

# support vector machines large margin classification

SVMs are considered to be the most powerful 'black box' learning algorithm, and by posing a cleverly-chosen optimization objective, one of the most widely used learning algorithms today

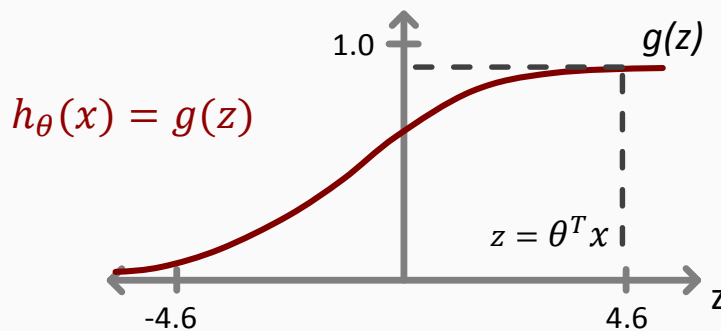
## large margin classification

### optimization objective

considerations among supervised learning algorithms are those of the amounts of data consumed and method in which they are implemented

compared to both logistic regression and neural networks, SVM can provide a cleaner and more powerful way of learning complex nonlinear functions

### alternate view of logistic regression



$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

$$z = \theta^T x$$

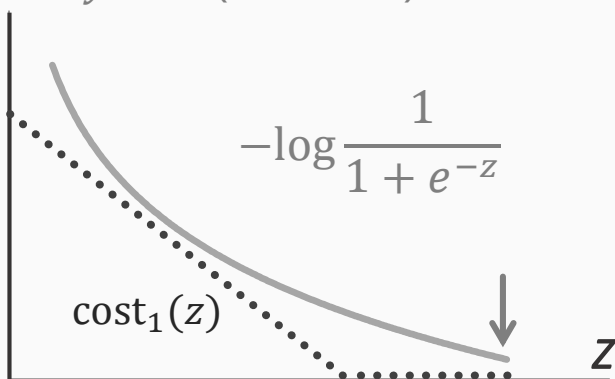
if  $y = 1$ , the desired output is  $h_\theta(x) \approx 1, \theta^T x \gg 0$

if  $y = 0$ , the desired output is  $h_\theta(x) \approx 0, \theta^T x \ll 0$

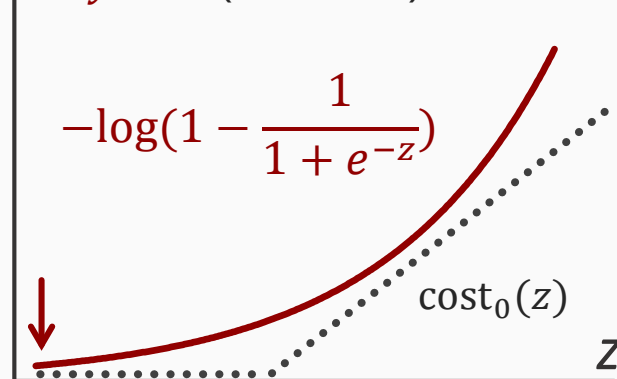
cost of example with a single observations  $(x, y)$ :

$$\begin{aligned} J(\theta) &= -[y \log h_\theta(x) + (1 - y) \log(1 - h_\theta(x))] \\ &= -y \log \frac{1}{1 + e^{-\theta^T x}} - (1 - y) \log \left( 1 - \frac{1}{1 + e^{-\theta^T x}} \right) \end{aligned}$$

if  $y = 1$  ( $\theta^T x \gg 0$ ):




if  $y = 0$  ( $\theta^T x \ll 0$ ):



## support vector machines

logistic regression:  $J(\theta) =$

$$\min_{\theta} \frac{1}{m} \left[ \sum_{i=1}^m y^{(i)} \underbrace{\left( -\log h_{\theta}(x^{(i)}) \right)}_{\text{cost}_1(\theta^T x^{(i)})} + (1 - y^{(i)}) \underbrace{\left( -\log (1 - h_{\theta}(x^{(i)})) \right)}_{\text{cost}_0(\theta^T x^{(i)})} \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$



**A**
**B**

support vector machine:

$$\min_{\theta} \quad \cancel{\frac{1}{m}} \quad \mathbf{C} \quad \sum_{i=1}^m y^{(i)} \text{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0(\theta^T x^{(i)}) + \frac{\mathbf{1}}{2} \cancel{\frac{\lambda}{m}} \sum_{j=1}^n \theta_j^2$$

svm will use a separate convention for applying  $\frac{1}{m}$ ; removing them will result in the same optimal value of  $\theta$

example

$$\min_u ((u - 5)^2) \times 10 + 1 \rightarrow u = 5$$

$$\min_u 10(u - 5)^2 + 10 \rightarrow u = 5$$

trading off weights for svm

logistic regression  $\rightarrow \mathbf{A} + \lambda \mathbf{B}$

svm  $\rightarrow \mathbf{CA} + \mathbf{B}$

$$\mathbf{C} = \frac{1}{\lambda}$$

overall optimization objective function for support vector machine:

$$\min_{\theta} \frac{1}{m} C \sum_{i=1}^m y^{(i)} \text{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0(\theta^T x^{(i)}) + \frac{1}{2} \sum_{j=1}^n \theta_j^2$$

minimizing the above function will result in the parameters learned by the svm; unlike logistic regression, the svm will not output a probability

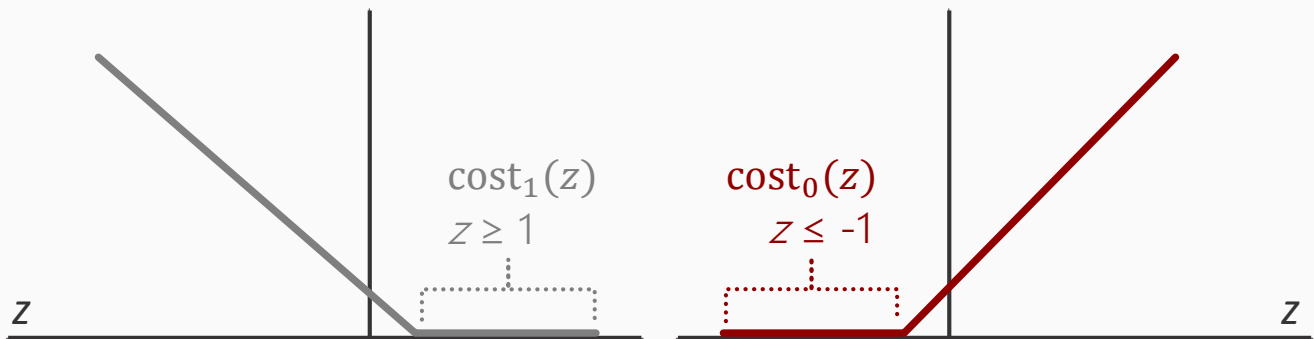
support vector machines output value of 1 or 0 after minimizing the cost function:

$$h_{\theta} x = \begin{cases} 1 & \text{if } \theta^T x \text{ is } \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

## large margin intuition

support vector machines are sometimes referred to as large margin classifiers:

$$\min_{\theta} \frac{1}{m} C \sum_{i=1}^m y^{(i)} \text{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0(\theta^T x^{(i)}) + \frac{1}{2} \sum_{j=1}^n \theta_j^2$$



if  $y = 1$ , the desired output is  $\theta^T x \geq 1$  (not just  $\geq 0$ )       $\theta^T x \geq \cancel{0} \quad 1$

if  $y = 0$ , the desired output is  $\theta^T x \leq -1$  (not just  $< 0$ )       $\theta^T x \leq \cancel{0} \quad -1$

$\theta^T x$  ideally should be considerably larger than or less than 0; consequentially,  $\theta^T x$  should be greater than +1 or less than -1: **built in safety margin factor**

svm decision boundary

$$\min_{\theta} \frac{1}{m} C \sum_{i=1}^m y^{(i)} \text{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0(\theta^T x^{(i)}) + \frac{1}{2} \sum_{j=1}^n \theta_j^2$$

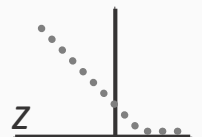
= 0

if  $C$  is a very large value (i.e.  $C = 100,000$ ), the optimization objective will be highly motivated to make the first term in the objective = 0 during minimization

whenever  $y^{(i)} = 1$ :

to ensure  $\text{cost}_1(z)$  then:  $\theta^T x \geq 1$

$$\cancel{\min C \times 0} + \frac{1}{2} \sum_{j=1}^n \theta_j^2$$

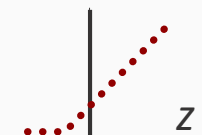


whenever  $y^{(i)} = 0$

to ensure  $\text{cost}_0(z)$  then:  $\theta^T x \leq -1$

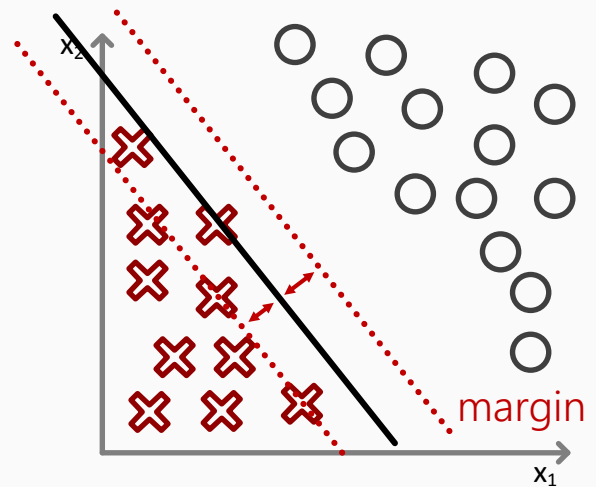
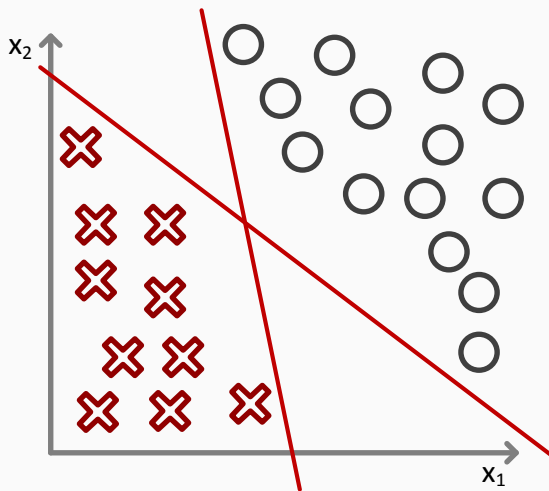
subject to:  $\theta^T x \geq 1$  if  $y^{(i)} = 1$

and  $\theta^T x \leq -1$  if  $y^{(i)} = 0$



## svm decision boundary: linearly separable case

linear boundaries can exist as many functions to classify the data as shown below (left):

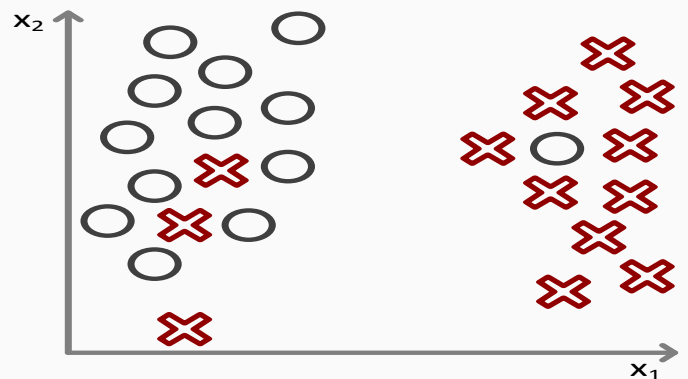


the above examples (left) are not natural boundaries and are typically not good choices for a learning algorithm. **support vector machines** will instead determine a more robust decision boundary (right). the svm decision boundary performs better at separating the positive(+) and negative(-) examples than the latter. mathematically, the svm has a much **larger margin** (shown in dotted line) which represents some minimum distance from any of the training examples: thus referred to as a **large margin classifier**

example

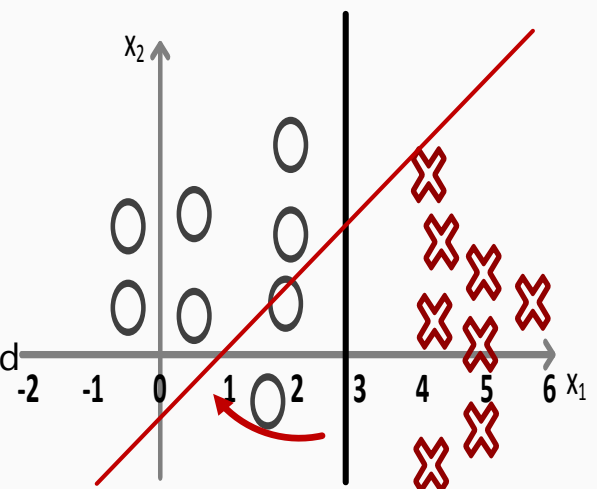
in the training set (right), "x" denotes positive examples ( $y=1$ ) and "o" denotes negative examples ( $y=0$ ). training an svm to predict 1 when  $\theta_0 + \theta_1 x_1 + \theta_2 x_2 \geq 0$  might return values for  $\theta_0$ ,  $\theta_1$ , and  $\theta_2$  of:

$$\theta_0 = -3, \theta_1 = 1, \text{ and } \theta_2 = 0$$



## large margin classifier in presence of outliers

if  $C$  is very large, the svm will adopt the new decision boundary (red line). a small or nominal  $C$  value will maintain the optimal decision boundary (black line). the svm will also choose the optimal decision boundary in the presence of nonlinear data (outliers  $\text{red 'x'}$   $\text{black 'o'}$  within the clustered data). note:  $C$  plays a role similar to  $\frac{1}{\lambda}$



# mathematics behind large margin classification

how the optimization problem of support vector machines lead to large margin classifiers

vector inner product

with two vectors  $u$  and  $v$ :

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$u^T v = ?$$

$\|u\|$  = euclidean length of vector  $u$

$$= \sqrt{u_1^2 + u_2^2} \in \mathbb{R}$$

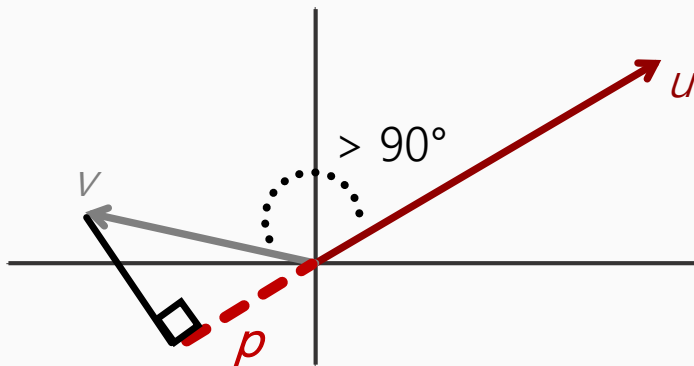
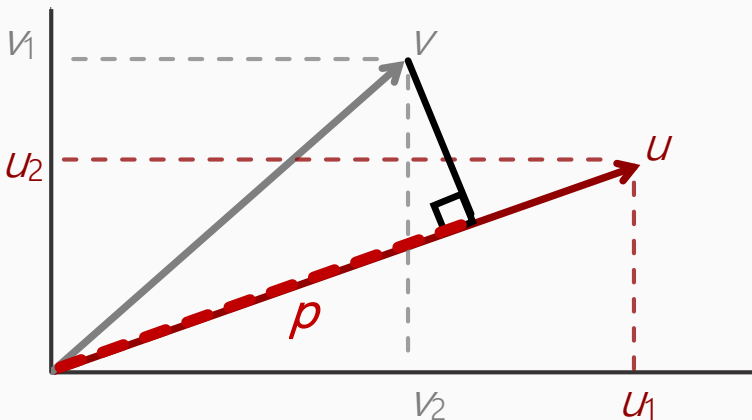
take an orthogonal (90°) projection of  $v$  down to  $u$  and measure the resulting length (red dashed line) of  $p$

$p$  = length of the projection  $v$  onto  $u$

$$u^T v = p \times \|u\| \quad p \in \mathbb{R}$$

$$= u_1 v_1 + u_2 v_2$$

computing  $v^T u$  would yield same result  
note that  $p$  can be negative (shown left)



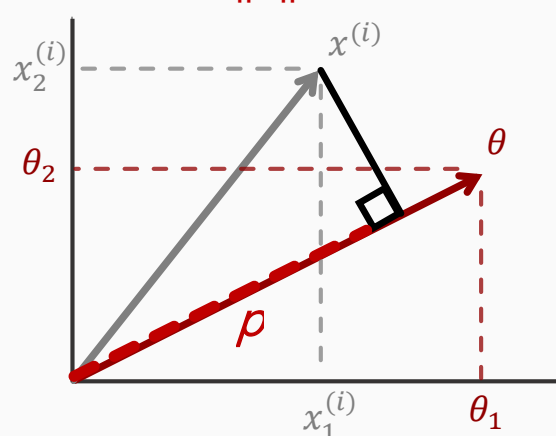
svm decision boundary

$$\min_{\theta} \frac{1}{2} \sum_{j=1}^n \theta_j^2 = \frac{1}{2} (\theta_1^2 + \theta_2^2) = \frac{1}{2} \left( \underbrace{\sqrt{\theta_1^2 + \theta_2^2}}_{= \|\theta\|} \right)^2 = \frac{1}{2} \|\theta\|^2$$

subject to:  $\theta^T x \geq 1$  if  $y^{(i)} = 1$   
and  $\theta^T x \leq -1$  if  $y^{(i)} = 0$

simplification:  $\theta_0 = 0, n = 2$

$$\begin{aligned} \theta^T x^{(i)} &= p^{(i)} \times \|\theta\| \\ &= \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} \end{aligned}$$



$$\begin{aligned} &u^T v \\ &\downarrow \downarrow \\ &\theta^T x^{(i)} = ? \end{aligned}$$

considering the training example below with simplification  $\theta_0 = 0$   
the simplification only utilizes svm decision boundaries that pass through the origin

determine the decision boundary that the svm will select:

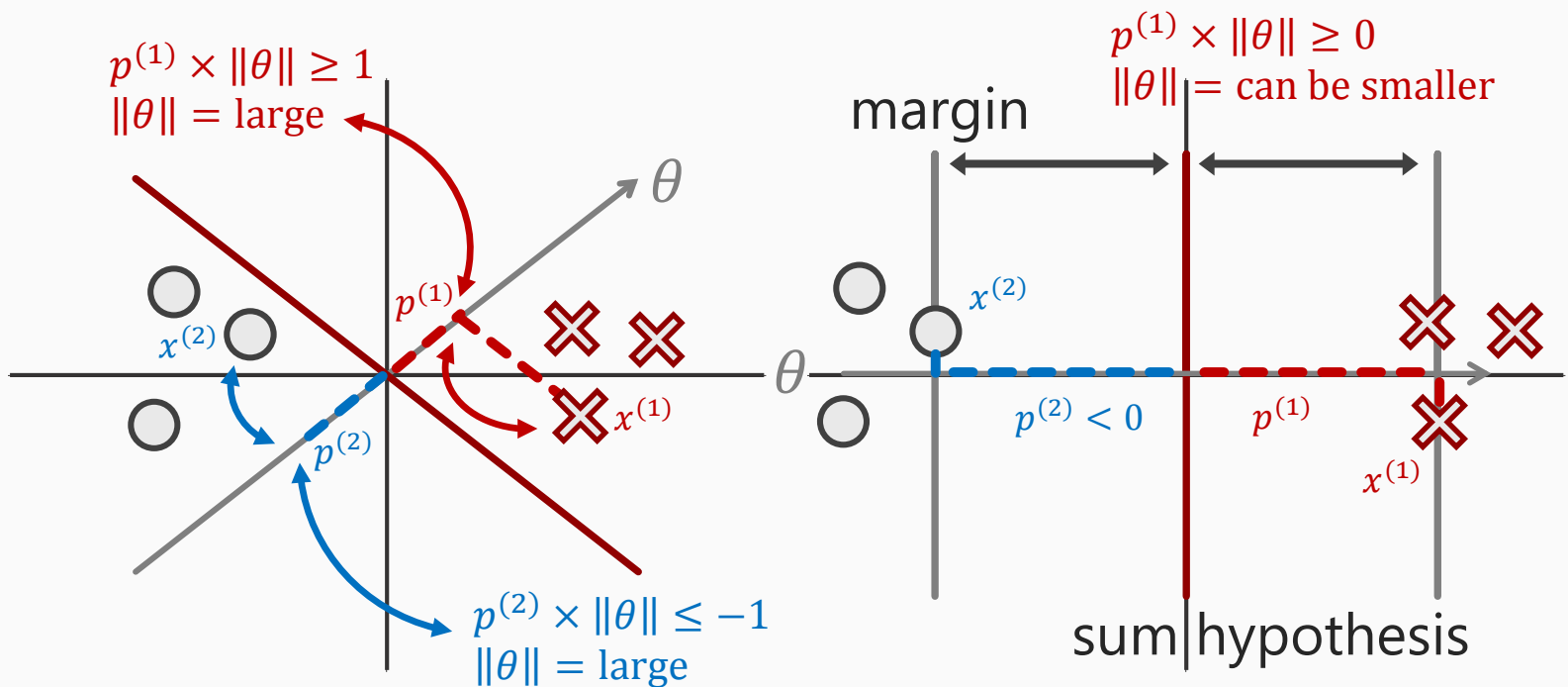
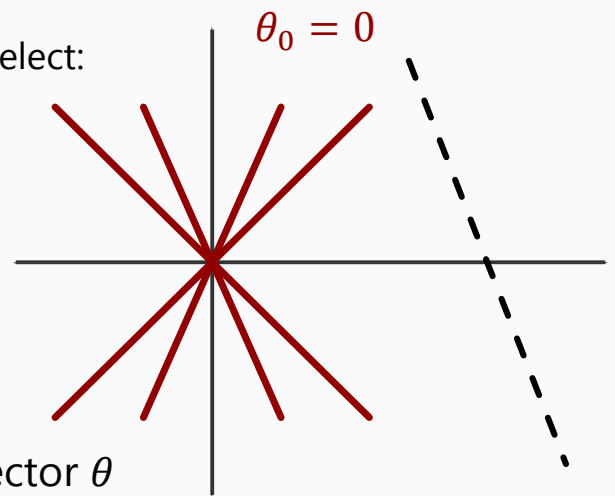
$$\min_{\theta} \frac{1}{2} \sum_{j=1}^n \theta_j^2 = \frac{1}{2} \|\theta\|^2$$

subject to:  $p^{(i)} \times \|\theta\| \geq 1$  if  $y^{(i)} = 1$

$p^{(i)} \times \|\theta\| \leq -1$  if  $y^{(i)} = -1$

where  $p^{(i)}$  is the projection of  $x^{(i)}$  onto the vector  $\theta$

with simplification for illustration  $\theta_0 = 0$  (intersects origin as seen above)



the red line in the left figure represents a poor decision boundary choice that is avoided through the use of svm. the grey line represents the svm tactics through projection from the points  $x^{(1)}$  and  $x^{(2)}$ . in this example,  $\|\theta\|$  and  $\|\theta\|$  would need to be large considering that  $p^{(1)}$  and  $p^{(2)}$  are small. the red line in the right figure represents a correct decision boundary choice that is achieved through the use of svm. it is shown the projections from the data  $x^{(1)}$  and  $x^{(2)}$  create much larger lengths of  $p^{(1)}$  and  $p^{(2)}$  creating a larger margin which all for a smaller value for  $\|\theta\|$  and  $\|\theta\|$