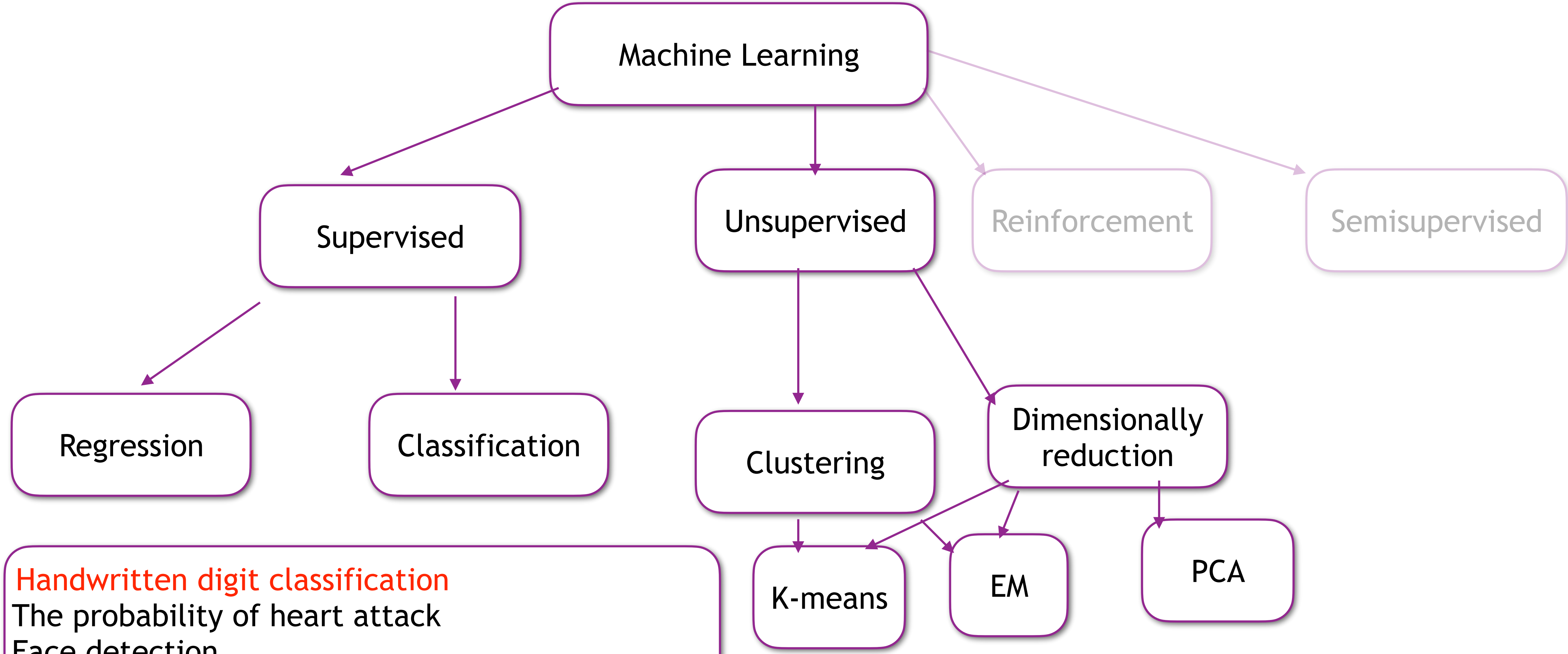
- http://cs229.stanford.edu/notes/cs229-notes3.pdf
- https://www.svm-tutorial.com/
- https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-the-linearly-separable-case-1.html
- Advanced: https://svmtutorial.online/download.php?file=SVM_tutorial.pdf

Lecture
Support Vector Machines

PROF. LINDA SELLIE

SOME SLIDES FROM PROF. RANGAN

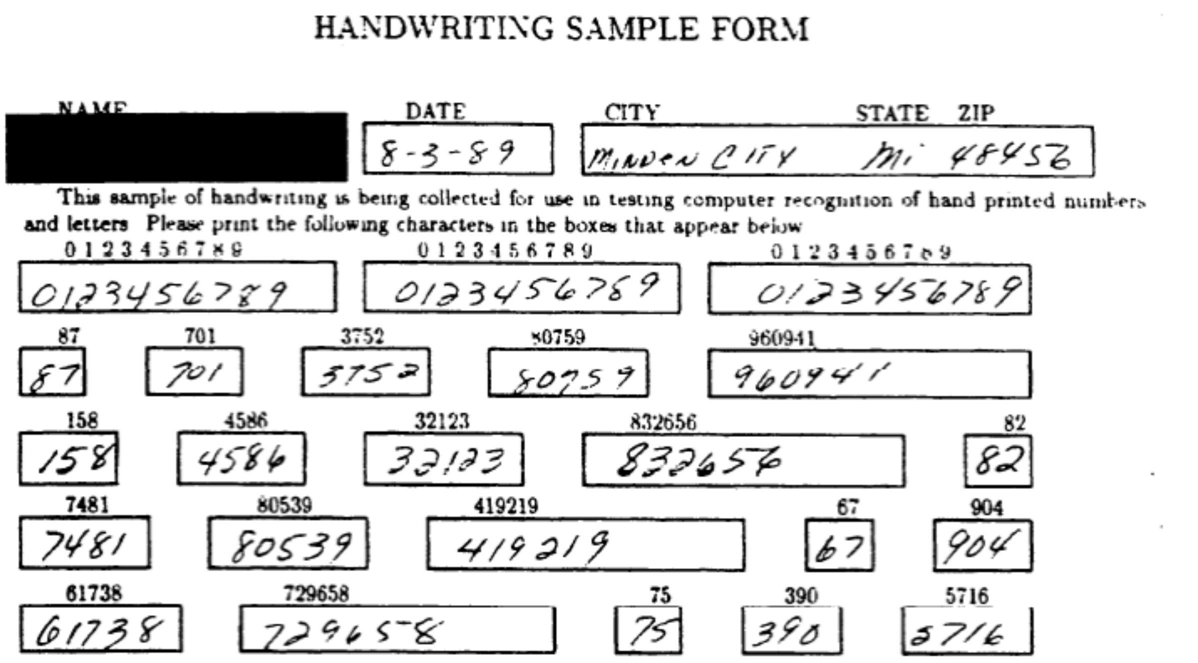


Handwritten digit classification
The probability of heart attack
Face detection
Face recognition
Getting into a college based on GPA, ACT< AP scores
Email is spam/ not spam

Learning objectives:

- Understand the idea behind the geometric margin, functional margin, and canonical weights
- Create an objective function 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
- 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.

0123456789 | 0123456789 | 0123456789

87 | 701 | 3752 | 80759 | 960941

158 | 4586 | 32123 | 832656 | 82

7481 | 80539 | 419219 | 67 | 904

61738 | 729658 | 75 | 390 | 5716

From Patrick J. Grother, NIST Special Database, 1995

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

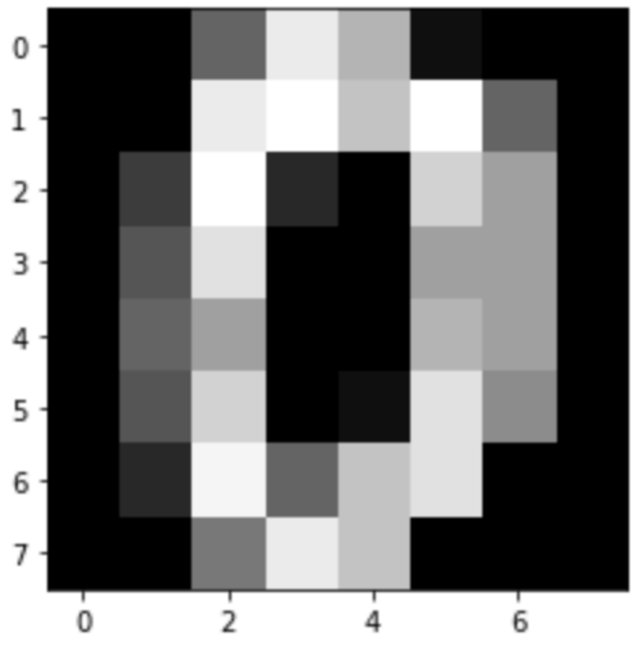
- What does one example look like?

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

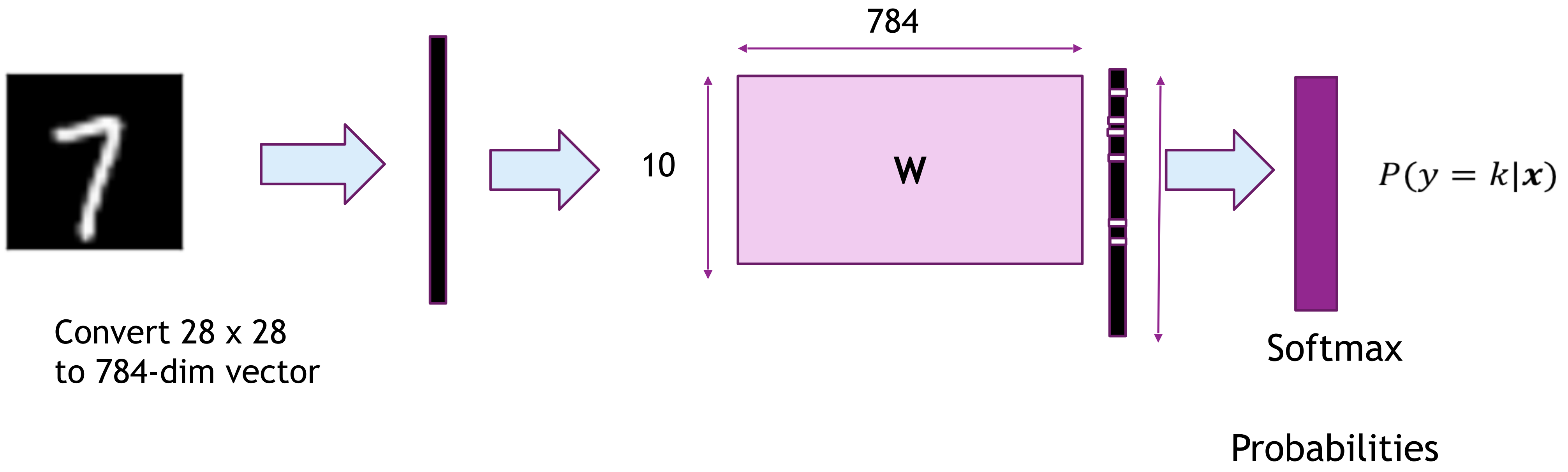
```
[[ 0.   0.   5.  13.   9.   1.   0.   0.]
 [ 0.   0.  13.  15.  10.  15.   5.   0.]
 [ 0.   3.  15.   2.   0.  11.   8.   0.]
 [ 0.   4.  12.   0.   0.   8.   8.   0.]
 [ 0.   5.   8.   0.   0.   9.   8.   0.]
 [ 0.   4.  11.   0.   1.  12.   7.   0.]
 [ 0.   2.  14.   5.  10.  12.   0.   0.]
 [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

Flatten

```
[[ 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.]]
```



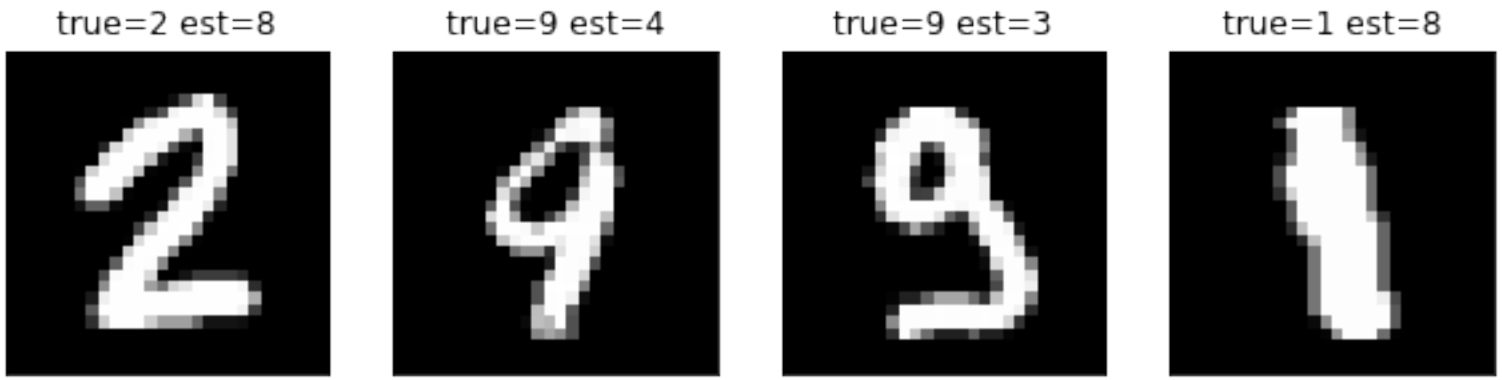
Recap: Logistic Classifier



- 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
- <u>Can we do better?</u>
- 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

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

New Notation

Previously we used

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

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

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.

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

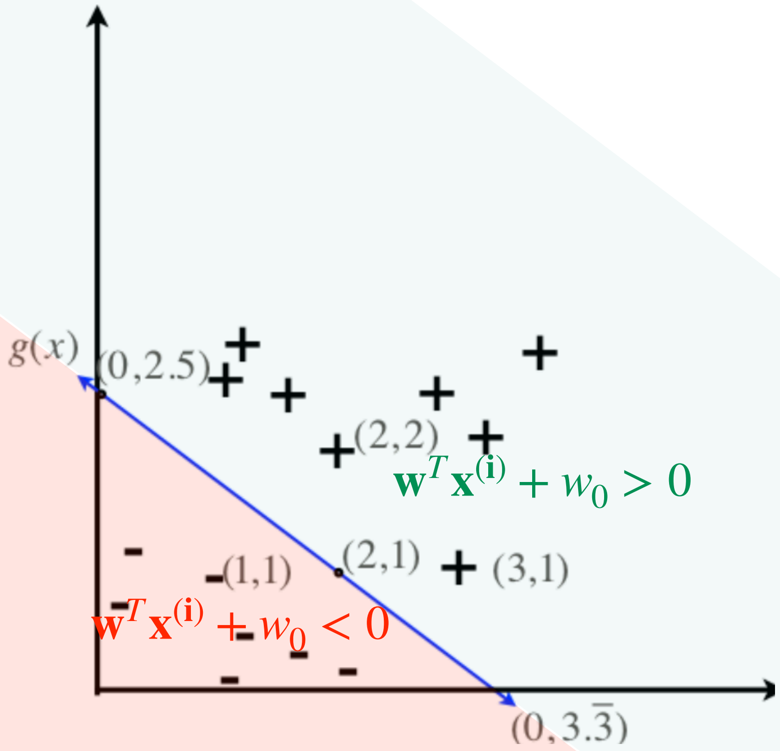
$$y \in \{-1,1\}$$

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

We use the hyperplane to classify a point x

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

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



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

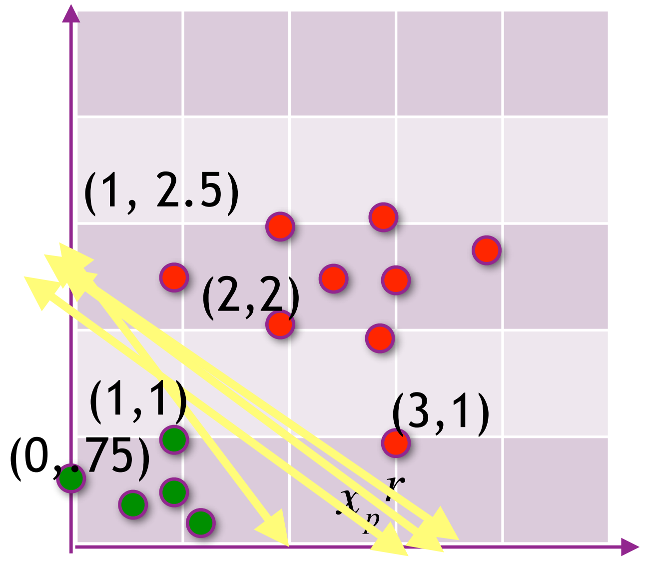
$$(3,4)\mathbf{x} - 10 < 0$$

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

Predicting using a hyperplane

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

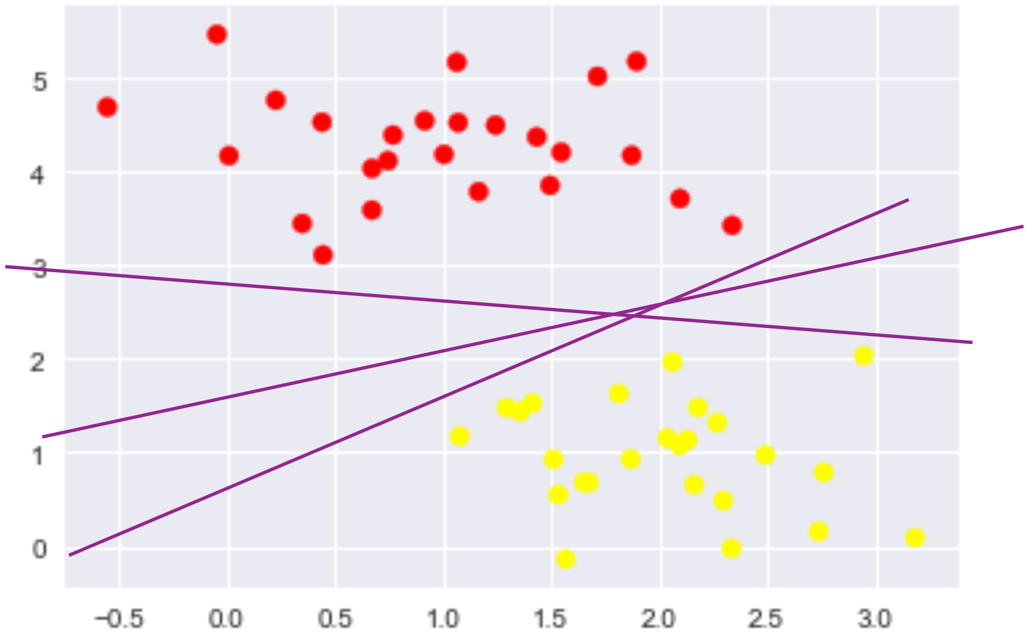
$$Predict\ -1\ \text{if}\ \ \mathbf{w}^\mathrm{T}\mathbf{x} + w_0 < 0$$



Which hyperplane?

Suppose there is a hyperplane that separates the training data

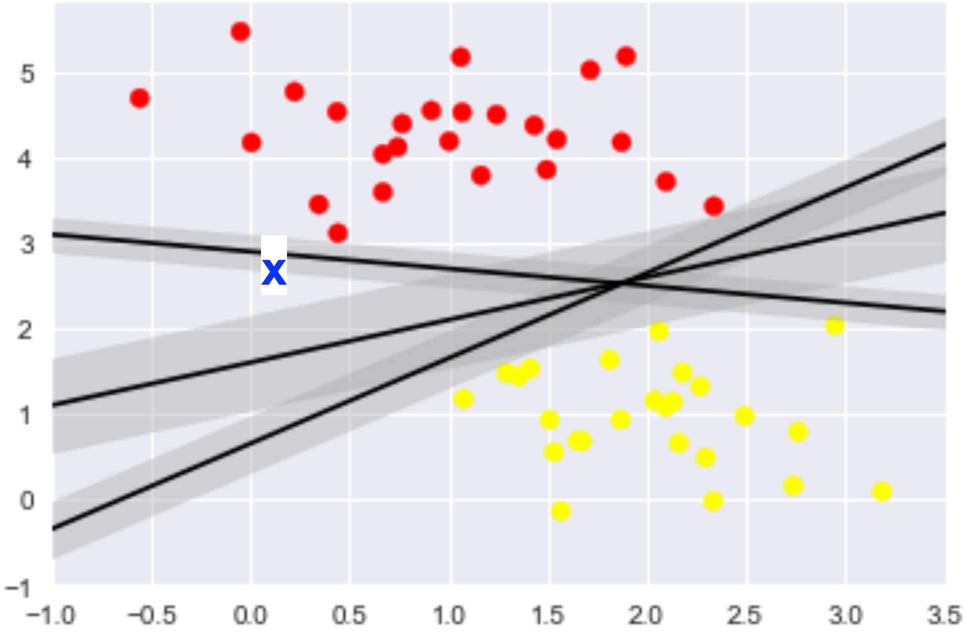
Which hyperplane is the “best”?

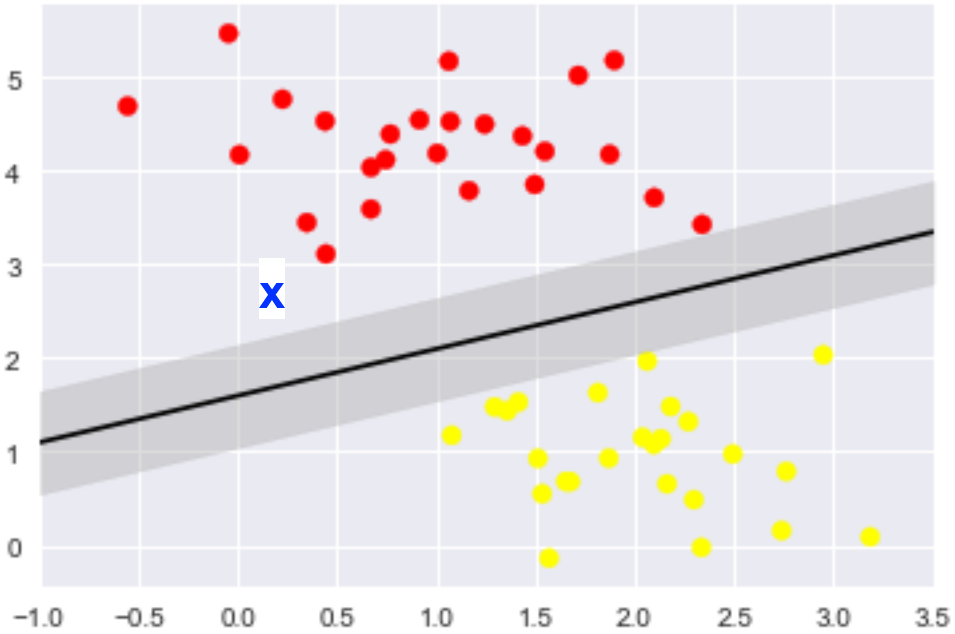


Which line is best? Why????

Which line should we use?

How should we classify a new point x?

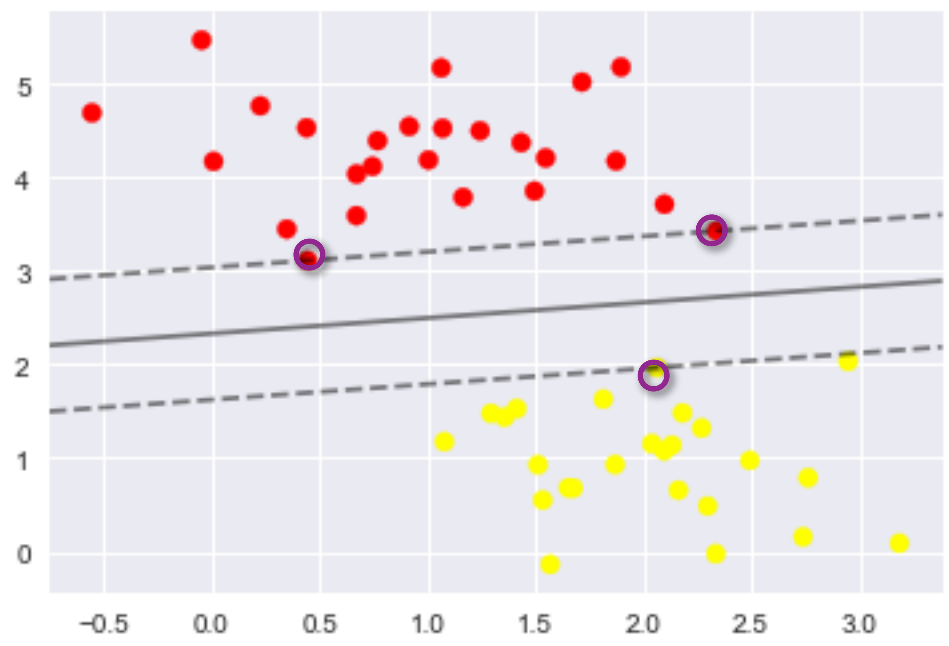




The one that makes intuitive sense is the line that has the maximum margin

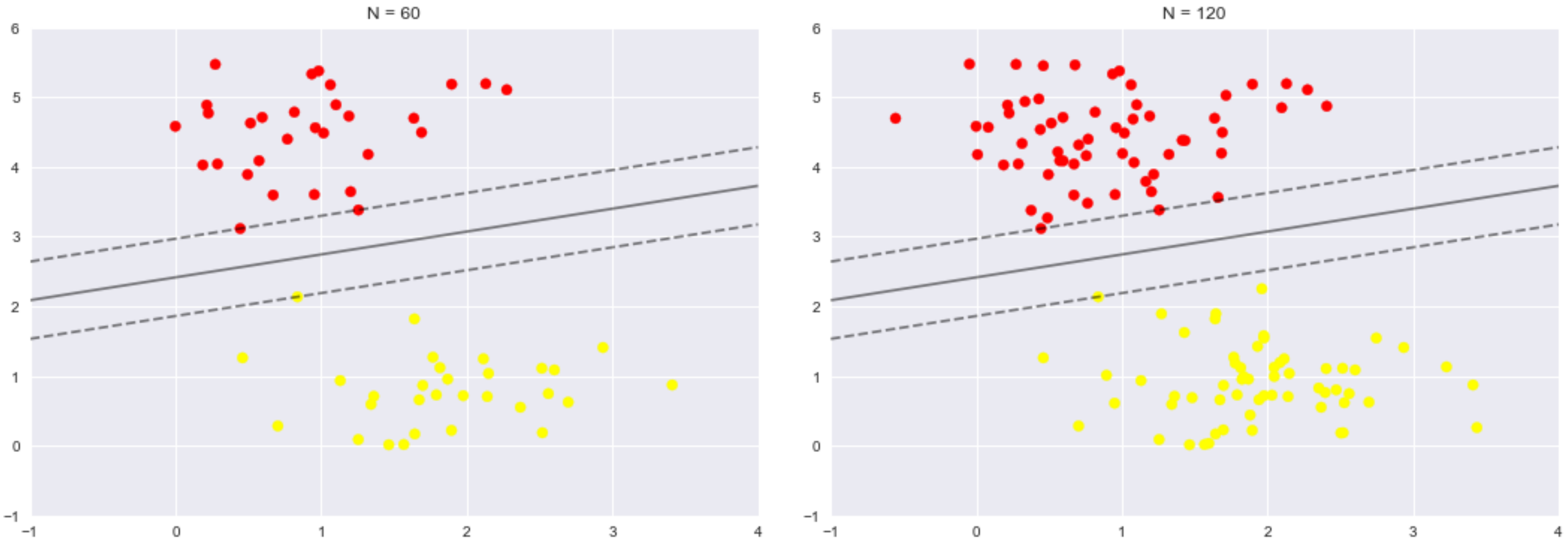
Now we can feel more comfortable predicting the point x

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
  - ★ 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 ideas is that the best hyperplane has the largest margin.

First observation:

For any hyperplane, we can find the distance of the closest training example to the hyperplane

Geometric Margin

+1

-1

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

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

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

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

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

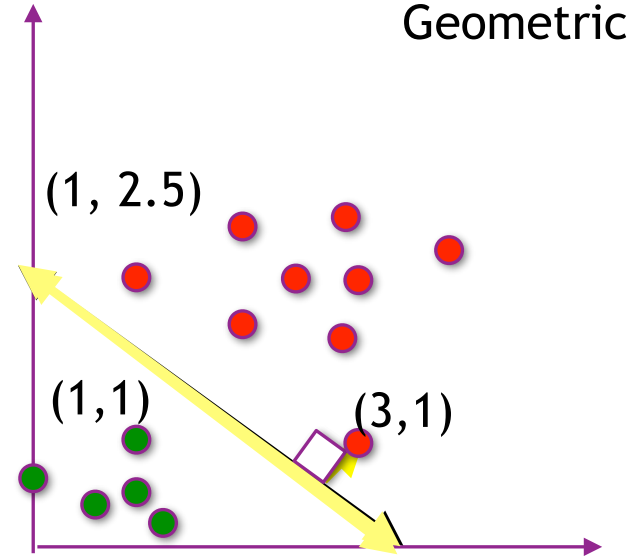
$$= (1)(3)/5$$

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

$$= (1)(3)/5$$

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

$$= (-1)(-3)/5$$



$(-10 + 3x_1 + 4x_2) = 0$

All other points have a geometric margin which is larger than 3/5

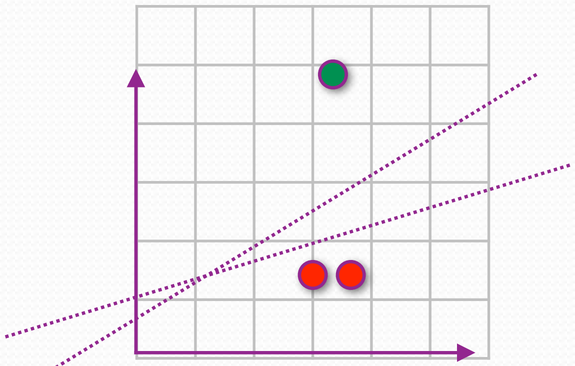
if $\|\mathbf{w}\|_2 = 1$ i.e $\mathbf{w} \coloneqq \mathbf{w}/\|\mathbf{w}\|_2$ we don't need to divide by $\|\mathbf{w}\|_2$

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

Second observation:

We can compare two hyperplanes by comparing their geometric margins.

Which hyperplane has a larger margin:



- Given the following data:

```
x(1) = [3.2   4.7]        y(1) = -1
x(2) = [3.5   1.4 ]       y(2) = 1
x(3) = [3.    1.4 ]       y(3) = 1
```

- What is the geometric margin for:

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

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

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

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

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

$\gamma_g = 0.9$

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

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

$$(-1)\left([1/3 \quad -1]\begin{bmatrix} 3.2 \\ 4.7 \end{bmatrix} + 1)\right)/\sqrt{1/9+1} = 2.5$$

$$(1)\left([1/3 \quad -1]\begin{bmatrix} 3.5 \\ 1.4 \end{bmatrix} + 1)\right)/\sqrt{1/9+1} = 0.7$$

$$(1)\left([1/3 \quad -1]\begin{bmatrix} 3 \\ 1.4 \end{bmatrix} + 1)\right)/\sqrt{1/9+1} = 0.6$$

$\gamma_g = 0.6$

$y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + w_0)/\|\mathbf{w}\|_2 \geq 0.6$

Objective function

Goal

$$\max_{w_0, \boldsymbol{w}} \ \gamma_g$$

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

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

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.

We want our constrained object function to return a unique $\mathbf{w}, w_0$

Crazy idea next!

Instead of requiring $\mathbf{w}$ to be a unit vector (i.e. $||\mathbf{w}|| = 1$),
We will define the idea of a "functional margin" and require that to be 1.

Steps to understanding "functional margin":

1. Simple observation: *rescaling* the parameters doesn't change the *decision boundary*.
2. How to rescale the parameters/weights, so the functional margin is 1. We call these weights/parameters *canonical weights*.

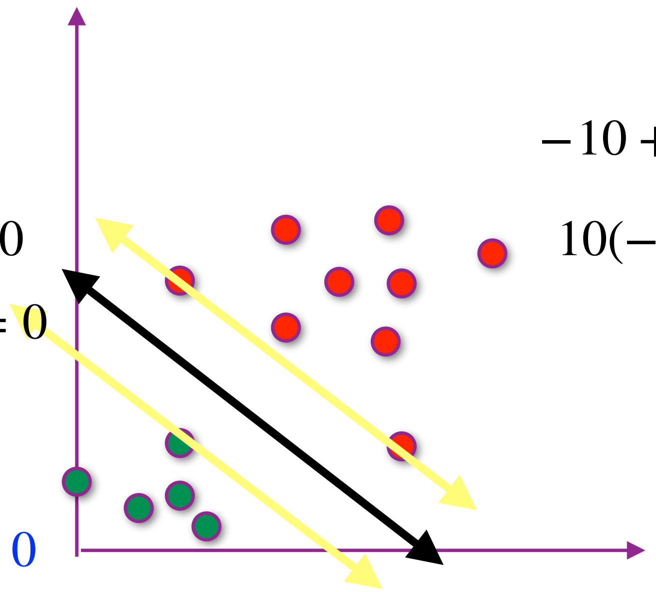
Step 1. We can write a hyperplane in many ways

$-10 + 3x_1 + 4x_2 > 0$

$10(-10 + 3x_1 + 4x_2) > 0$

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

$-100 + 30x_1 + 40x_2 = 10(-10 + 3x_1 + 4x_2) = 0$



$-10 + 3x_1 + 4x_2 < 0$

$10(-10 + 3x_1 + 4x_2) < 0$

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

Step 1 conclusion

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

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

Step 2

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

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

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

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

$$\gamma_f = \min\{\gamma_f^{(1)}, \gamma_f^{(2)}, \ldots, \gamma_f^{(N)}\}$$

Step 2

Canonical weights

+1

-1

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

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

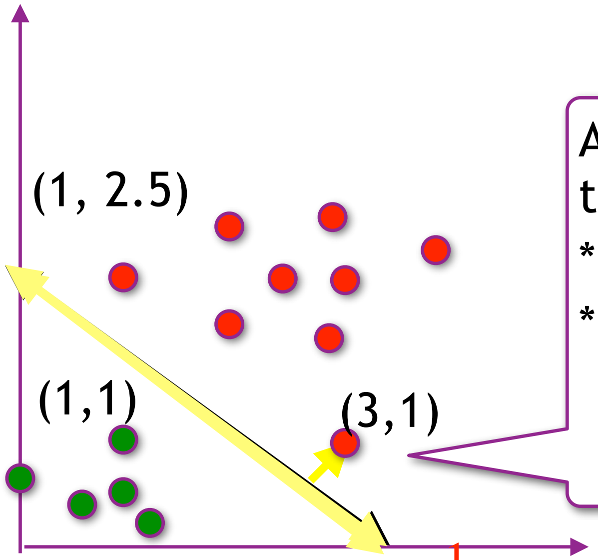
Functional margin of $(\mathbf{w}, w_0)$ with respect to ***a set*** S is

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

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

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

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



$$\frac{1}{3}(-10 + 3x_1 + 4x_2) = 0$$

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

* functional margin of 1
* Euclidean distance for this point is $\dfrac{1}{\|\mathbf{w}\|_2}$

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

We can make $\gamma_f = 1$

The canonical weights:

$$\mathbf{w} = \begin{bmatrix}1\\4/3\end{bmatrix}, w_0 = -10/3$$

Step 2 conclusion

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

Canonical weights are when the functional margin is 1 for the set of training examples

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

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

Next: Many equivalent versions of our objective function

Hard-Margin SVM

$\gamma_f = \gamma_g$ when $\|\mathbf{w}\|_2 = 1$

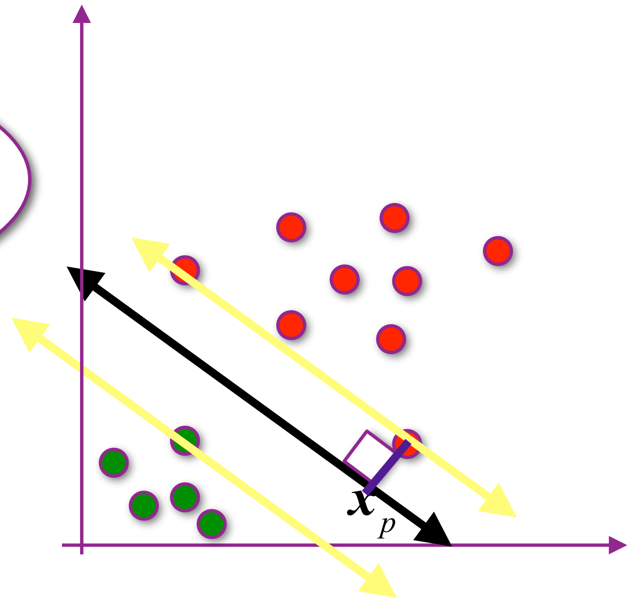
1)

Constrained optimization problem:

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

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

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



$\dfrac{\gamma_f}{\|\mathbf{w}\|_2}$ = Geometric margin

2)

Another formulation:

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

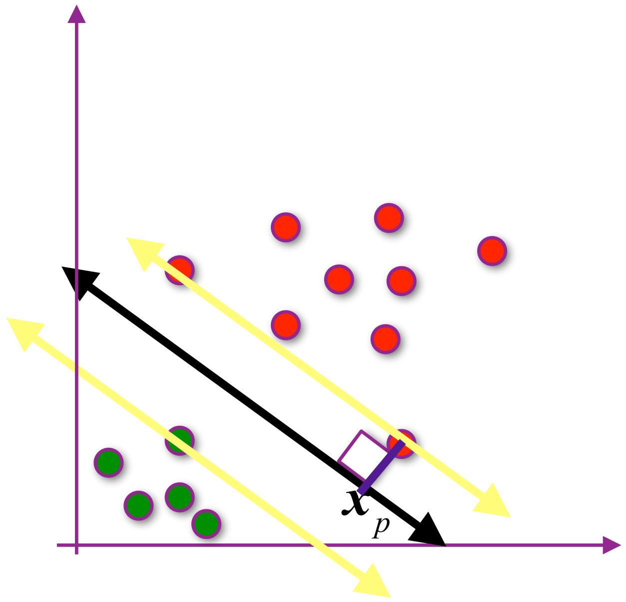
Subject to $y^{(i)}(\mathbf{w}^T \mathbf{x}^{(i)} + w_0) \geq \gamma_f$ for all $i \in \{1, \ldots, N\}$

Hard-Margin SVM

$$\frac{\gamma_f}{\|\mathbf{w}\|_2} = \gamma_g$$

2) $$\max_{\mathbf{w}, w_0} \frac{\gamma_f}{\|\mathbf{w}\|_2} = r$$

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



Canonical weights!!!

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

notice that $\max_{\mathbf{w}, w_0} \frac{\gamma_f}{\|\mathbf{w}\|_2}$ equals $\max_{\mathbf{w}, w_0} \frac{\gamma_f / \gamma_f}{\|\mathbf{w}/\gamma_f\|_2}$

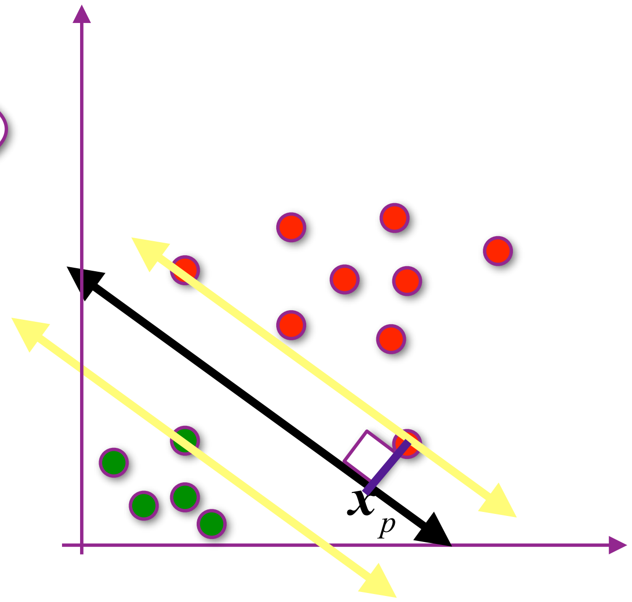
Subject to $y^{(i)}(\frac{\mathbf{w}^T}{\gamma_f}\mathbf{x}^{(i)} + \frac{w_0}{\gamma_f}) \geq \frac{\gamma_f}{\gamma_f}$ for all $i \in \{1, \ldots, N\}$

Hard-Margin SVM

Canonical weights!!!

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

Subject to $y^{(i)}(\frac{\mathbf{w}^T}{\gamma_f}\mathbf{x}^{(i)} + \frac{w_0}{\gamma_f}) \geq \frac{\gamma_f}{\gamma_f}$ for all $i \in \{1,\ldots,N\}$



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

Using this idea we rewrite the formula as

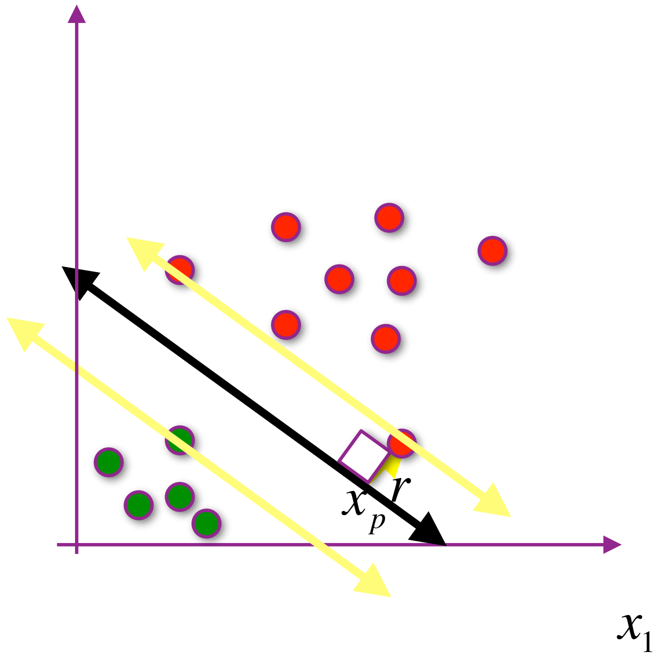
$$\max_{w_0,\mathbf{w}} 1/\|\mathbf{w}\|_2$$

Now $\gamma_f = 1$

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

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

Hard-Margin SVM



3) 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 $i = 1, \ldots, N$

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

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

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

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

Objective function is convex and points satisfying constraints are convex

Solvable in polynomial time!

A constrained quadratic optimization problem!

Example Hard-Margin SVM

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

The constrained quadratic optimization function is:

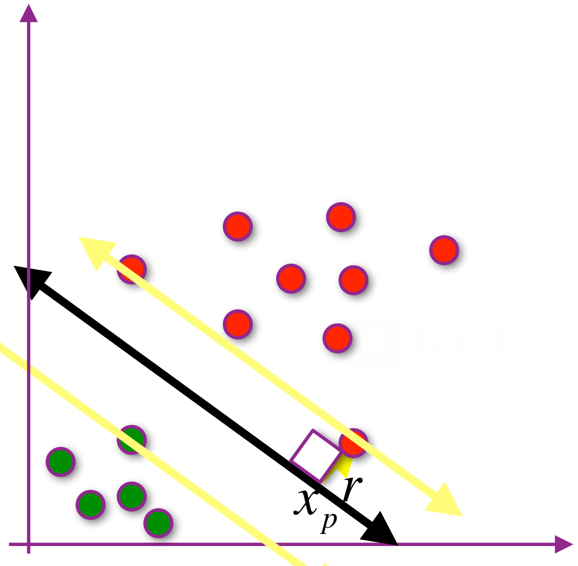
$$\min_{w_0,\boldsymbol{w}} \left|\left|\mathbf{w}\right|\right|_2^2 = w_1^2 + w_2^2$$

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

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

$$\vdots$$

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

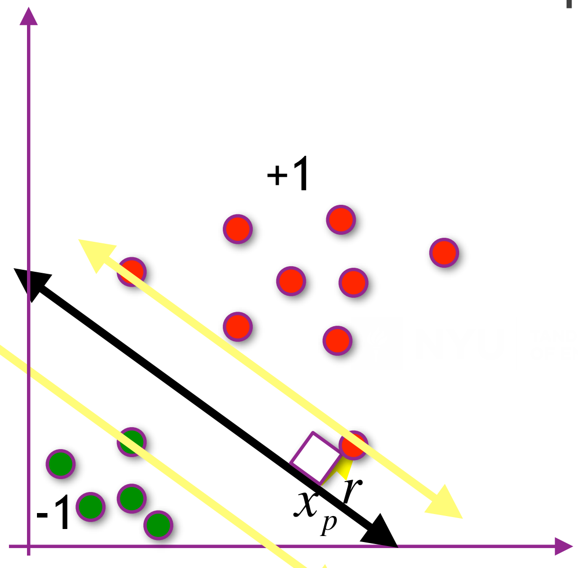


Example Hard-Margin SVM

$(\mathbf{x}^T, y)$: ((1, 2.5),1), ((2, 2),1), ((3,1),1),…,((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

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

No!

We could scale w and w_0 to find a smaller $\|\mathbf{w}\|_2^2$

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