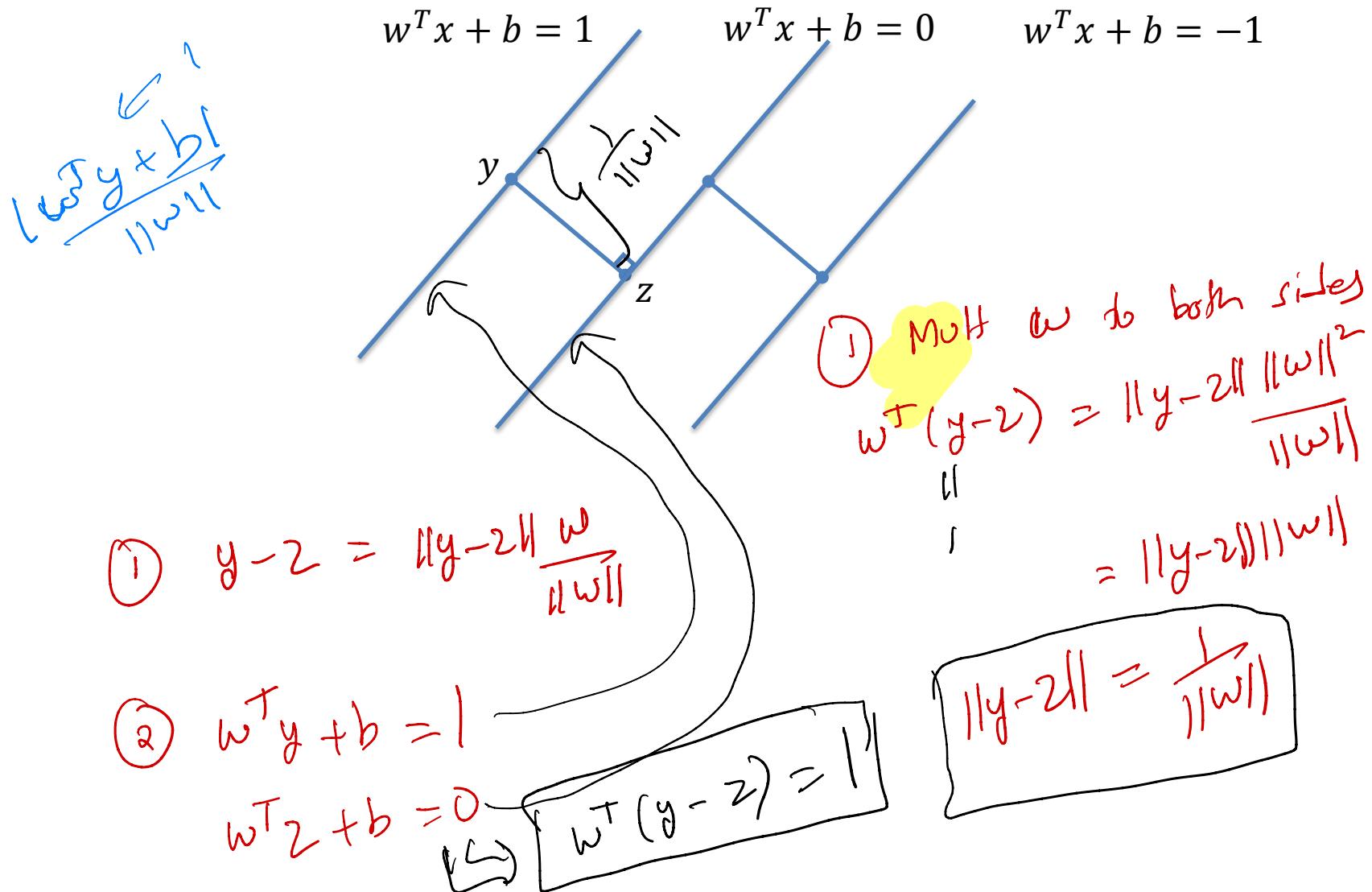


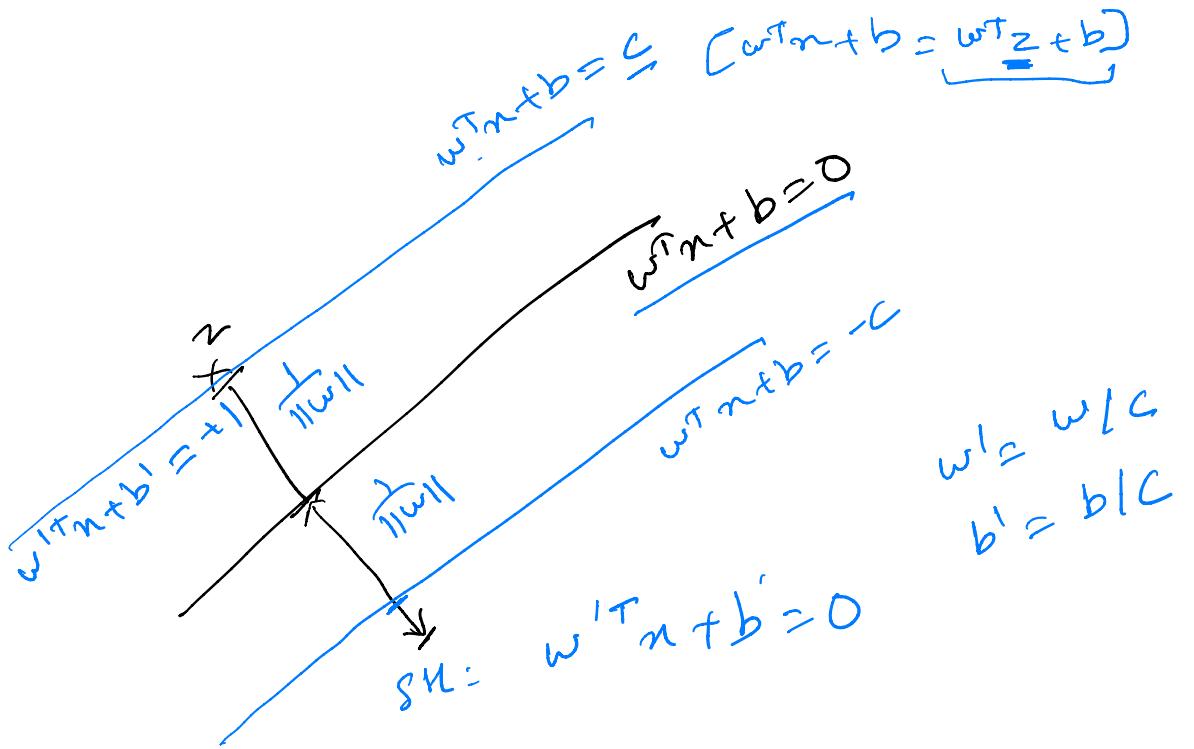
$$\# m = 1 : M \quad \# \text{Data} \quad y^{(m)} [w^T x^{(m)} + b] \geq 1$$

-1  $\rightarrow w^T x^{(m)} + b \leq -1$  ↗

+1  $\rightarrow w^T x^{(m)} + b \geq 1$  ↘

# What is the Margin?





# SVMs



- This analysis yields the following optimization problem

$$\max_{w,b} \frac{1}{\|w\|}$$

[margin]

such that

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i$$

[constraint]

- Or, equivalently,

$$\min_{w,b} \|w\|^2$$

↳ convex

↳ convex

such that

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i$$

[Constrained opt problem]  
Constrained Convex Opt Problem

# SVMs



such that

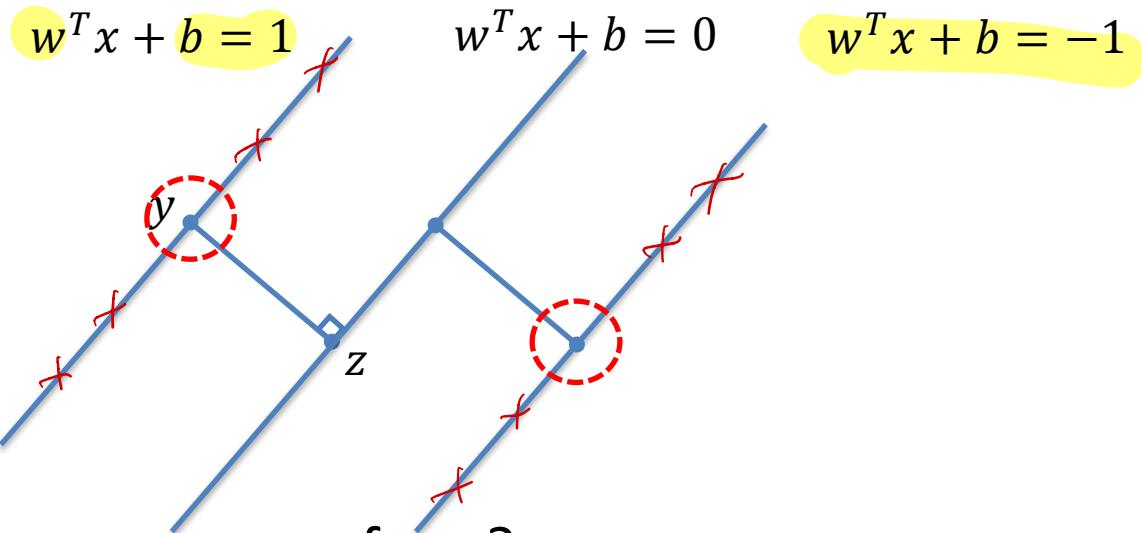
$$\min_{w,b} \|w\|^2 \quad \leftarrow \text{Objective.}$$

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \text{ for all } i \quad \leftarrow \text{Constraint.}$$

- This is a standard quadratic programming problem
  - Falls into the class of convex optimization problems
  - Can be solved with many specialized optimization tools (e.g., quadprog() in MATLAB)

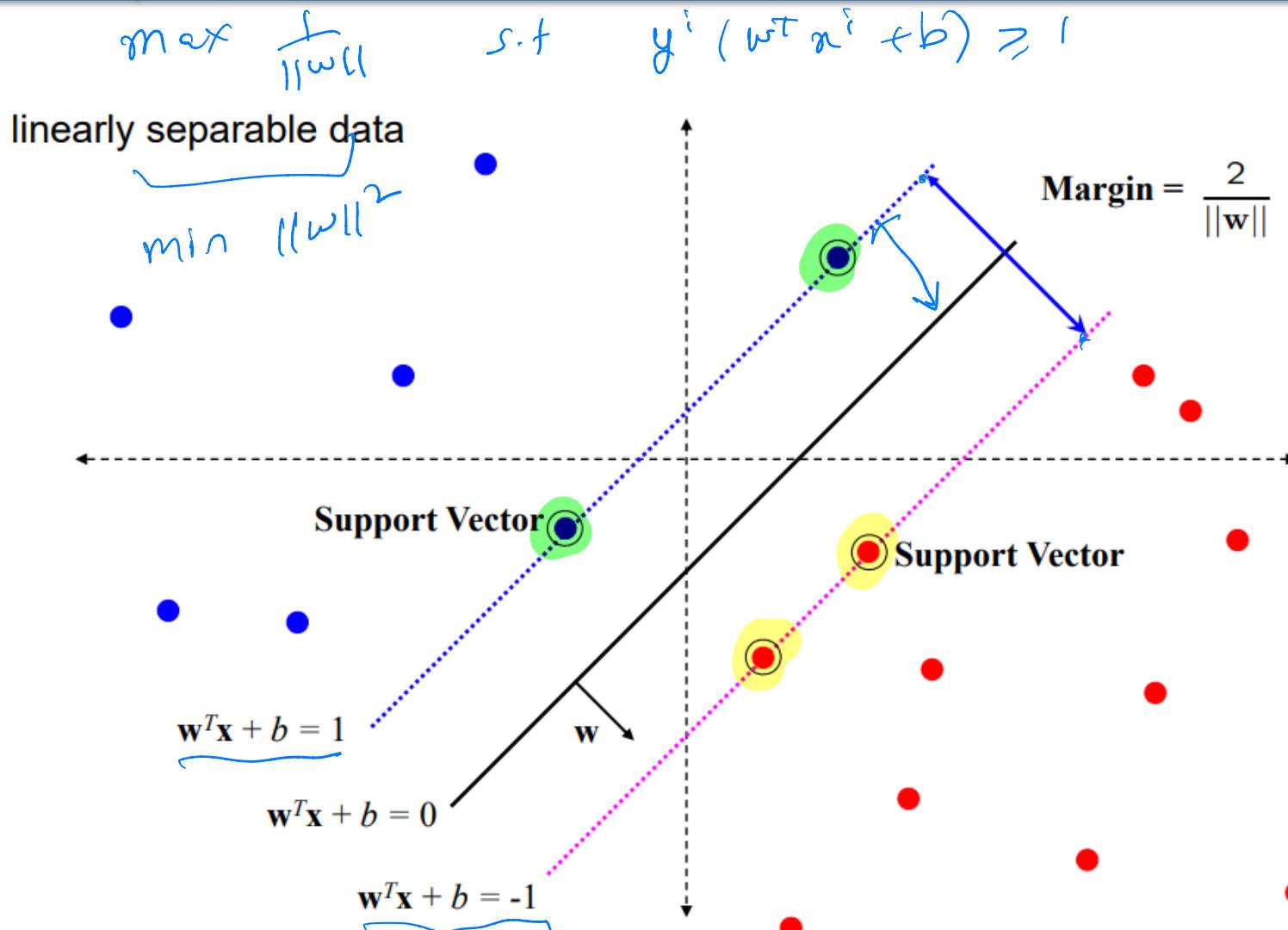
$$\sum_{i=1}^M L(f(x_i, w, b), y_i)$$

# Support Vectors



- Where does the name come from?
  - The set of all data points such that  $y^{(i)}(w^T x^{(i)} + b) = 1$  are called **support vectors**
  - The SVM classifier is completely determined by the support vectors (you could delete the rest of the data and get the same answer)

# Putting Everything Together



# SVMs

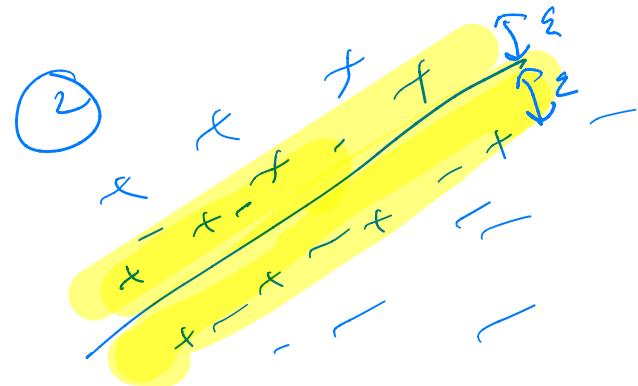


$$w^T \phi(x) + b$$

- What if the data isn't linearly separable?



$$\phi(x) = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ 1 \end{bmatrix}$$



- What if we want to do more than just binary classification (i.e., if  $y \in \{1,2,3\}$ )?

# SVMs



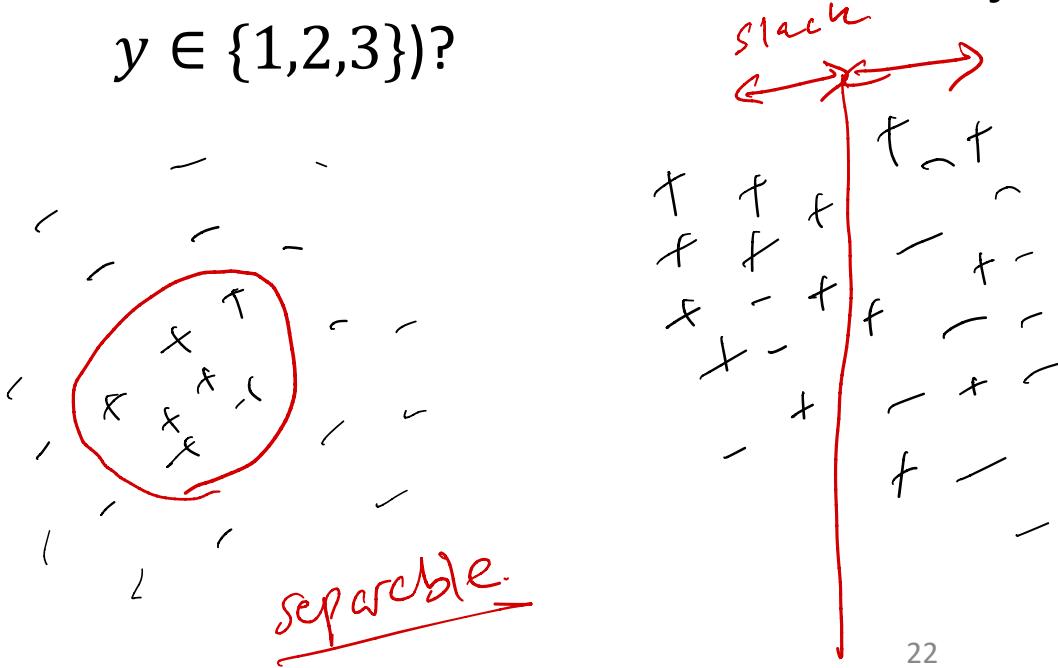
- What if the data isn't linearly separable?

- Higher order (polynomial features)

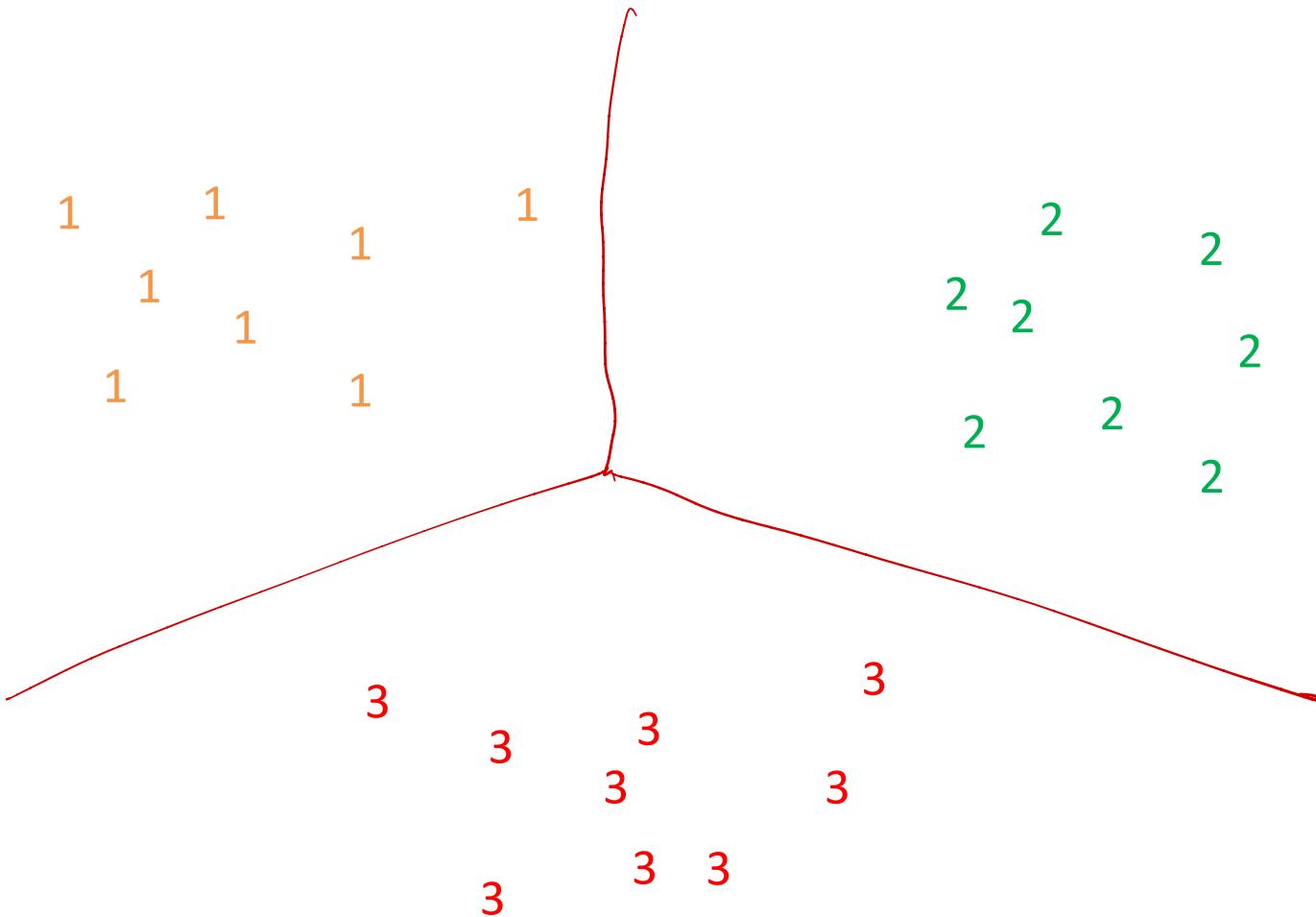
Linear Regression  
Perception

- Relax the constraints (coming soon)

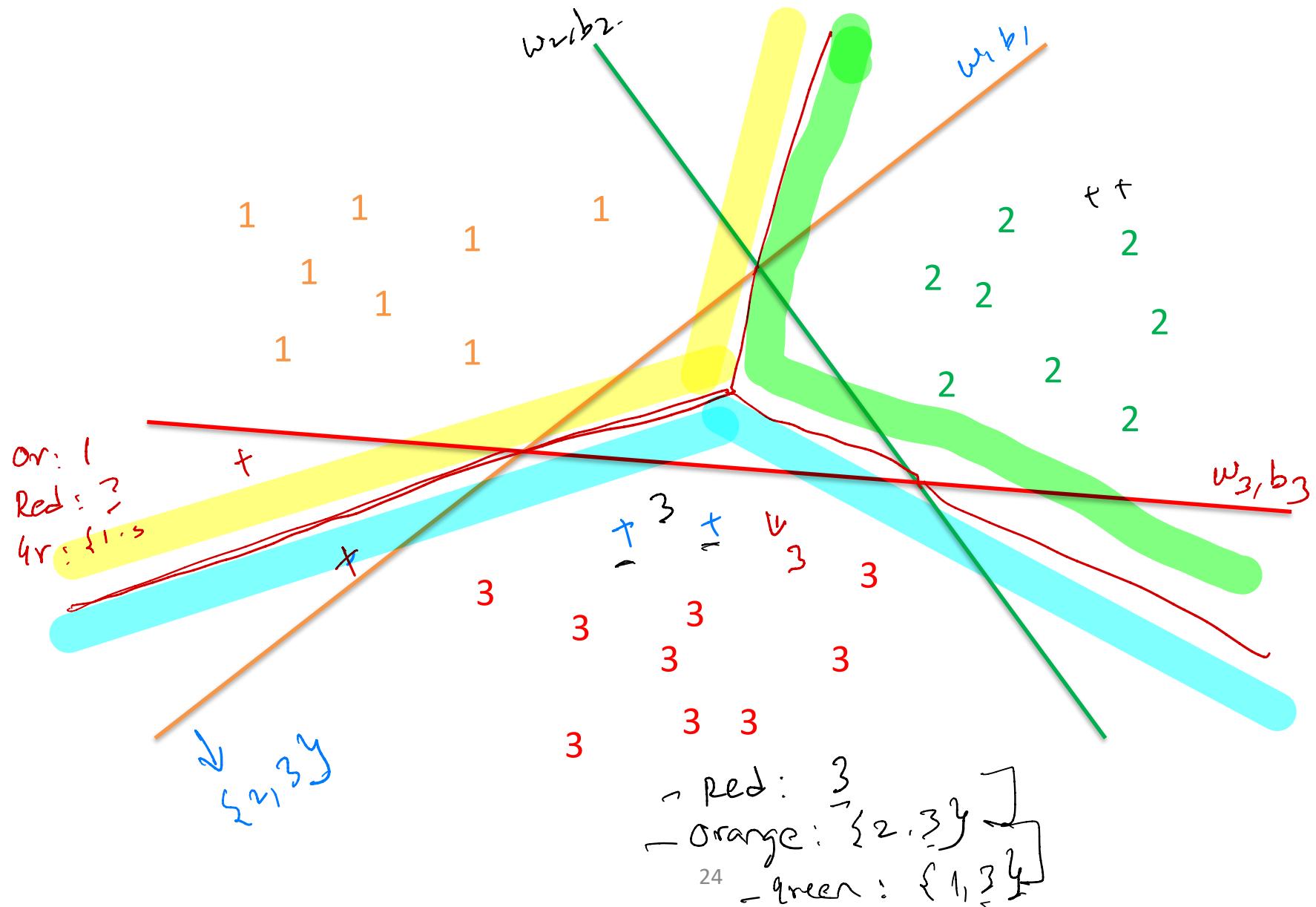
- What if we want to do more than just binary classification (i.e., if  $y \in \{1,2,3\}$ )?



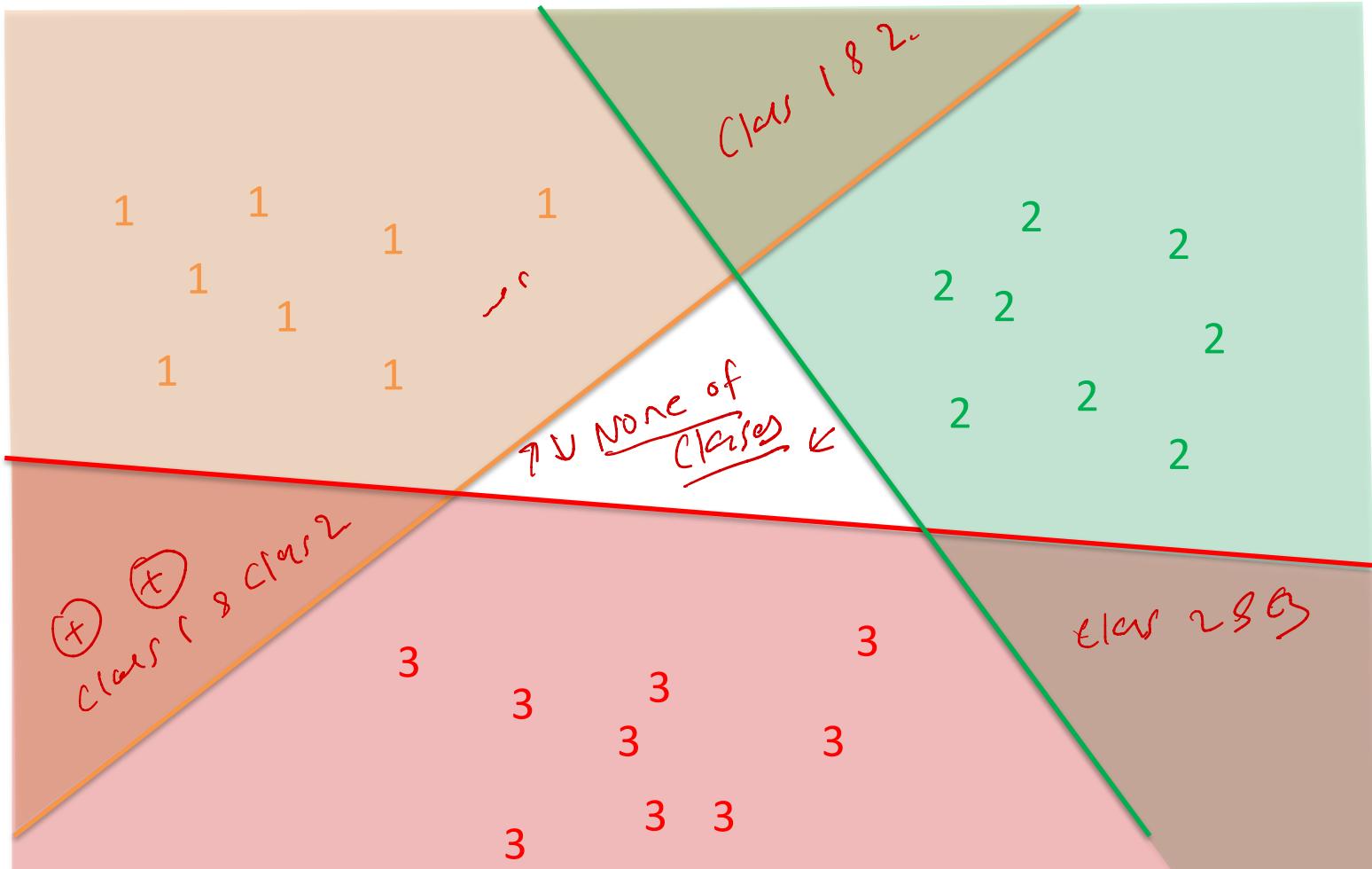
# Multiclass Classification



# One-Versus-All SVMs



# One-Versus-All SVMs



Regions correctly classified by exactly one classifier

# One-Versus-All SVMs

- Compute a classifier for each label versus the remaining labels (i.e., an SVM with the selected label as plus and the remaining labels changed to minuses)

- Let  $f^k(x) = w^{(k)^T}x + b^{(k)}$  be the classifier for the  $k^{th}$  label

- For a new datapoint  $x$ , classify it as

$$k' \in \operatorname{argmax}_k f^k(x)$$

$n:$

$$\begin{aligned} f^1 &= w_1 x + b_1 \\ f^2 &= w_2 x + b_2 \\ f^3 &= w_3 x + b_3 \end{aligned}$$

- Drawbacks:

- If there are  $L$  possible labels, requires learning  $L$  classifiers over the entire data set

$$L = 10^0$$

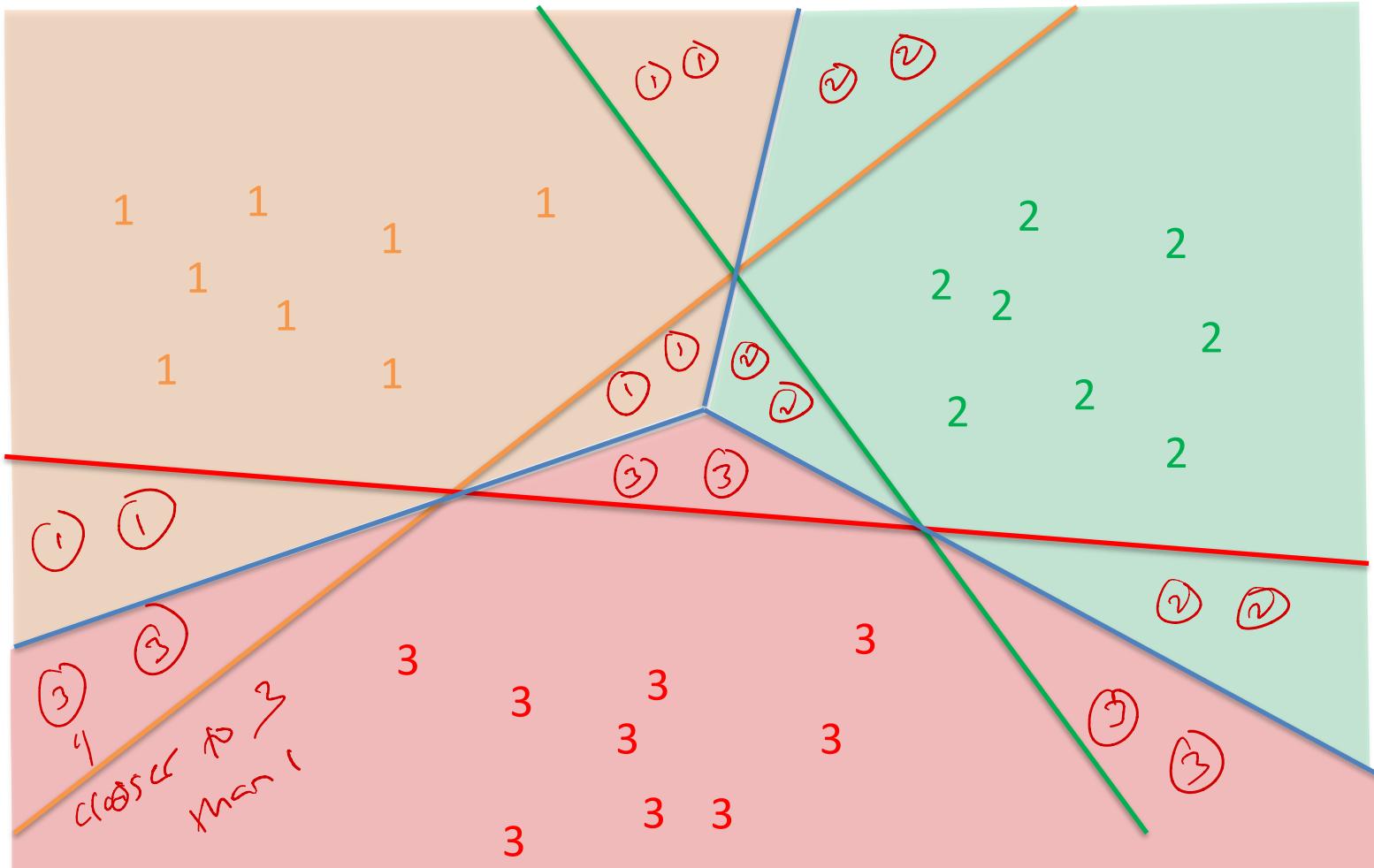
$$\text{cost} \approx L^{\frac{N \times n}{\text{feat}}}$$

26

Extreme Classif<sup>0</sup>

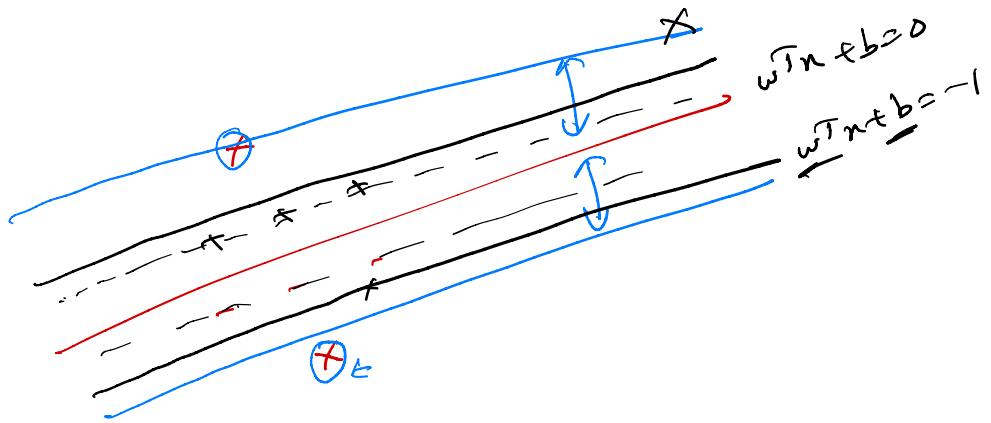
$$L = 10^6$$

# One-Versus-All SVMs



Regions in which points are classified by highest value of  $w^T x + b$

$$\arg \max_k w^{(k)T} x + b^{(k)}$$



# One-Versus-One SVMs (Class )



- Alternative strategy is to construct a classifier for all possible pairs of labels
- Given a new data point, can classify it by majority vote (i.e., find the most common label among all of the possible classifiers)
- If there are  $L$  labels, requires computing  $\binom{L}{2}$  different classifiers each of which uses only a fraction of the data
- Drawbacks: Can overfit if some pairs of labels do not have a significant amount of data (plus it can be computationally expensive)

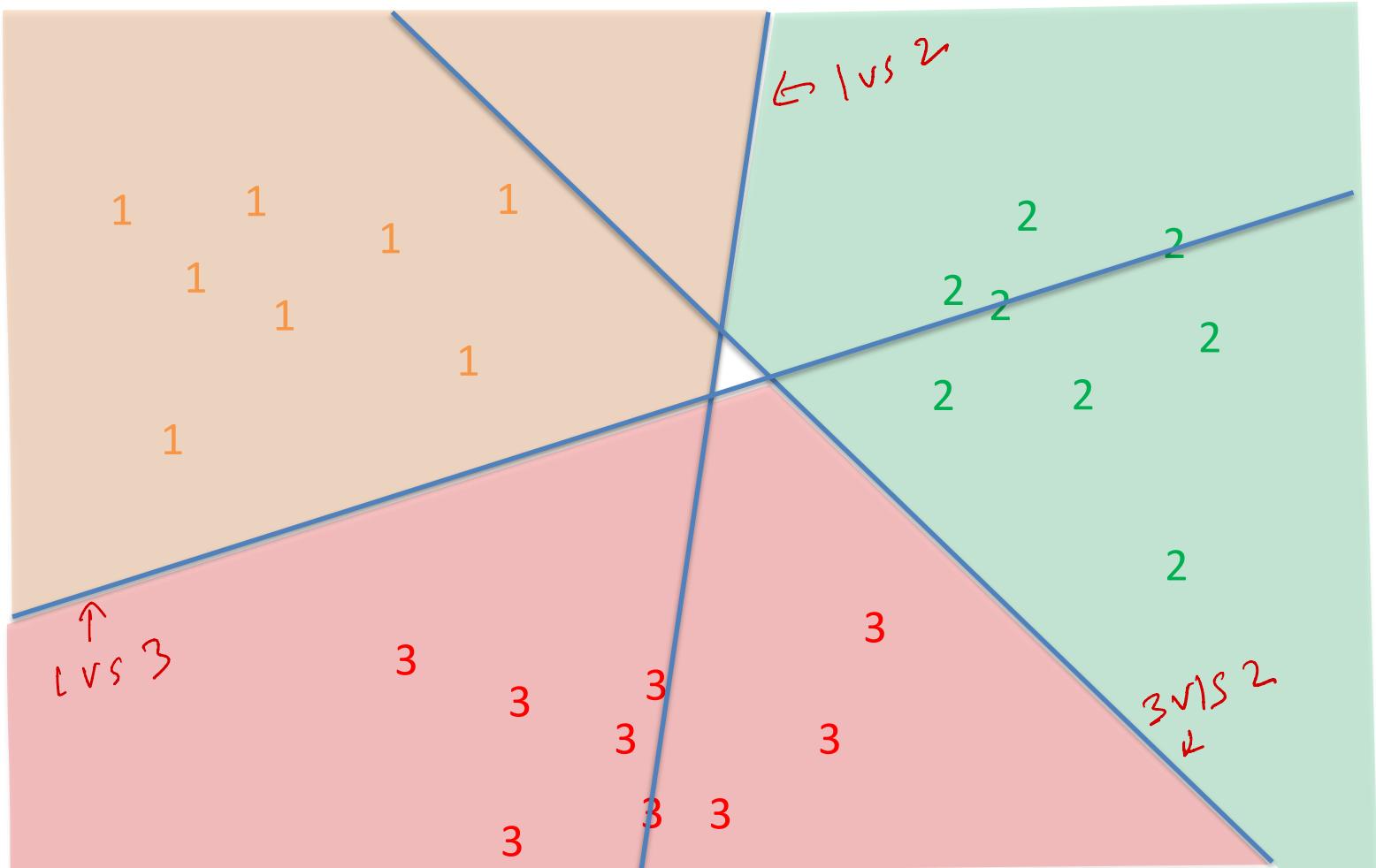
$$\frac{2^M}{L}$$

$$\binom{L}{2} \approx \frac{L(L-1)}{2}$$

Diagram illustrating the number of classifiers required:

The diagram shows a list of pairs of labels:  $(1, 2), (1, 3), (1, 4), \dots, (1, k)$  and  $(2, 3), (2, 4), \dots, (2, k)$ . These pairs are grouped together with a bracket, indicating they are part of a larger set. To the right of the bracket, the formula  $\binom{k}{2}$  is written, representing the total number of such pairs.

# One-Versus-One SVMs



Regions determined by majority vote over the classifiers