

# Do not distribute course material

You may not and may not allow others to reproduce or distribute lecture notes and course materials publicly whether or not a fee is charged.

# Lecture Support Vector Machines

- <http://cs229.stanford.edu/notes/cs229-notes3.pdf>
- <https://www.svm-tutorial.com/>
- <https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-the-linearly-separable-case-1.html>
- Advanced: [https://svmtutorial.online/download.php?file=SVM\\_tutorial.pdf](https://svmtutorial.online/download.php?file=SVM_tutorial.pdf)

PROF. LINDA SELLIE

SOME SLIDES FROM PROF. RANGAN

SOME APPROACHES TAKEN ARE FROM CSU 18-661

# Learning objectives:

- ❑ Understand the idea behind the geometric margin, functional margin, and canonical weights
- ❑ Create an objective function for to find the hyperplane with the largest margin for linearly separable data
- ❑ Understand the hinge loss penalty
- ❑ Modify the objective function to allow for non-linearly separable data
- ❑ Understand the the trade off between the two terms in the soft margin objective function
- ❑ Know how to create a kernel function
- ❑ Understand the importance of the kernel function
- ❑ Able to use a the dual hypothesis for prediction
- ❑ Know which vectors are support vectors

# MNIST Digit Classification

## HANDWRITING SAMPLE FORM

NAME [REDACTED] DATE 8-3-89 CITY MINDEN CITY STATE MI ZIP 48456

This sample of handwriting is being collected for use in testing computer recognition of hand printed numbers and letters. Please print the following characters in the boxes that appear below.

0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
0123456789	0123456789	0123456789
87	701	3752
87	701	5752
158	4586	32123
158	4586	32123
7481	80539	419219
7481	80539	419219
61738	729658	75
61738	729658	75

- ❑ Problem: Recognize hand-written digits
- ❑ Originally problem:
  - Census forms
  - Automated processing
- ❑ Classic machine learning problem
- ❑ Benchmark

From Patrick J. Grother, NIST Special Database, 1995

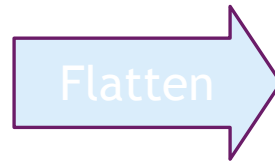
0 0 0 0 0 0 0 0 0 0  
1 1 1 1 1 1 1 1 1 1  
2 2 2 2 2 2 2 2 2 2  
3 3 3 3 3 3 3 3 3 3  
4 4 4 4 4 4 4 4 4 4  
5 5 5 5 5 5 5 5 5 5  
6 6 6 6 6 6 6 6 6 6  
7 7 7 7 7 7 7 7 7 7  
8 8 8 8 8 8 8 8 8 8  
9 9 9 9 9 9 9 9 9 9



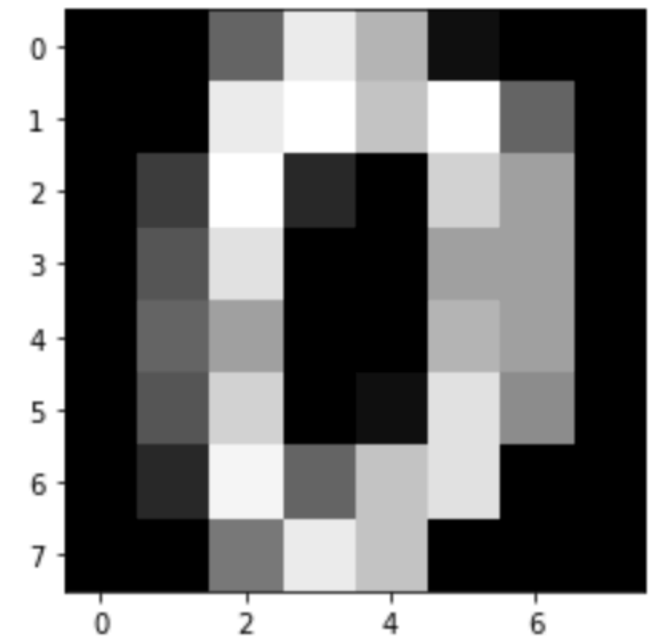
❑ What does one example look like?

❑ Images can be represented as 2D matrices or 1D vectors

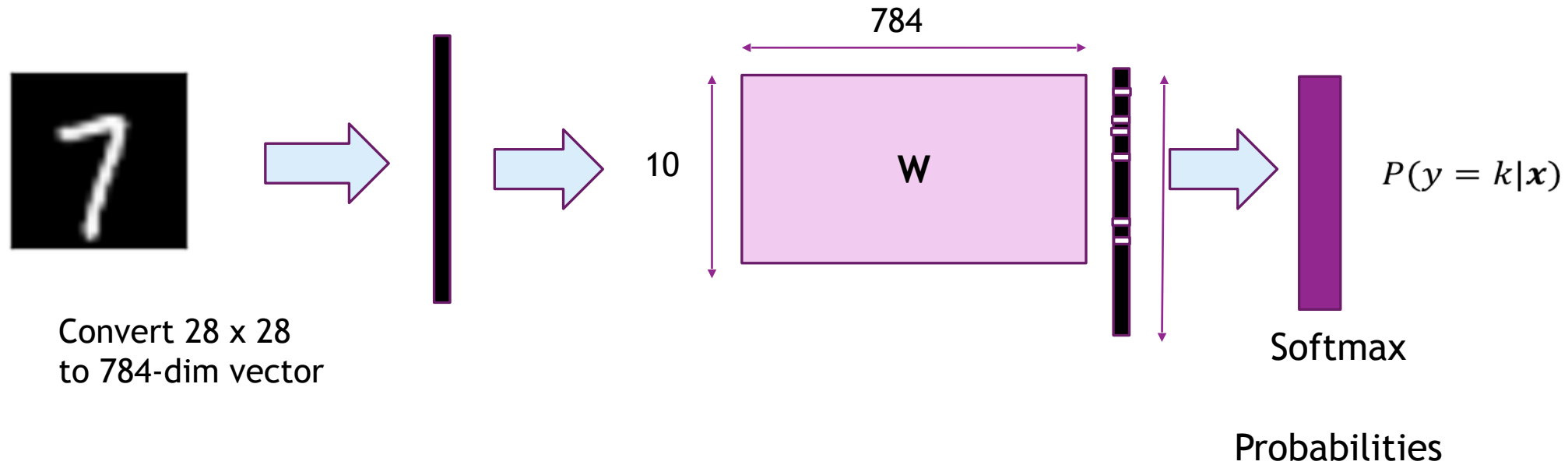
```
[[ 0.  0.  5. 13.  9.  1.  0.  0.]  
 [ 0.  0. 13. 15. 10. 15.  5.  0.]  
 [ 0.  3. 15.  2.  0. 11.  8.  0.]  
 [ 0.  4. 12.  0.  0.  8.  8.  0.]  
 [ 0.  5.  8.  0.  0.  9.  8.  0.]  
 [ 0.  4. 11.  0.  1. 12.  7.  0.]  
 [ 0.  2. 14.  5. 10. 12.  0.  0.]  
 [ 0.  0.  6. 13. 10.  0.  0.  0.]]
```



```
[[ 0.]  
 [ 0.]  
 [ 5.]  
 [13.]  
 [ 9.]  
 [ 1.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [13.]  
 [15.]  
 [10.]  
 [15.]  
 [ 5.]  
 [ 0.]  
 [ 0.]  
 [ 3.]  
 [15.]  
 [ 2.]  
 [ 0.]  
 [11.]  
 [ 8.]  
 [ 0.]  
 [ 0.]  
 [ 4.]  
 [12.]  
 [ 0.]  
 [ 0.]  
 [ 8.]  
 [ 8.]  
 [ 0.]  
 [ 0.]  
 [ 5.]  
 [ 8.]  
 [ 0.]  
 [ 0.]  
 [ 9.]  
 [ 8.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [ 4.]  
 [11.]  
 [ 0.]  
 [ 1.]  
 [12.]  
 [ 7.]  
 [ 0.]  
 [ 0.]  
 [ 2.]  
 [14.]  
 [ 5.]  
 [10.]  
 [12.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [ 6.]  
 [13.]  
 [10.]  
 [ 0.]  
 [ 0.]  
 [ 0.]  
 [ 0.]]
```



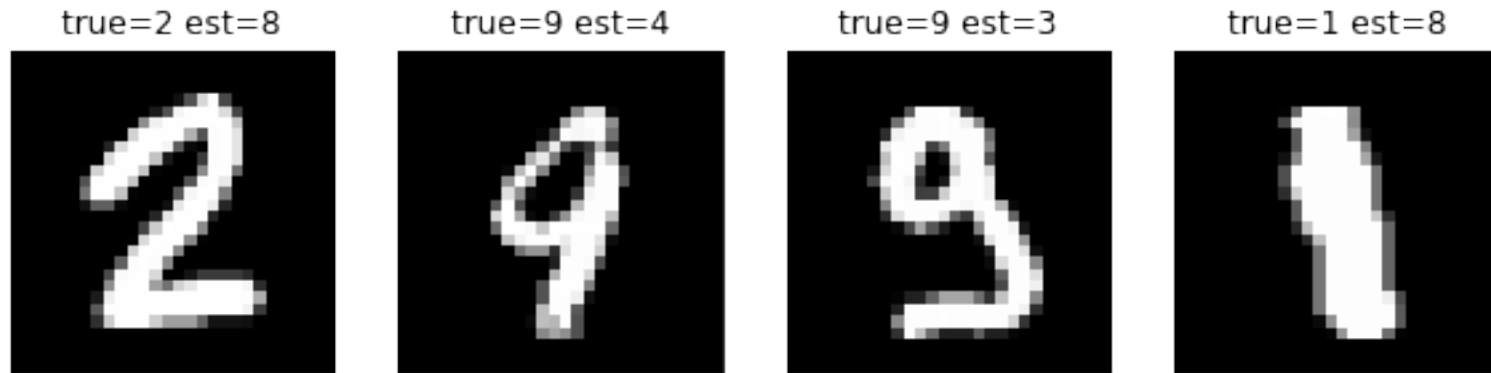
# Recap: Logistic Classifier



- Will select  $\hat{y} = \arg \max_k P(y = k|x) = \arg \max_k z_k$ 
  - Output  $z_k$  which is largest
- When is  $z_k$  large?

# Try a Logistic Classifier Performance

- Accuracy = 93%. (Or around 89-91% if using a smaller amount of data)
- Can we do better?
- Some of the errors





# MNIST: Widely-Used Benchmark

---

- ❑ We will look at SVM today
- ❑ Not the best algorithm for handwritten digit. See the result of different approaches here: <http://yann.lecun.com/exdb/mnist/>
- ❑ But quite good 98.5% accuracy (or around 93% on a smaller training set)
- ❑ ...and illustrates the main points

On the small dataset we can transform the features and get better performance!

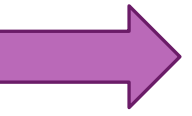
# SVM:

- ❑ Scales better with high-dimensional data
- ❑ Generalizes well to many nonlinear models

# Learning objectives:

- ❑ Understand the idea behind the geometric margin, functional margin, and canonical weights
- ❑ Create an objective function for to find the hyperplane with the largest margin for linearly separable data
- ❑ Understand the hinge loss penalty
- ❑ Modify the objective function to allow for non-linearly separable data
- ❑ Understand the trade off between the two terms in the soft margin objective function
- ❑ Know how to create a kernel function
- ❑ Understand the importance of the kernel function
- ❑ Able to use the dual hypothesis for prediction
- ❑ Know which vectors are support vectors

# Outline

- 
- Notation change, intuition, and finding how to compare hyperplanes - **mathematically how do compare hyperplanes to find the one with the maximum margin. Can we turn this way of comparing hyperplanes into an objective function**
  - Support vector machines
    - ★ hard margin - **find the constrained objective function when the data is linearly separable**
    - ★ Dealing with non-linear data - “Soft” margins for SVM - **New constrained objective function for the case where the data is not linearly separable**
    - ★ Pegasos algorithm. **Optimizer for soft margin SVM**
    - ★ dual formula - **a clever trick**  $g(\mathbf{x}) = \text{sign} \left( w_0 + \sum_{i \in I} \alpha^{(i)} y^{(i)} \mathbf{x}^{(i)T} \mathbf{x} \right)$
    - ★ Dealing with non-linear data - feature transformation with the kernel trick - **Show two popular feature maps**

# New Notation

Previously we used

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix} \quad x^{(i)} = \begin{bmatrix} 1 \\ x_1^{(i)} \\ \vdots \\ x_d^{(i)} \end{bmatrix}$$

$$y \in \{0,1\}$$

This lecture, we separate the intercept term from the other weights. The mathematics of this lecture makes easier. We change notation to make this clearer.

$$w_0 \quad \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix} \quad x^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_d^{(i)} \end{bmatrix}$$

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

$$g(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + w_0) = \begin{cases} +1 & \text{if } \mathbf{w}^T \mathbf{x} + w_0 > 0 \\ -1 & \text{if } \mathbf{w}^T \mathbf{x} + w_0 < 0 \end{cases}$$

# What is a hyperplane?

In  $p$ -dimensions, a hyperplane is a flat subspace of dimension  $p-1$

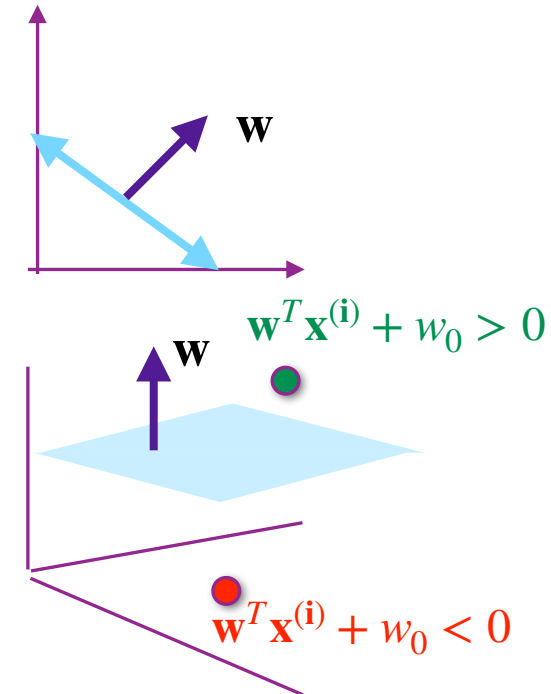
The mathematical definition of a hyperplane:  $\forall \mathbf{x}^{(i)}, w_0 + \mathbf{w}^T \mathbf{x}^{(i)} = 0$

In 2-dimensions, the hyperplane is a line

In 2-dimensions, the hyperplane is defined by  $\forall \mathbf{x}^{(i)}, w_0 + w_1 x_1^{(i)} + w_2 x_2^{(i)} = 0$

In 3-dimensions, the hyperplane is a plane

In 3-dimensions, the hyperplane is defined by  $\forall \mathbf{x}^{(i)}, w_0 + w_1 x_1^{(i)} + w_2 x_2^{(i)} + w_3 x_3^{(i)} = 0$



# We use the hyperplane to classify a point $x$

Predict 1 if  $\mathbf{w}^T \mathbf{x} + w_0 > 0$

Predict -1 if  $\mathbf{w}^T \mathbf{x} + w_0 < 0$

The hyperplane is defined by all the points that satisfy

$$(3,4)\mathbf{x} - 10 = 0$$

e.g.  $(3,4)(2,1)^T - 10 = 0$

$$(3,4)(0,2.5)^T - 10 = 0$$

All the points above the line are positive

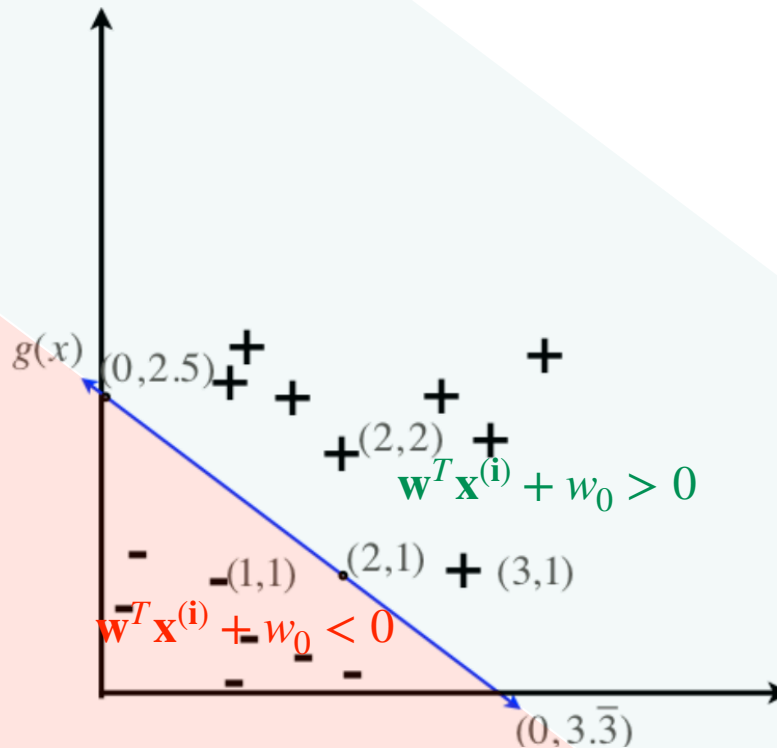
$$(3,4)\mathbf{x} - 10 > 0$$

e.g.  $(3,4)(2,2)^T - 10 = 4$

All the points below the line are negative

$$(3,4)\mathbf{x} - 10 < 0$$

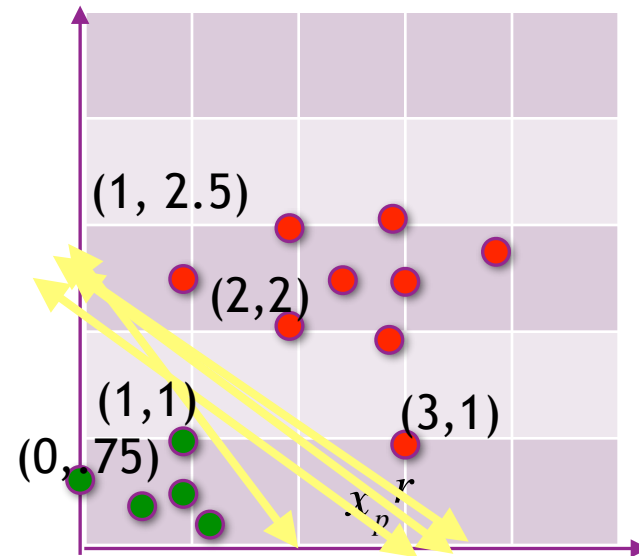
e.g.  $(3,4)(1,1)^T - 10 = -3$



## Predicting using a hyperplane

*Predict 1 if  $\mathbf{w}^T \mathbf{x} + w_0 > 0$*

*Predict -1 if  $\mathbf{w}^T \mathbf{x} + w_0 < 0$*

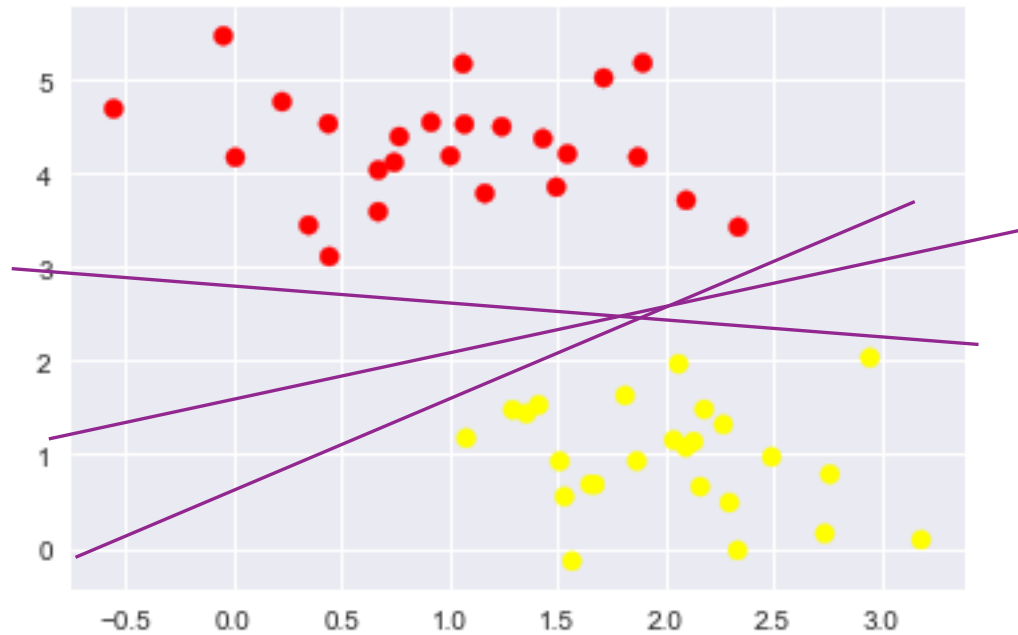


Which hyperplane?



# Suppose there is a hyperplane that separates the training data

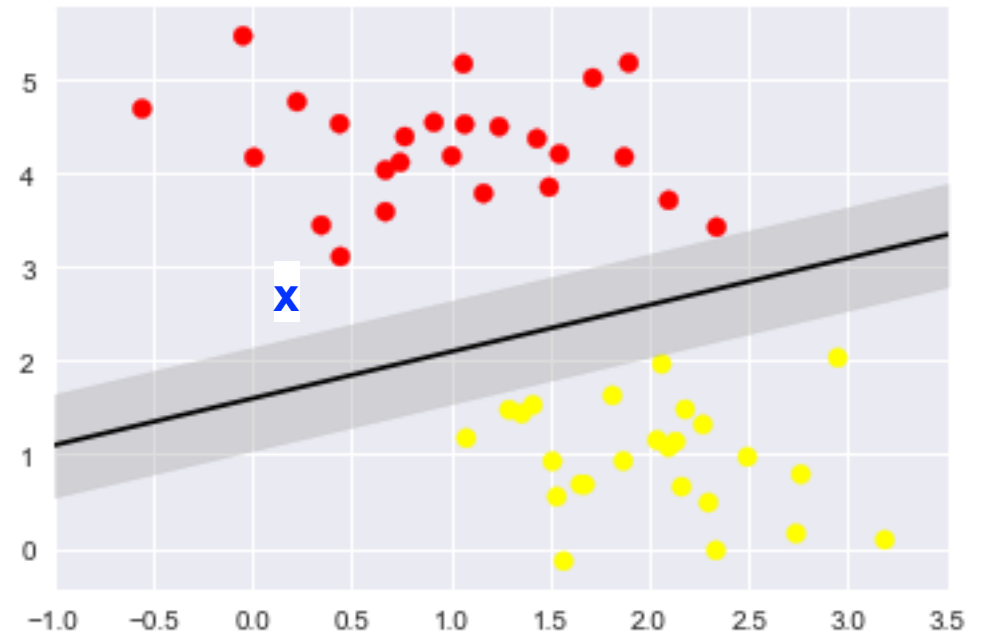
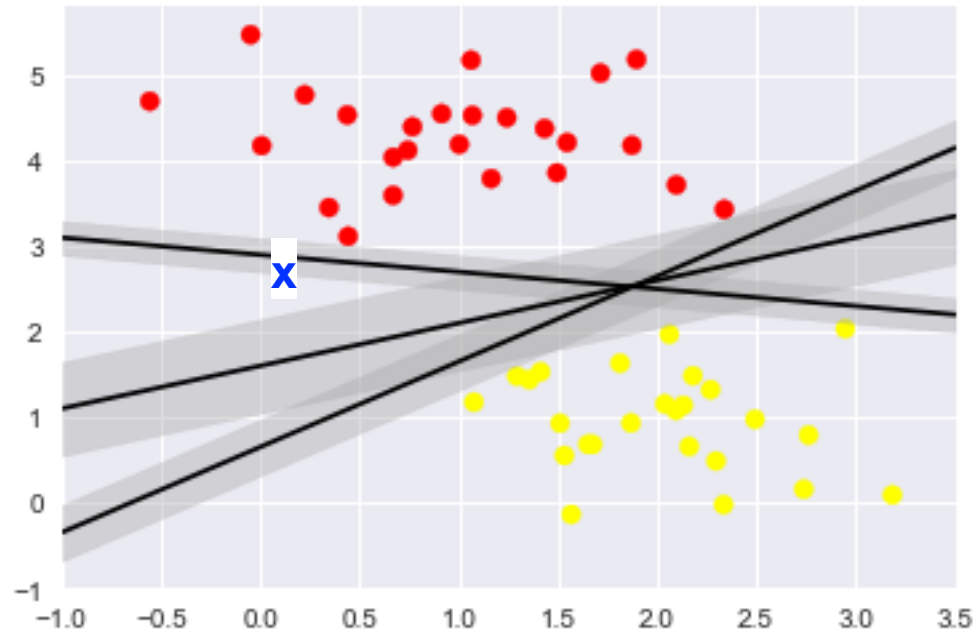
Which hyperplane is the “best”?



# Which line is best? Why????

**Which line should we use?**

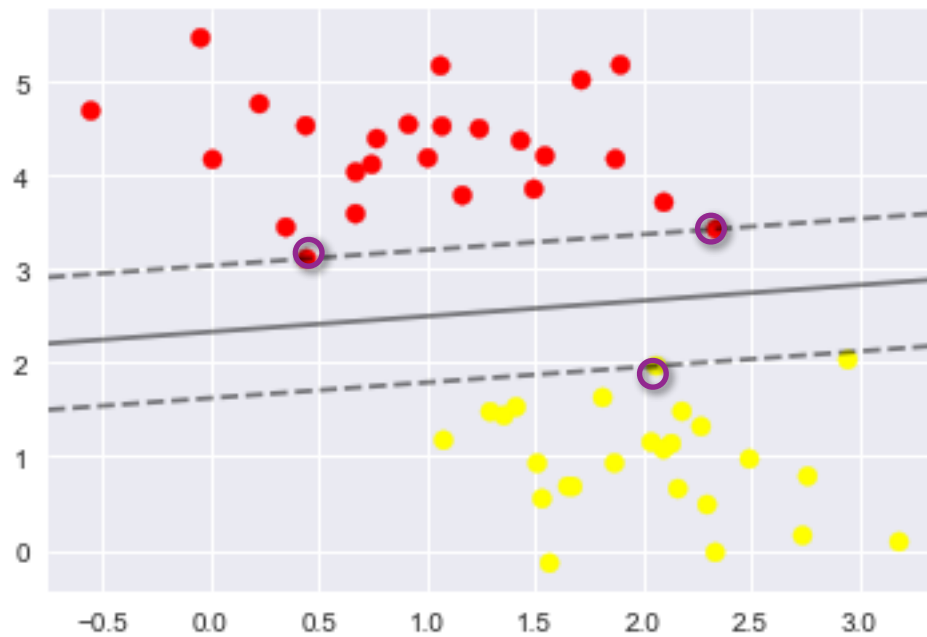
**How should we classify a new point  $x$ ?**



**The one that makes intuitive sense is the line that has the maximum margin**

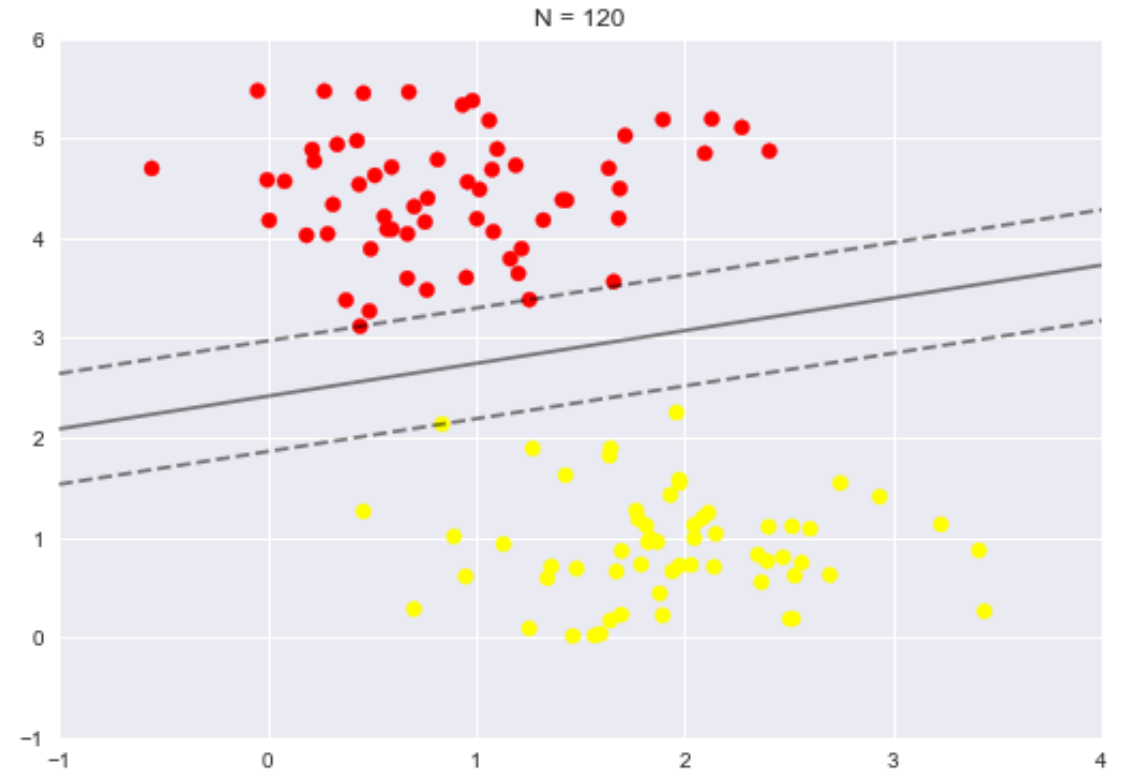
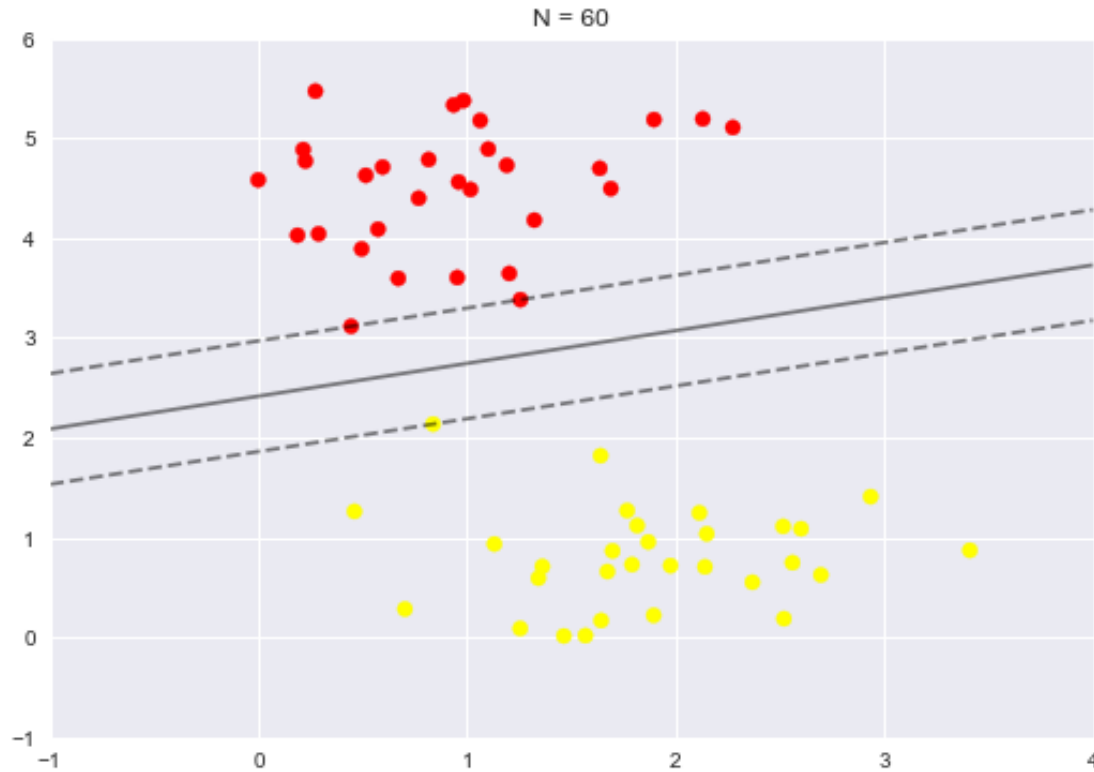
**Now we can feel more comfortable predicting the point  $x$**

**These are the points that prevent a larger margin**



**Note that only the points on the margin affect the decision boundary.**

Changing by adding, deleting points outside the margin will not affect the hyperplane



# Outline

□ Notation change, intuition, and finding how to compare hyperplanes - **mathematically how do compare hyperplanes to find the one with the maximum margin. Can we turn this way of comparing hyperplanes into an objective function**

□ Support vector machines



★ hard margin - **find the constrained objective function when the data is linearly separable**

★ Dealing with non-linear data - “Soft” margins for SVM - **New constrained objective function for the case where the data is not linearly separable**

★ Pegasos algorithm. **Optimizer for soft margin SVM**

★ dual formula - **a clever trick**  $g(\mathbf{x}) = \text{sign} \left( w_0 + \sum_{i \in I} \alpha^{(i)} y^{(i)} \mathbf{x}^{(i)T} \mathbf{x} \right)$

★ Dealing with non-linear data - feature transformation with the kernel trick - **Show two popular feature maps**

In the hard margin case, we will  
assume the data is linearly separable

How can we turn our intuition into an objective function?

Let us start by finding a way to compare the hyperplanes mathematically.

Our idea is that the best hyperplane has the largest margin.



Math background:

Shortest distance of a point to a hyperplane

# Another way to describe a point

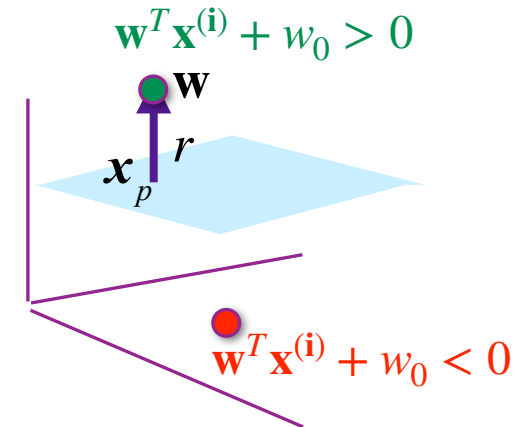
$$\mathbf{w} = [w_1, w_2, \dots, w_d]^T$$

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|_2}$$

$$\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2 + \dots + w_d^2} \text{ is the length of the vector } \mathbf{w}$$

$\mathbf{w}/\|\mathbf{w}\|_2$  converts  $\mathbf{w}$  into a unit vector. E.g.  $(3,4)^T/5 = (3/5, 4/5)$

$$\forall \mathbf{x}^{(i)}, w_0 + w_1 x_1^{(i)} + w_2 x_2^{(i)} + w_3 x_3^{(i)} = 0$$



$\mathbf{w} = [w_1, w_2, \dots, w_d]^T$      $\|\mathbf{w}\|_2^2 = w_1^2 + w_2^2 + \dots + w_d^2$      $\mathbf{w}/\|\mathbf{w}\|_2$  converts  $\mathbf{w}$  into a unit vector. E.g.  $(3,4)^T/5 = (3/5, 4/5)$

$\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2 + \dots + w_d^2}$  is the length of the vector  $\mathbf{w}$

## Computing the signed distance from a point to the hyperplane

$\mathbf{x}$ , a point

$\mathbf{x}_p$ , the normal projection of  $\mathbf{x}$  onto  $\mathbf{w}$ ,

Note that

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|_2}$$

$$z(\mathbf{x}) = w_0 + \mathbf{w}^T \mathbf{x} = w_0 + \mathbf{w}^T \left( \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|_2} \right)$$

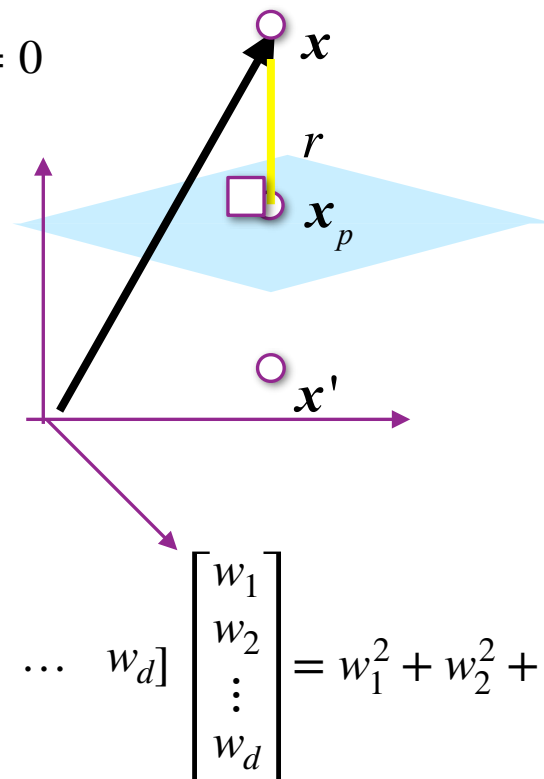
$$= \underbrace{w_0 + \mathbf{w}^T \mathbf{x}_p}_{= r \|\mathbf{w}\|_2} + \mathbf{w}^T r \frac{\mathbf{w}}{\|\mathbf{w}\|_2}$$

observe that  $\mathbf{w}^T \mathbf{w} = \|\mathbf{w}\|_2^2 = [w_1 \ w_2 \ \dots \ w_d] \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix} = w_1^2 + w_2^2 + \dots + w_d^2$

Consequently:  $r = \frac{z(\mathbf{x})}{\|\mathbf{w}\|_2}$

Note that  $r$  can be positive or negative depending on which side of the hyperplane  $\mathbf{x}$  lies

For any  $\mathbf{x}$ ,  $\mathbf{w}^T \mathbf{x} + w_0 = 0$



$\mathbf{w} = [w_1, w_2, \dots, w_d]^T$      $\|\mathbf{w}\|_2^2 = w_1^2 + w_2^2 + \dots + w_d^2$      $\mathbf{w}/\|\mathbf{w}\|_2$  converts  $\mathbf{w}$  into a unit vector. E.g.  $(3,4)^T/5 = (3/5, 4/5)$

$\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2 + \dots + w_d^2}$  is the length of the vector  $\mathbf{w}$

## Computing the signed distance from a point to the hyperplane

$\mathbf{x}$ , a point

$\mathbf{x}_p$ , the normal projection of  $\mathbf{x}$  onto  $\mathbf{w}$ ,

Note that

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|_2}$$

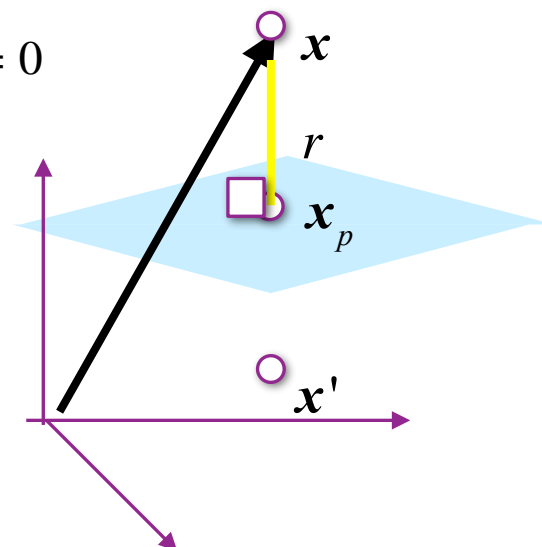
$$z(\mathbf{x}) = w_0 + \mathbf{w}^T \mathbf{x} = w_0 + \mathbf{w}^T \left( \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|_2} \right)$$

$$= w_0 + \mathbf{w}^T \mathbf{x}_p + \mathbf{w}^T r \frac{\mathbf{w}}{\|\mathbf{w}\|_2}$$

$$= r \|\mathbf{w}\|_2$$

observe that  $\mathbf{w}^T \mathbf{w} = \|\mathbf{w}\|_2^2 = [w_1 \ w_2 \ \dots \ w_d] \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix} = w_1^2 + w_2^2 + \dots + w_d^2$

For any  $\mathbf{x}$ ,  $\mathbf{w}^T \mathbf{x} + w_0 = 0$



Consequently:  $r = \frac{z(\mathbf{x})}{\|\mathbf{w}\|_2}$

**unsigned** distance  $\frac{|\mathbf{w}^T \mathbf{x} + w_0|}{\|\mathbf{w}\|_2}$  of  $\mathbf{x}$  to the hyperplane  $\mathbf{w}^T \mathbf{x} + w_0 = 0$

# Geometric Margin

Signed distance point to hyperplane:  $\frac{(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)}{\|\mathbf{w}\|_2}$

geometric margin of a point:  $\gamma^{(i)} = \frac{y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)}{\|\mathbf{w}\|_2}$

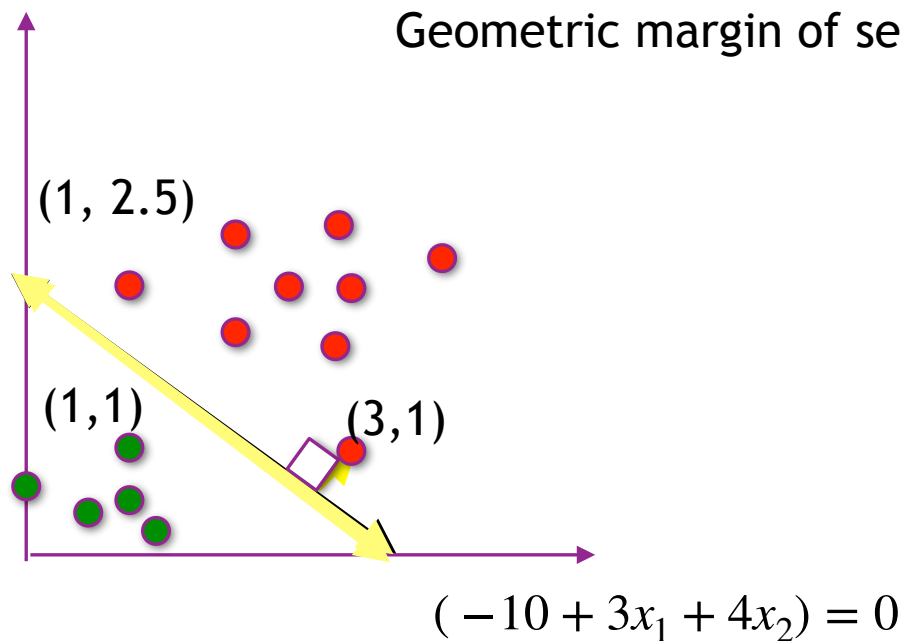
Geometric margin of set:  $\gamma = \min\{\gamma^{(1)}, \gamma^{(2)}, \dots, \gamma^{(N)}\}$

$$= \min_i \frac{y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)}{\|\mathbf{w}\|_2}$$

$$\begin{aligned} \gamma^{(1)} &= (1) \left( [3 \ 4] \begin{bmatrix} 1 \\ 2.5 \end{bmatrix} - 10 \right) / \sqrt{3^2 + 4^2} \\ &= (1)(3)/5 \end{aligned}$$

$$\begin{aligned} \gamma^{(2)} &= (1) \left( [3 \ 4] \begin{bmatrix} 3 \\ 1 \end{bmatrix} - 10 \right) / \sqrt{3^2 + 4^2} \\ &= (1)(3)/5 \end{aligned}$$

$$\begin{aligned} \gamma^{(N)} &= (-1) \left( [3 \ 4] \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 10 \right) / \sqrt{3^2 + 4^2} \\ &= (-1)(-3)/5 \end{aligned}$$



if  $\|\mathbf{w}\|_2 = 1$  i.e  $\mathbf{w} := \mathbf{w} / \|\mathbf{w}\|_2$  we don't need to divide by  $\|\mathbf{w}\|_2$

Goal find hyperplane ( $\|\mathbf{w}\|_2 = 1$ ) which has the largest  $\gamma$  (i.e.)  $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)}) = \gamma^{(i)} \geq \gamma$

# Which hyperplane has a larger margin:

- Given the following data:

$$\mathbf{x}^{(1)} = \begin{bmatrix} 3.2 & 4.7 \end{bmatrix}$$

$$y^{(1)} = -1$$

$$\mathbf{x}^{(2)} = \begin{bmatrix} 3.5 & 1.4 \end{bmatrix}$$

$$y^{(2)} = 1$$

$$\mathbf{x}^{(3)} = \begin{bmatrix} 3. & 1.4 \end{bmatrix}$$

$$y^{(3)} = 1$$

- What is the geometric margin for:

$$w_0 = 1/2 \quad \mathbf{w} = \begin{bmatrix} 2/3 \\ -1 \end{bmatrix}$$

$$\underbrace{y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) / \|\mathbf{w}\|_2}_{(-1) \left( \begin{bmatrix} 2/3 & -1 \end{bmatrix} \begin{bmatrix} 3.2 \\ 4.7 \end{bmatrix} + 1/2 \right) / \sqrt{4/9 + 1}} = 1.7$$

$$(1) \left( \begin{bmatrix} 2/3 & -1 \end{bmatrix} \begin{bmatrix} 3.5 \\ 1.4 \end{bmatrix} + 1/2 \right) / \sqrt{4/9 + 1} = 1.2$$

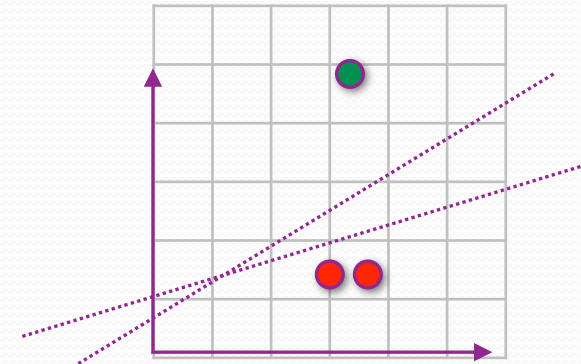
$$(1) \left( \begin{bmatrix} 2/3 & -1 \end{bmatrix} \begin{bmatrix} 3 \\ 1.4 \end{bmatrix} + 1/2 \right) / \sqrt{4/9 + 1} = 0.9$$

$\gamma = 0.9$  ✓

Goal is to find  $\mathbf{w}, w_0$  that has the largest  $\gamma$  such that  $y^{(i)}(\mathbf{w}^T \mathbf{x} + w_0) / \|\mathbf{w}\|_2 \geq \gamma$

$$y^{(i)}(\mathbf{w}^T \mathbf{x} + w_0) / \|\mathbf{w}\|_2 \geq 0.9$$

$$y^{(i)}(\mathbf{w}^T \mathbf{x} + w_0) / \|\mathbf{w}\|_2 \geq 0.5$$



$$w_0 = 1 \quad \mathbf{w} = \begin{bmatrix} 1/3 \\ -1 \end{bmatrix}$$

$$(-1) \left( \begin{bmatrix} 1/3 & -1 \end{bmatrix} \begin{bmatrix} 3.2 \\ 4.7 \end{bmatrix} + 1 \right) / \sqrt{4/9 + 1} = 2.2$$

$$(1) \left( \begin{bmatrix} 1/3 & -1 \end{bmatrix} \begin{bmatrix} 3.5 \\ 1.4 \end{bmatrix} + 1 \right) / \sqrt{4/9 + 1} = 0.6$$

$$(1) \left( \begin{bmatrix} 1/3 & -1 \end{bmatrix} \begin{bmatrix} 3 \\ 1.4 \end{bmatrix} + 1 \right) / \sqrt{4/9 + 1} = 0.5$$

$\gamma = 0.5$

# Objective function

Goal

$$\max_{w_0, \mathbf{w}} \gamma$$

subject to  $y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq \gamma$  for all  $i=1, \dots, N$

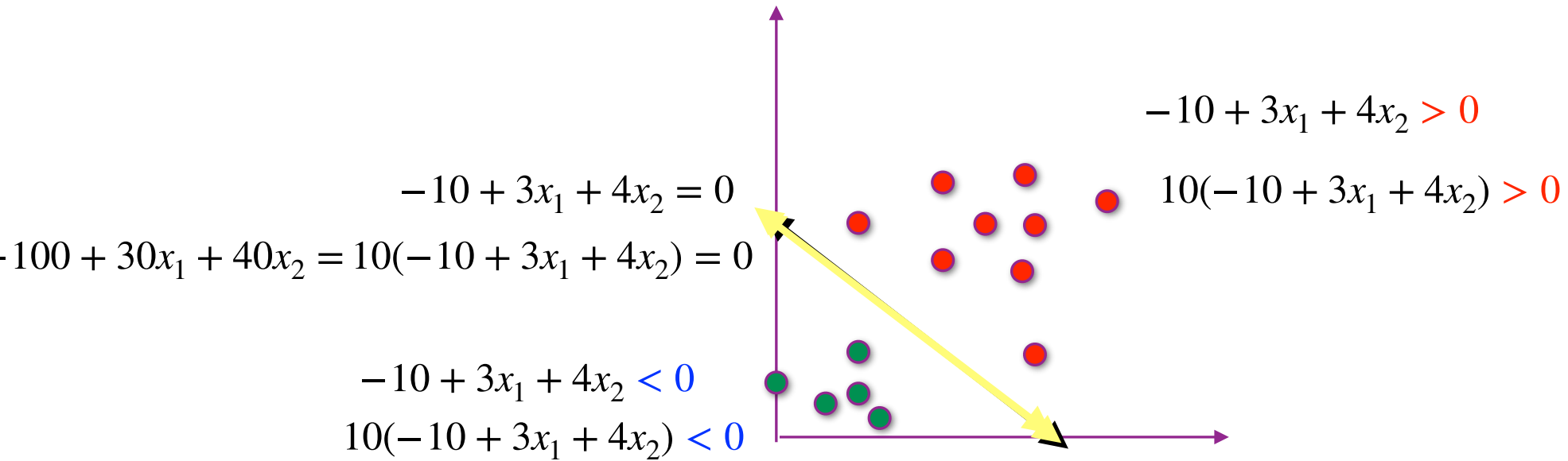
$$\|\mathbf{w}\|_2 = 1$$

Not yet the  
form we need

Difficult to work with constraints that are not linear. Let us write our objective function in a different way.



Pair share: Do we change the classification if we multiply  $-10 + 3x_1 + 4x_2 = 0$  by 10?



Rescaling the parameters doesn't change the line (decision boundary)!

$$\mathbf{w}^T \mathbf{x} + w_0 = 0 = c\mathbf{w}^T \mathbf{x} + cw_0$$

# What is another way to constrain the problem so that we get a unique solution?

$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) = 1$  for the closest point to the hyperplane

Functional margin of  $(\mathbf{w}, w_0)$  with respect to a point  $\mathbf{x}^{(i)}$  is

$$\gamma^{(i)} = y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0)$$

*Functional margin* of  $(\mathbf{w}, w_0)$  with respect to a set  $S$  is

$$\gamma = \min\{\gamma^{(1)}, \gamma^{(2)}, \dots, \gamma^{(N)}\}$$

For any hyperplane that separates the data we can make its functional margin any value we want.

How can we make the functional margin 1?

$$\min_i y^{(i)}(\mathbf{w}^T \mathbf{x} + w_0) = 1$$

Pair share: given a hyperplane  $\mathbf{w} = (3,4)^T$ ,  $w_0 = -10$  which has a functional margin of 3, rescale the parameters so the functional margin is 1

# Functional Margin

Functional margin of  $(\mathbf{w}, w_0)$  with respect to a point  $\mathbf{x}^{(i)}$  is

$$\gamma^{(i)} = y^{(i)} \left( \frac{\mathbf{w}^T \mathbf{x}^{(i)}}{3} + \frac{w_0}{3} \right)$$

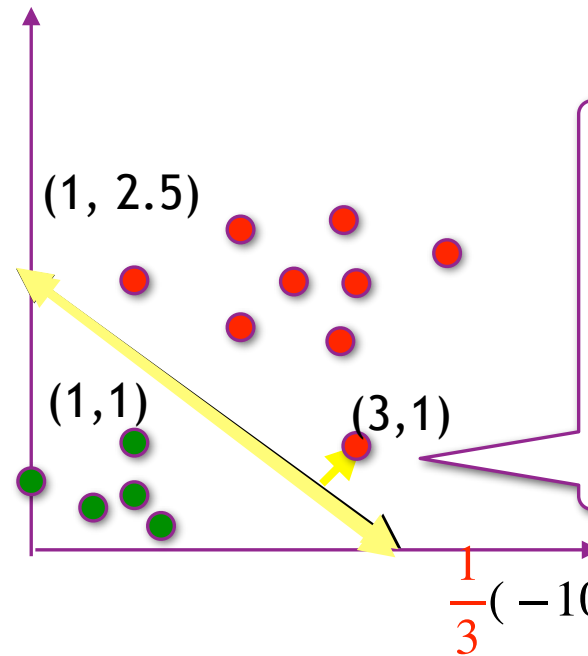
Functional margin of  $(\mathbf{w}, w_0)$  with respect to a set  $S$  is

$$\gamma = \min \{ \gamma^{(1)}, \gamma^{(2)}, \dots, \gamma^{(N)} \}$$

$$\begin{aligned} \gamma^{(2)} &= (1) \left( \frac{[3 \ 4]}{3} \begin{bmatrix} 3 \\ 1 \end{bmatrix} - \frac{10}{3} \right) \\ &= (1) \frac{(3)}{3} \end{aligned}$$

$$\begin{aligned} \gamma^{(1)} &= (1) \left( \frac{[3 \ 4]}{3} \begin{bmatrix} 1 \\ 2.5 \end{bmatrix} - \frac{10}{3} \right) \\ &= (1) \frac{(3)}{3} \end{aligned}$$

$$\begin{aligned} \gamma^{(N)} &= (-1) \left( \frac{[3 \ 4]}{3} \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \frac{10}{3} \right) \\ &= (-1) \frac{(-3)}{3} \end{aligned}$$



After scaling, the points closest to the decision boundary have:

\* functional margin of 1

\* Euclidean distance of  $\frac{1}{\|\mathbf{w}\|_2}$

$$-10 + 3x_1 + 4x_2 = 0 = -10/3 + 3/3x_1 + 4/3x_2$$

We can make  $\gamma = 1$   
The canonical weights

Next: Many equivalent versions of  
our objective function

# Hard-Margin SVM

Constrained optimization problem:

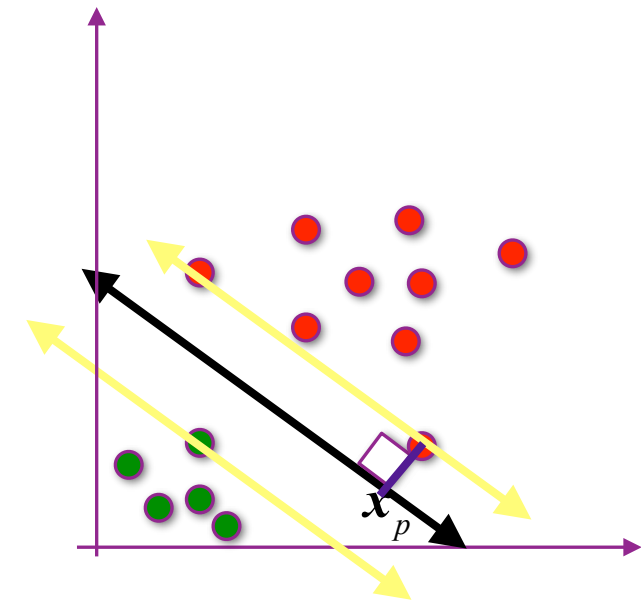
$$\begin{aligned} & \max_{w_0, \mathbf{w}} \gamma \\ & \text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq \gamma \text{ for all } i=1, \dots, N \\ & \|\mathbf{w}\|_2 = 1 \end{aligned}$$

$$\frac{\gamma}{\|\mathbf{w}\|_2} = \text{Geometric margin}$$

Another formulation:

$$\max_{w_0, \mathbf{w}} \frac{\gamma}{\|\mathbf{w}\|_2} = r$$

$$\text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq \gamma \text{ for all } i=1, \dots, N$$



Idea: we can rescale our margin to anything we want by rescaling our coefficients

notice that  $\max \gamma / \|\mathbf{w}\|_2$  equals  $\max(\gamma / \gamma) / \|\mathbf{w} / \gamma\|_2$  subject to  $y^{(i)}(w_0 / \gamma + \mathbf{w}^T / \gamma \mathbf{x}^{(i)}) \geq \gamma / \gamma$  for all  $i=1, \dots, N$

We set  $w_0 := w_0 / \gamma$ , and  $\mathbf{w} := \mathbf{w} / \gamma$  Notice we now want to  $\max 1 / \|\mathbf{w}\|_2$

Using this idea we rewrite the formula as

$$\max_{w_0, \mathbf{w}} 1 / \|\mathbf{w}\|_2 \quad \text{now } \gamma = 1$$

$$\frac{1}{\|\mathbf{w}\|_2} = \text{margin}$$

$$\text{Subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1 \text{ for all } i = 1, \dots, N$$

# Hard-Margin SVM

Constrained optimization problem:

$$\begin{aligned} & \max_{w_0, \mathbf{w}} \gamma \\ & \text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq \gamma \text{ for all } i=1, \dots, N \\ & \|\mathbf{w}\|_2 = 1 \end{aligned}$$

$$\frac{\gamma}{\|\mathbf{w}\|_2} = \text{Geometric margin}$$

Another formulation:

$$\max_{w_0, \mathbf{w}} \frac{\gamma}{\|\mathbf{w}\|_2} = r$$

$$\text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq \gamma \text{ for all } i=1, \dots, N$$

Canonical weights!!!

Idea: we can rescale our margin to anything we want by rescaling our coefficients

notice that  $\max \gamma / \|\mathbf{w}\|_2$  equals  $\max(\gamma / \gamma) / \|\mathbf{w} / \gamma\|_2$  subject to  $y^{(i)}(w_0 / \gamma + \mathbf{w}^T / \gamma \mathbf{x}^{(i)}) \geq \gamma / \gamma$  for all  $i=1, \dots, N$

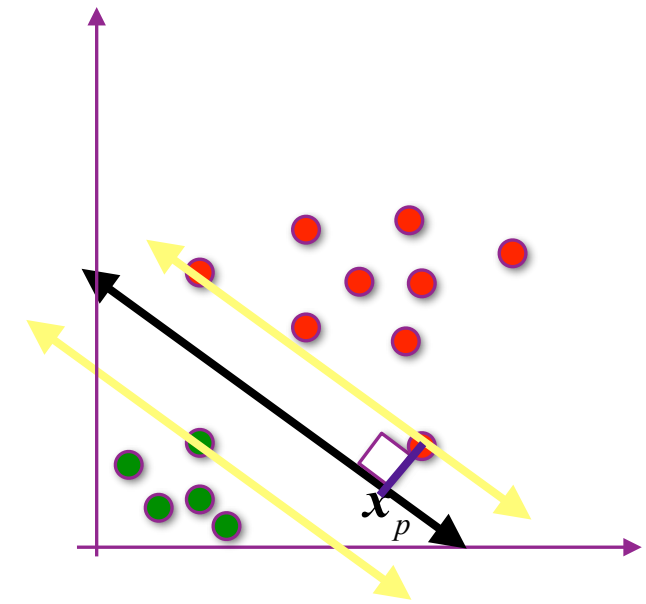
We set  $w_0 := w_0 / \gamma$ , and  $\mathbf{w} := \mathbf{w} / \gamma$  Notice we now want to  $\max 1 / \|\mathbf{w}\|_2$

Using this idea we rewrite the formula as

$$\max_{w_0, \mathbf{w}} 1 / \|\mathbf{w}\|_2 \quad \text{now } \gamma = 1$$

$$\frac{1}{\|\mathbf{w}\|_2} = \text{margin}$$

$$\text{Subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1 \text{ for all } i = 1, \dots, N$$





# Hard-Margin SVM

Constrained optimization problem:

$$\max_{w_0, \mathbf{w}} 1/\|\mathbf{w}\|_2$$

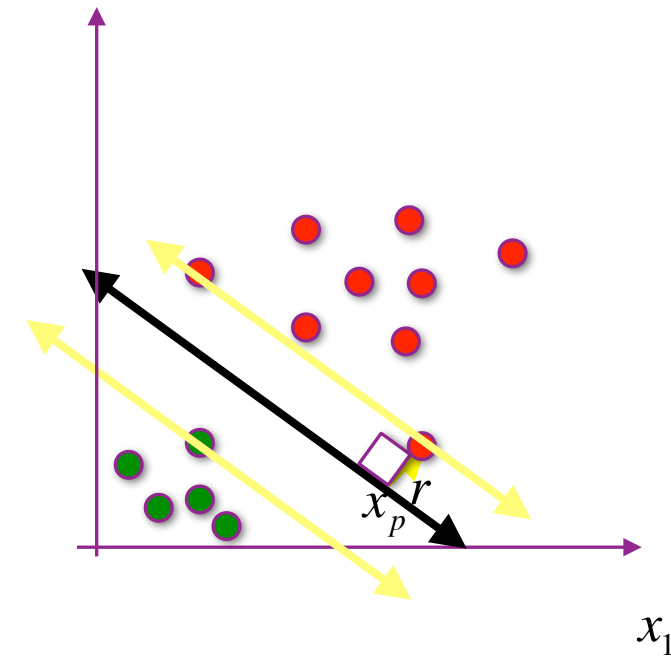
Subject to  $y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1$  for all  $i = 1, \dots, N$

Notice  $\max 1/\|\mathbf{w}\|_2$  is the same as  $\min \|\mathbf{w}\|_2$

Notice  $\min \|\mathbf{w}\|_2$  is the same as  $\min \|\mathbf{w}\|_2^2$

$$\min \|\mathbf{w}\|_2^2 = \min(w_1^2 + w_2^2 + \dots + w_d^2)$$

Subject to  $y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1$  for all  $i = 1, \dots, N$



Solvable in polynomial time!

Objective function is convex and points satisfying constraints is convex

A constrained quadratic optimization problem!

# Example Hard-Margin SVM

$(\mathbf{x}^T, y)$ :  $((1, 2.5), 1), ((2, 2), 1), ((3, 1), 1), \dots, ((0, 0.75), -1), ((1, 1), -1)\}$

The constrained quadratic optimization function is:

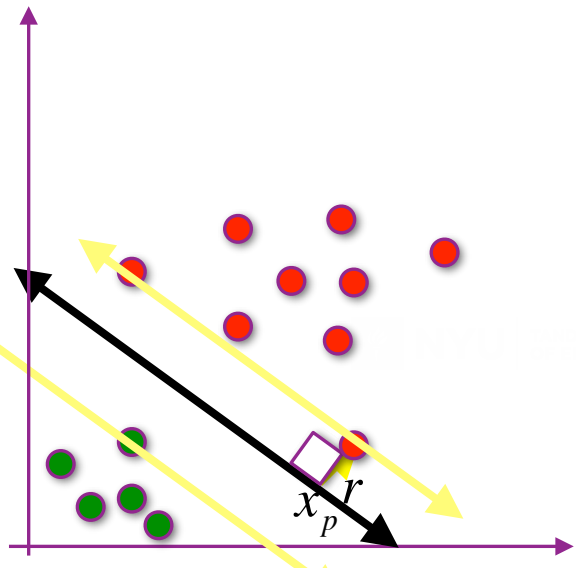
$$\min_{w_0, \mathbf{w}} \|\mathbf{w}\|_2^2 = w_1^2 + w_2^2$$

$$\text{subject to } (1) \left( w_0 + \mathbf{w}^T \begin{bmatrix} 1 \\ 2.5 \end{bmatrix} \right) \geq 1$$

$$(1) \left( w_0 + \mathbf{w}^T \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right) \geq 1$$

$$\vdots$$

$$(-1) \left( w_0 + \mathbf{w}^T \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right) \geq 1$$



# Example Hard-Margin SVM

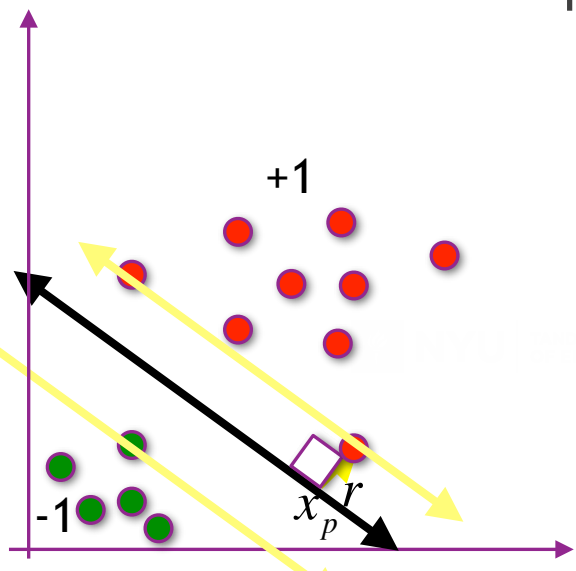
$(\mathbf{x}^T, y)$ :  $((1, 2.5), 1), ((2, 2), 1), ((3, 1), 1), \dots, ((0, 0.75), -1), ((1, 1), -1)\}$

The optimal hyperplane is:  $\mathbf{w} = (1, 4/3)^T$ ,  $w_0 = -10/3$

- $f(\mathbf{x}) = (1, 4/3)\mathbf{x} - 10/3$
- Predict +1 if  $f(\mathbf{x}) > 0$
- Predict -1 if  $f(\mathbf{x}) < 0$

Two types of training data:

- $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) = 1$ . Points on the margin called **support vectors**
- $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) > 1$ . If we remove these points, the solution doesn't change



Could it be possible for

$y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) > 1$  for all  $i=1, \dots, N$  and

No!

We could scale  $w$  and find a  
smaller  $\|\mathbf{w}\|_2^2$

So the functional margin  
with respect to the set of  
training examples will  
be 1

# Outline

□ Notation change, intuition, and finding how to compare hyperplanes - **mathematically how do compare hyperplanes to find the one with the maximum margin. Can we turn this way of comparing hyperplanes into an objective function**

□ Support vector machines

★ hard margin - **find the constrained objective function when the data is linearly separable**

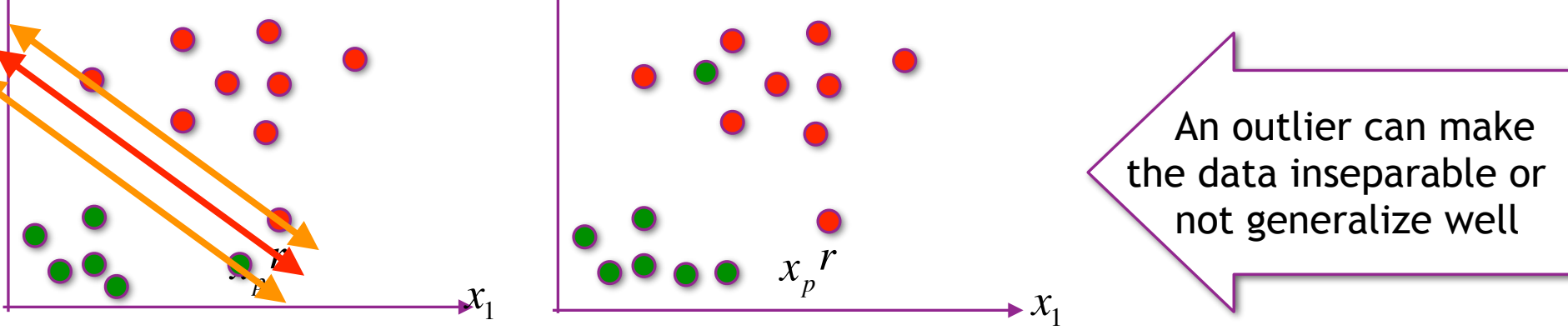


★ Dealing with non-linear data - “Soft” margins for SVM - **New constrained objective function for the case where the data is not linearly separable**

★ Pegasos algorithm. **Optimizer for soft margin SVM**

★ dual formula - **a clever trick**  $g(\mathbf{x}) = \text{sign} \left( w_0 + \sum_{i \in I} \alpha^{(i)} y^{(i)} \mathbf{x}^{(i)T} \mathbf{x} \right)$

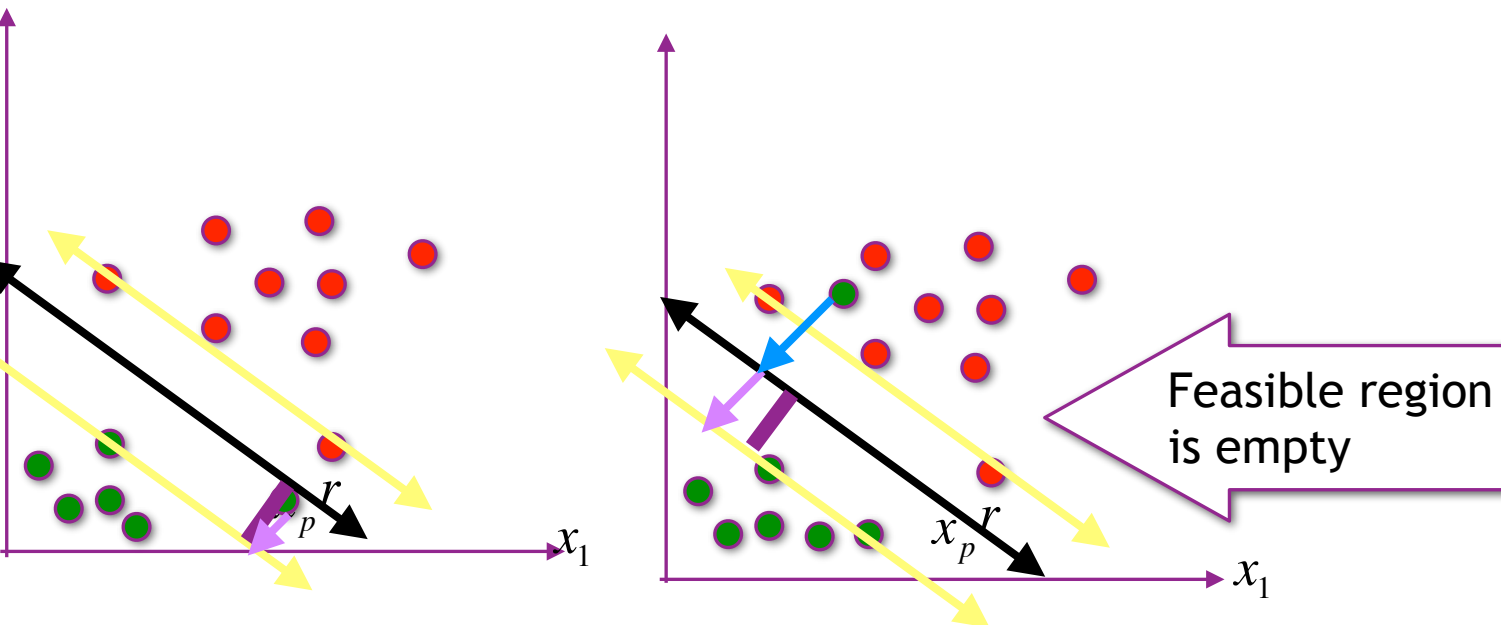
★ Dealing with non-linear data - feature transformation with the kernel trick - **Show two popular feature maps**



# What if the data isn't linearly separable

WE CAN MAKE OUR MODEL MORE FLEXIBLE BY ADDING A COST FUNCTION FOR THE POINTS THAT ARE MISCLASSIFIED

# Soft-Margin SVM



How can we still find the optimal hyperplane where we allow for a few points to either be misclassified or within the margin?

We could incur a cost  $\xi^{(i)}$  for how far the  $\mathbf{x}^{(i)}$  is away from the margin.

We will create a slack variable  $\xi^{(i)}$  for each training example  $\mathbf{x}^{(i)}$

The hyperplane must satisfy

$$y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1 - \xi^{(i)}$$

$$\xi^{(i)} =$$

$$+$$

$$\xi^{(i)} =$$

What function should we use for

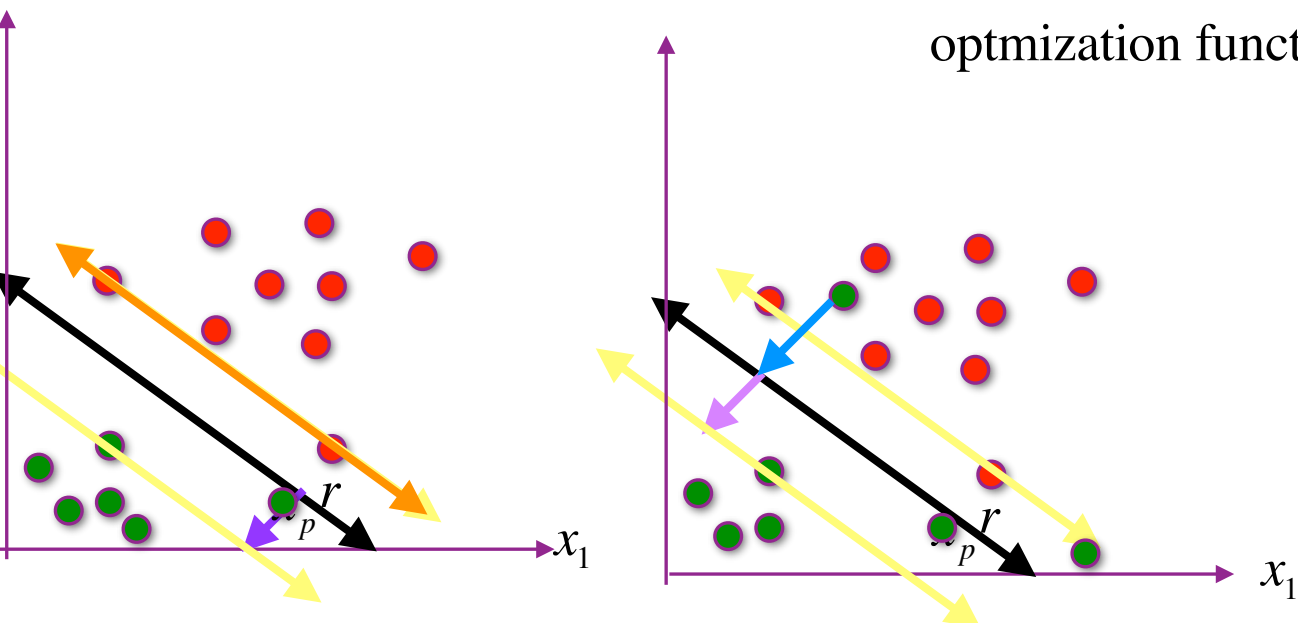
# Which cost function?

SHOULD ALL THE POINTS BE CHARGED, OR ONLY THOSE THAT ARE INSIDE THE MARGIN OR INCORRECTLY CLASSIFIED?





# Soft-Margin SVM

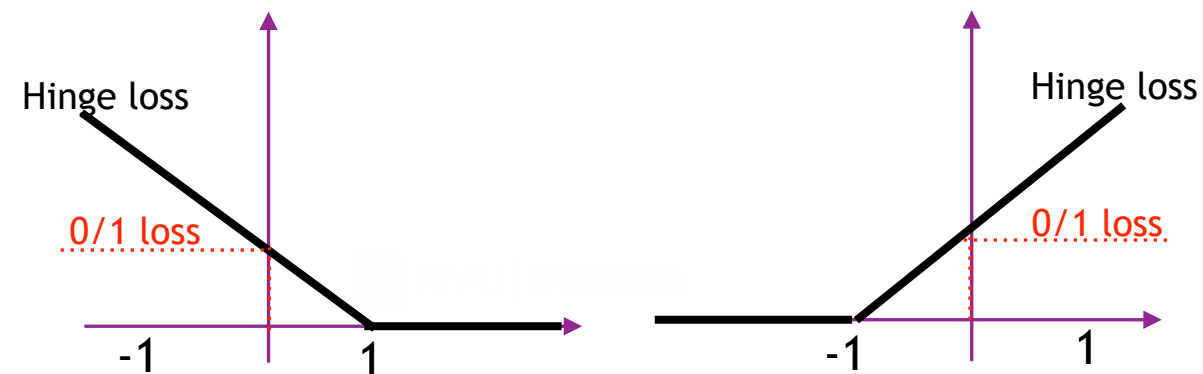


$$\min_{w_0, \mathbf{w}, \{\xi^{(i)}\}_{i=1}^N} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^N \xi^{(i)}$$

subject to  $y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1 - \xi^{(i)}$  for all  $i=1, \dots, N$

$$\xi^{(i)} \geq 0$$

$C$  is a tunable parameter. Gives relative importance of the error term



$$\xi^{(i)} = \begin{cases} 0 & y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq 1 \\ 1 - y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) & \text{otherwise} \end{cases}$$

# optimization function

$$\min_{w_0, \mathbf{w}, \{\xi^{(i)}\}_{i=1}^N} \|\mathbf{w}\|_2^2 + \sum_{i=1}^N \xi^{(i)}$$

$$\text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1 - \xi^{(i)}$$

$$\xi^{(i)} \geq 0$$

Pair share: What do you know about the functional margin for  $\mathbf{x}$  if:

1)  $\xi \geq 1$

2)  $0 < \xi < 1$

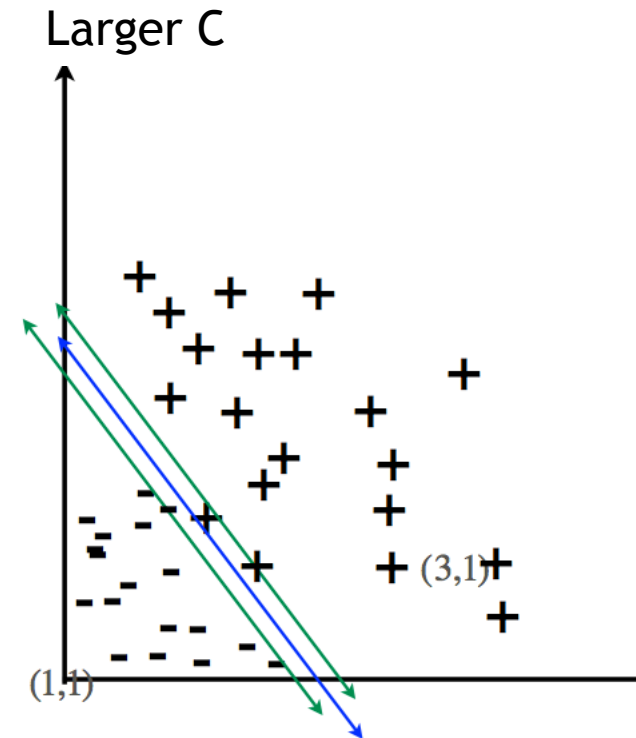
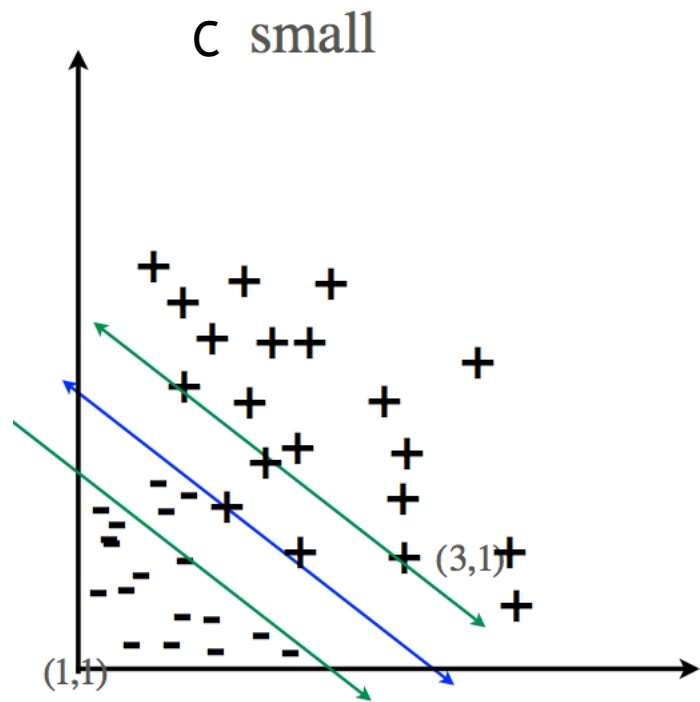
3)  $\xi = 0$

Pair share: Do you think that  $\sum_{i=1}^N \xi^{(i)}$  is an upper bound on the number of training errors?

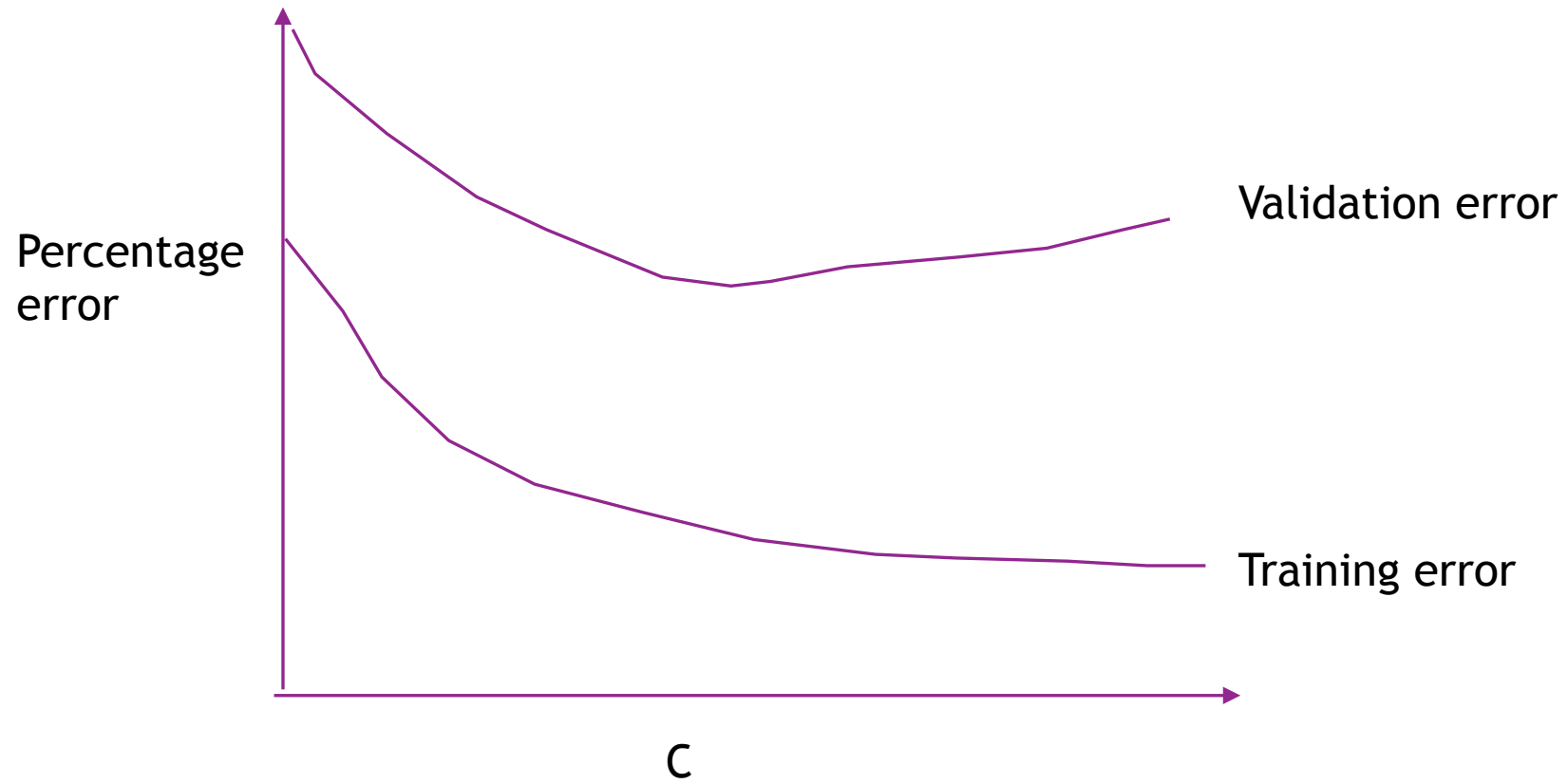
Pair share:

1) What happens to the margin if I make  $C$  large?

2) What happens to the margin if I make  $C$  small?

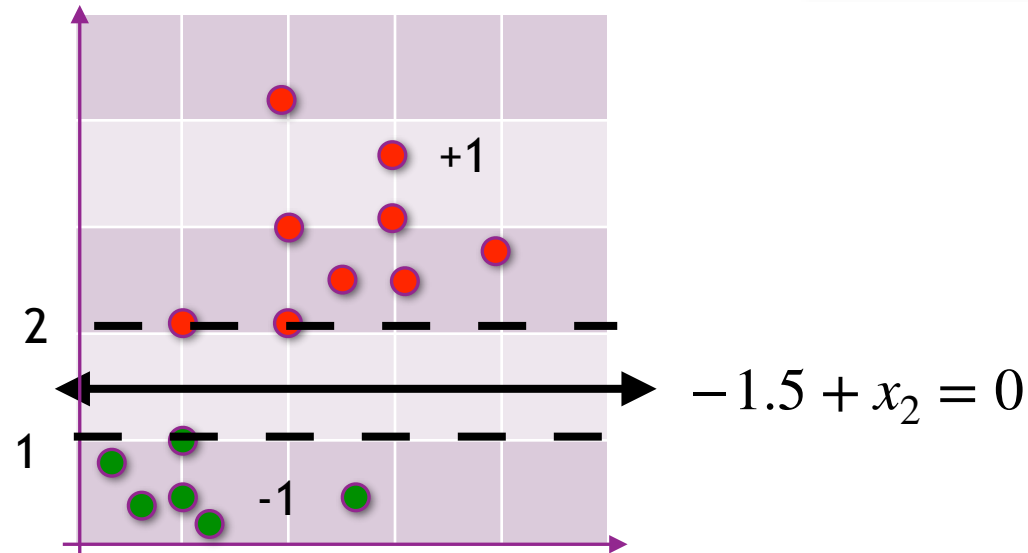


What if  $C = \infty$ ?



# Example

Pair share: How can modify our decision boundary to have a functional margin of 1?

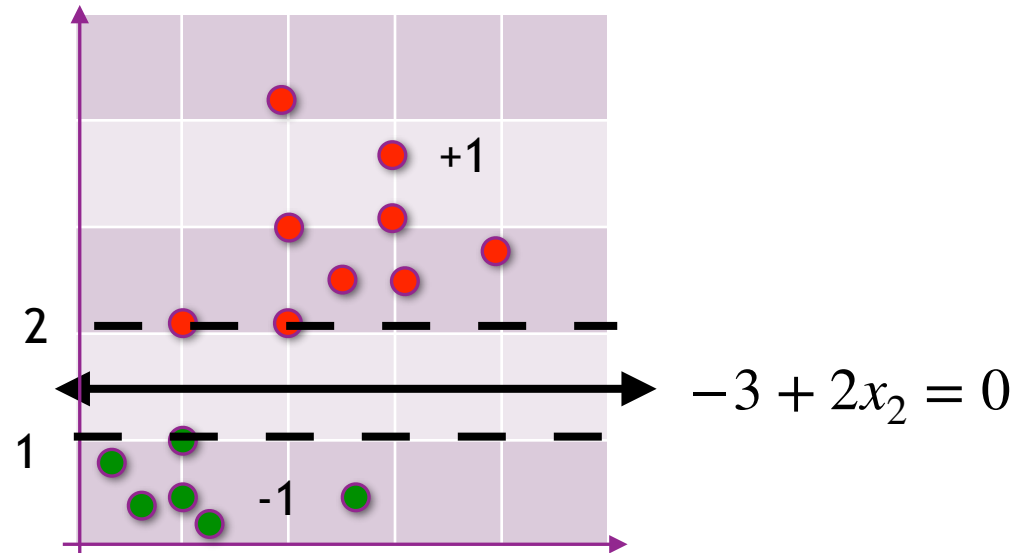


Decision boundary is  $\mathbf{w} = [0, 1]^T$ ,  $w_0 = -1.5$

Is this the form we wanted?

The support vectors are supposed to have a functional margin of 1:  $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) = 1$

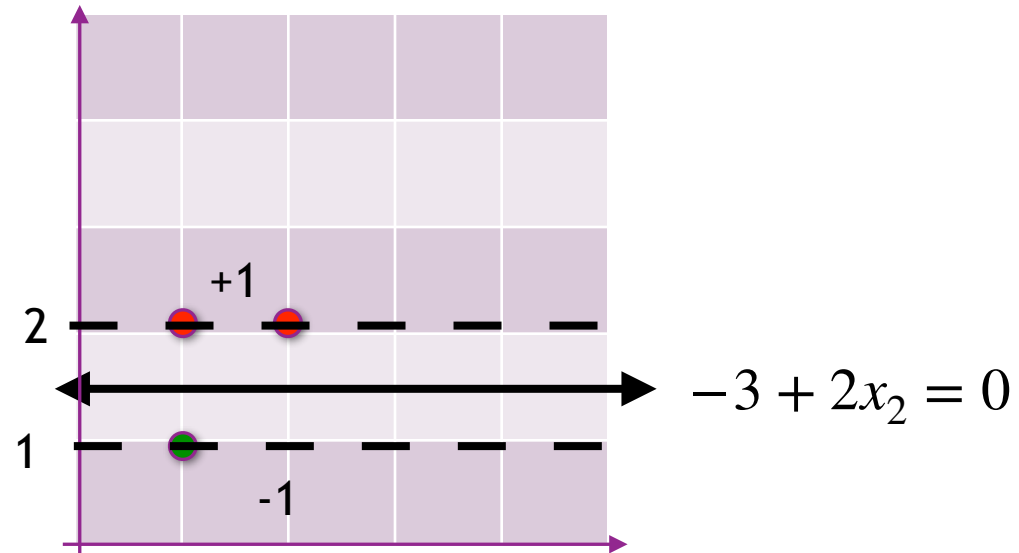
# Example



Decision boundary is  $\mathbf{w} = [0, 2]^T$ ,  $w_0 = -3$

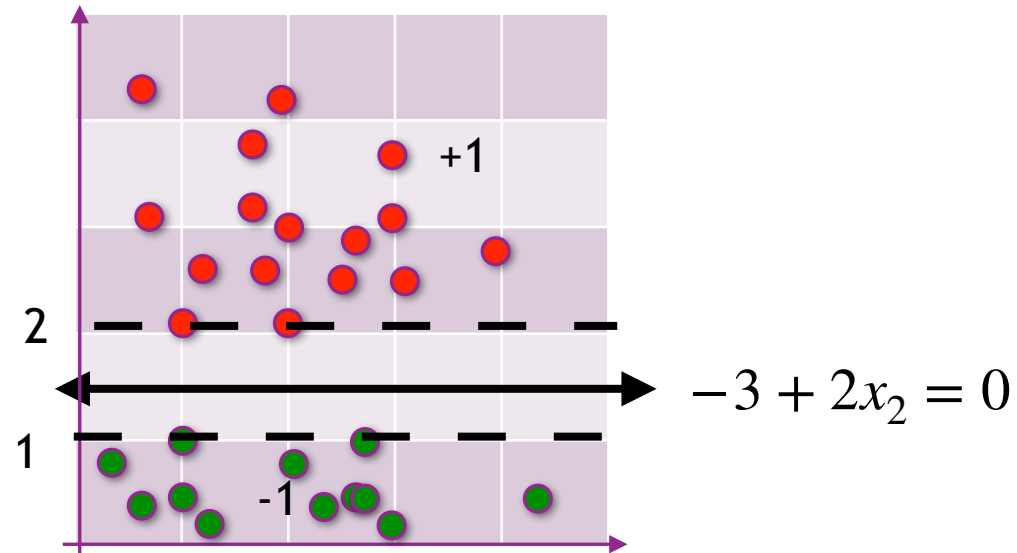
The support vectors have a functional margin of 1:  $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) = 1$

# Example



The boundary doesn't change if I remove points with a functional margin  $> 1$

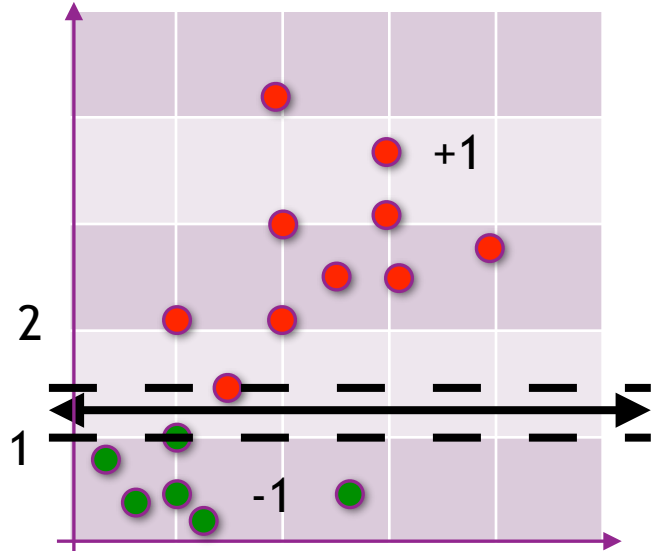
# Example



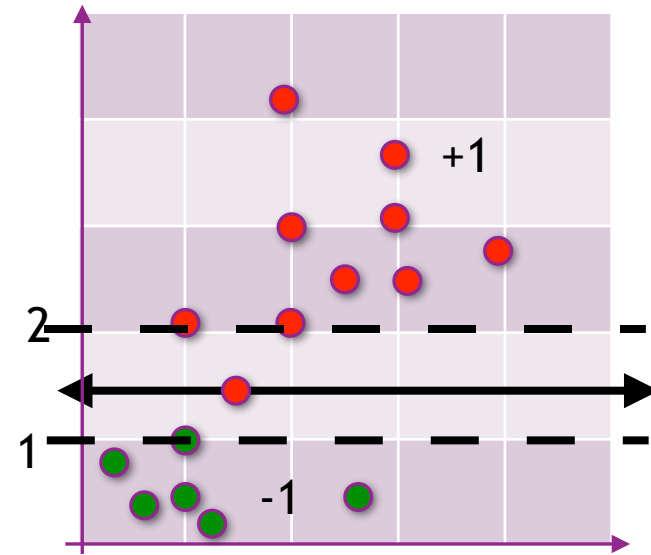
The solution doesn't change if I add points whose functional margin is  $\geq 1$



# Example



Our margin becomes smaller if we have an outlier



$$\min_{w_0, \mathbf{w}, \{\xi^{(i)}\}_{i=1}^N} \|\mathbf{w}\|_2^2 + \sum_{i=1}^N \xi^{(i)}$$

$$\text{subject to } y^{(i)}(w_0 + \mathbf{w}^T \mathbf{x}^{(i)}) \geq 1 - \xi^{(i)}$$