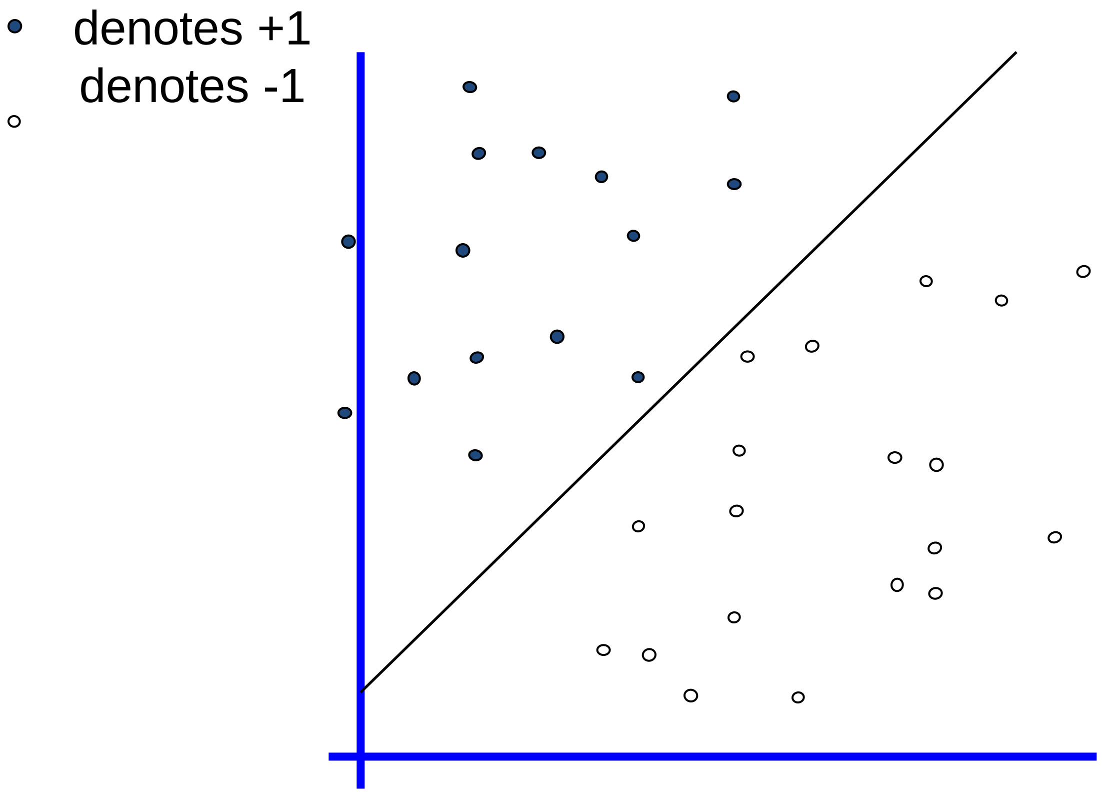
Support Vector Machines

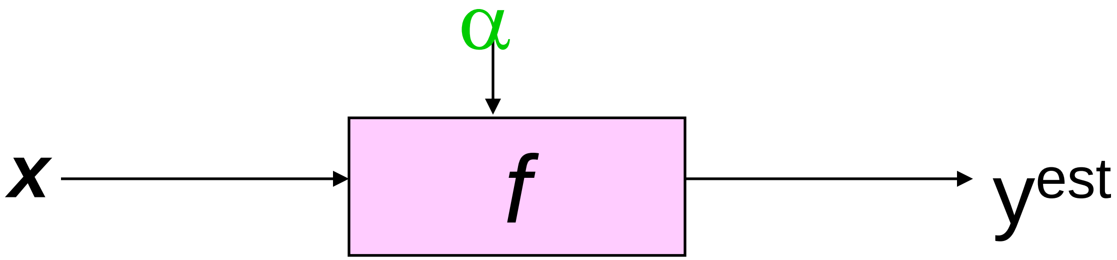
Dr. Rahul Kottath

History of SVM

- SVM is related to statistical learning theory
- SVM was first introduced in 1992
- SVM becomes popular because of its success in handwritten digit recognition
  - 1.1% test error rate for SVM. This is the same as the error rates of a carefully constructed neural network, LeNet 4.
- SVM is now regarded as an important example of “kernel methods”, one of the key area in machine learning

Linear Classifiers



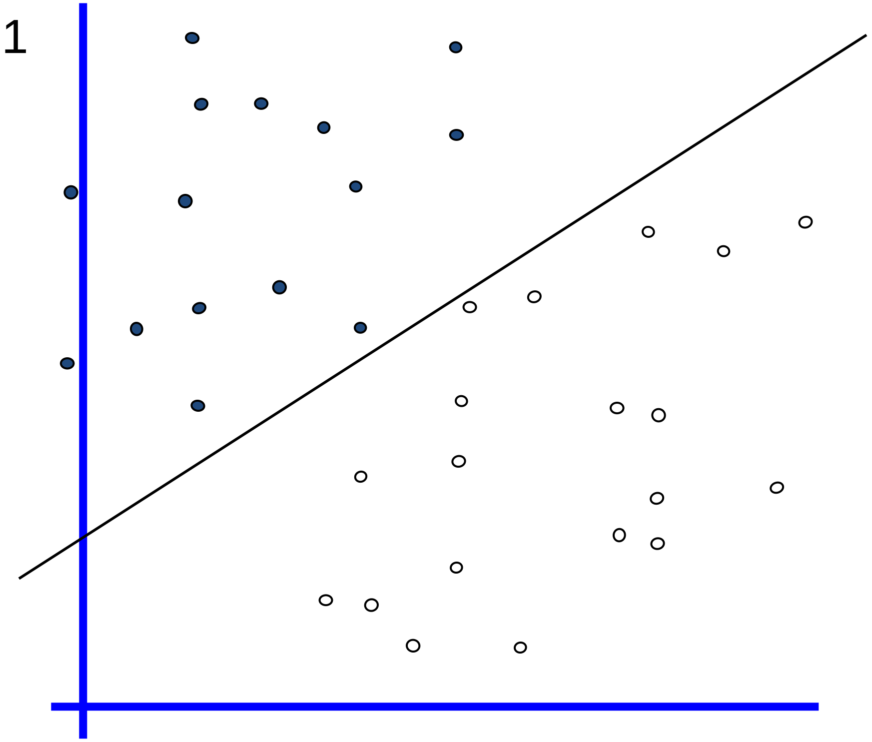


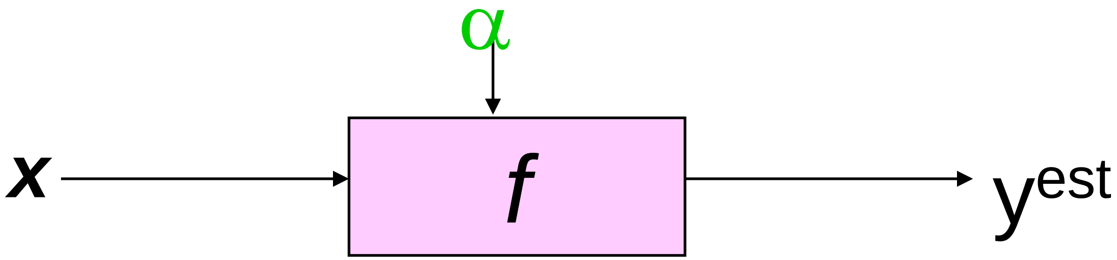
$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{w},b) = sign(\boldsymbol{w}.\ \boldsymbol{x} - b)$

How would you classify this data?

Linear Classifiers

- denotes +1
- denotes -1

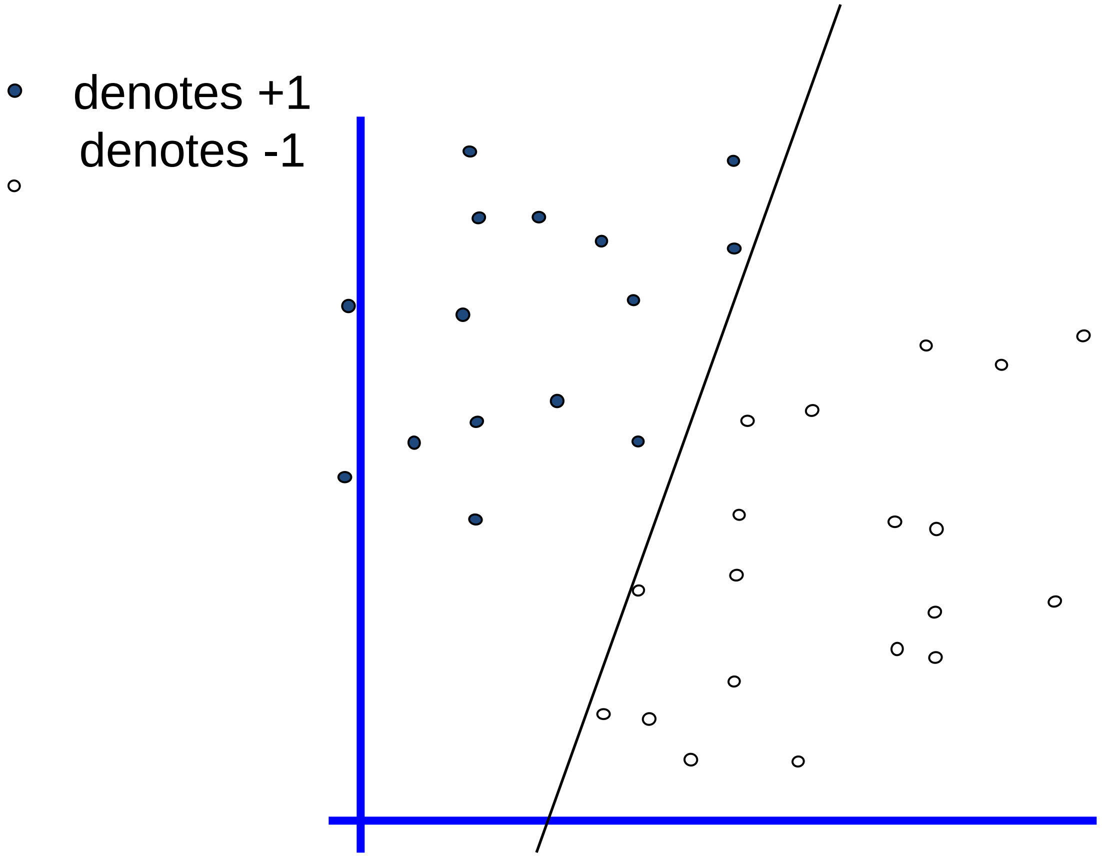


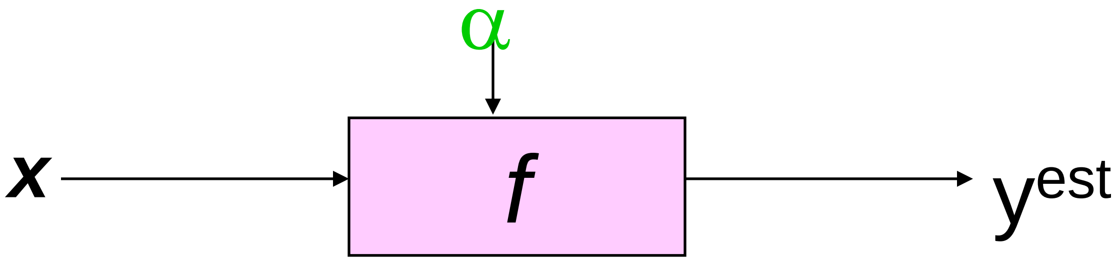


$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$

How would you classify this data?

Linear Classifiers



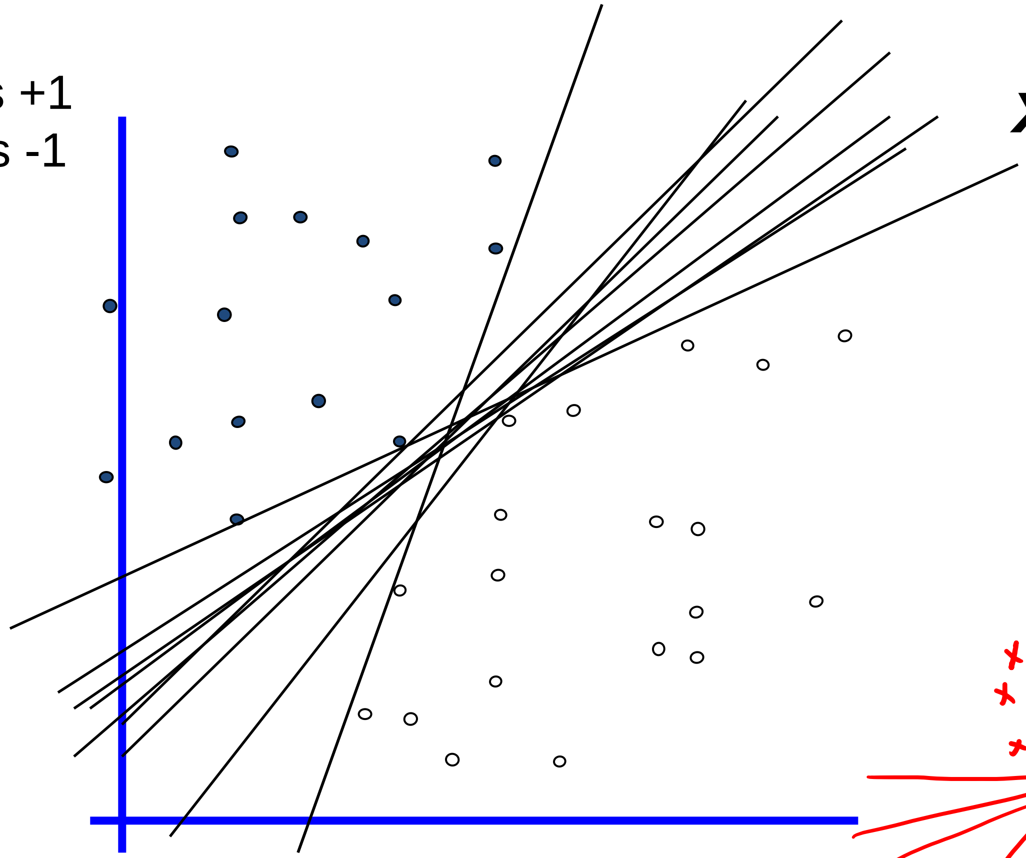


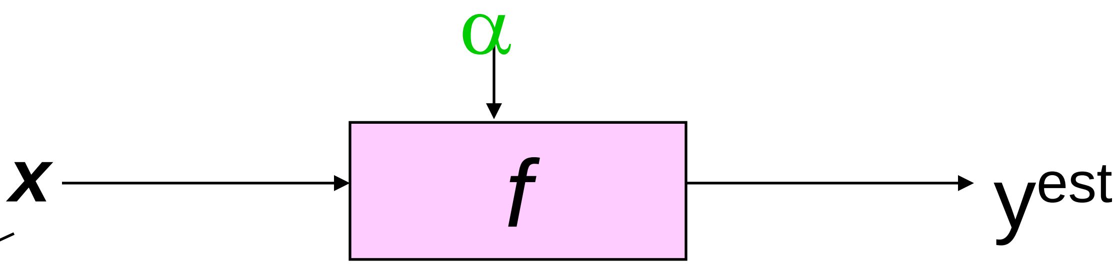
$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}\cdot\mathbf{x} - b)$

How would you classify this data?

Linear Classifiers

- denotes +1
- denotes -1

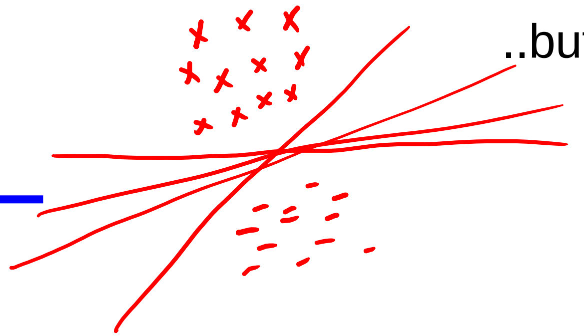




$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$

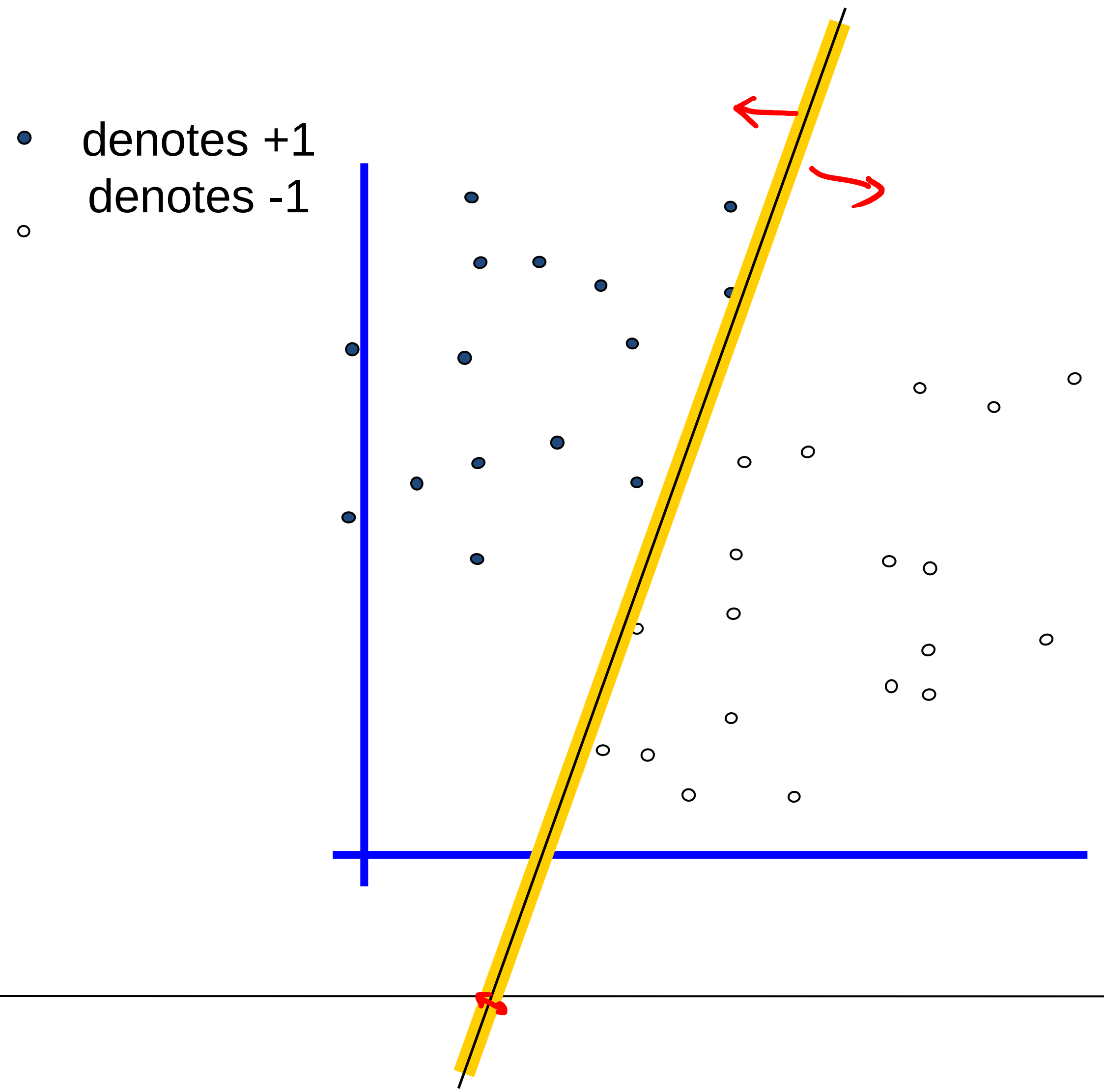
Any of these would be fine..

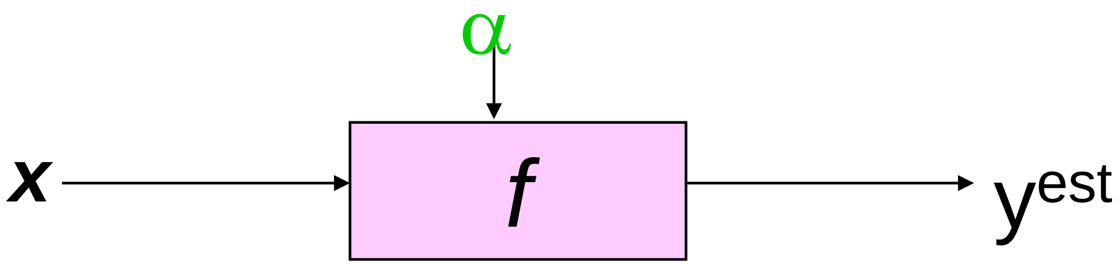
..but which is best?



Classifier Margin

$y = \underbrace{m}_{\text{slope}}x + \underbrace{c}_{\text{intercept}}$





$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w} \cdot \mathbf{x} - b)$

Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

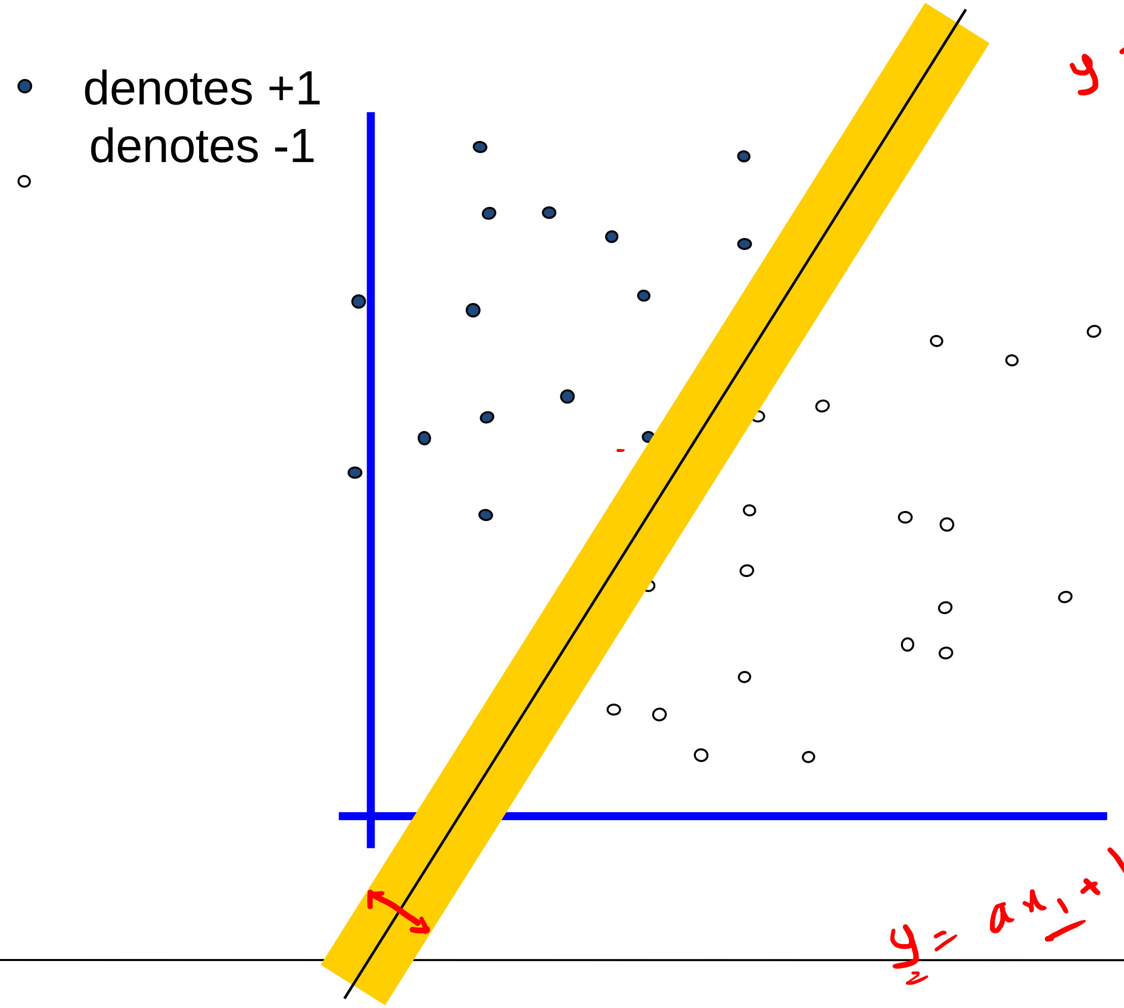
Classifier Margin

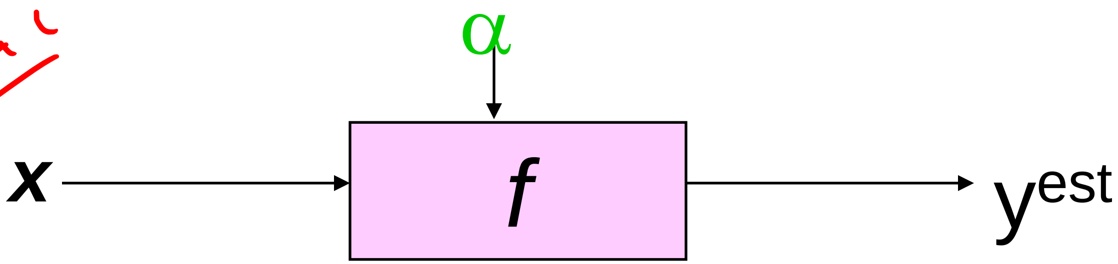
linear classified

$y = a_1x_1 + a_2x_2 + a_3x_3 + d$

$ax^2 + bx + c = 0$

$y = mx + c$





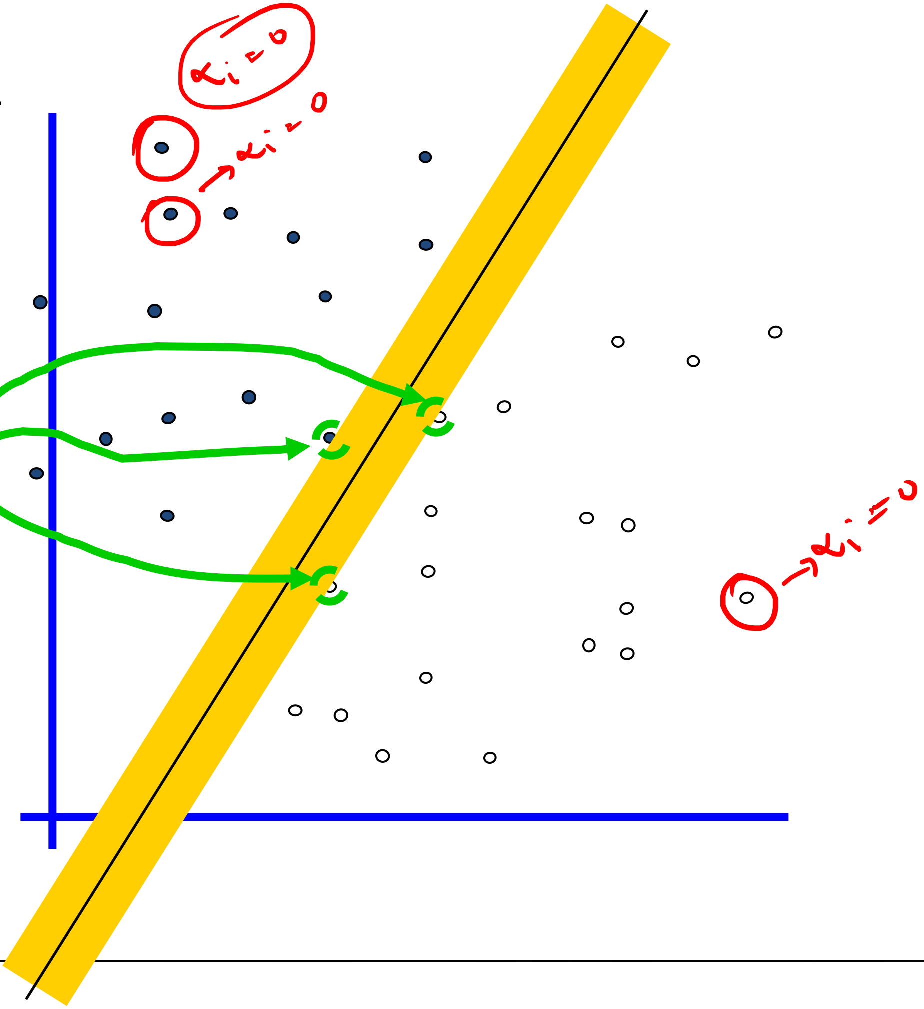
$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$

The maximum margin linear classifier is the linear classifier with the maximum margin. This is the simplest kind of SVM (Called an LSVM)

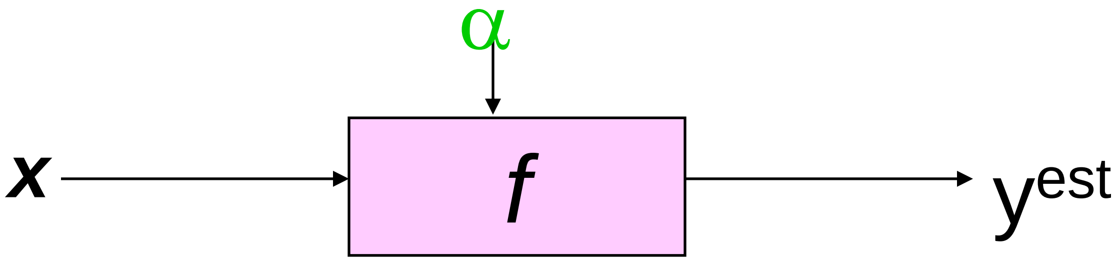
$y = ax_1 + bx_2 + c$

Classifier Margin

• denotes +1
° denotes -1



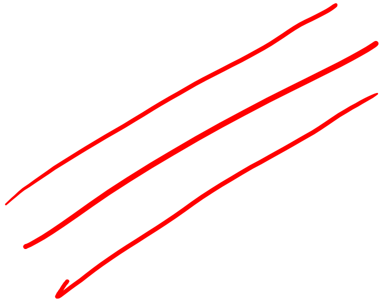
Support Vectors are those datapoints that the margin pushes up against

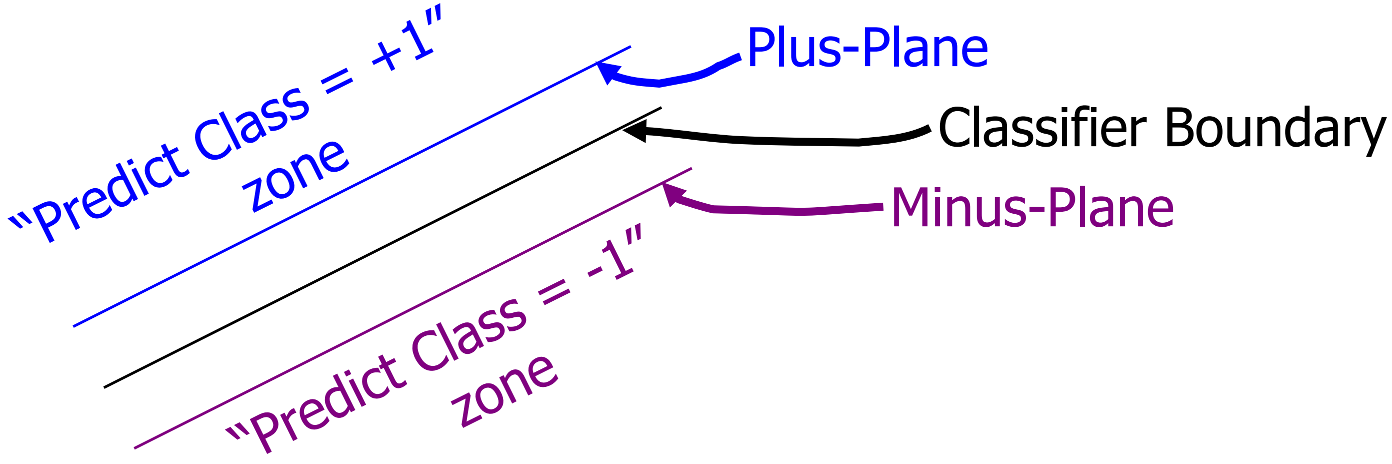


$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$

The maximum margin linear classifier is the linear classifier with the maximum margin. This is the simplest kind of SVM (Called an LSVM)

Specifying a line and margin





How do we represent this mathematically?

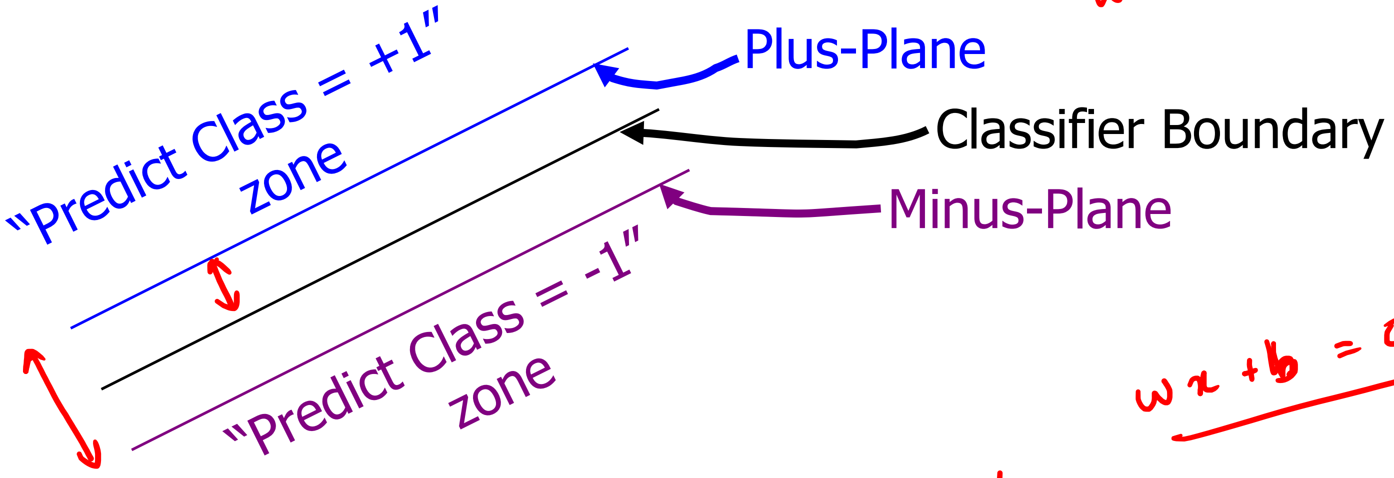
…in *m* input dimensions?

Specifying a line and margin

max → w

$wx+b > 0$ $+1$

$wx+b < 0$ -1



$wx + b = 0$

$y = mx + b$

$wx+b = +1$

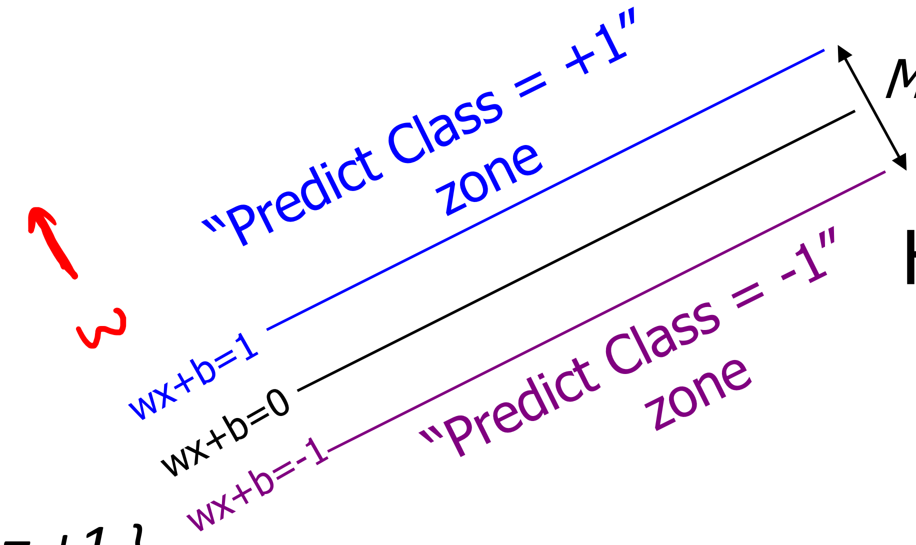
$wx+b = -1$

Plus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} . \boldsymbol{x} + b = +1 \}$

Minus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} . \boldsymbol{x} + b = -1 \}$

Classify as..	+1	if	$\boldsymbol{w} . \boldsymbol{x} + b >= 1$
	-1	if	$\boldsymbol{w} . \boldsymbol{x} + b <= -1$
	Universe explodes	if	$-1 < \boldsymbol{w} . \boldsymbol{x} + b < 1$

Computing the margin width



M = Margin Width

How do we compute *M* in terms of ***w*** and *b*?

Plus-plane = { $\mathbf{x}$: $\mathbf{w}$. $\mathbf{x}$ + b = +1 }

Minus-plane = { $\mathbf{x}$: $\mathbf{w}$. $\mathbf{x}$ + b = -1 }

Claim: The vector **w** is perpendicular to the Plus Plane. Why?

Computing the margin width

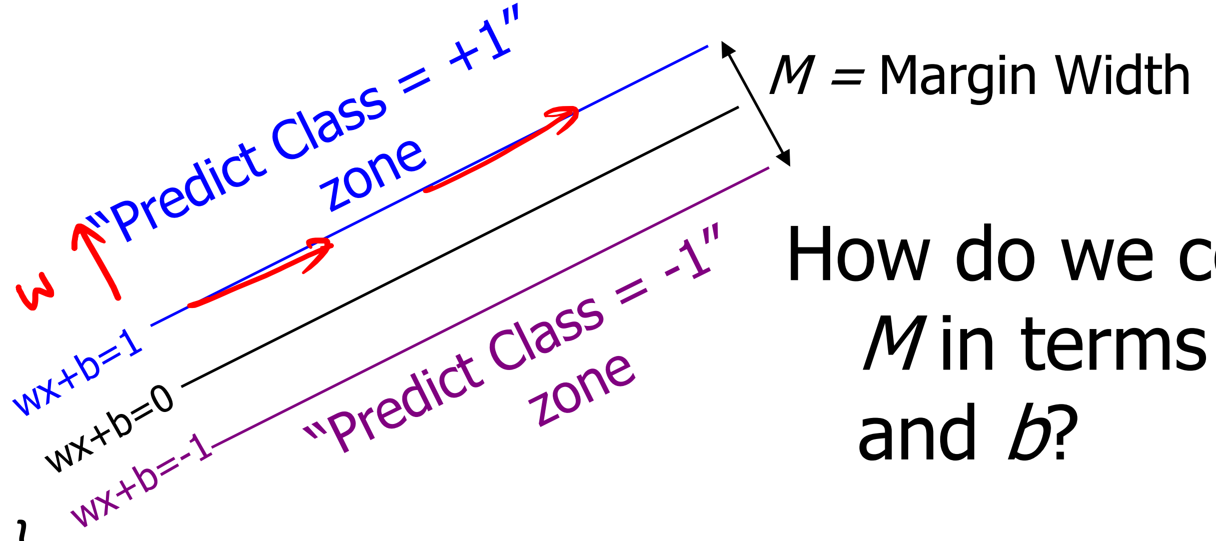
$\vec{a} \cdot \vec{b} = 0$

$a^T b = 0$

$\hat{a}\, b = 0$

$w^T x + b$

$w(u-v) = 0$



How do we compute M in terms of $\boldsymbol{w}$ and b?

Plus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \cdot \boldsymbol{x} + b = +1 \}$

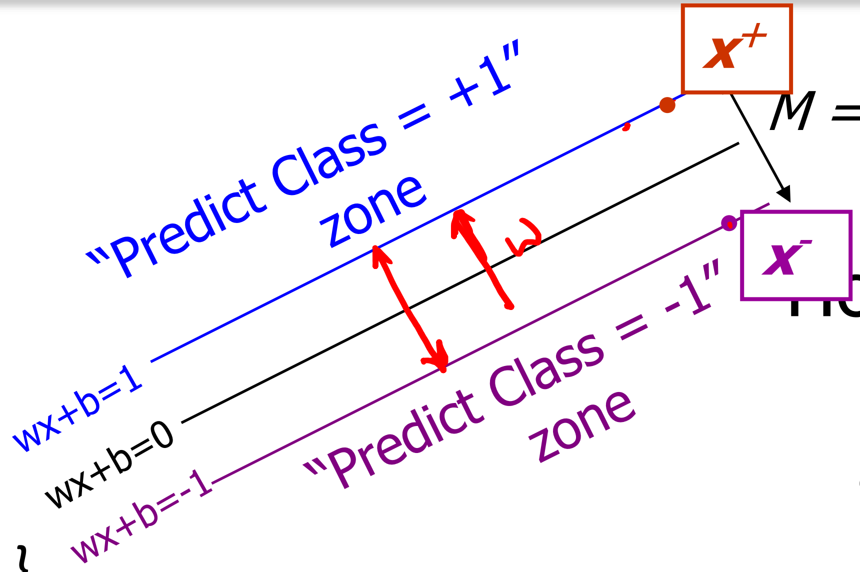
Minus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \cdot \boldsymbol{x} + b = -1 \}$

Claim: The vector **w** is perpendicular to the Plus Plane. Why?

And so of course the vector **w** is also perpendicular to the Minus Plane

Let **u** and **v** be two vectors on the Plus Plane. What is $\boldsymbol{w} \cdot (\boldsymbol{u} - \boldsymbol{v})$?

Computing the margin width

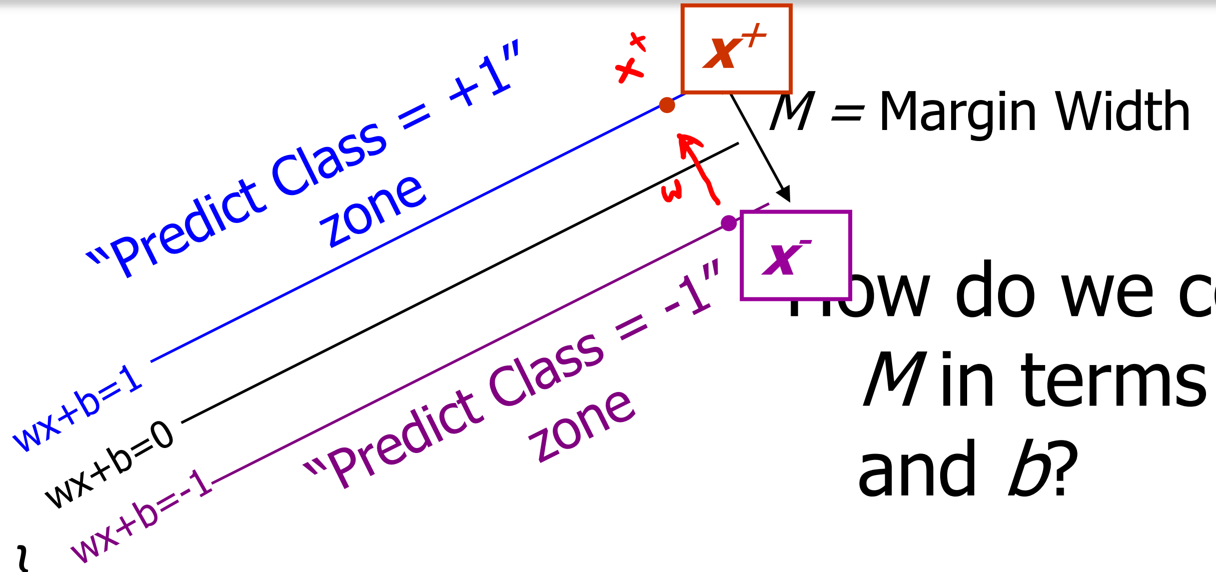


M = Margin Width

How do we compute M in terms of $\boldsymbol{w}$ and b?

- Plus-plane = $\{ \mathbf{x} : \mathbf{w} \cdot \mathbf{x} + b = +1 \}$
- Minus-plane = $\{ \mathbf{x} : \mathbf{w} \cdot \mathbf{x} + b = -1 \}$
- The vector **w** is perpendicular to the Plus Plane
- Let $\boldsymbol{x}^-$ be any point on the minus plane
- Let $\boldsymbol{x}^+$ be the closest plus-plane-point to $\boldsymbol{x}^-$.

Computing the margin width



How do we compute M in terms of $\boldsymbol{w}$ and b?

Plus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + b = +1 \}$

Minus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + b = -1 \}$

The vector **w** is perpendicular to the Plus Plane

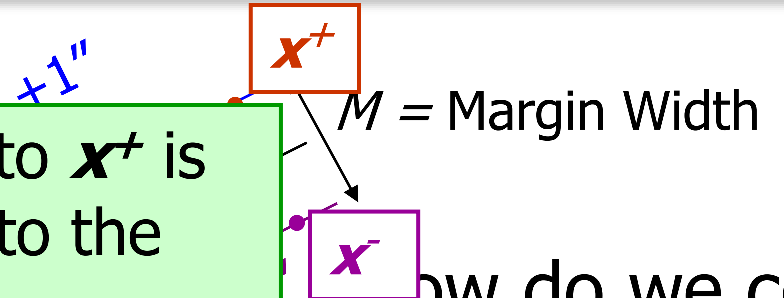
Let $\boldsymbol{x}^-$ be any point on the minus plane

Let $\boldsymbol{x}^+$ be the closest plus-plane-point to $\boldsymbol{x}^-$.

$x^+ = x^- + \underline{\lambda} w$

Claim: $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda \boldsymbol{w}$ for some value of λ. Why?

Computing the margin width



How do we compute M in terms of $\boldsymbol{w}$ and b?

The line from $\boldsymbol{x}^-$ to $\boldsymbol{x}^+$ is perpendicular to the planes.

So to get from $\boldsymbol{x}^-$ to $\boldsymbol{x}^+$ travel some distance in direction $\boldsymbol{w}$.

Plus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + b =$

Minus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + b = -1 \}$

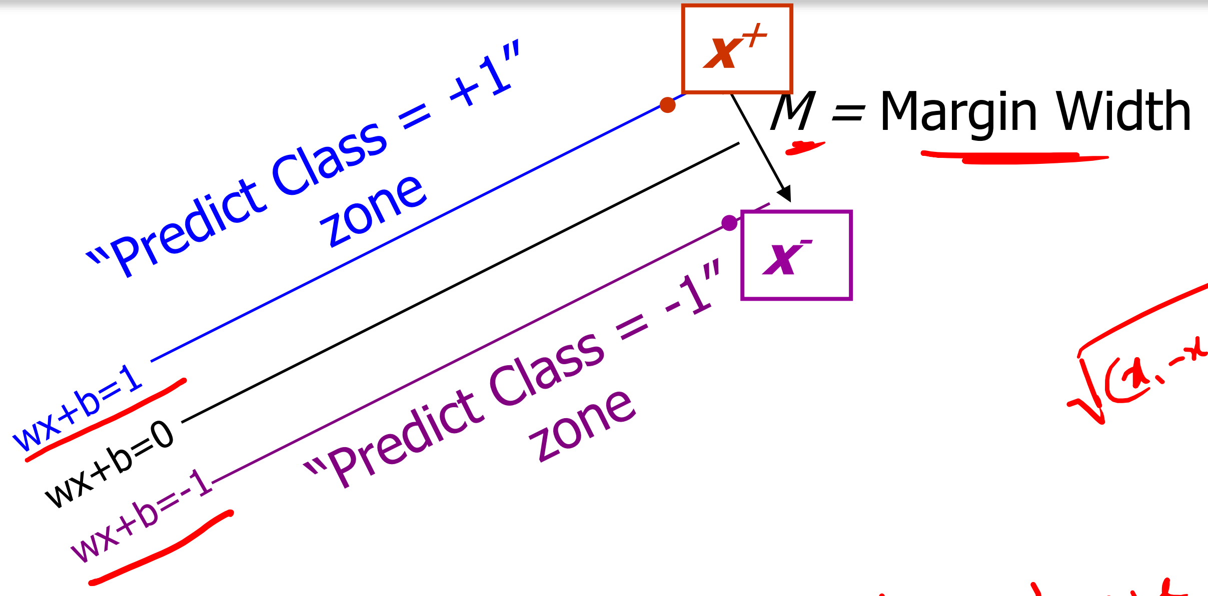
The vector **w** is perpendicular to the Plus Plane

Let $\boldsymbol{x}^-$ be any point on the minus plane

Let $\boldsymbol{x}^+$ be the closest plus-plane-point to $\boldsymbol{x}^-$.

Claim: $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda \boldsymbol{w}$ for some value of λ. Why?

Computing the margin width



$\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$

$|x|$ distance

What we know:

$\boldsymbol{w} \cdot \boldsymbol{x}^+ + b = +1$

$\boldsymbol{w} \cdot \boldsymbol{x}^- + b = -1$

$\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda \boldsymbol{w}$

$|\boldsymbol{x}^+ - \boldsymbol{x}^-| = M$

It's now easy to get M in terms of $\boldsymbol{w}$ and b

Computing the margin width

What we know:

- $\mathbf{w} \cdot \mathbf{x}^+ + b = +1$
- $\mathbf{w} \cdot \mathbf{x}^- + b = -1$
- $\mathbf{x}^+ = \mathbf{x}^- + \lambda \mathbf{w}$
- $|\mathbf{x}^+ - \mathbf{x}^-| = M$

It's now easy to get M in terms of $\mathbf{w}$ and b

$$\mathbf{w} \cdot (\mathbf{x}^- + \lambda \mathbf{w}) + b = 1$$

=>

$$\mathbf{w} \cdot \mathbf{x}^- + b + \lambda \mathbf{w} \cdot \mathbf{w} = 1$$

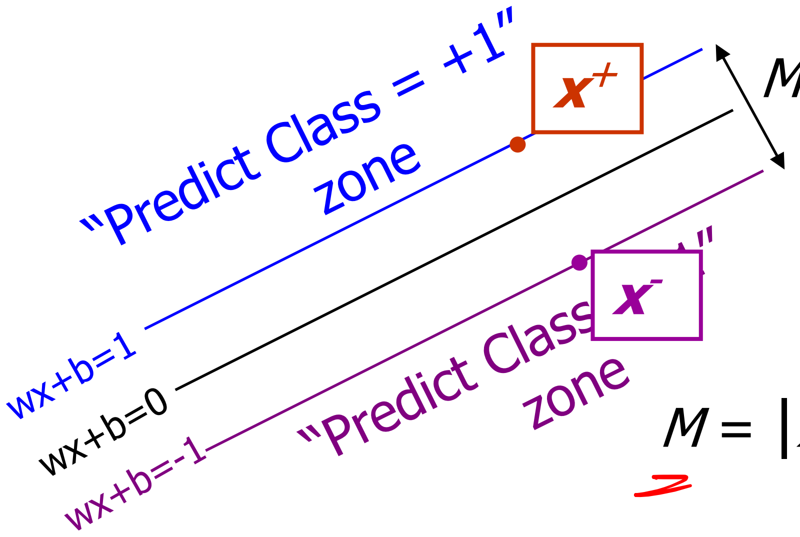
=>

$$-1 + \lambda \mathbf{w} \cdot \mathbf{w} = 1$$

=>

$$\lambda = \frac{2}{\mathbf{w}.\mathbf{w}}$$

Computing the margin width



M = Margin Width = $\dfrac{2}{\sqrt{\mathbf{w}.\mathbf{w}}}$

$a^2 + b^2$

$ai + bj$ $= (ai + bj).(ai$

What we know:

$\mathbf{w} \cdot \mathbf{x}^+ + b = +1$

$\mathbf{w} \cdot \mathbf{x}^- + b = -1$

$\mathbf{x}^+ = \mathbf{x}^- + \lambda \mathbf{w}$

$|\mathbf{x}^+ - \mathbf{x}^-| = M$

$$\lambda = \frac{2}{\mathbf{w}.\mathbf{w}}$$

$$M = |\mathbf{x}^+ - \mathbf{x}^-| = |\lambda \mathbf{w}| =$$

$$= \lambda |\mathbf{w}| = \lambda \sqrt{\mathbf{w}.\mathbf{w}}$$

$$= \frac{2\sqrt{\mathbf{w}.\mathbf{w}}}{\mathbf{w}.\mathbf{w}} = \frac{2}{\sqrt{\mathbf{w}.\mathbf{w}}}$$

Finding the Decision Boundary

The decision boundary can be found by solving the following constrained optimization problem

$$\text{Minimize } \frac{1}{2}||\mathbf{w}||^2$$

$$\text{subject to } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \qquad \forall i$$

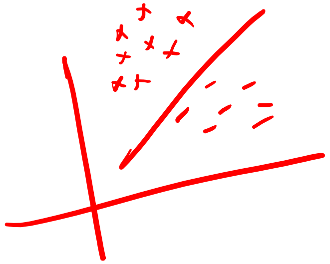
$+1$: $w^T x + b \geq +1$

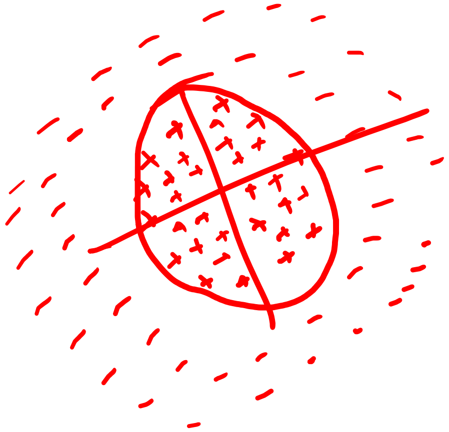
-1: $w^T x + b \leq -1$

SVM

Next step... Optional

- Converting SVM to a form we can solve
  - Dual form
- Allowing a few errors
  - Soft margin
- Allowing nonlinear boundary
  - Kernel functions





$y^2 = ax^2 + bx + c$

$ay^2 + bx^2 = r^2$

$x^2 + y^2 = r^2$

$= r^2 \rightarrow -ve$

$> r^2$

$< r^2 \rightarrow +ve$

The Dual Problem (we ignore the derivation)

The new objective function is in terms of α_i only

It is known as the dual problem: if we know **w**, we know all α_i; if we know all α_i, we know **w**

The original problem is known as the primal problem

The objective function of the dual problem needs to be maximized!

The dual problem is therefore:

f(x)

$$\max.\ W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

constraints

$$\text{subject to } \alpha_i \geq 0, \qquad \sum_{i=1}^{n} \alpha_i y_i = 0$$

Properties of α_i when we introduce the Lagrange multipliers

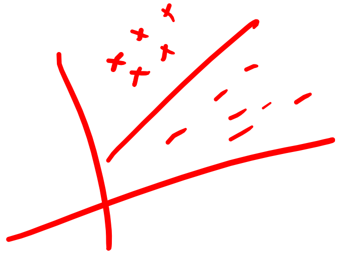
The result when we differentiate the original Lagrangian w.r.t. b

The Dual Problem

$$\max.\ \ W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\text{subject to } \alpha_i \geq 0, \sum_{i=1}^{n} \alpha_i y_i = 0$$

Optimization



(x_i, y_i)

This is a quadratic programming (QP) problem

A global maximum of α_i can always be found

w can be recovered by $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$

Characteristics of the Solution

Many of the α_i are zero

w is a linear combination of a small number of data points

This "sparse" representation can be viewed as data compression as in the construction of knn classifier

$\mathbf{x}_i$ with non-zero α_i are called support vectors (SV)

The decision boundary is determined only by the SV

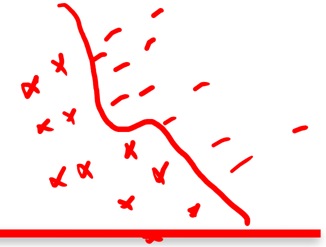
Let t_j (j=1, ..., s) be the indices of the s support vectors. We can write $\mathbf{w} = \sum_{j=1}^{s} \alpha_{t_j} y_{t_j} \mathbf{x}_{t_j}$

For testing with a new data **z**

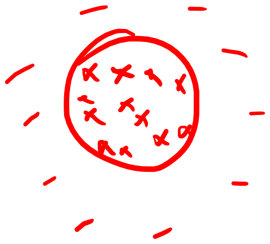
Compute $\mathbf{w}^T\mathbf{z} + b = \sum_{j=1}^{s} \alpha_{t_j} y_{t_j} (\mathbf{x}_{t_j}^T \mathbf{z}) + b$ and classify **z** as class 1 if the sum is positive, and class 2 otherwise

Note: **w** need not be formed explicitly

Extension to Non-linear Decision Boundary



- So far, we have only considered large-margin classifier with a linear decision boundary
- How to generalize it to become nonlinear?
- Key idea: transform $\mathbf{x}_i$ to a higher dimensional space to "make life easier"
  - Input space: the space the point $\mathbf{x}_i$ are located
  - Feature space: the space of $\phi(\mathbf{x}_i)$ after transformation

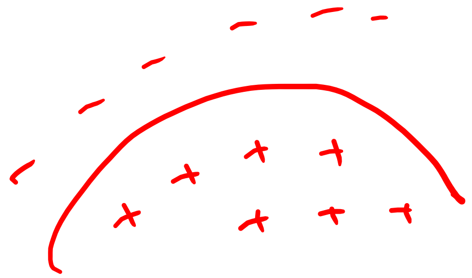


The Kernel Trick

- Recall the SVM optimization problem

$$\text{max.}\ W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\text{subject to } C \geq \alpha_i \geq 0, \sum_{i=1}^{n} \alpha_i y_i = 0$$



- The data points only appear as inner product
- As long as we can calculate the inner product in the feature space, we do not need the mapping explicitly
- Many common geometric operations (angles, distances) can be expressed by inner products
- Define the kernel function K by $\quad K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$

Examples of Kernel Functions

- Polynomial kernel with degree *d*

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^d$$

- Radial basis function kernel with width σ (Rbf)

$$K(\mathbf{x}, \mathbf{y}) = \exp(-||\mathbf{x} - \mathbf{y}||^2 / (2\sigma^2))$$

  - Closely related to radial basis function neural networks
  - The feature space is infinite-dimensional

- Sigmoid with parameter κ and θ

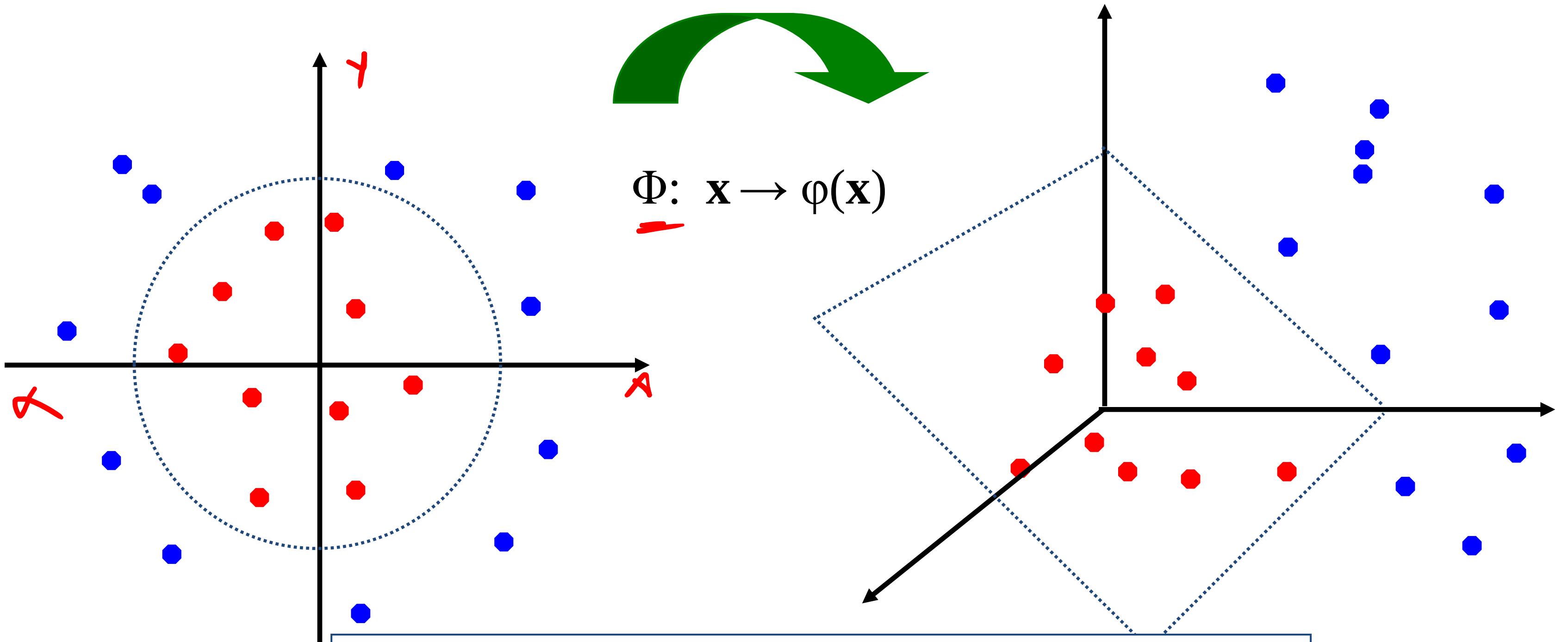
$$K(\mathbf{x}, \mathbf{y}) = \tanh(\kappa \mathbf{x}^T \mathbf{y} + \theta)$$

Non-linear SVMs: Feature spaces

General idea: the original input space can always be mapped to some higher-dimensional feature space where the training set is separable:

2D

SVM α



sample

$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$

$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$

Conclusion

- Choosing the Kernel Function
  - Probably the trickiest part of using SVM.
- SVM is a useful alternative to neural networks
- Two key concepts of SVM: maximize the margin and the kernel trick
- Many SVM implementations are available on the web for you to try on your data set!

Python Packages needed

- pandas
  - Data Analytics
- numpy
  - Numerical Computing
- matplotlib.pyplot
  - Plotting graphs
- sklearn
  - Classification and Regression Classes

Implementation Using sklearn

Let's go to Jupyter Notebook!