UVA CS 4774: Machine Learning

S4: Lecture 20: Support Vector Machine (Basics)

Module I

Dr. Yanjun Qi

University of Virginia

Department of Computer Science

Three major sections for classification

- We can divide the large variety of classification approaches into roughly three major types

1. Discriminative
   directly estimate a decision rule/boundary
   e.g., support vector machine, decision tree, ~~logistic regression,~~
   e.g~~. neural networks (NN), deep NN~~

2. Generative:
   build a generative statistical model
   e.g., ~~Bayesian networks, Naïve Bayes classifier~~

3~~. Instance based classifiers~~
   - Use observation directly (no models)
   ~~- e.g. K nearest neighbors~~

Today: Basic Support Vector Machine

Data

↓

Task

↓

Representation

↓

Score Function

↓

Search/Optimization

↓

Models, Parameters

Tabular

↓

Classification

↓

Kernel Trick Func K(xi, xj)

$$K(\mathbf{x}, z) := \Phi(\mathbf{x})^T \Phi(z)$$

↓

Margin + Hinge Loss

↓

QP with Dual form

$$w = \sum_i \alpha_i x_i y_i$$

↓

Dual Weights

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{p} w_i^2 + C\sum_{i=1}^{n} \varepsilon_i$$

$$\text{subject to } \forall \mathbf{x}_i \in Dtrain : y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1 - \varepsilon_i$$

$$\max_\alpha \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^T \mathbf{x_j}$$

$$\sum_i \alpha_i y_i = 0, \qquad \alpha_i \geq 0 \qquad \forall i$$

Today

- ❑ Support Vector Machine (SVM)
  - ✓ History of SVM
  - ✓ Large Margin Linear Classifier
  - ✓ Define Margin (M) in terms of model parameter
  - ✓ Optimization to learn model parameters (w, b)
  - ✓ Linearly Non-separable case
  - ✓ Optimization with dual form
  - ✓ Nonlinear decision boundary
  - ✓ Multiclass SVM

History of SVM

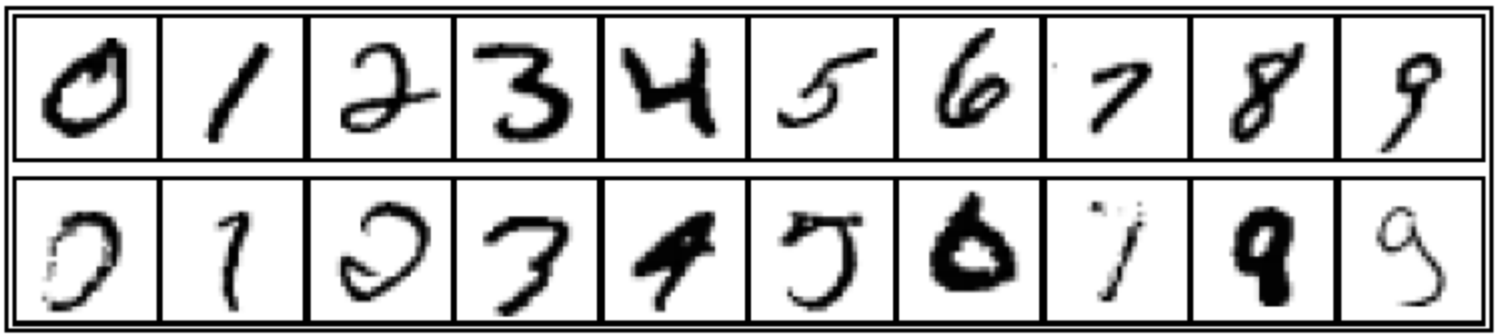
Theoretically sound / Impactful

- SVM is inspired from statistical learning theory [3]
- SVM was first introduced in 1992 [1]
- SVM becomes popular because of its success in handwritten digit recognition (1994)
  - 1.1% test error rate for SVM.
  - The same as the error rates of a carefully constructed neural network, LeNet 4.
    - Section 5.11 in [2] or the discussion in [3] for details
- Regarded as an important example of "kernel methods", arguably the hottest area in machine learning 20 years ago

[1] B.E. Boser et al. A Training Algorithm for Optimal Margin Classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 5 144-152, Pittsburgh, 1992.
[2] L. Bottou et al. Comparison of classifier methods: a case study in handwritten digit recognition. Proceedings of the 12th IAPR International Conference on Pattern Recognition, vol. 2, pp. 77-82, 1994.
[3] V. Vapnik. The Nature of Statistical Learning Theory. 2nd edition, Springer, 1999.

Handwritten digit recognition

→ MNIST (SVM) 1994



3-nearest-neighbor = 2.4% error
400–300–10 unit MLP = 1.6% error
LeNet: 768–192–30–10 unit MLP = 0.9% error

best (kernel machines, vision algorithms) ≈ 0.6% error

Mixed National Institute of Standards and Technology (MNIST)

- Is a database and evaluation setup for handwritten digit recognition
- Contains 60,000 training and 10,000 test instances of hand-written digits, encoded as 28×28 pixel grayscale images
- The data is a re-mix of an earlier NIST dataset in which adults generated the training data and high school students generated the test set
- Lets compare the performance of different methods

MNIST

Classifier	Test Error Rate (%)	References
Linear classifier (1-layer neural net)	12.0	LeCun et al. (1998)
K-nearest-neighbors, Euclidean (L2)	5.0	LeCun et al. (1998)
2-Layer neural net, 300 hidden units, mean square error	4.7	LeCun et al. (1998)
Support vector machine, Gaussian kernel	1.4	MNIST Website
Convolutional net, LeNet-5 (no distortions)	**0.95**	LeCun et al. (1998)
Methods using distortions		
Virtual support vector machine, deg-9 polynomial, (2-pixel jittered and deskewing)	0.56	DeCoste and Scholkopf (2002)
Convolutional neural net (elastic distortions)	0.4	Simard, Steinkraus, and Platt (2003)
6-Layer feedforward neural net (on GPU) (elastic distortions)	**0.35**	Ciresan, Meier, Gambardella, and Schmidhuber (2010)
Large/deep convolutional neural net (elastic distortions)	**0.35**	Ciresan, Meier, Masci, Maria Gambardella, and Schmidhuber (2011)
Committee of 35 convolutional networks (elastic distortions)	0.23	Ciresan, Meier, and Schmidhuber (2012)

Applications of SVMs

- Computer Vision
- Text Categorization
- Ranking (e.g., Google searches)
- Handwritten Character Recognition
- Time series analysis
- Bioinformatics
- ……….

→Lots of very successful applications!!!

Early History

- In 1950 English mathematician Alan Turing wrote a landmark paper titled "Computing Machinery and Intelligence" that asked the question: "Can machines think?"
- Further work came out of a 1956 workshop at Dartmouth sponsored by John McCarthy. In the proposal for that workshop, he coined the phrase a "study of artificial intelligence"
- 1950s
  - Samuel's checker player : start of machine learning
  - Selfridge's Pandemonium
- **1952-1969: Enthusiasm:** Lots of work on neural networks
- 1970s: Expert systems, Knowledge bases to add on rule-based inference

Adapted From Prof. Raymond J. Mooney's slides

Early History

- 1980s :
  - Advanced decision tree and rule learning
  - Valiant's PAC Learning Theory

- 1990s:
  - Reinforcement learning (RL)
  - Ensembles: Bagging, Boosting, and Stacking
  - Bayes Net learning
  - Convolutional neural network (CNN) and Recurrent neural network (RNN) were invented
  - SVM

- 2000s ~2010s
  - Support vector machines (becoming popular and dominating)
  - Kernel methods
  - Graphical models
  - Statistical relational learning
  - Transfer learning
  - Sequence labeling
  - Collective classification and structured outputs

Adapted From Prof. Raymond J. Mooney's slides

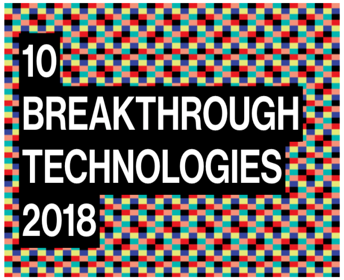
MIT Technology Review

10 Breakthrough Technologies 2013

Think of the most frustrating, intractable, or simply annoying problems you can imagine. Now think about what technology is doing to fix them. That's what we did in coming up with our annual list of 10 Breakthrough Technologies. We're looking for technologies that we believe will expand the scope of human possibilities.

10 Breakthrough Technologies 2017

These technologies all have staying power. They will affect the economy and our politics, improve medicine, or influence our culture. Some are unfolding now; others will take a decade or more to develop. But you should know about all of them right now.



Deep Learning

CNN/RNN
1990s

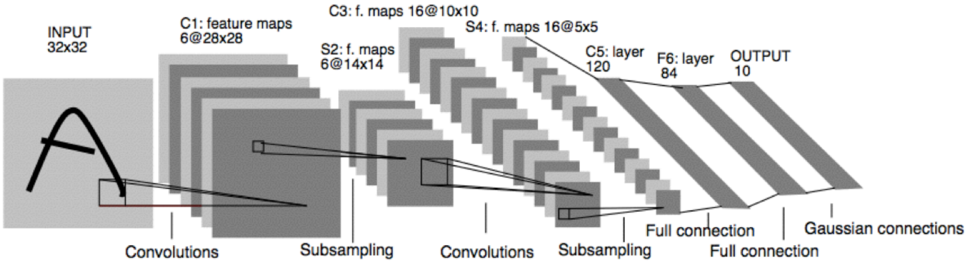
Deep Reinforcement Learning

1990s~2010s

Generative Adversarial Network (GAN)

2010s

- 1952-1969 Enthusiasm: Lots of work on neural networks
- 1990s: Convolutional neural network (CNN) and Recurrent neural network (RNN) were invented



Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86(11): 2278–2324, 1998.

Reason of the recent breakthroughs of deep learning:

① Low Bias

High Variance

Plenty of (Labeled) Data	Advanced Computer Architecture that fits DNNs	Powerful DNN platforms / Libraries

② XgBoost

Tree Based

reduce variance

X_1	X_2	X_3	Y

$$f : \boxed{X} \longrightarrow \boxed{Y}$$

{-1, 1}

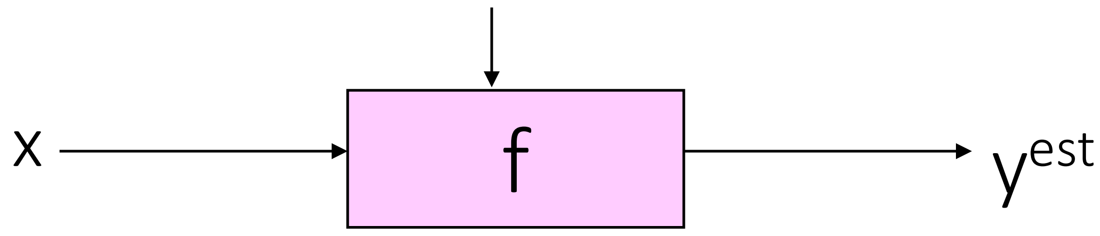
Output as Binary Class: only two possibilities

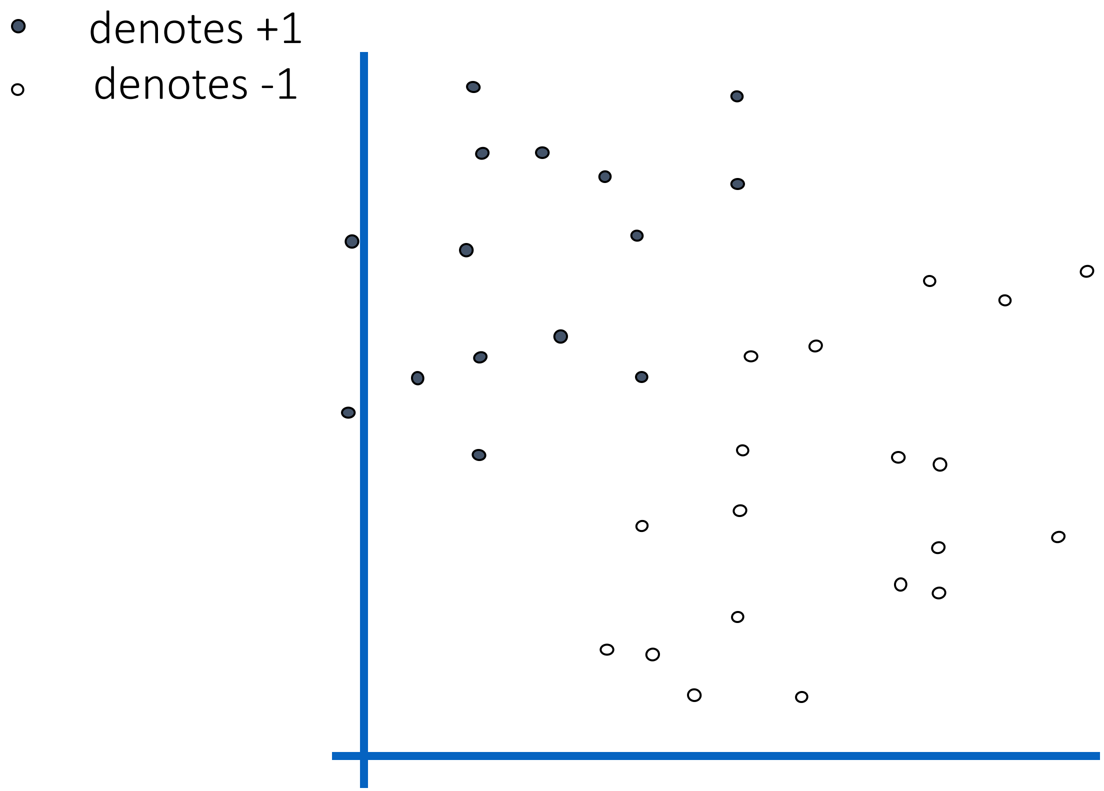
- **Data**/*points/instances/examples/samples/records*: [rows]
- **Features**/*attributes/dimensions/independent variables/covariates/predictors/regressors*: [columns, except the last]
- **Target**/*outcome/response/label/dependent variable*: special column to be predicted [last column]

Today

- ❑ Support Vector Machine (SVM)
  - ✓ History of SVM
  - ✓ Large Margin Linear Classifier
  - ✓ Define Margin (M) in terms of model parameter
  - ✓ Optimization to learn model parameters (w, b)
  - ✓ Linearly Non-separable case
  - ✓ Optimization with dual form
  - ✓ Nonlinear decision boundary
  - ✓ Multiclass SVM

Linear Classifiers

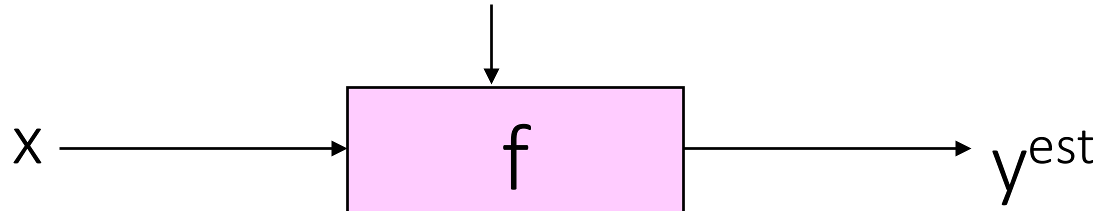


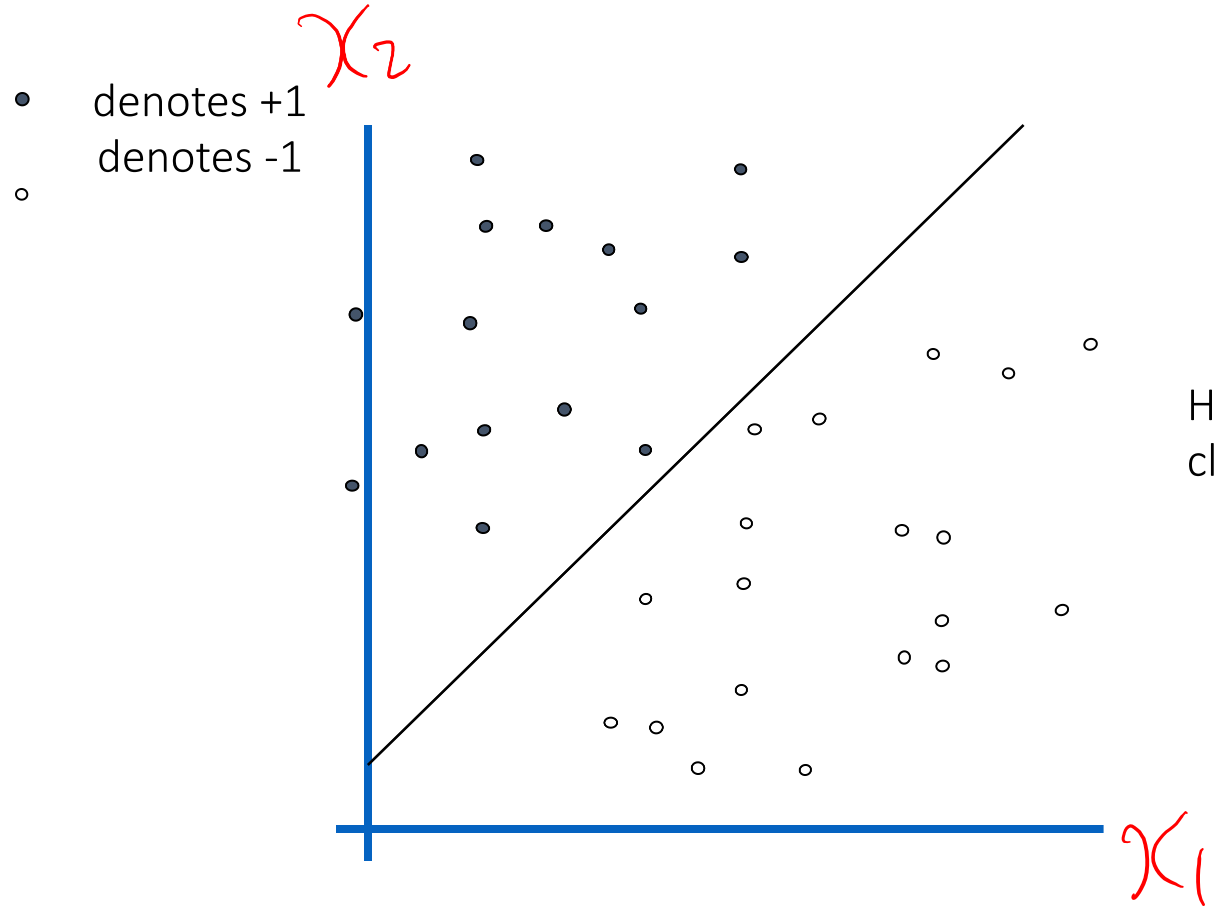


How would you classify this data?

Credit: Prof. Moore

Linear Classifiers

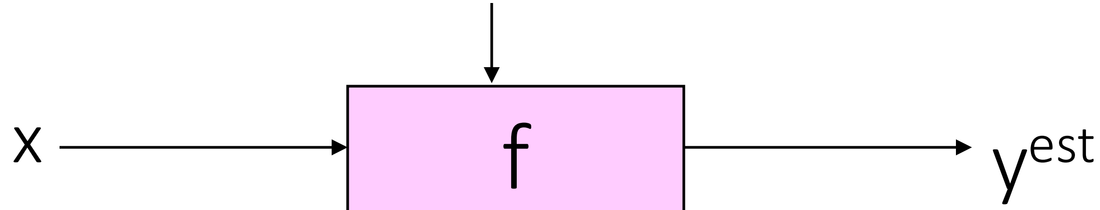


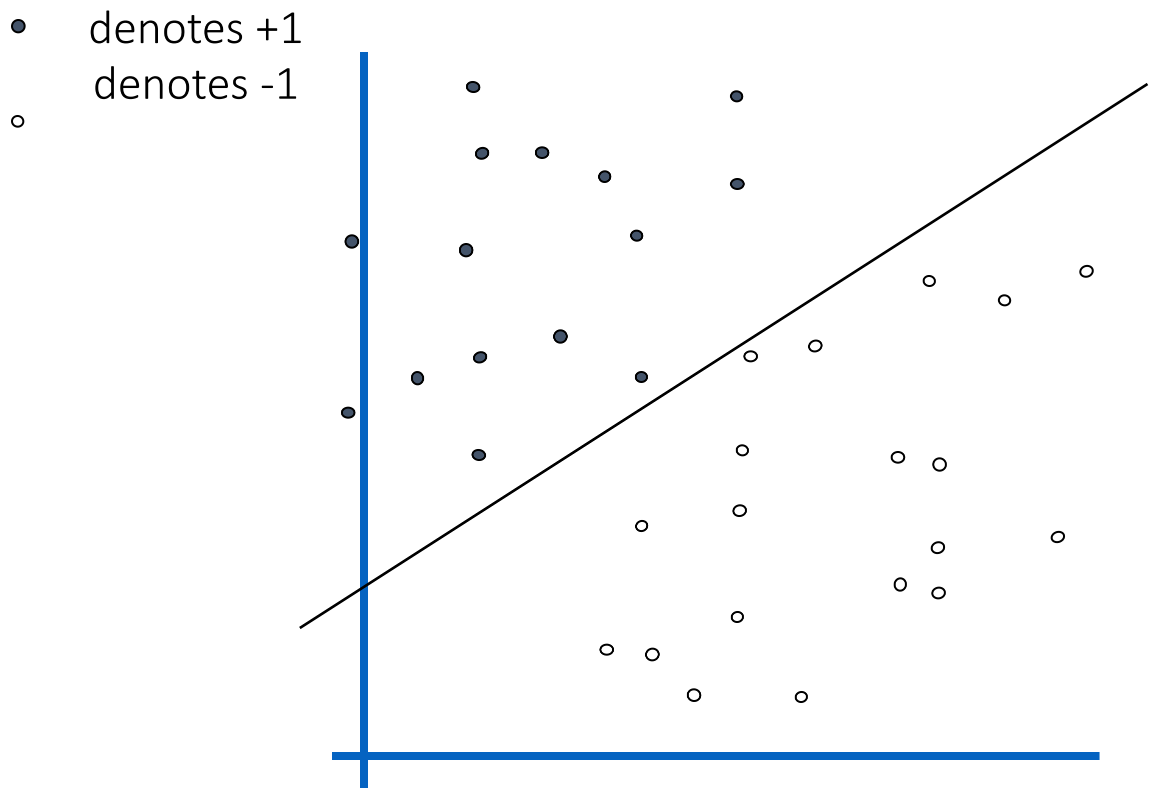


How would you classify this data?

Credit: Prof. Moore

Linear Classifiers

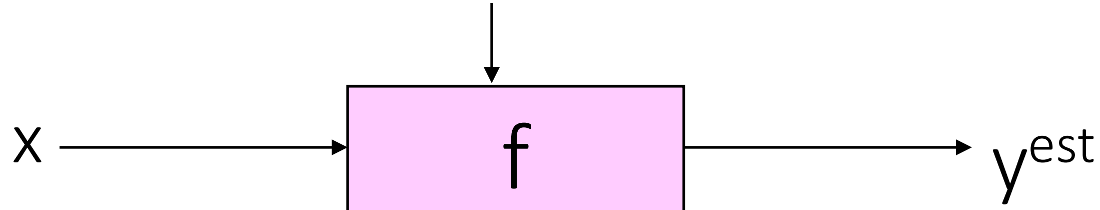


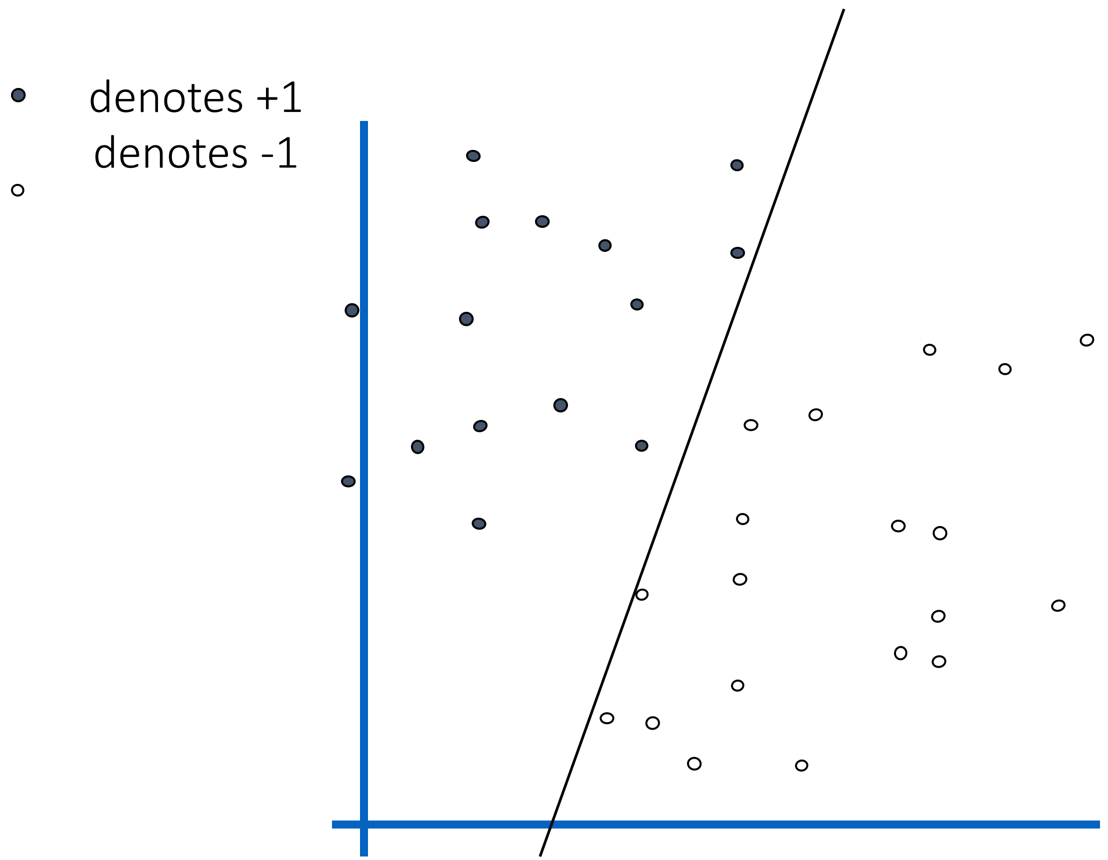


How would you classify this data?

Credit: Prof. Moore

Linear Classifiers

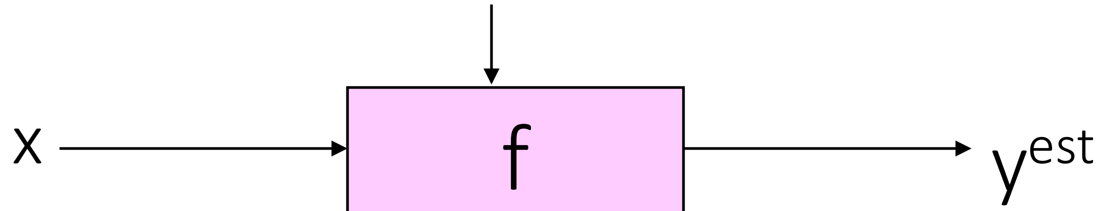




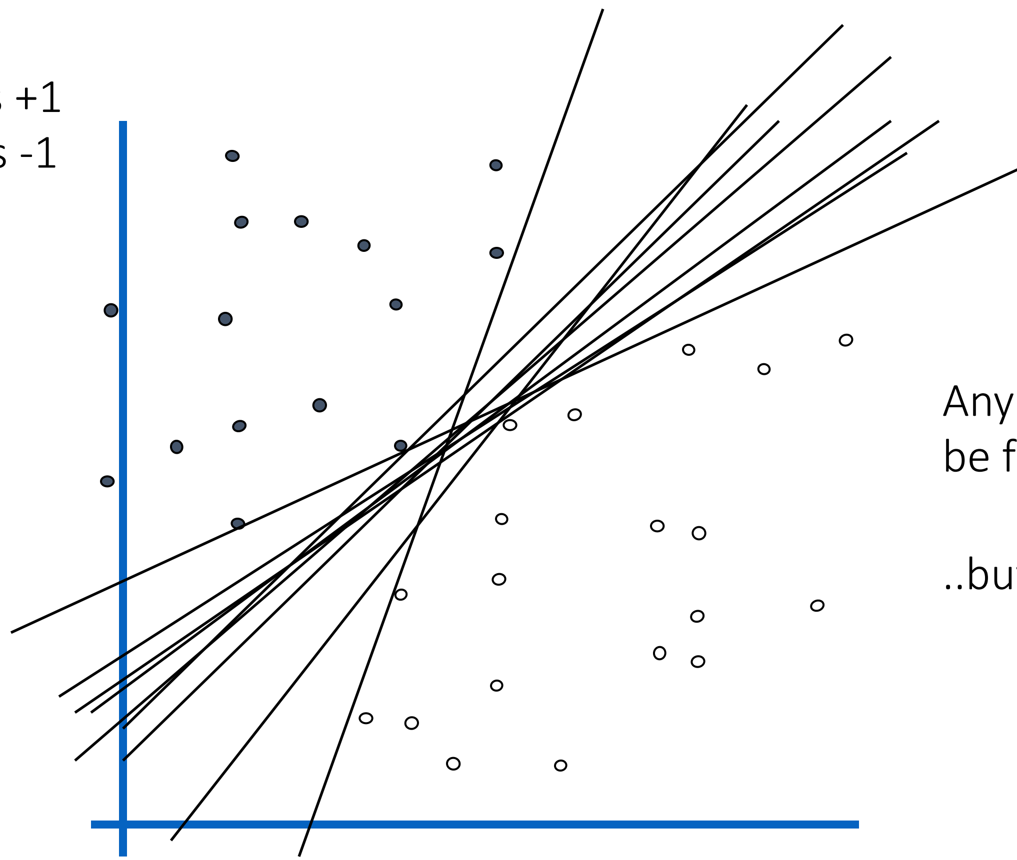
How would you classify this data?

Credit: Prof. Moore

Linear Classifiers



• denotes +1
○ denotes -1

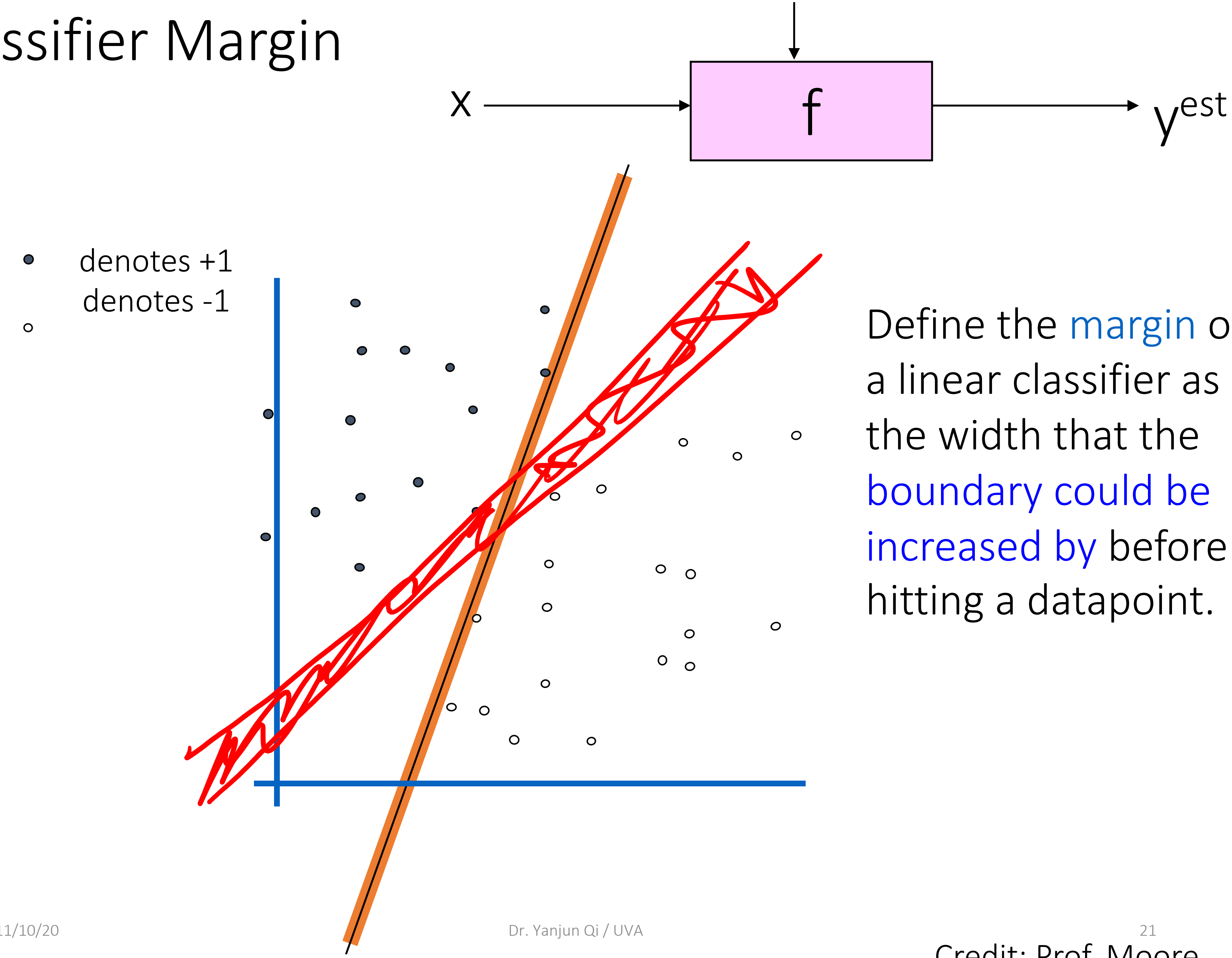


Any of these would be fine..

..but which is best?

Credit: Prof. Moore

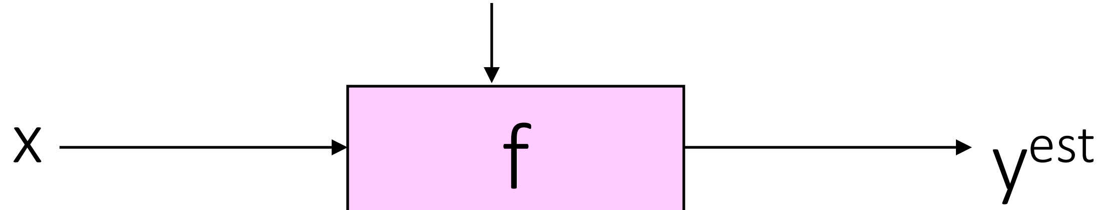
Classifier Margin

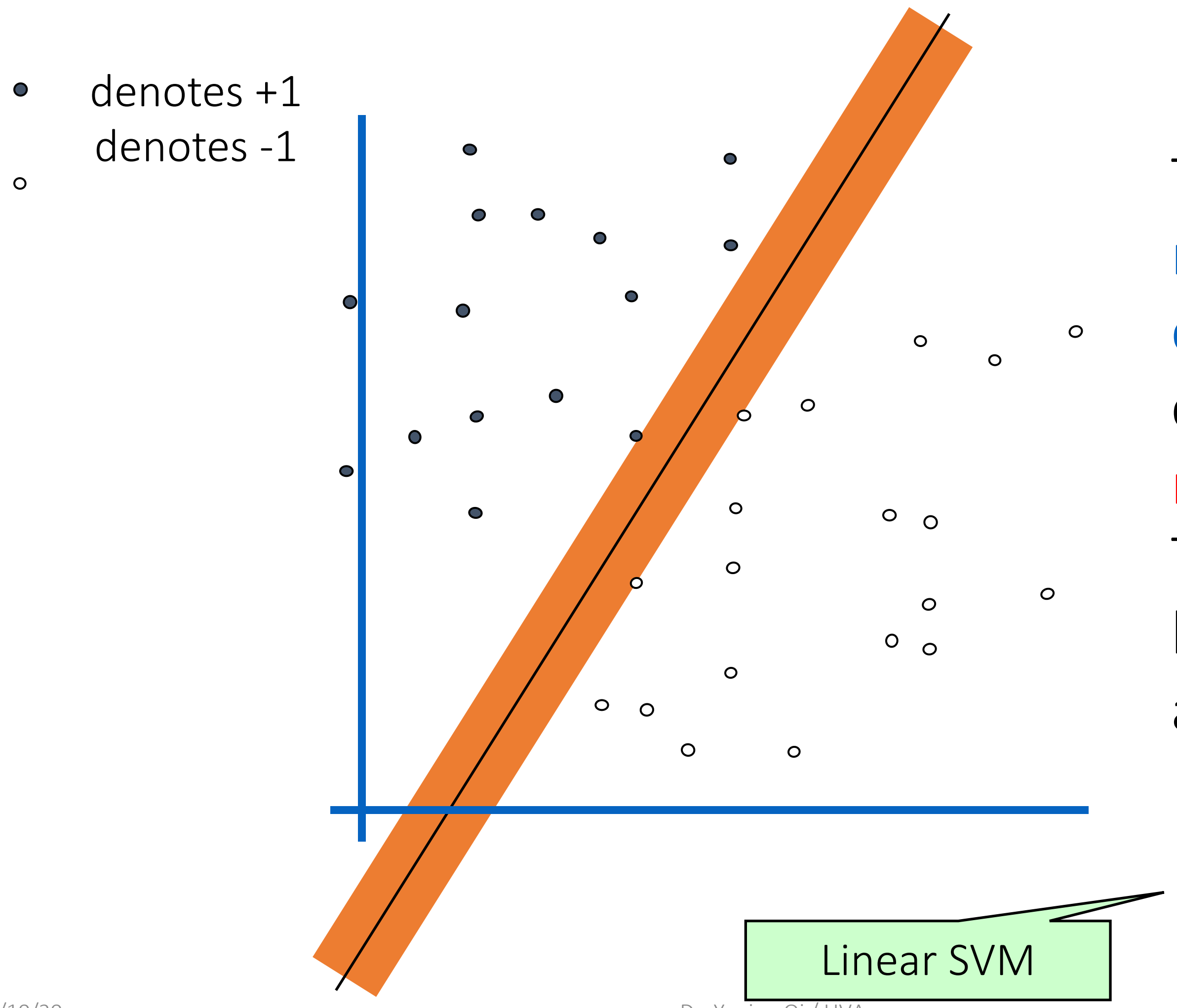


Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

Credit: Prof. Moore

Maximum Margin

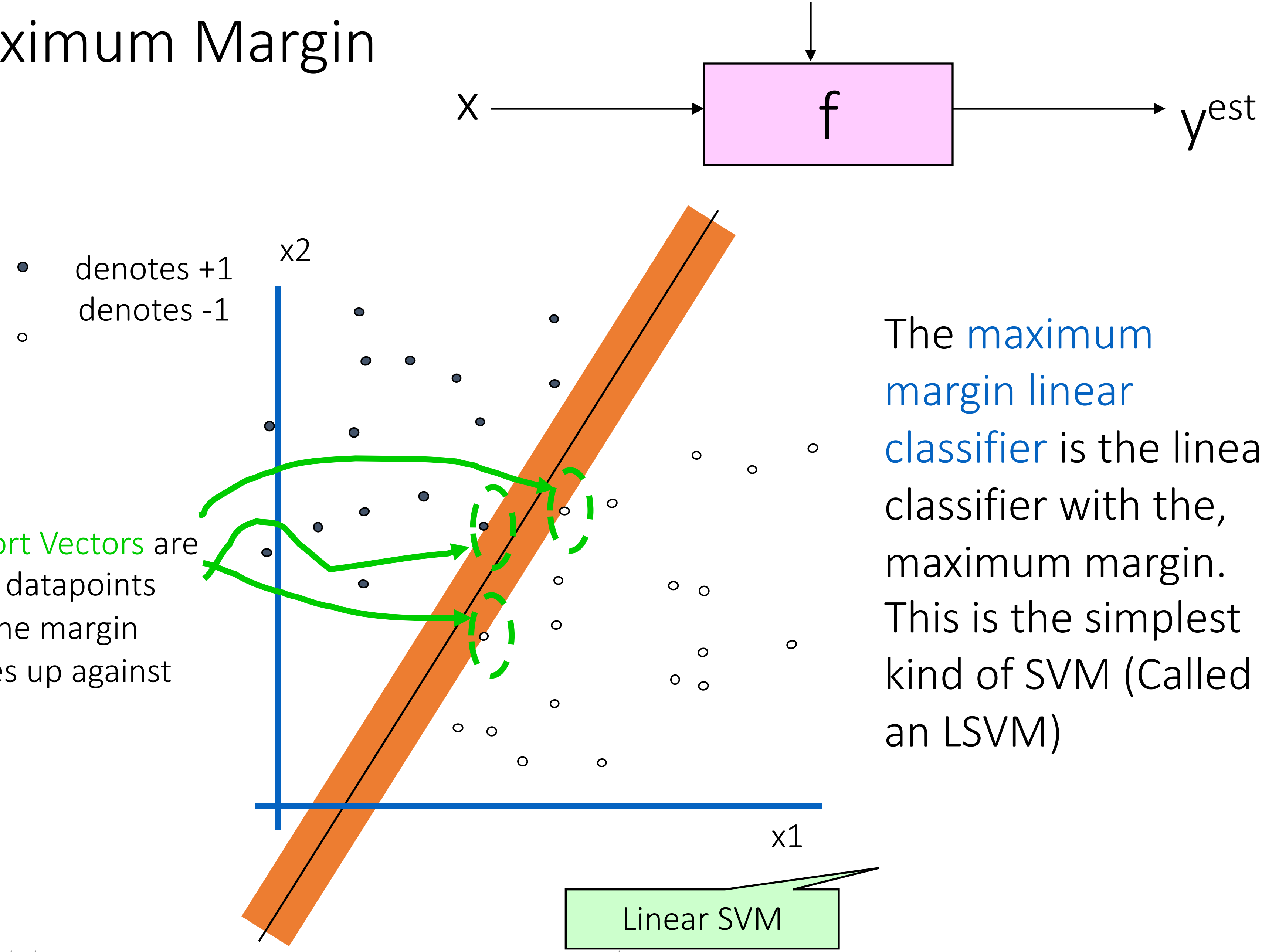




The maximum margin linear classifier is the linear classifier with the, maximum margin. This is the simplest kind of SVM (Called an LSVM)

Credit: Prof. Moore

Maximum Margin

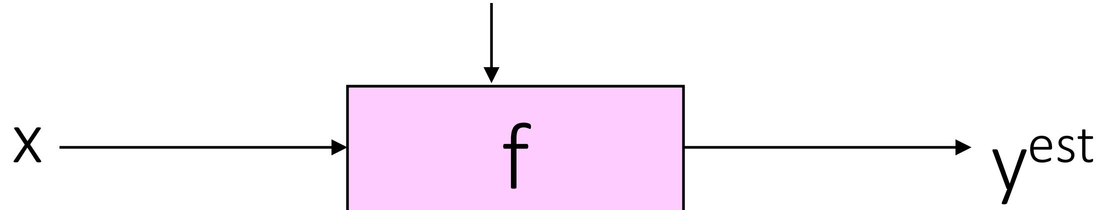


Support Vectors are those datapoints that the margin pushes up against

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

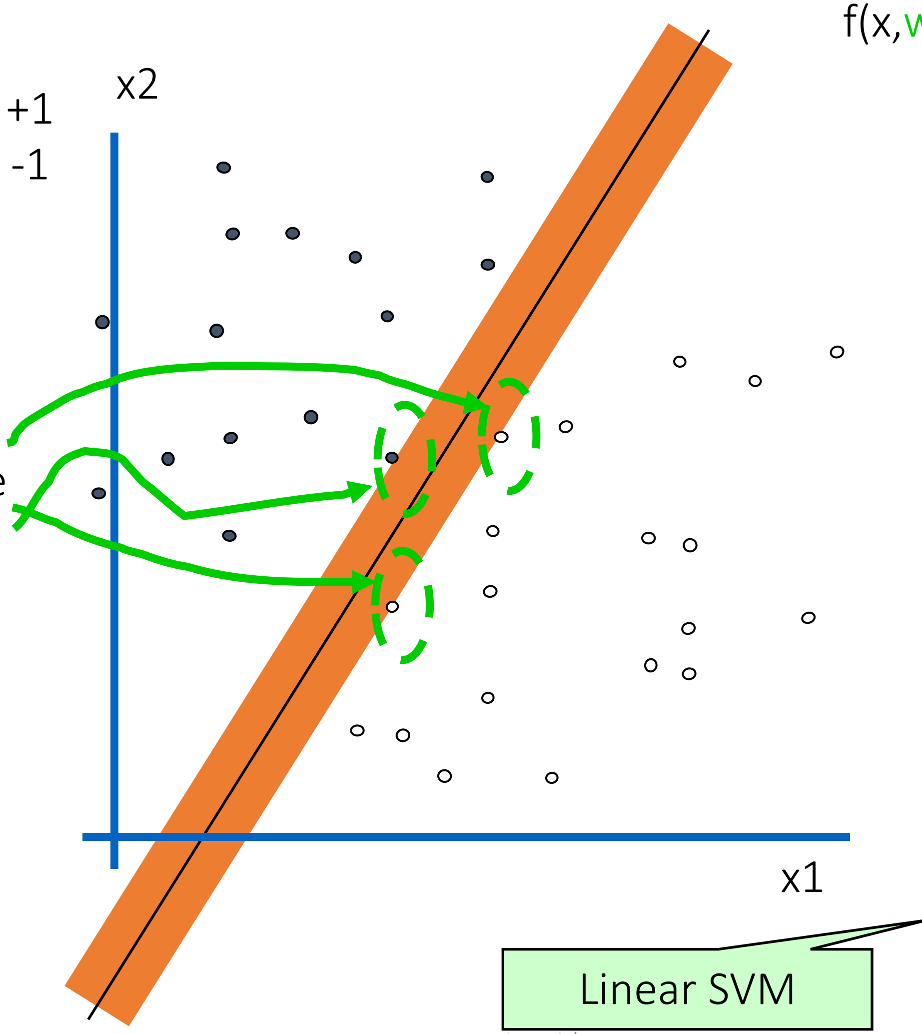
Credit: Prof. Moore

Maximum Margin



$f(x,w,b) = \mathrm{sign}(w^T x + b)$

- denotes +1
- denotes -1



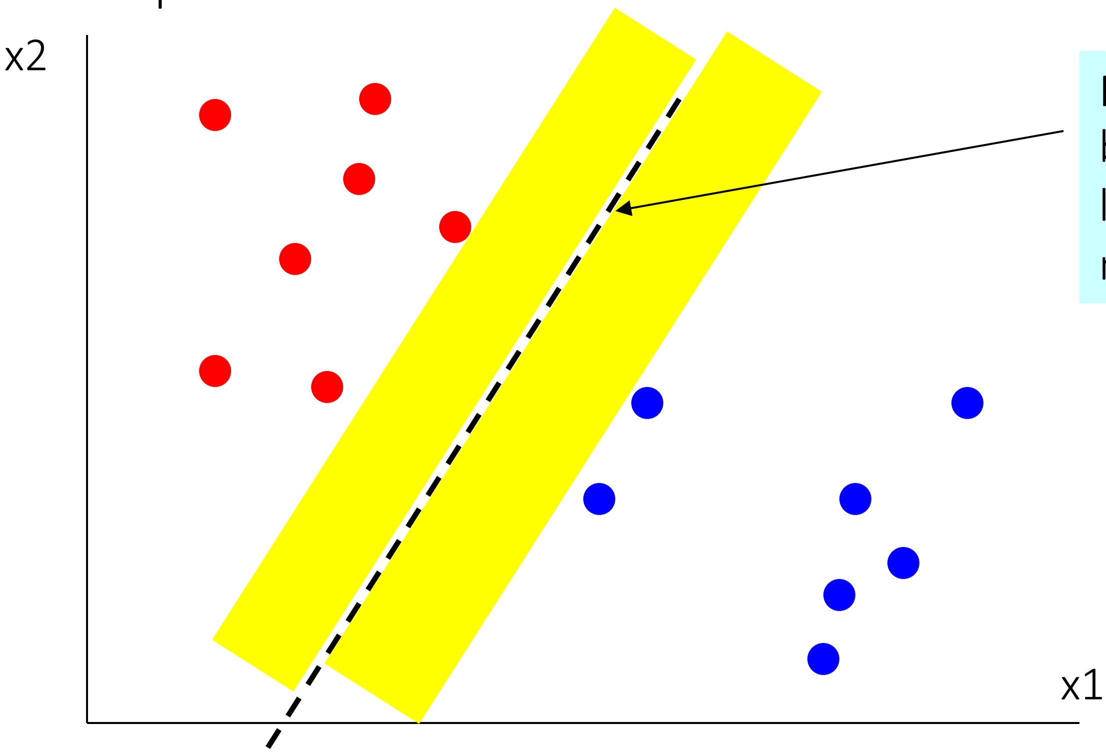
Support Vectors are those datapoints that the margin pushes up against

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

Credit: Prof. Moore

Max margin classifiers

- Instead of fitting all points, focus on boundary points
- Learn a boundary that leads to the largest margin from both sets of points



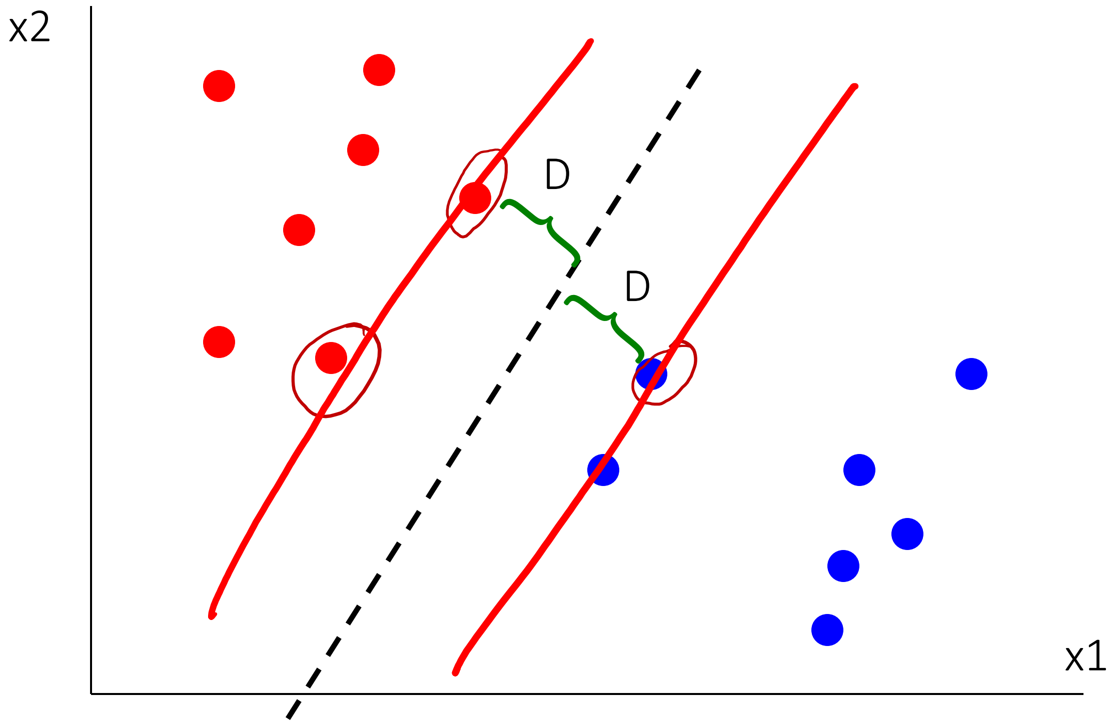
From all the possible boundary lines, this leads to the largest margin on both sides

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

Credit: Prof. Moore

Max margin classifiers

- Instead of fitting all points, focus on boundary points
- Learn a boundary that leads to the largest margin from points on both sides



Why MAX margin?

- Intuitive, ‘makes sense’
- Some theoretical support (using VC dimension)
- Works well in practice

Credit: Prof. Moore

Thank You

Thank you

UVA CS 4774: Machine Learning

S4: Lecture 20: Support Vector Machine (Basics)

Module II

Dr. Yanjun Qi

University of Virginia

Department of Computer Science

Today: Basic Support Vector Machine

Data
↓
Task
↓
Representation
↓
Score Function
↓
Search/Optimization
↓
Models, Parameters

Tabular
↓
Classification
↓
Kernel Trick Func K(xi, xj)

$$K(\mathbf{x}, z) := \Phi(\mathbf{x})^T \Phi(z)$$

↓
Margin + Hinge Loss
↓
QP with Dual form

$$w = \sum_i \alpha_i x_i y_i$$

↓
Dual Weights

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{p} w_i^2 + C\sum_{i=1}^{n} \varepsilon_i$$

$$\text{subject to } \forall \mathbf{x}_i \in Dtrain: y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1 - \varepsilon_i$$

$$\max_\alpha \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^T \mathbf{x_j}$$

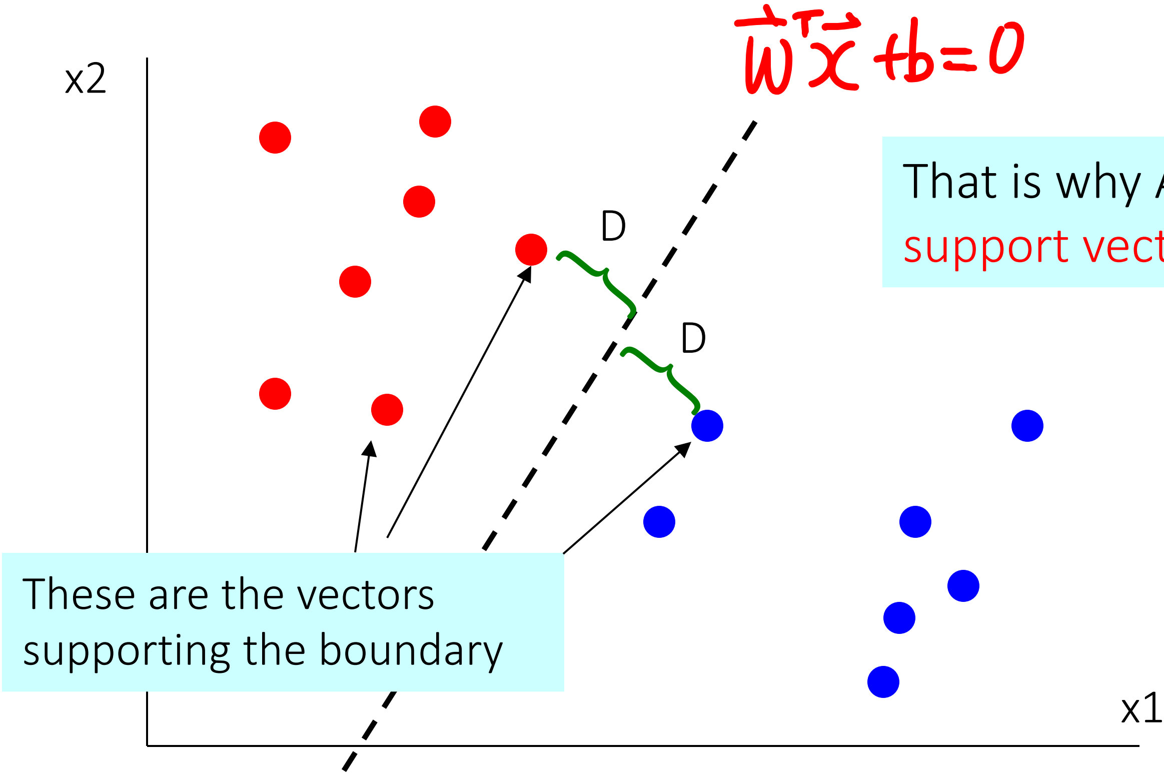
$$\sum_i \alpha_i y_i = 0, \qquad \alpha_i \geq 0 \qquad \forall i$$

Today

- ❑ Supervised Classification
- ❑ Support Vector Machine (SVM)
  - ✓ History of SVM
  - ✓ Large Margin Linear Classifier
  - ✓ Define Margin (M) in terms of model parameter
  - ✓ Optimization to learn model parameters (w, b)
  - ✓ Linearly Non-separable case
  - ✓ Optimization with dual form
  - ✓ Nonlinear decision boundary
  - ✓ Multiclass SVM

Max margin classifiers

- Instead of fitting all points, focus on boundary points
- Learn a boundary that leads to the largest margin from points on both sides

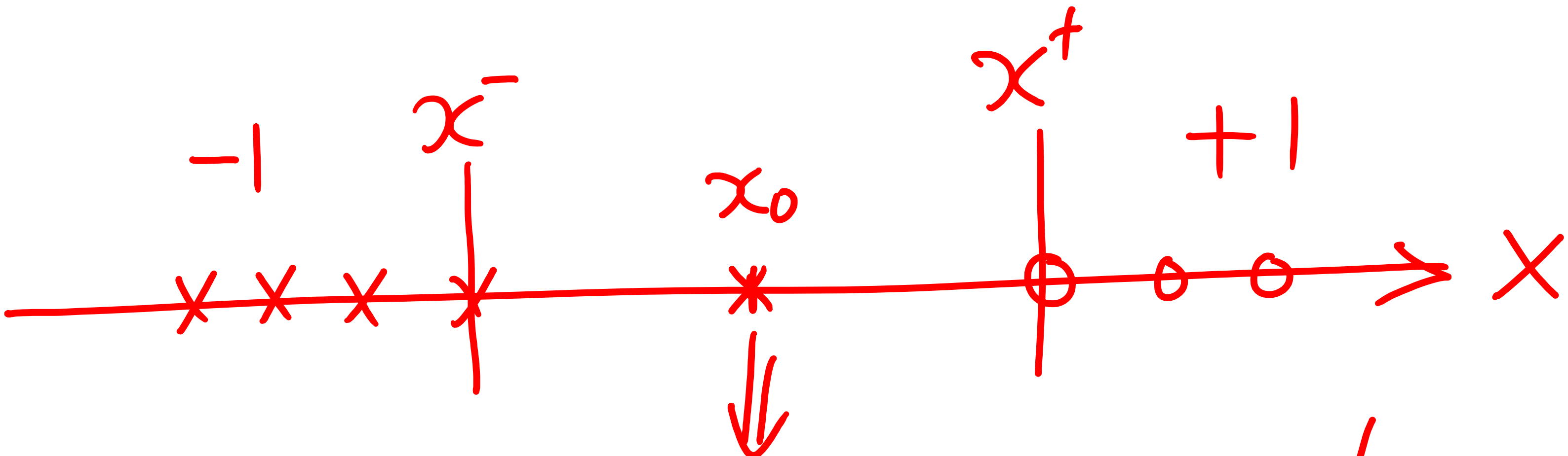


That is why Also known as linear support vector machines (SVMs)

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

Credit: Prof. Moore

$R^1 : \{x\} \to \{+1, -1\}$



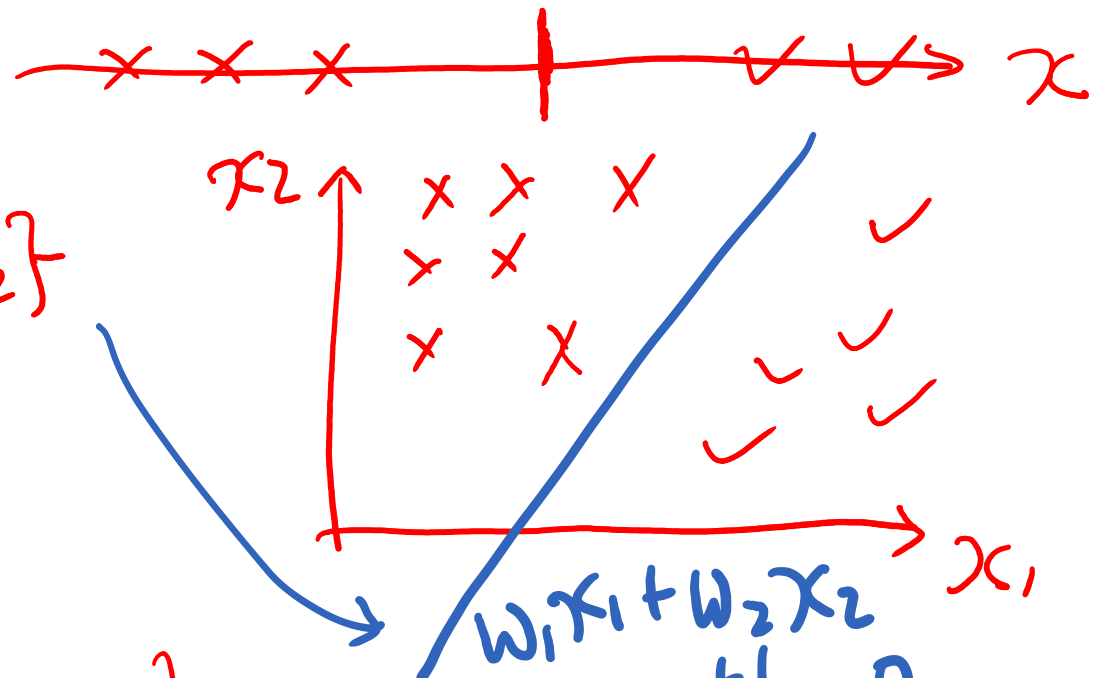
$$\begin{cases} \vec{w}^T \vec{x}_0 + b = 0 \\ w^T x^- + b = -1 \\ w^T x^+ + b = 1 \end{cases}$$

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

How to represent a Linear Decision Boundary?

$R^1 : \{x\}$

$x = x_0 = -\frac{b}{w}$



$R^2 : \{x_1, x_2\}$

$w_1 x_1 + w_2 x_2 + b = 0$

$R^p : \{x_1, x_2, \ldots, x_p\}$: $\vec{w}^T \vec{x} + b = 0$

Review : **Affine Hyperplanes**

- https://en.wikipedia.org/wiki/Hyperplane
- Any hyperplane can be given in coordinates as the solution of a single linear (algebraic) equation of degree 1.

$$\left[a_1 x_1 + a_2 x_2 + a_3 x_3 + \cdots + a_p x_p = -b \right], \text{ at least one } a_i \neq 0$$

$\Rightarrow$ e.g. classification Boundary $\quad \vec{w}^T \vec{x} + b = 0$

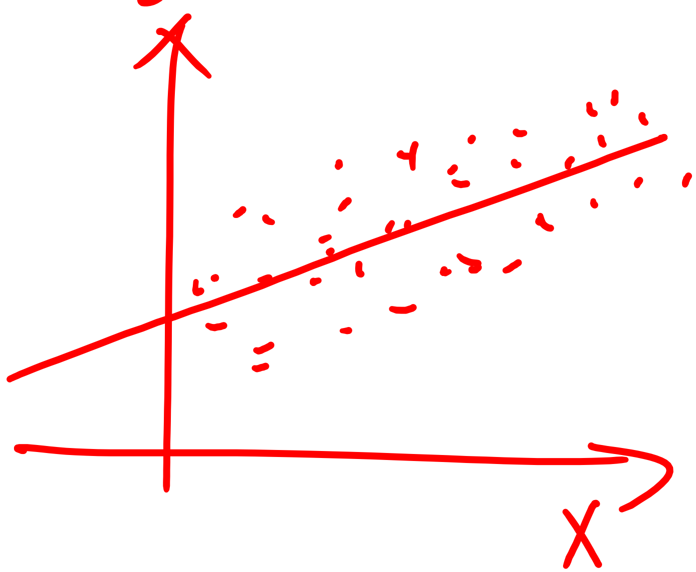
$$\begin{cases} \vec{x} \in R^p \\ b \in R \end{cases}$$

Q: How does this connect to linear regression?

[Regression]

$y \in \mathbb{R}^1$

$\vec{X} \in \mathbb{R}^p$, $p=1$ e.g.



$y = w^T x + w_0$

$-y + w^T x + w_0 = 0$

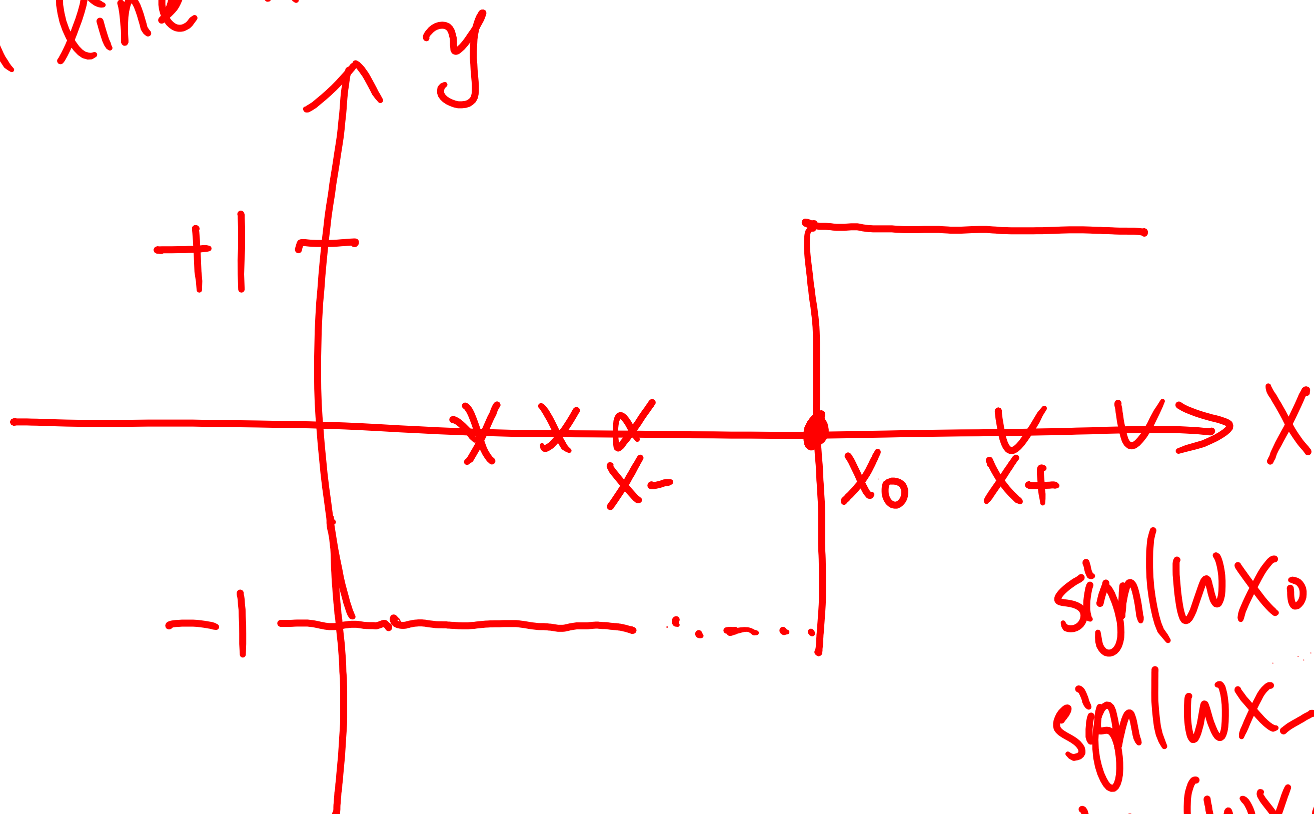
a line in (x, y) space
$p+1$

Binary Classification $y \in \{-1, 1\}$

Decision Boundary is a line in R^p

$X \in R^p$, $p=1$ e.g.

(x, y)



$$\mathrm{sign}(wx_0 + b) = 0$$
$$\mathrm{sign}(wx_- + b) = -1$$
$$\mathrm{sign}(wx_+ + b) = +1$$

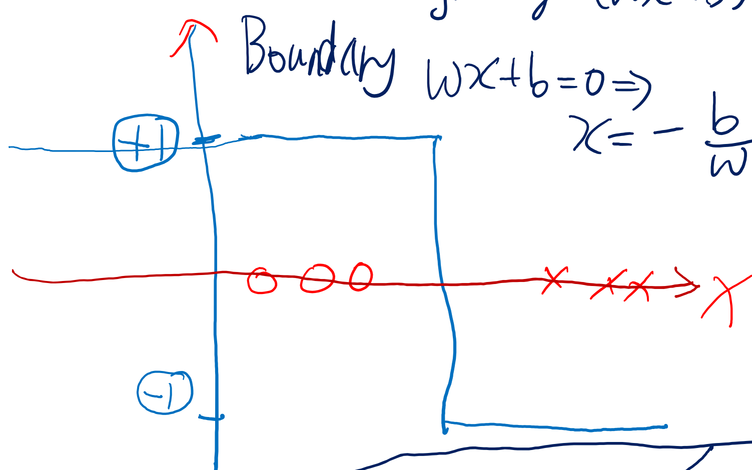
Binary Classification

$y \in \{-1, 1\}$

if ⇒ 1D [$x \in R$]

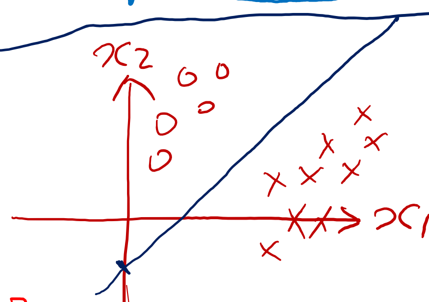
$y = sign(wx+b)$

Boundary $wx+b=0 \Rightarrow x = -\frac{b}{w}$



if ⇒ 2D [$x \in R^2$]

Boundary $w^T x + b = 0$



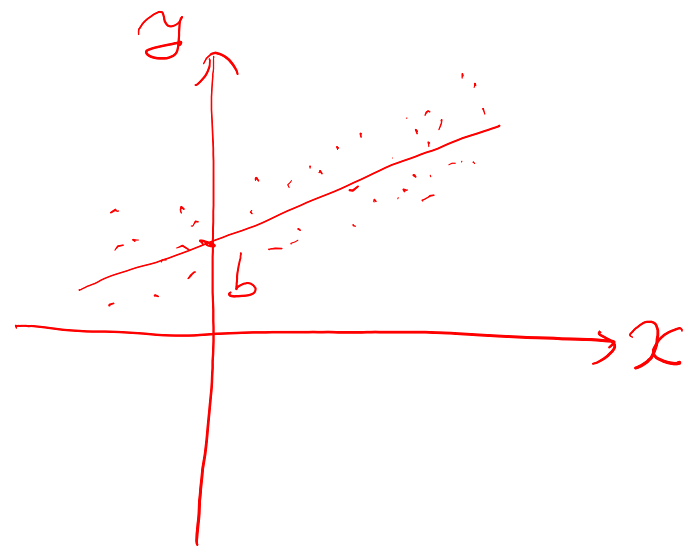
if ⇒ p dim [$x \in R^p$]

Boundary : hyperplane

Regression

$y \in R$

if ⇒ 1D $x \in R$



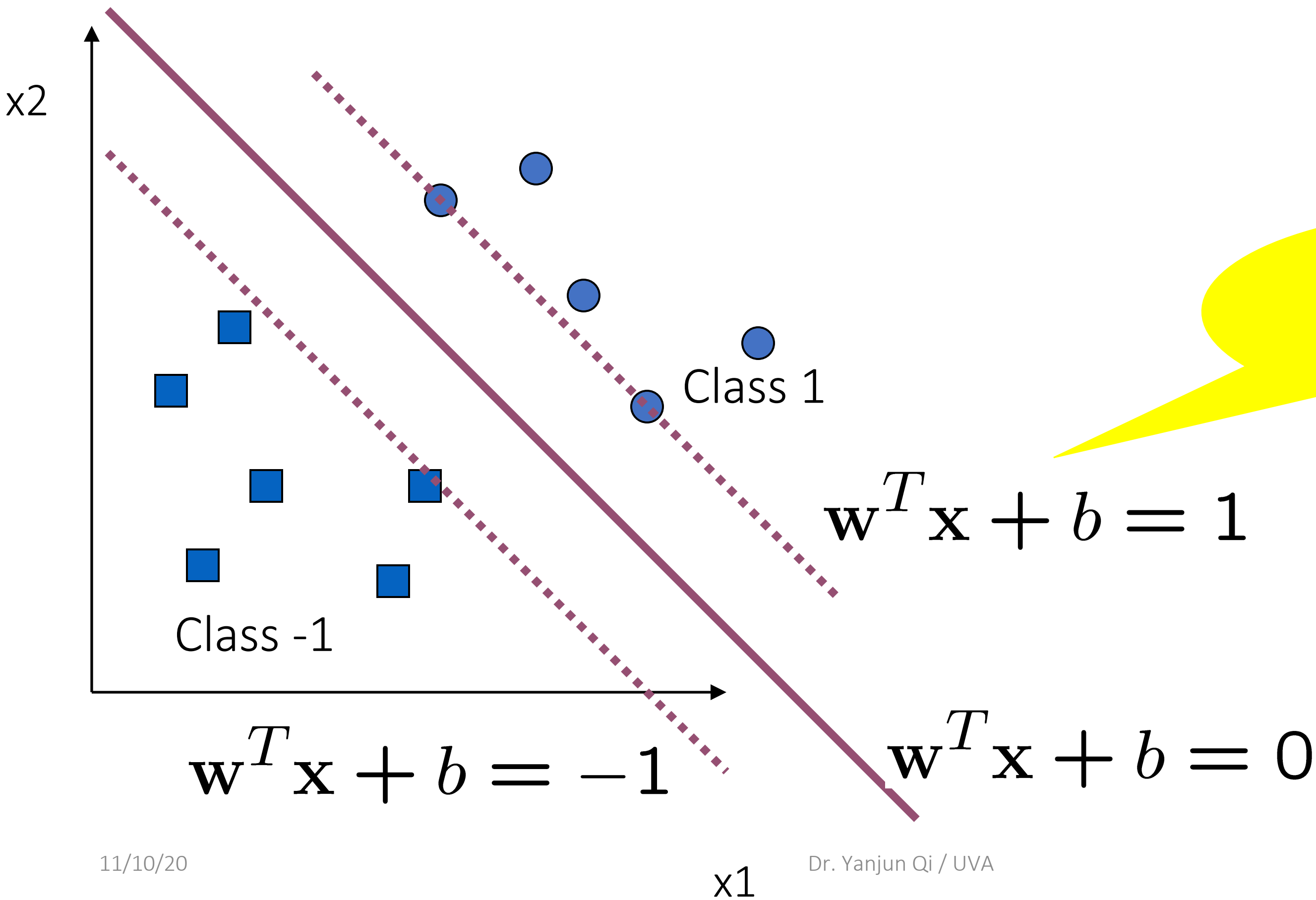
$y = wx + b$

$y = \vec{w}^T \vec{x} + b$ if $x \in R^p$

aka. $y - wx - b = 0$

linear-line in 2-D space $\{x, y\}$

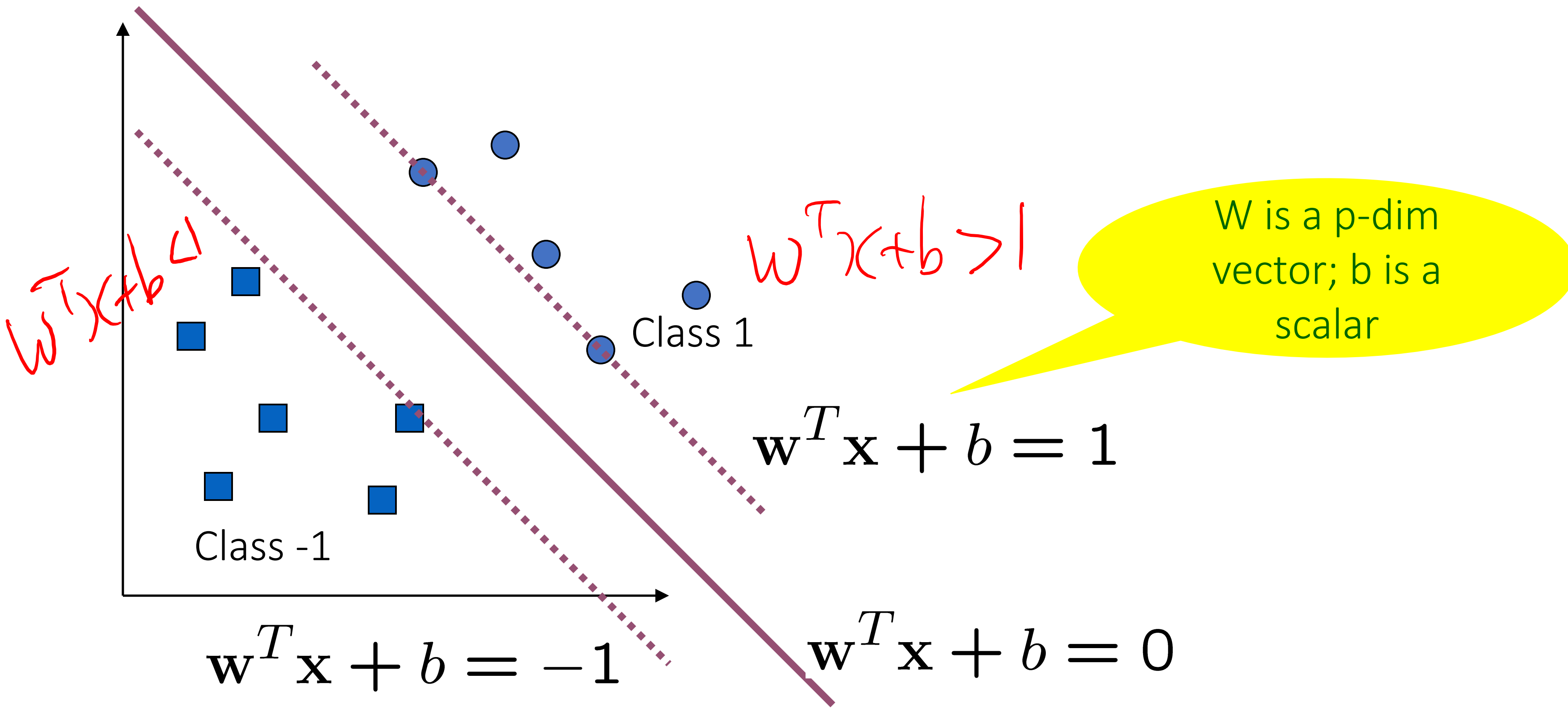
Max-margin & Decision Boundary



W is a p-dim vector; b is a scalar

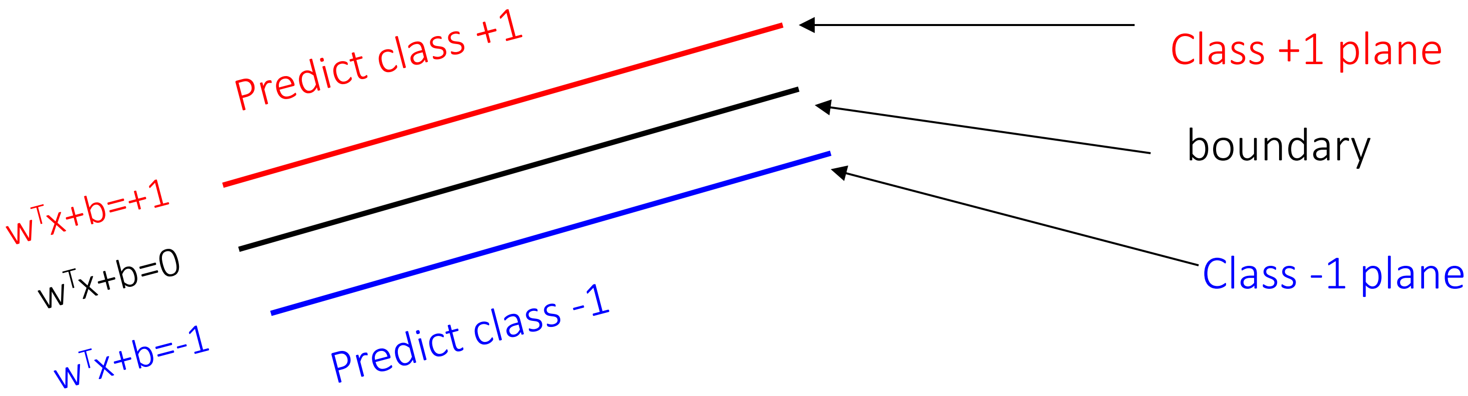
Max-margin & Decision Boundary $\mathbf{w}^T\mathbf{x} + b = 0$

- The decision boundary should be as far away from the data of both classes as possible



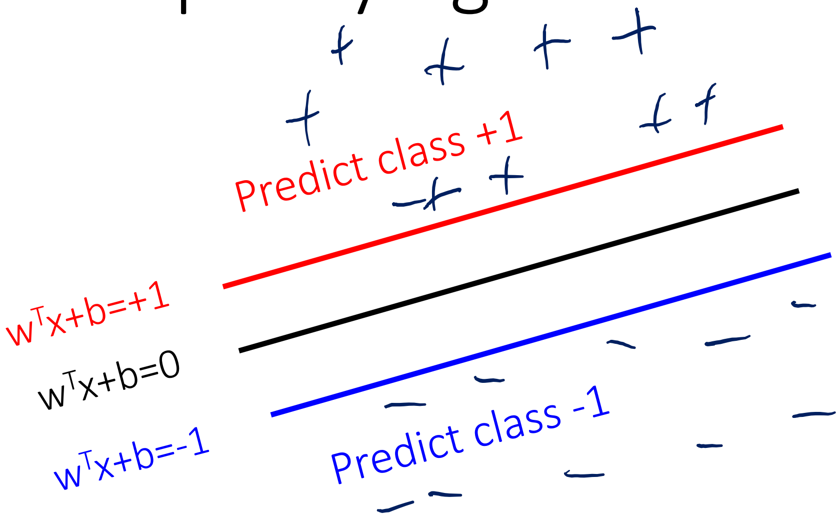
$f(x,w,b) = \text{sign}(w^T x + b)$

Specifying a max margin classifier



Classify as +1	if	$w^T x+b >= 1$
Classify as -1	if	$w^T x+b <= -1$
Undefined	if	$-1 < w^T x+b < 1$

Specifying a max margin classifier



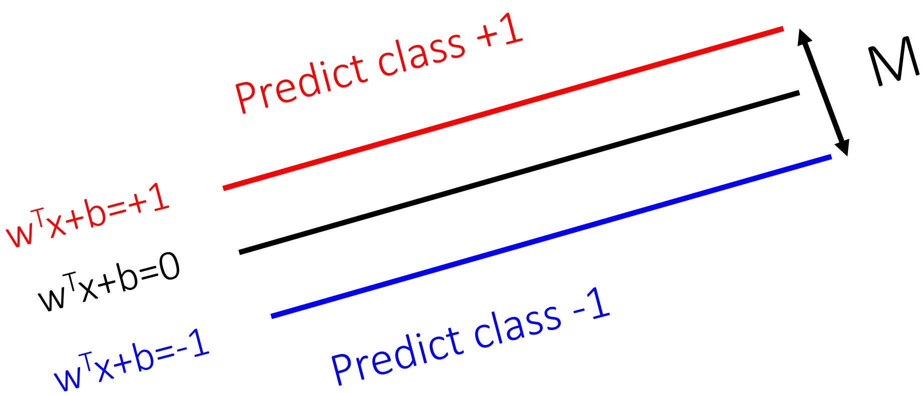
Is the linear separation assumption realistic?

We will deal with this shortly, but lets assume it for now

Classify as +1	if	$w^Tx+b >= 1$
Classify as -1	if	$w^Tx+b <= - 1$
Undefined	if	$-1 < w^Tx+b < 1$

Now Assuming such lines exist in our train

Maximizing the margin



Classify as +1 if $w^Tx+b >= 1$

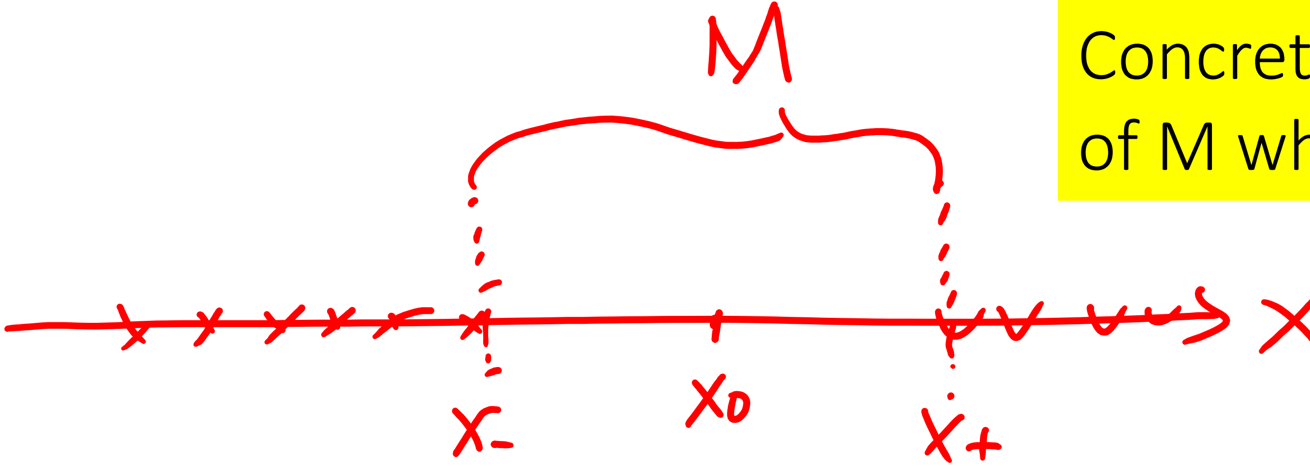
Classify as -1 if $w^Tx+b <= -1$

Undefined if $-1 < w^Tx+b < 1$

- Let us define the width of the margin by M
- How can we encode our goal of maximizing M in terms of our parameters (w and b)?
- Lets start with a few observations

See Concrete derivations of M in Extra slides when 2D

Concrete derivations of M when X is 1D



$p=1$

Length of the vector

$$\Rightarrow M = \left| \vec{x}_+ - \vec{x}_- \right|$$

$$= \left| \frac{2}{w} \right|$$

$$\Rightarrow M = \frac{2}{\sqrt{w^T w}}$$

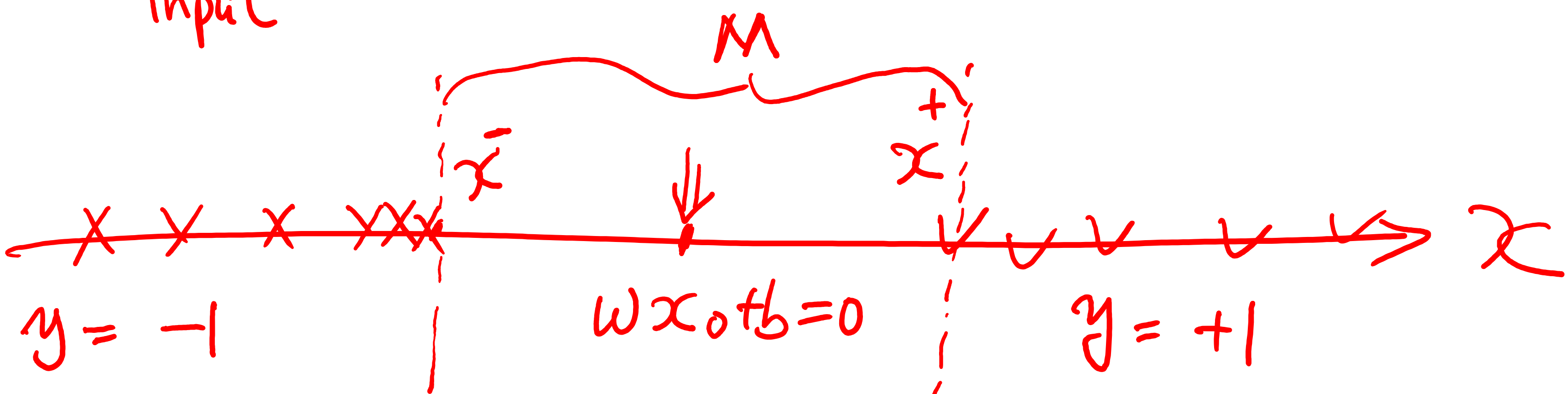
$$\begin{cases} w x_- + b = -1 \\ w x_+ + b = 1 \end{cases}$$

$$\Downarrow$$

$$w(x_+ - x_-) = 2$$

$$\Rightarrow \max M \Rightarrow \min w^T w$$

If 1-D input $\{x\} \rightarrow \{+1, -1\}$



$wx^- + b = -1$ support vector x^-

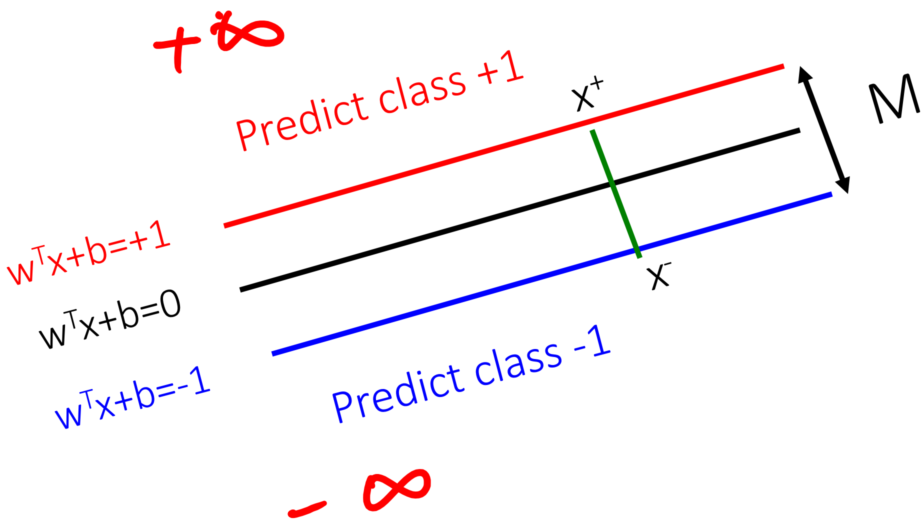
$wx^+ + b = 1$ supp. vector x^+

$$\Rightarrow \begin{cases} wx^- + b = -1 \\ wx^+ + b = 1 \end{cases} \Rightarrow w(x^+ - x^-) = 2$$

$$\text{margin } M = |x^+ - x^-| = \frac{2}{|w|}$$

(Interval when 1-D x)

Finding the optimal parameters



$$M = \frac{2}{\sqrt{w^T w}} = \frac{2}{||\mathbf{w}||}$$

We can now search for the optimal parameters by finding a solution that:

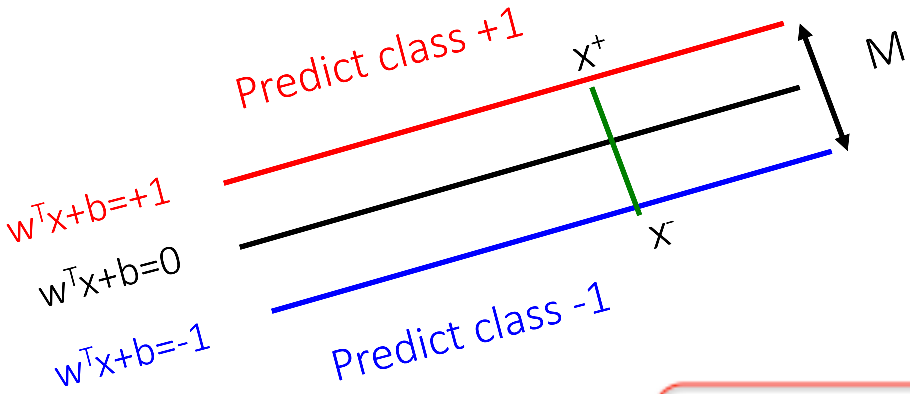
1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes $w^T w$)

Several optimization methods can be used: Gradient descent, OR SMO (see extra slides)

Today

- Support Vector Machine (SVM)
  - ✓ History of SVM
  - ✓ Large Margin Linear Classifier
  - ✓ Define Margin (M) in terms of model parameter
  - ✓ Optimization to learn model parameters (w, b)
  - ✓ Linearly Non-separable case
  - ✓ Optimization with dual form
  - ✓ Nonlinear decision boundary
  - ✓ Practical Guide

Optimization Step
i.e. learning optimal parameter for SVM

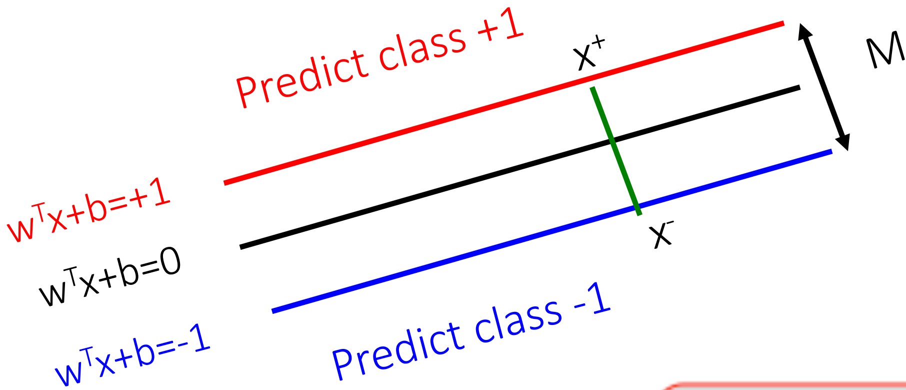


$$M = \frac{2}{\sqrt{w^T w}}$$

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

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes w^Tw)

Optimization Step
i.e. learning optimal parameter for SVM



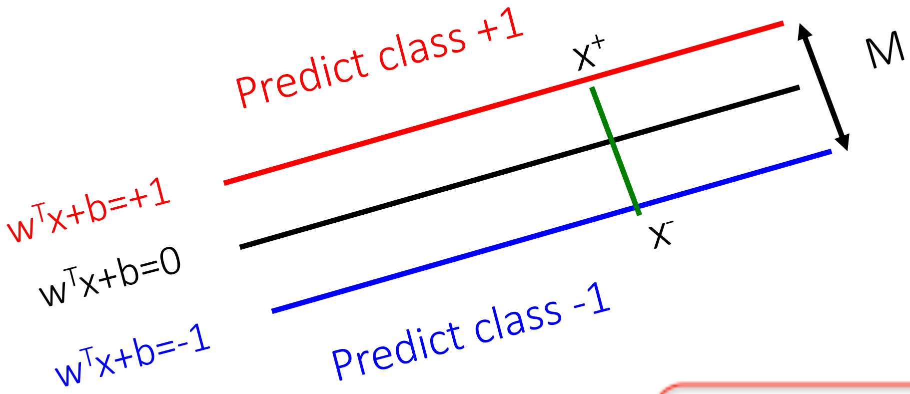
$$M = \frac{2}{\sqrt{w^T w}}$$

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes w^Tw)

Min $(w^Tw)/2$

subject to the following constraints:

Optimization Step
i.e. learning optimal parameter for SVM



$$M = \frac{2}{\sqrt{w^T w}}$$

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes w^Tw)

Min $(w^Tw)/2$

subject to the following constraints:

For all x in class + 1

$w^Tx+b >= 1$

For all x in class - 1

$w^Tx+b <= -1$

A total of n constraints if we have n training samples

Optimization Reformulation

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

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes $w^T w$)

Min $(w^T w)/2$

subject to the following constraints:

For all x in class + 1

$w^T x+b >= 1$ $y_i = 1$

For all x in class - 1

$w^T x+b <= -1$ $y_i = -1$

A total of n constraints if we have n input samples

$\rightarrow$ pos $y_i = 1$, $w^T x_i + b \geq 1$

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

$\rightarrow$ neg $y_i = -1$, $w^T x_i + b \leq -1$

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

Optimization Reformulation

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes w^Tw)

Min $(w^Tw)/2$

subject to the following constraints:

For all x in class + 1

$w^Tx+b \geq 1$

For all x in class - 1

$w^Tx+b \leq -1$

A total of n constraints if we have n input samples

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{p} w_i^2$$

$$\text{subject to } \forall \mathbf{x}_i \in Dtrain : y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1$$

$x_i^T w = w^T x_i$

Optimization Reformulation

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

1. Correctly classifies all points
2. Maximizes the margin (or equivalently minimizes $w^T w$)

$y_i \in \{+1, -1\}$

Quadratic Objective

Min $(w^T w)/2$

subject to the following constraints:

For all x in class + 1

$w^T x + b >= 1$

For all x in class - 1

$w^T x + b <= -1$

A total of n constraints if we have n input samples

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{p} w_i^2$$

$$\text{subject to } \forall \mathbf{x}_i \in Dtrain : \underbrace{y_i}_{1\times1} \big(\underbrace{\underbrace{\mathbf{w}^T}_{1\times p} \underbrace{\mathbf{x}_i}_{p\times1}}_{1\times1} + \underbrace{b}_{1\times1} \big) \geq 1$$

Quadratic programming i.e.,
- Quadratic objective
- Linear constraints

Today: Basic Support Vector Machine

Data → Task → Representation → Score Function → Search/Optimization → Models, Parameters

Tabular → Classification → Kernel Trick Func K(xi, xj) → Margin + Hinge Loss → QP with Dual form → Dual Weights

$$K(\mathbf{x}, z) := \Phi(\mathbf{x})^T \Phi(z)$$

$$w = \sum_i \alpha_i x_i y_i$$

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{p} w_i^2 + C\sum_{i=1}^{n} \varepsilon_i$$

$$\text{subject to } \forall \mathbf{x}_i \in Dtrain : y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1 - \varepsilon_i$$

$$\max_\alpha \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^T \mathbf{x_j}$$

$$\sum_i \alpha_i y_i = 0, \qquad \alpha_i \geq 0 \qquad \forall i$$

What Next?

- Support Vector Machine (SVM)
  - ✓ History of SVM
  - ✓ Large Margin Linear Classifier
  - ✓ Define Margin (M) in terms of model parameter
  - ✓ Optimization to learn model parameters (w, b)
  - ✓ Linearly Non-separable case (soft SVM)
  - ✓ Optimization with dual form
  - ✓ Nonlinear decision boundary
  - ✓ Practical Guide

Thank You

Thank you

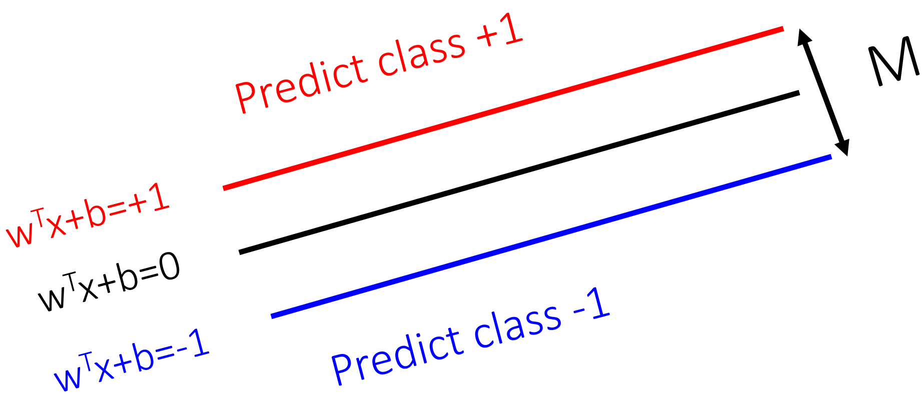
References

- Big thanks to Prof. Ziv Bar-Joseph and Prof. Eric Xing @ CMU for allowing me to reuse some of his slides
- Elements of Statistical Learning, by Hastie, Tibshirani and Friedman
- Prof. Andrew Moore @ CMU's slides
- Tutorial slides from Dr. Tie-Yan Liu, MSR Asia
- A Practical Guide to Support Vector Classification Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, 2003-2010
- Tutorial slides from Stanford "Convex Optimization I — Boyd & Vandenberghe

EXTRA

Concrete derivations of M in Extra slides when X is 2D

How to define the width of the margin by M (EXTRA)



Classify as +1 if $w^Tx+b >= 1$

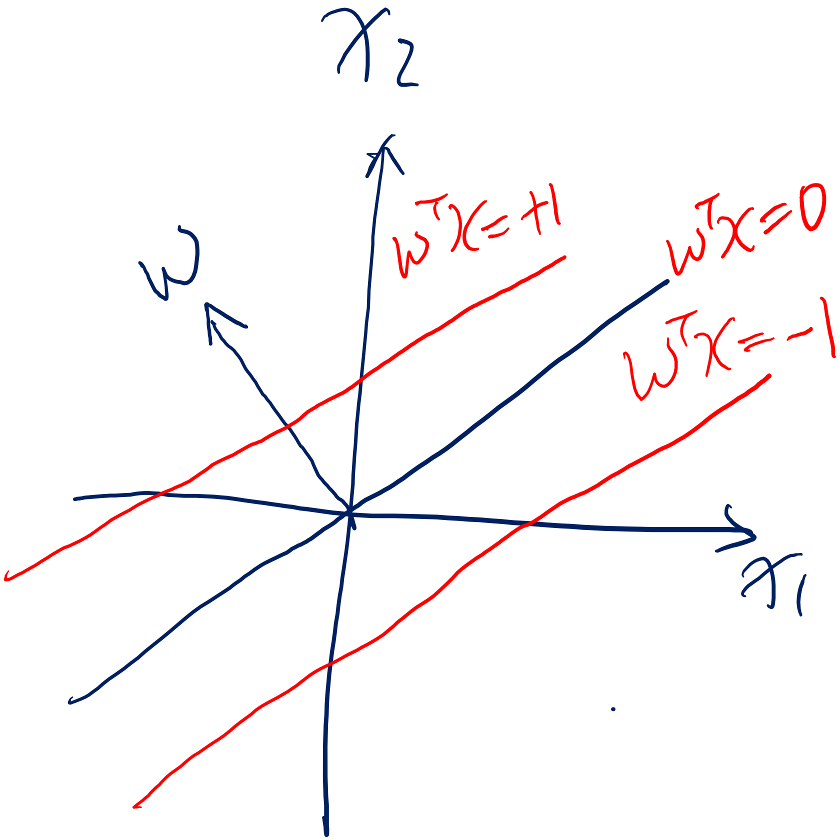
Classify as -1 if $w^Tx+b <= -1$

Undefined if $-1 < w^Tx+b < 1$

- Lets define the width of the margin by M
- How can we encode our goal of maximizing M in terms of our parameters (w and b)?
- Lets start with a few obsevrations

Concrete derivations of M see Extra

[Classification]



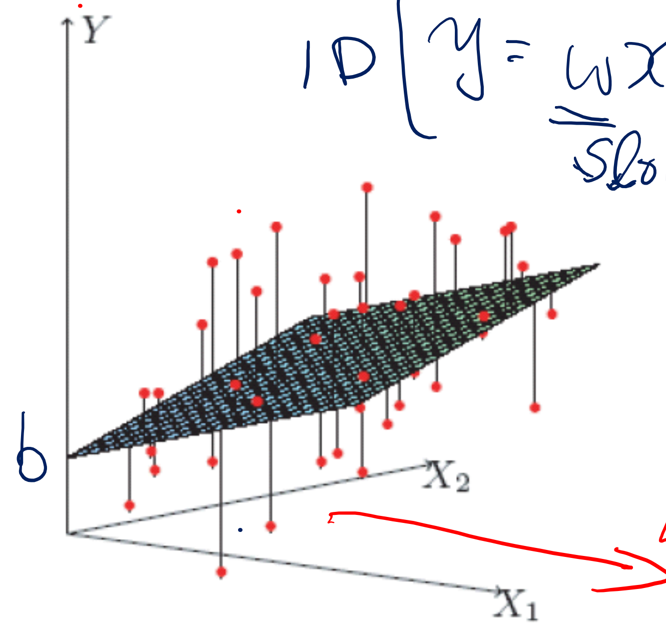
[Regression]

$$y = \vec{w}^T \vec{x} + b$$

$$\nabla x = \frac{\partial y}{\partial x} = \vec{w} \quad \text{Slope}$$

1D $[\, y = \underline{w}x + \underline{b} \,]$

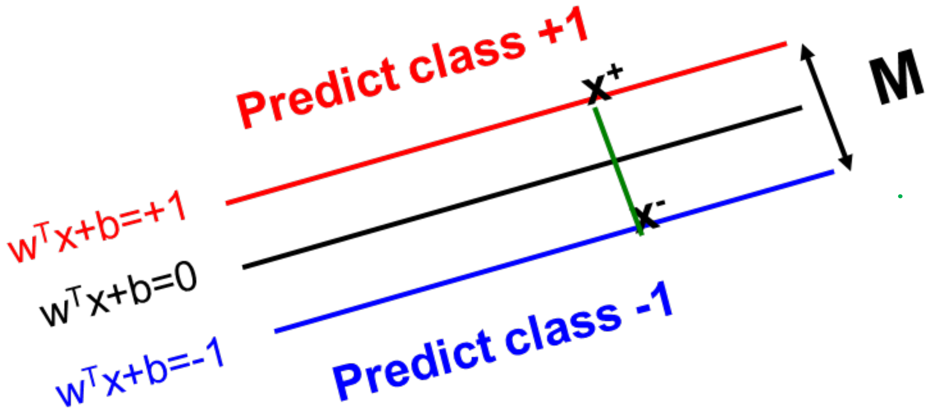
Slope intercept Bias



$\nabla x = w$

The gradient points in the direction of the greatest rate of increase of the function and its magnitude is the slope of the graph in that direction

Margin M



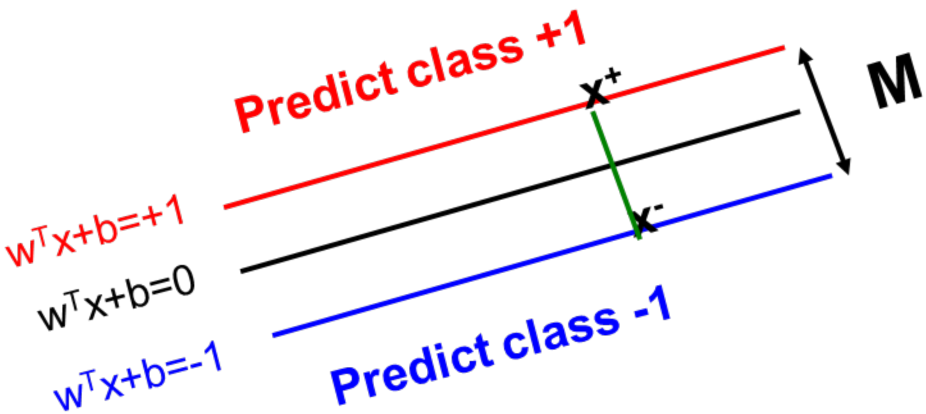
Classify as +1 if $w^Tx+b >= 1$

Classify as -1 if $w^Tx+b <= -1$

Undefined if $-1 < w^Tx+b < 1$

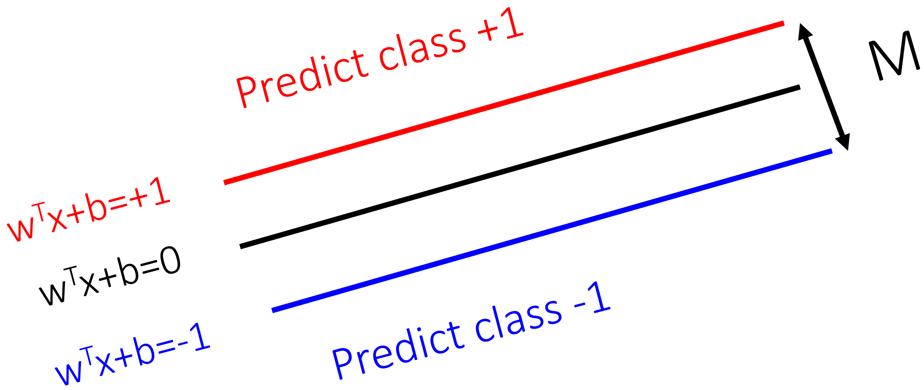
$M = |x^+ - x^-|$ length of vector $(x^+ - x^-)$

$\Rightarrow$ How to represent $(x^+ - x^-)$???



- $w^T x^+ + b = +1$
- $w^T x^- + b = -1$
- $M = | x^+ - x^- | = ?$

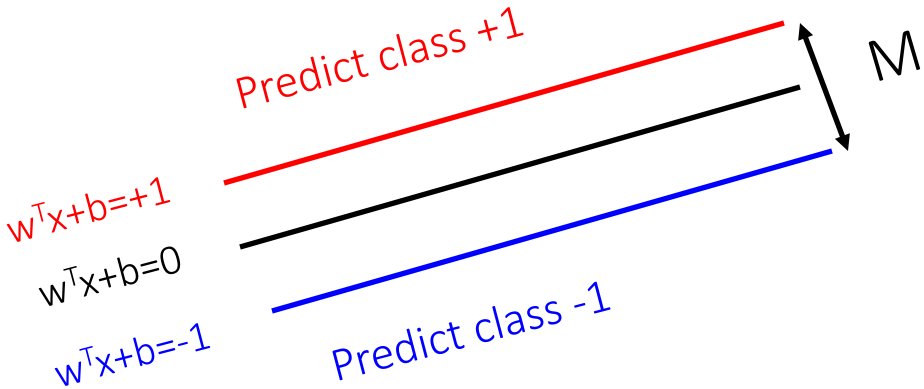
Maximizing the margin: observation-1



- Observation 1: the vector w is orthogonal to the +1 plane
- Why?

Corollary: the vector w is orthogonal to the -1 plane

Maximizing the margin: observation-1



Classify as +1 if $w^Tx+b >= 1$

Classify as -1 if $w^Tx+b <= -1$

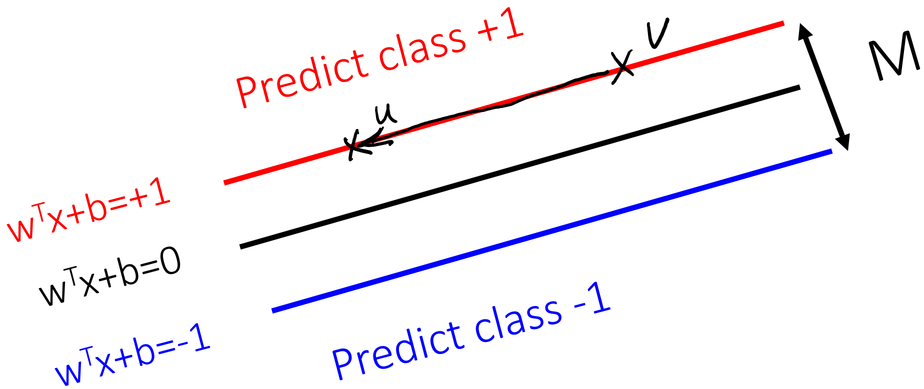
Undefined if $-1 < w^Tx+b < 1$

- Observation 1: the vector w is orthogonal to the +1 plane
- Why?

Let u and v be two points on the +1 plane, then for the vector defined by u and v we have $w^T(u-v) = 0$

Corollary: the vector w is orthogonal to the -1 plane

Maximizing the margin: observation-1



Classify as +1 if $w^Tx+b >= 1$

Classify as -1 if $w^Tx+b <= -1$

Undefined if $-1 < w^Tx+b < 1$

- Observation 1: the vector w is orthogonal to the +1 plane
- Why?

→ vector $(u-v)$ shown above

→ $w^T(u-v) = w^Tu - w^Tv = (1-b) - (1-b) = 0$

$\Rightarrow$ w orthogonal to $(u-v)$

Let u and v be two points on the +1 plane, then for the vector defined by u and v we have $w^T(u-v) = 0$

Corollary: the vector w is orthogonal to the -1 plane

Review : Vector Product, Orthogonal, and Norm

For two vectors x and y,

$$x^T y$$

is called the (inner) vector product.

x and y are called orthogonal if

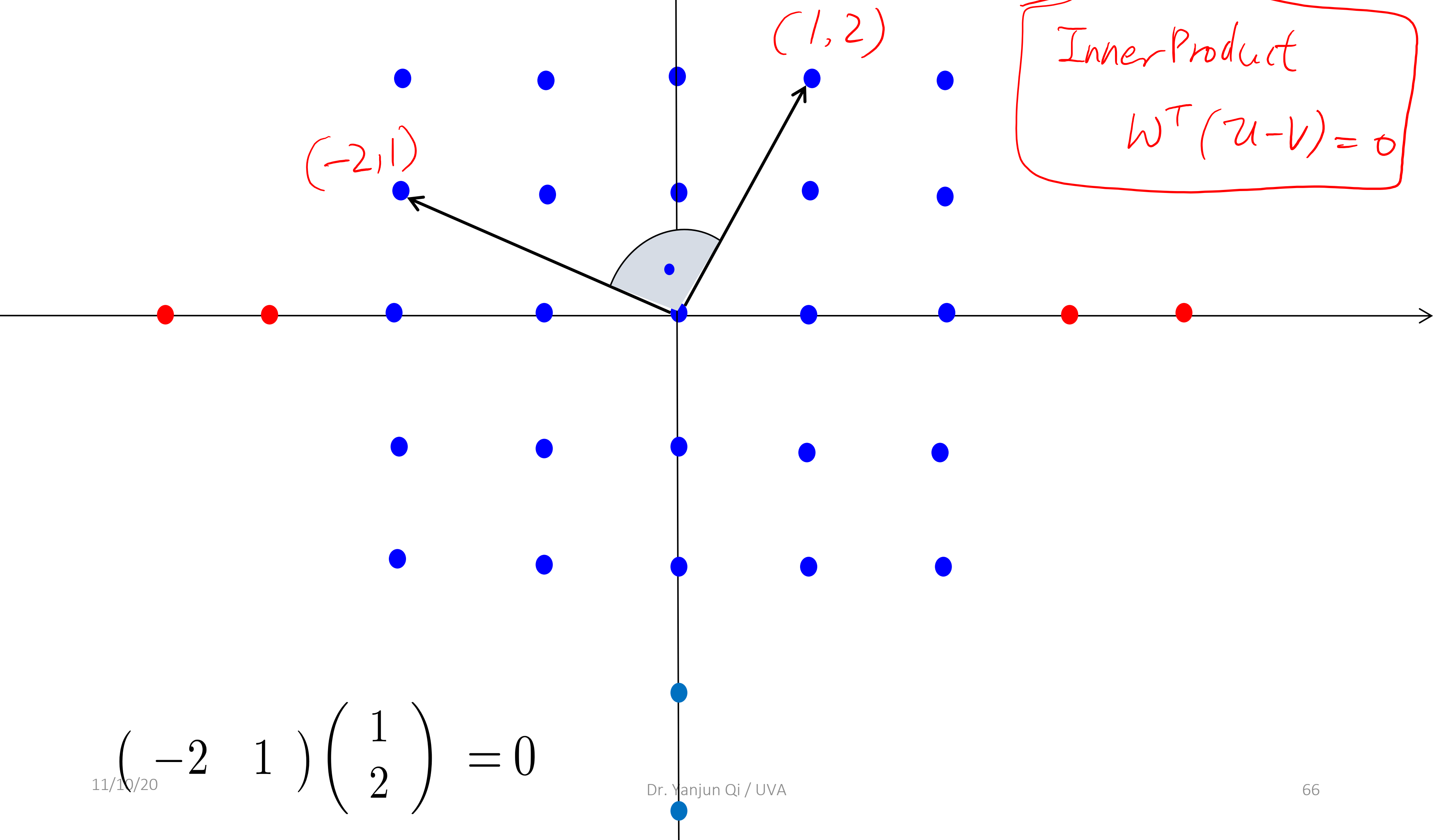
$$x^T y = 0$$

The square root of the product of a vector with itself,

$$\sqrt{x^T x}$$

is called the 2-norm ($|x|_2$), can also write as $|x|$

Observation 1
➔ Review : Orthogonal



Inner Product

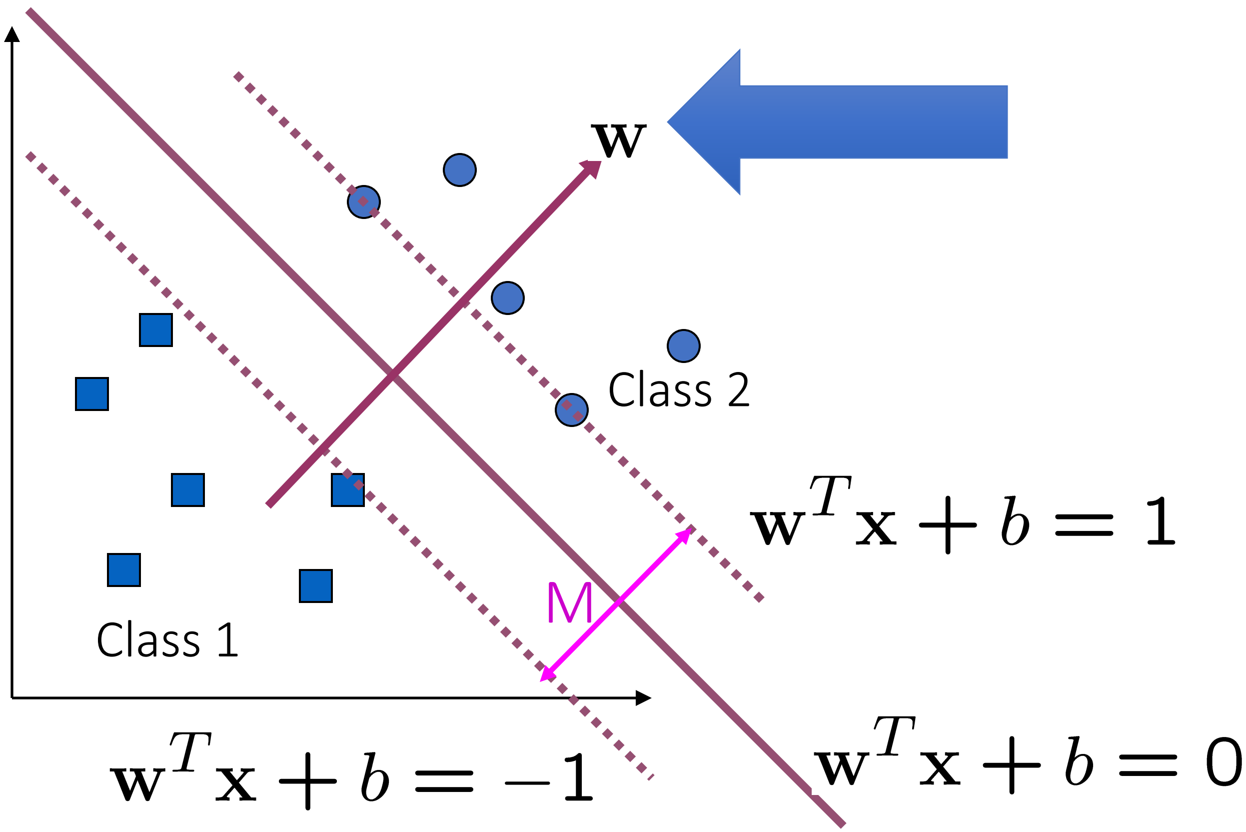
$w^T(u-v)=0$

$$\begin{pmatrix} -2 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = 0$$

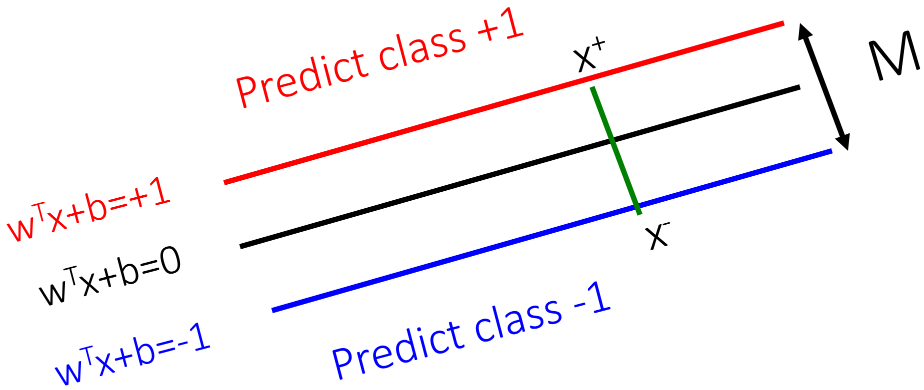
Maximizing the margin: observation-1

- Observation 1: the vector w is orthogonal to the +1 plane

-1 plane
0 -plane



Maximizing the margin: observation-2



Classify as +1 if $w^Tx+b >= 1$

Classify as -1 if $w^Tx+b <= -1$

Undefined if $-1 < w^Tx+b < 1$

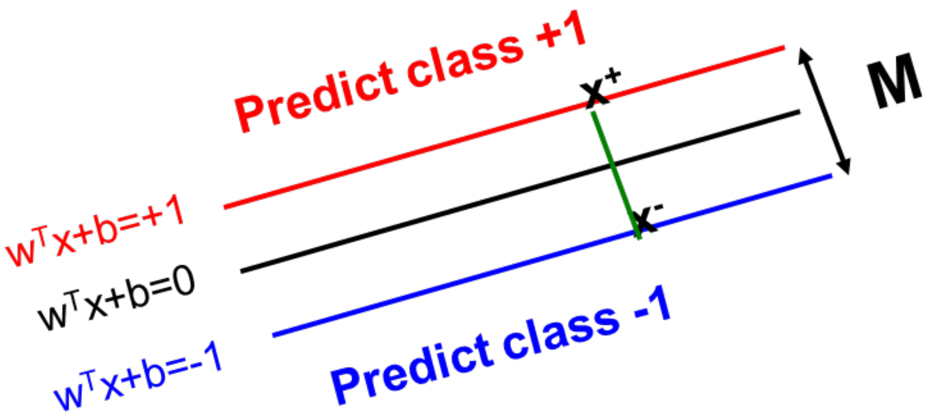
- Observation 1: the vector w is orthogonal to the +1 and -1 planes
- Observation 2: if x^+ is a point on the +1 plane and x^- is the closest point to x^+ on the -1 plane then

$$x^+ = \lambda w + x^-$$

Since w is orthogonal to both planes we need to 'travel' some distance along w to get from x^+ to x^-

Concrete derivations of M in Extra slides when X is 2D

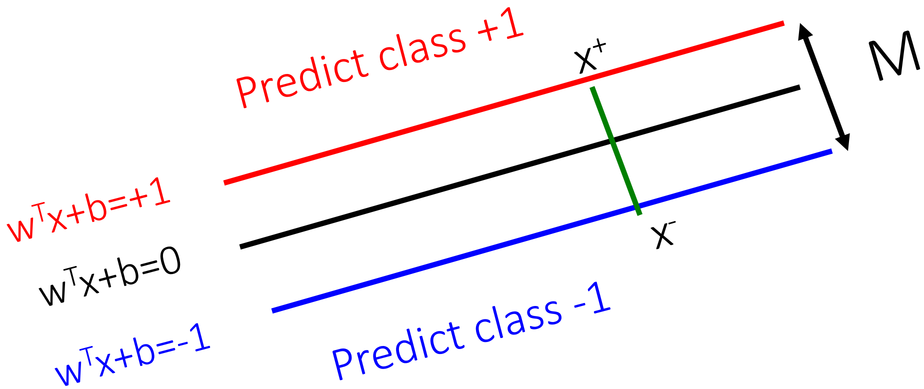
Putting it together



- $w^T x^+ + b = +1$
- $w^T x^- + b = -1$
- $M = | x^+ - x^- | = ?$
- $x^+ = \lambda w + x^-$

Concrete derivations of M in Extra slides when X is 2D

Putting it together



- $w^T x^+ + b = +1$
- $w^T x^- + b = -1$
- $x^+ = \lambda w + x^-$
- $| x^+ - x^- | = M$

We can now define M in terms of w and b

$$M = |x^+ - x^-|$$

$$= |\lambda w|$$

$$= \lambda |w|$$

$$= \lambda \sqrt{w^T w}$$

$$= \frac{2}{w^T w} \sqrt{w^T w}$$

$$\left[= \frac{2}{\sqrt{w^T w}} \right]$$

$$w^T x^+ + b = 1$$

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

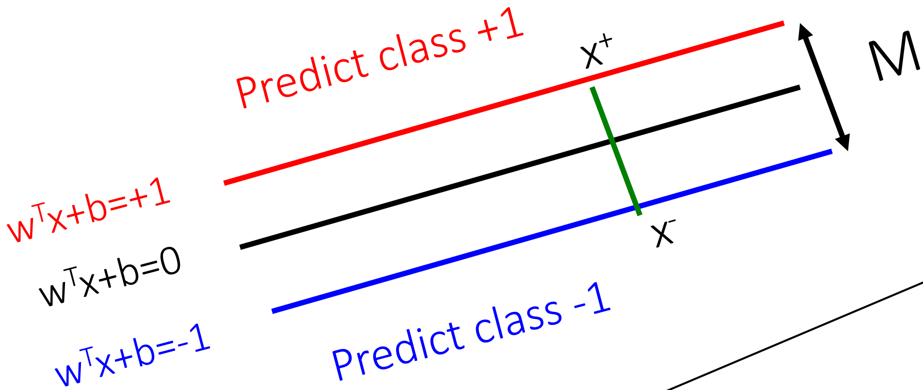
$$\lambda w^T w + \underbrace{w^T x^- + b}_{\Rightarrow -1} = 1$$

$$\lambda w^T w = 2$$

$$\Rightarrow \quad \lambda = \frac{2}{w^T w}$$

Concrete derivations of M in Extra slides when X is 2D

Putting it together



- $w^T x^+ + b = +1$
- $w^T x^- + b = -1$
- $x^+ = \lambda w + x^-$
- $| x^+ - x^- | = M$

$w^T x^+ + b = +1$

=>

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

=>

$w^T x^- + b + \lambda w^T w = +1$

=>

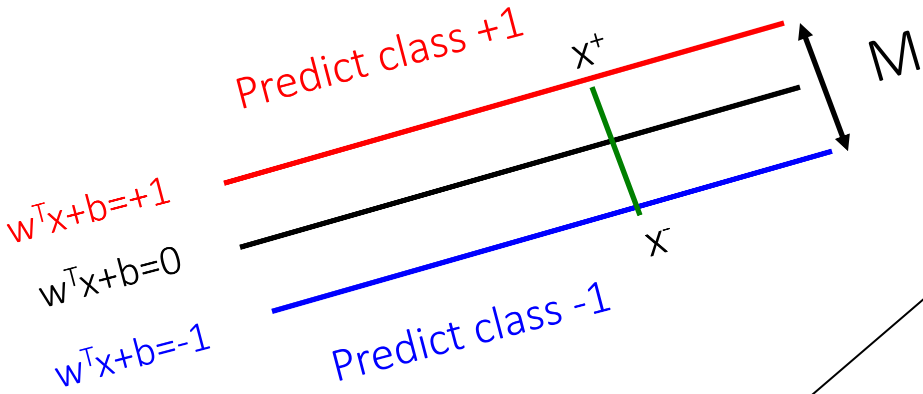
$-1 + \lambda w^T w = +1$

=>

$\lambda = 2/w^T w$

We can now define M in terms of w and b

Putting it together



- $w^T x^+ + b = +1$
- $w^T x^- + b = -1$
- $x^+ = \lambda w + x^-$
- $| x^+ - x^- | = M$
- $\lambda = 2/w^T w$

$M = |x^+ - x^-|$

=>

$$\mathbf{M} = |\lambda \mathbf{w}| = \lambda |\mathbf{w}| = \lambda \sqrt{\mathbf{w}^T \mathbf{w}}$$

=>

$$\mathbf{M} = 2 \frac{\sqrt{\mathbf{w}^T \mathbf{w}}}{\mathbf{w}^T \mathbf{w}} = \frac{2}{\sqrt{\mathbf{w}^T \mathbf{w}}}$$

$\max(M)$

$\Rightarrow \min(w^T w)$

We can now define M in terms of w and b