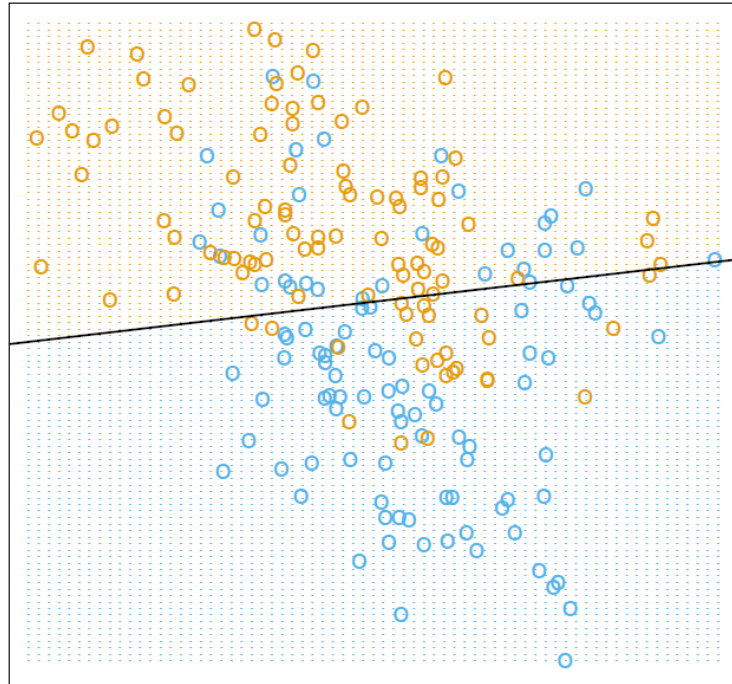


# Linear Classifiers

Linear Regression of 0/1 Response



Some slides from:  
Andrew Ng

CSC411: Machine Learning and Data Mining, Winter 2018

Michael Guerzhoy and Lisa Zhang

# Classification vs. Regression

- Classification: for the example  $(x_1, x_2, \dots, x_n)$  predict the label  $y$  (e.g., face recognition)
- Regression: for the example  $(x_1, x_2, \dots, x_n)$  predict a real number  $y$  (e.g., house price prediction)

# Classification with two classes

- If there are only two classes, transform, e.g.,  
orange => 1  
blue => 0  
to turn the classification problem into a regression problem

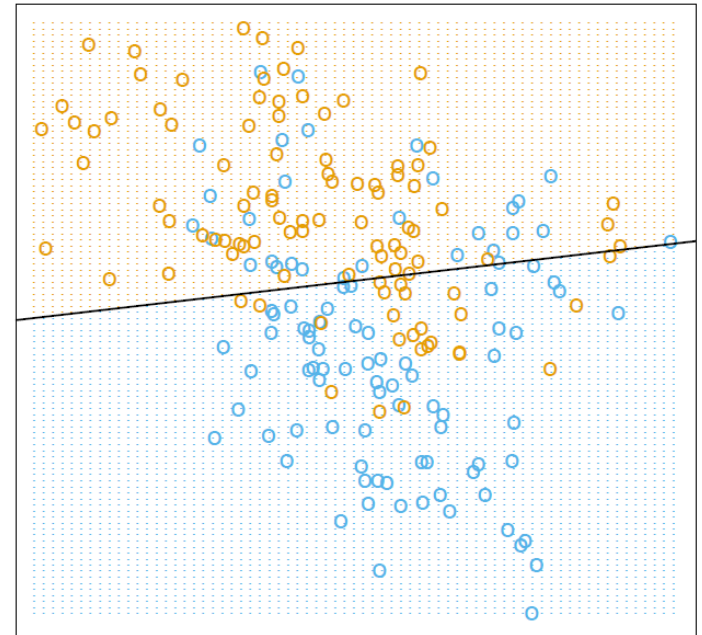
- Find the best

$$h_{\theta}(x) = \theta^T x$$

- Predict:

$$\begin{cases} 1, & h_{\theta}(x) > 0.5 \\ 0, & \text{otherwise} \end{cases}$$

Linear Regression of 0/1 Response



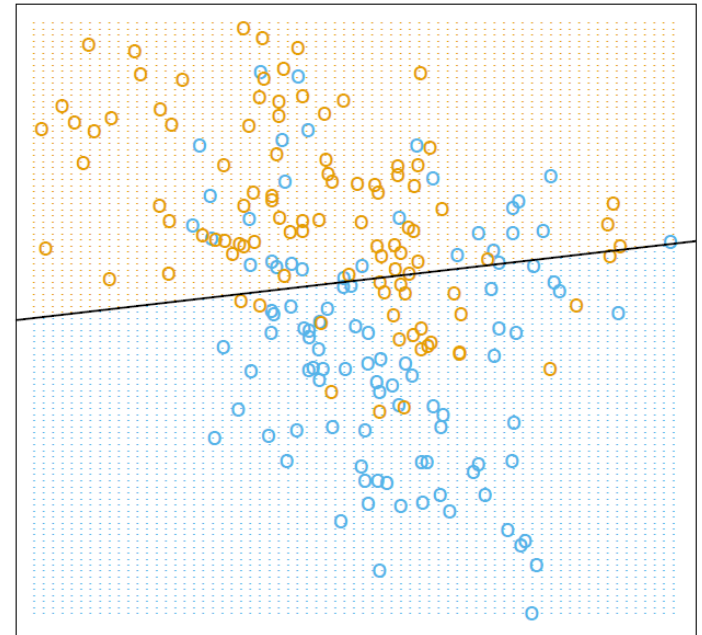
$$\theta_1 x_1 + \theta_2 x_2 \text{ (can add in } \theta_0 \text{)}$$

What is the equation of the decision boundary?

# But what about the loss function?

(Loss function = cost function)

Linear Regression of 0/1 Response



What is the equation of the decision boundary?

# Attempt #1:

- Quadratic loss, as in Linear Regression.

$$\sum_{i=1}^m (y^{(i)} - \theta^T x^{(i)})^2$$

- What is the problem with this loss function?

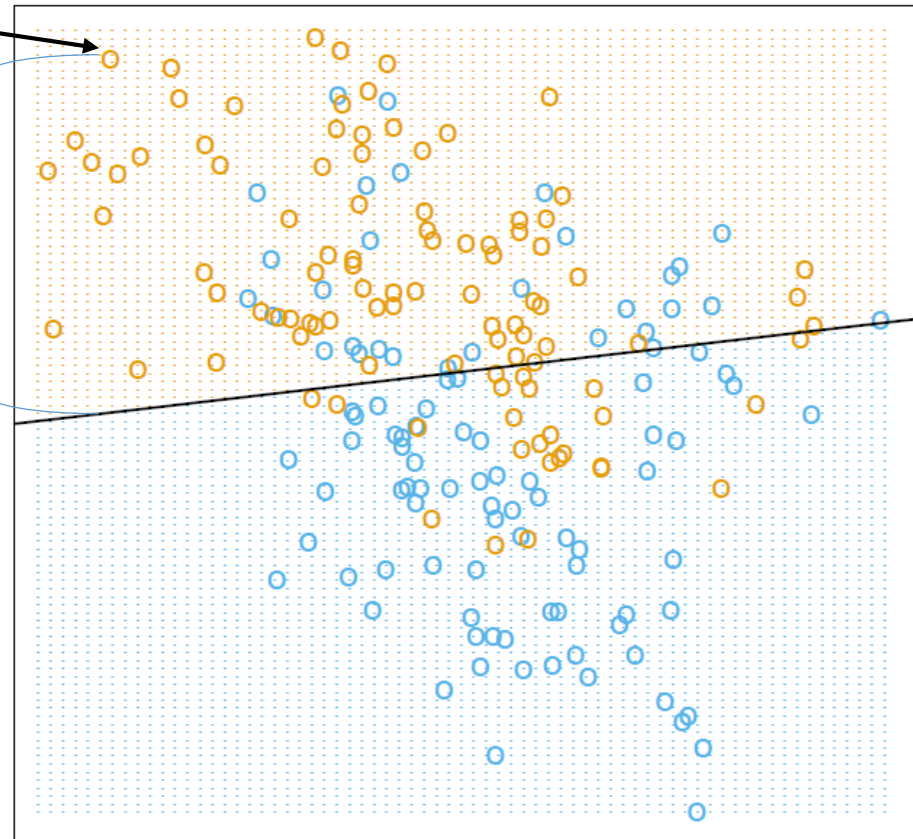
# Attempt #1:

How much does this training data contribute to the loss?

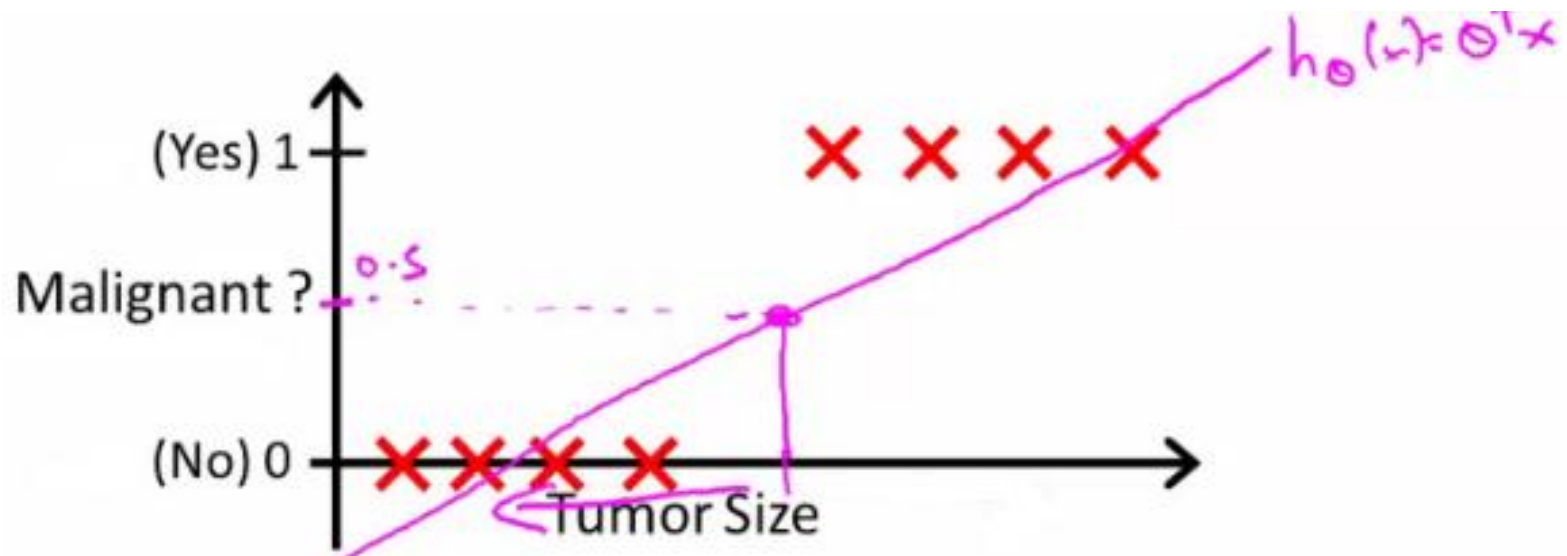
A lot!

Quadratic loss penalizes data well within the decision boundary.

Linear Regression of 0/1 Response



# Example in 1D



Even with perfect classification,  
Loss is still nonzero (and can be high!)

# Attempt #2:

- Classification error or 0-1 loss.

$$\sum_{i=1}^m I[y^{(i)}, t^{(i)}]$$

$t^{(i)} = \begin{cases} 1, & h_{\theta}(x^{(i)}) > 0.5 \\ 0, & \text{otherwise} \end{cases}$

- Where  $I$  is the indicator function:

$$I[y, t] = \begin{cases} 1, & y = t \\ 0, & \text{otherwise} \end{cases}$$

- What is the problem with this loss function?

Not continuous.

Hard to optimize.

Cannot use gradient descent (Why?

On the board)

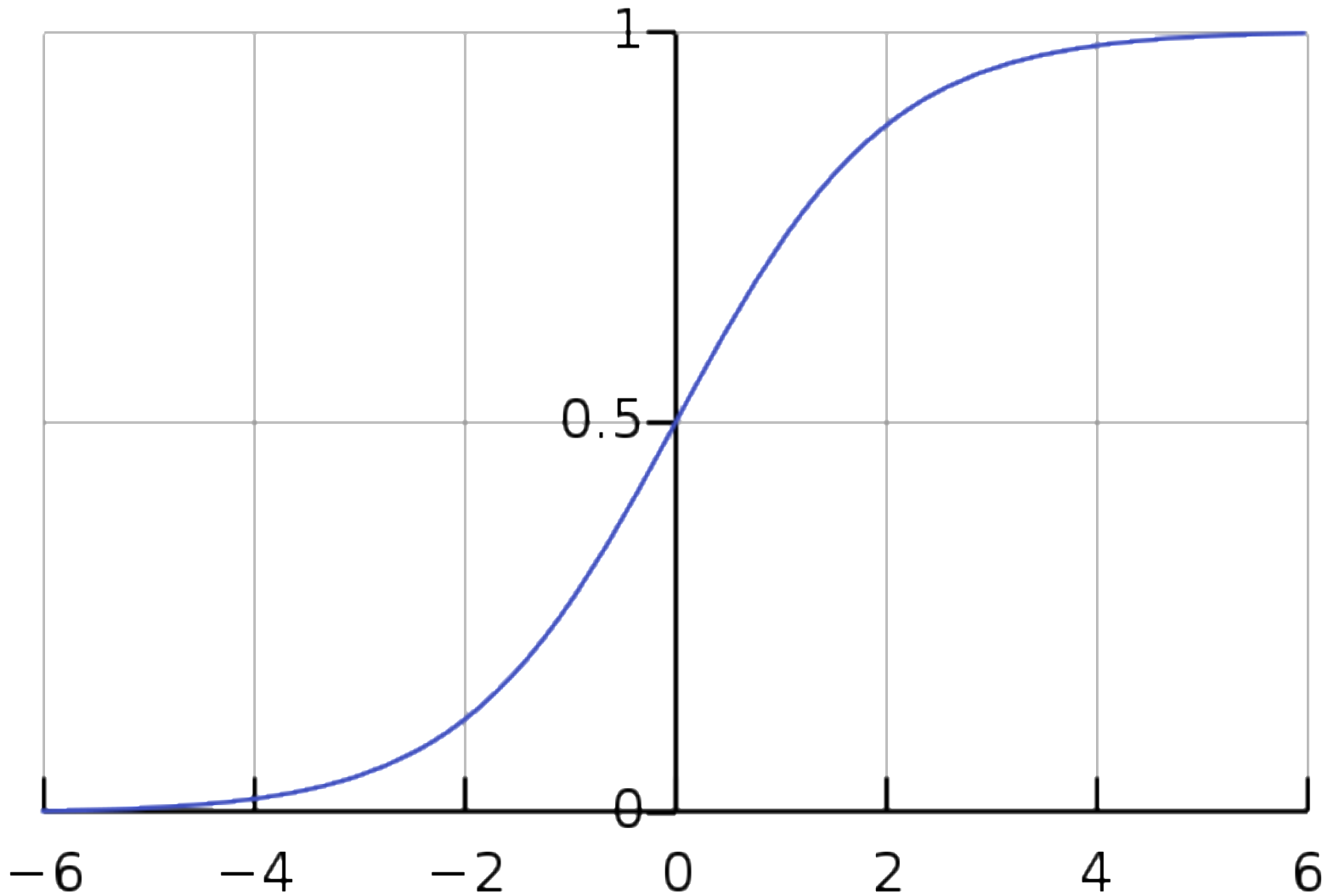


# Attempt #3:

- Problem with linear regression (quadratic loss): **Predictions are allowed to take arbitrary real values!**
- Problem with linear regression (0-1 loss): **Hard to optimize!**
- Apply a nonlinearity or activation function: **sigmoid function:**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

# Sigmoid function



# Revised Setup

- If there are only two classes, transform, e.g.,  
orange => 1  
blue => 0  
to turn the classification problem into a regression problem

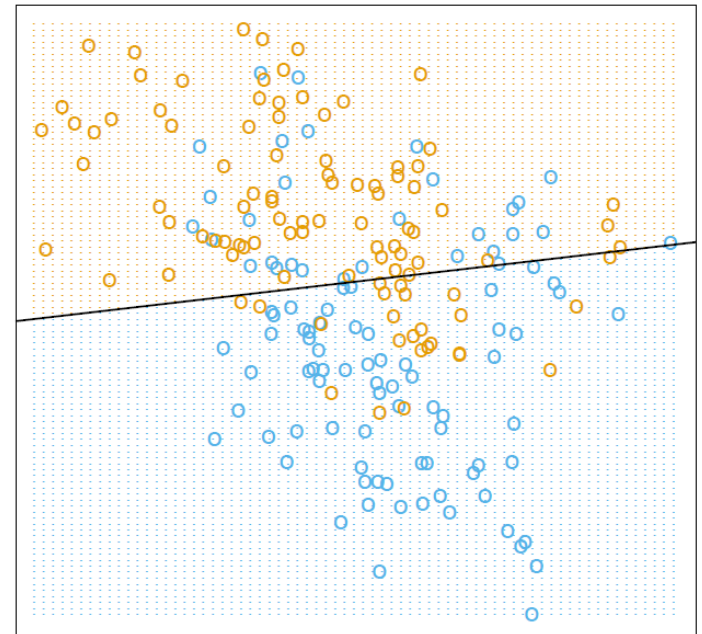
- Model:

$$h_{\theta}(x) = \sigma(\theta^T x)$$

- Where

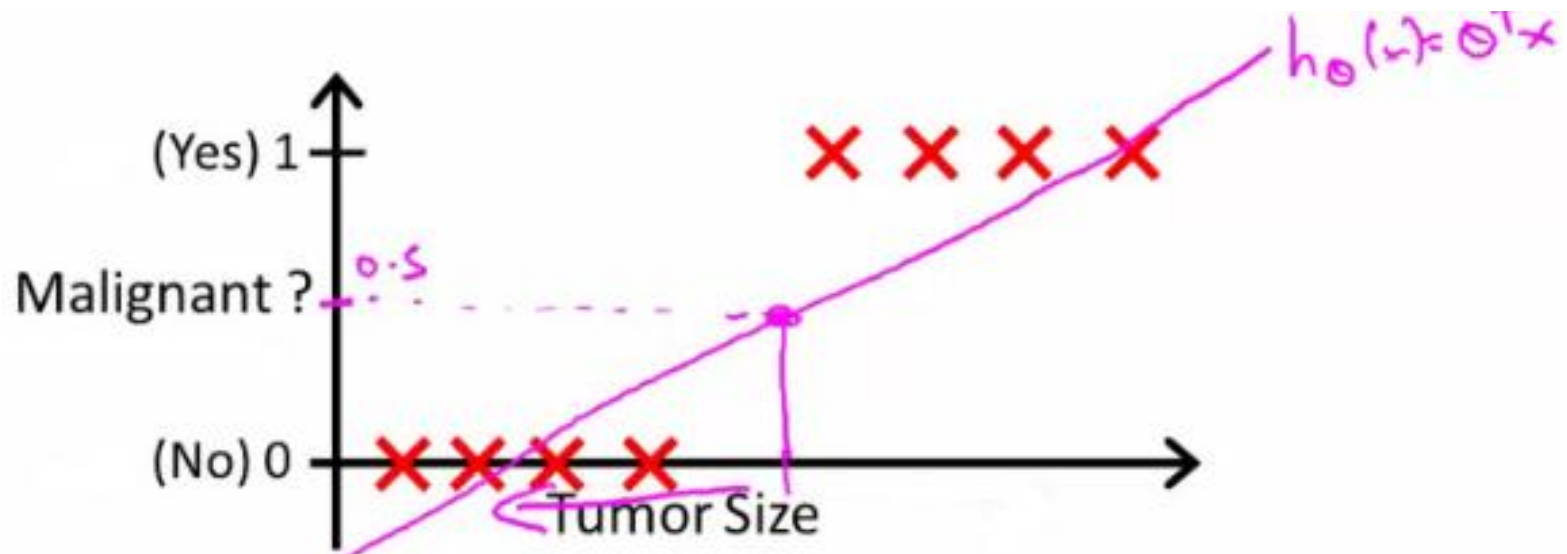
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Linear Regression of 0/1 Response



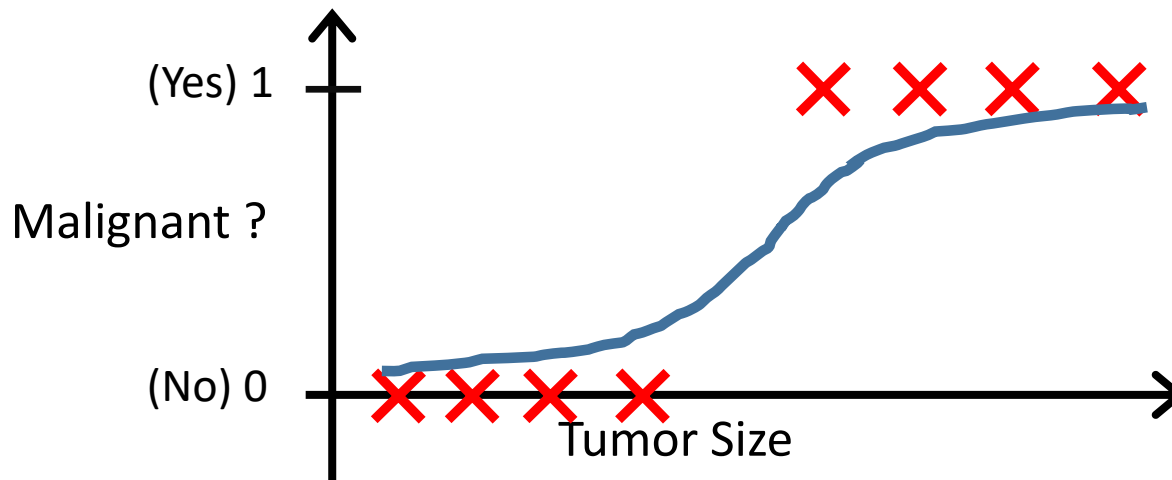
What is the equation of the decision boundary?

# Reminder: linear prediction in 1D



Even with perfect classification,  
Loss is still nonzero (and can be high!)

# Example in 1D: applying the sigmoid



# What about the loss?

- Square Loss?

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m \left( y^{(i)} - \sigma(\theta^T \mathbf{x}^{(i)}) \right)^2$$

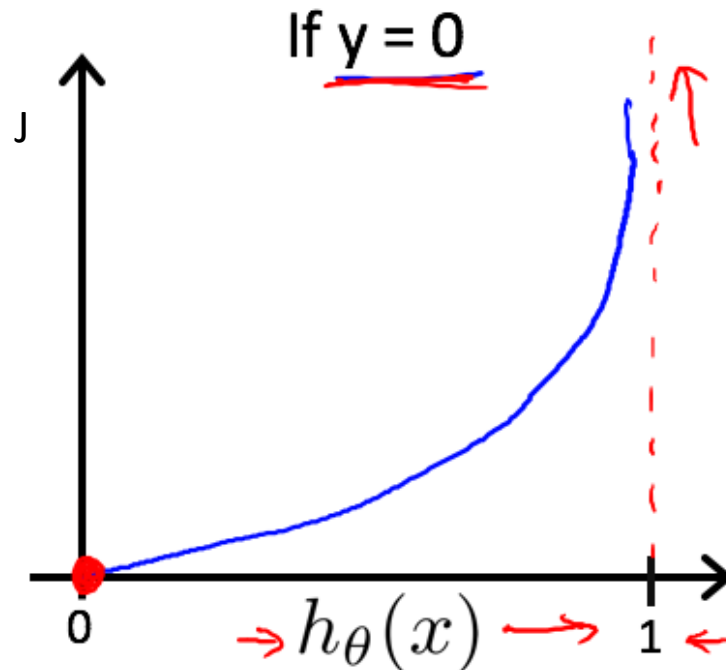
- On the board:
  - If  $h_{\theta}(\mathbf{x}) = \sigma(\theta^T \mathbf{x})$  is very close to 0 or 1, then the gradient of the loss is close to zero!
  - Why is that a problem?
  - Summary:

$$\nabla J(\theta) = \frac{1}{m} \sum_{i=1}^m \left( y - \sigma(\theta^T \mathbf{x}^{(i)}) \right) \sigma(\theta^T \mathbf{x}^{(i)}) \left( 1 - \sigma(\theta^T \mathbf{x}^{(i)}) \right) \mathbf{x}^{(i)}$$

# What loss should we use?

- We will use **Cross Entropy Loss**

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m -y^{(i)} \log \sigma(\theta^T \mathbf{x}^{(i)}) + (1 - y^{(i)}) (-\log(1 - \sigma(\theta^T \mathbf{x}^{(i)})))$$

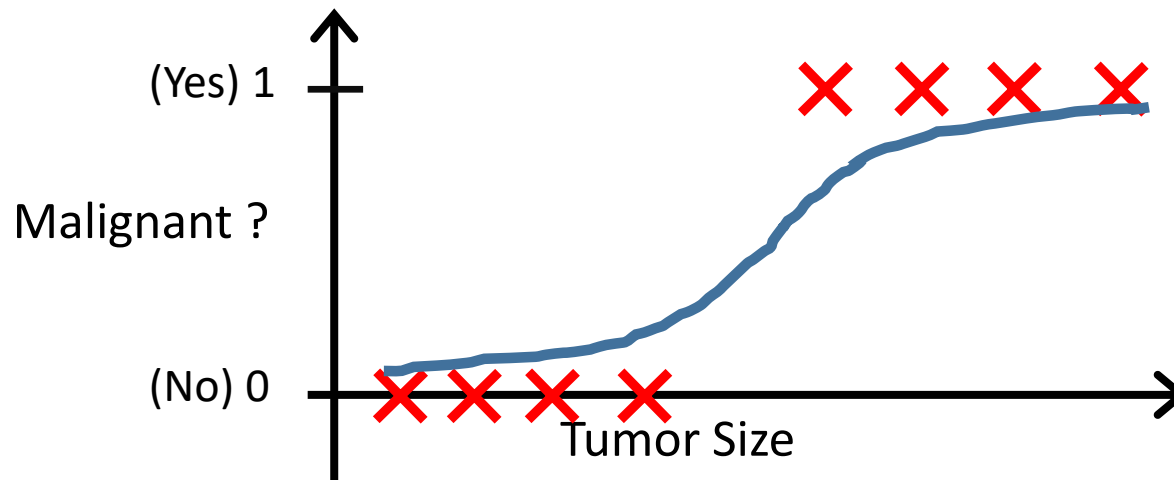


# Why Cross Entropy?

- Where does this come from?
- As in Linear Regression, we will use a probabilistic interpretation



# Predictions look like probabilities



Logistic Regression


# Logistic Regression

- Assume the data is generated according to

$$y^{(i)} = 1 \text{ with probability } \frac{1}{1 + \exp(-\theta^T x^{(i)})}$$

$$y^{(i)} = 0 \text{ with probability } \frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})}$$

- This can be written concisely as:

odds 

$$\frac{P(y^{(i)}=1|x^{(i)},\theta)}{P(y^{(i)}=0|x^{(i)},\theta)} = \exp(\theta^T x^{(i)})$$

(exercise)

# Logistic Regression: Likelihood

- $$P(y^{(i)} = 1 | x^{(i)}, \theta) = \left( \frac{1}{1 + \exp(-\theta^T x^{(i)})} \right)^{y^{(i)}} \left( \frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})} \right)^{1 - y^{(i)}}$$

(just a trick that works because  $y^{(i)}$  is either 1 or 0)

- $$P(y|x, \theta) = \prod_{i=1}^m \left( \frac{1}{1 + \exp(-\theta^T x^{(i)})} \right)^{y^{(i)}} \left( \frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})} \right)^{1 - y^{(i)}}$$
- $$\log P(y|x, \theta) = \sum_{i=1}^m y^{(i)} \log \left( \frac{1}{1 + \exp(-\theta^T x^{(i)})} \right) + (1 - y^{(i)}) \log \left( \frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})} \right)$$

# Logistic Regression: Learning and Testing

- Learning: find the  $\theta$  that maximizes the log-likelihood:

$$\sum_{i=1}^m y^{(i)} \log\left(\frac{1}{1 + \exp(-\theta^T x^{(i)})}\right) + (1 - y^{(i)}) \log\left(\frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})}\right)$$

- For  $x$  in the test set, compute

$$P(y = 1|x, \theta) = \frac{1}{1 + \exp(-\theta^T x)}$$

- Predict that  $y = 1$  if  $P(y = 1|x, \theta) > .5$

# Logistic Regression: Decision Surface

- Predict  $y = 1$  if  $\frac{1}{1+\exp(-\theta^T x)} > 0.5$

$$\Leftrightarrow -\theta^T x < 0$$

$$\Leftrightarrow \theta^T x > 0$$

- The decision surface is  $\theta^T x = 0$ , a hyperplane

# Logistic Regression

- Outputs the probability of the datapoint's belonging to a certain class:

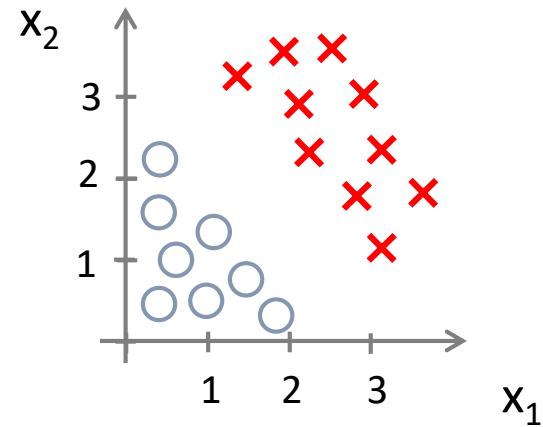
$$y^{(i)} = 1 \text{ with probability } \frac{1}{1 + \exp(-\theta^T x^{(i)})}$$

$$y^{(i)} = 0 \text{ with probability } \frac{\exp(-\theta^T x^{(i)})}{1 + \exp(-\theta^T x^{(i)})}$$

(compare with linear regression)

- Linear decision surface
- Probably the first thing you would try in a real-world setting for a classification task

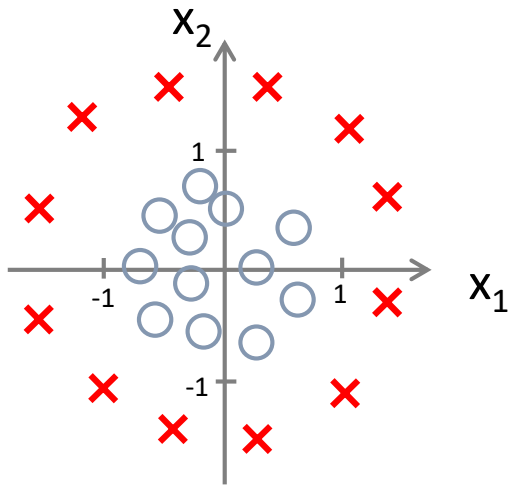
# Decision boundary shapes



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

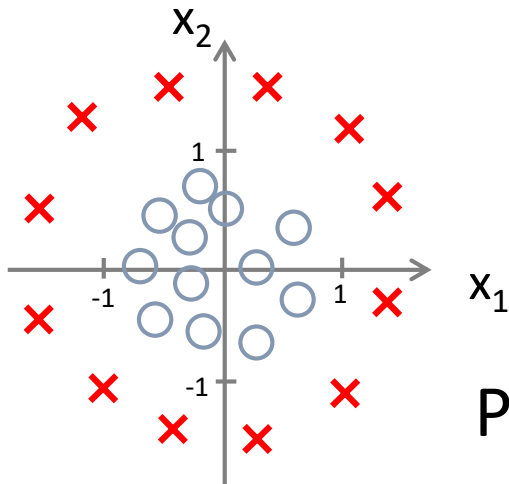
Predict  $y = 1$  if  $-3 + x_1 + x_2 \geq 0$

# Decision boundary shapes





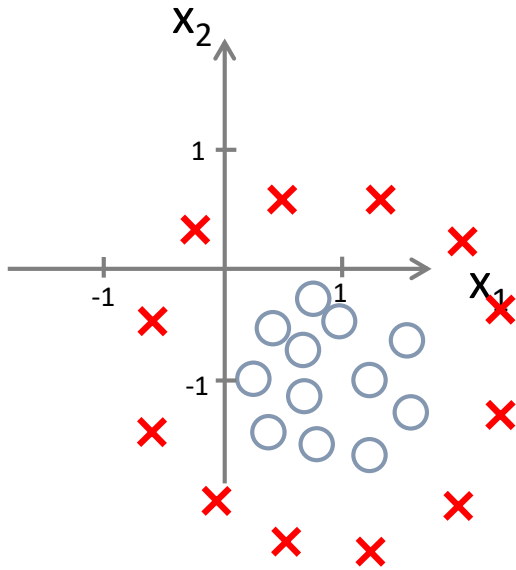
# Decision boundary shapes



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

Predict  $y = 1$  if  $-1 + x_1^2 + x_2^2 \geq 0$

# What is the equation for a good decision boundary?



# Multiclass Classification

Email foldering/tagging : Work, Friends, Family, Hobby

$y = 1$     $y = 2$     $y = 3$     $y = 4$

Features:  $x_1$ : 1 if “extension” is in the email, 0 otherwise

$x_2$ : 1 if “dog” is in the email, 0 otherwise

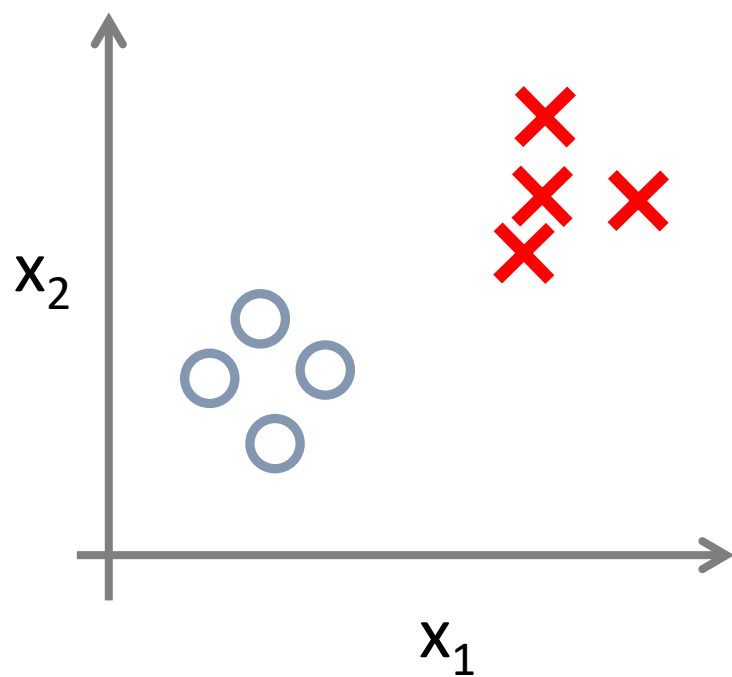
...

Medical diagrams: Not ill, Cold, Flu

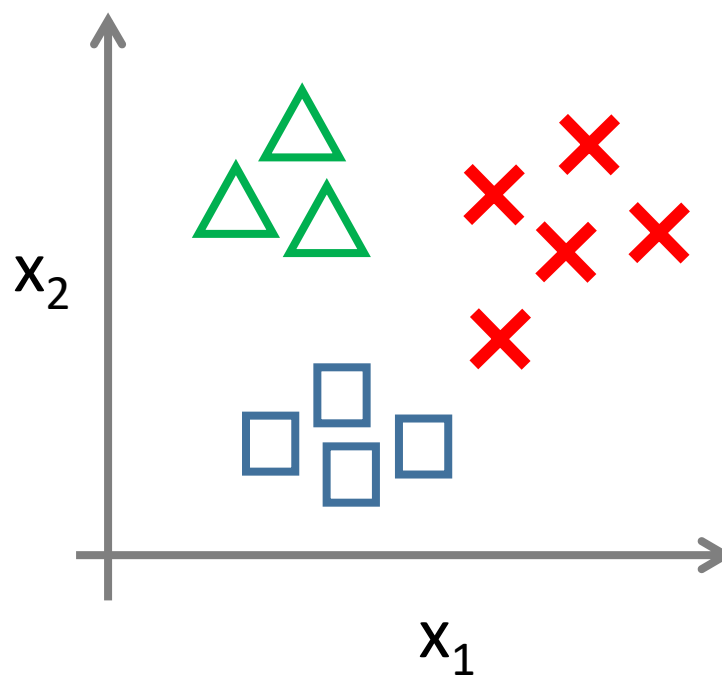
$y = 1$     $y = 2$     $y = 3$

Features: temperature, cough presence, ...

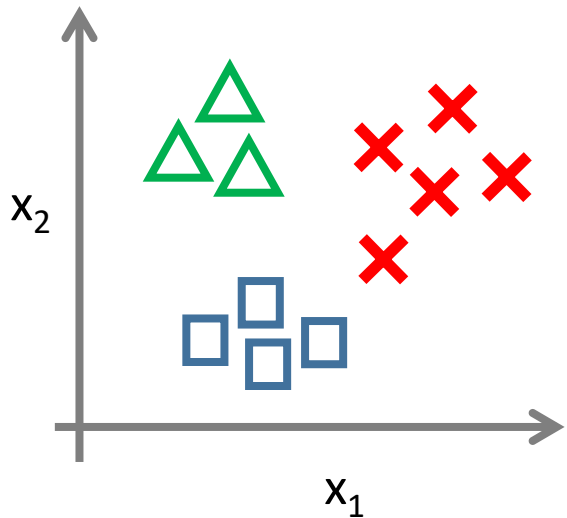
Binary classification:





Multi-class classification:




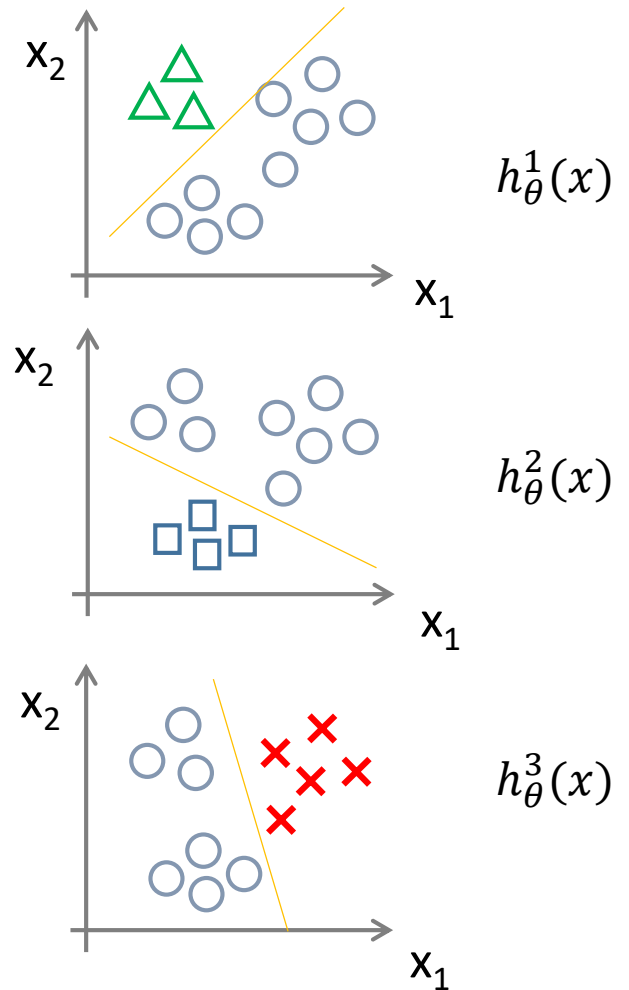
## One-vs-all (one-vs-rest):



Class 1: 

Class 2: 

Class 3: 



Output the  $i$  such that  $h_{\theta}^i(x)$  is the largest  
(Idea: a large  $h_{\theta}^i(x)$  means that the classifier is “sure”)