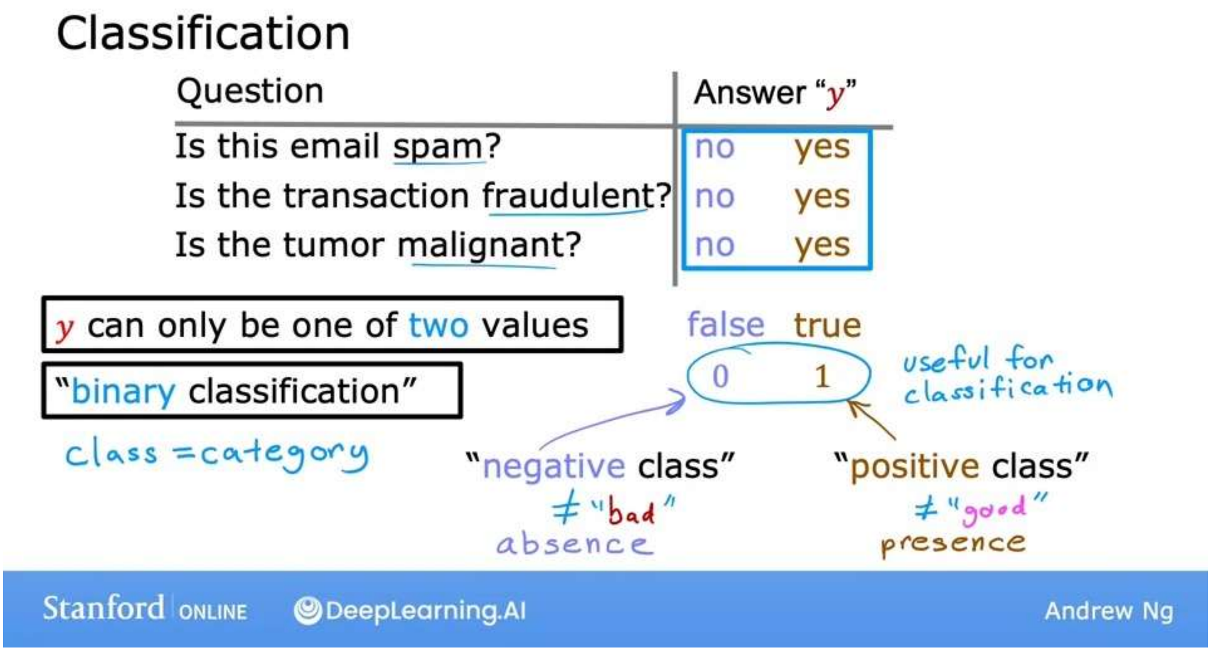
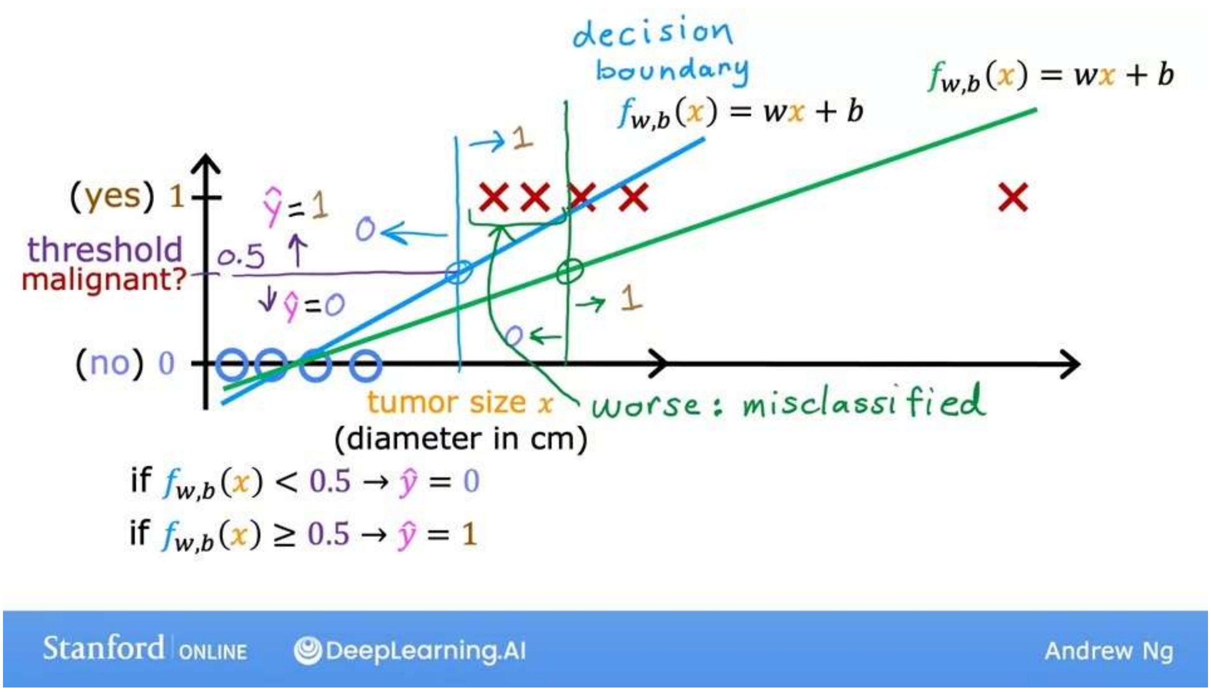
Supervised Machine Learning: Regression and Classification

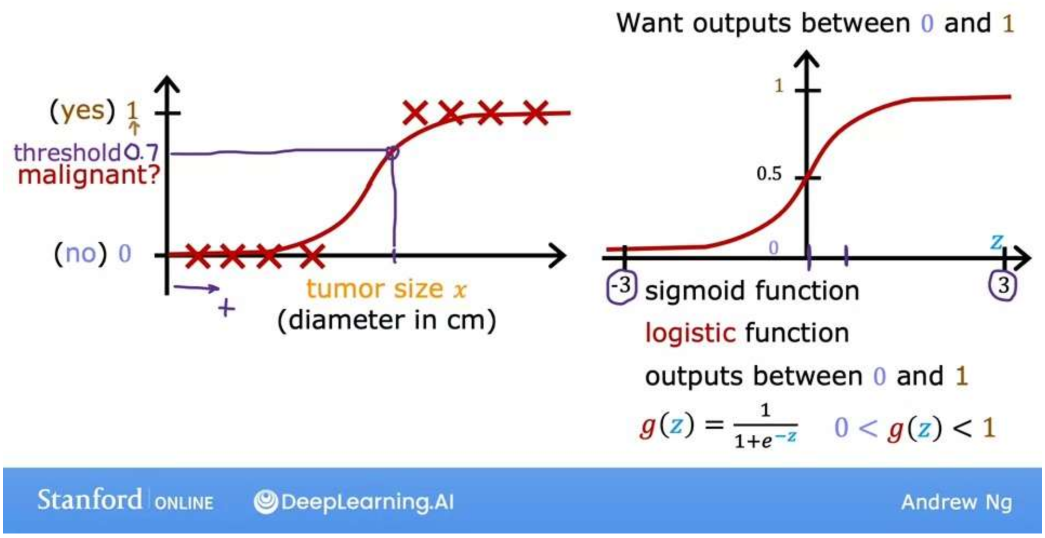
Week 3

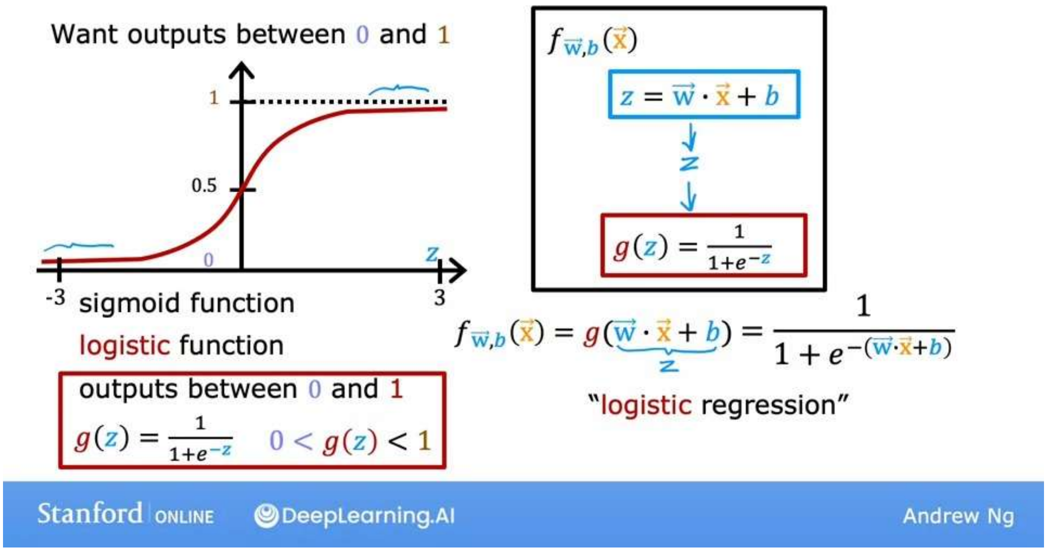
Classification with logistic regression

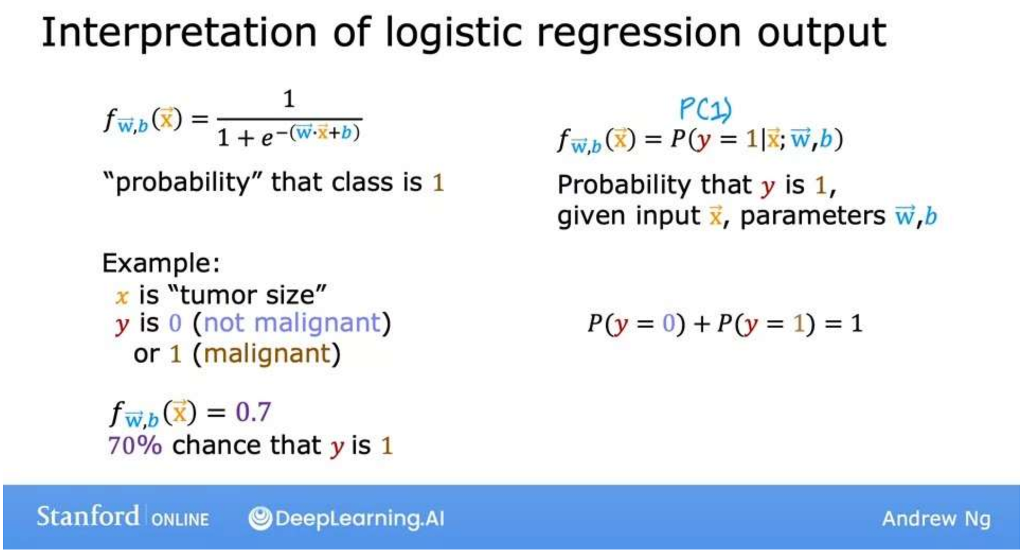




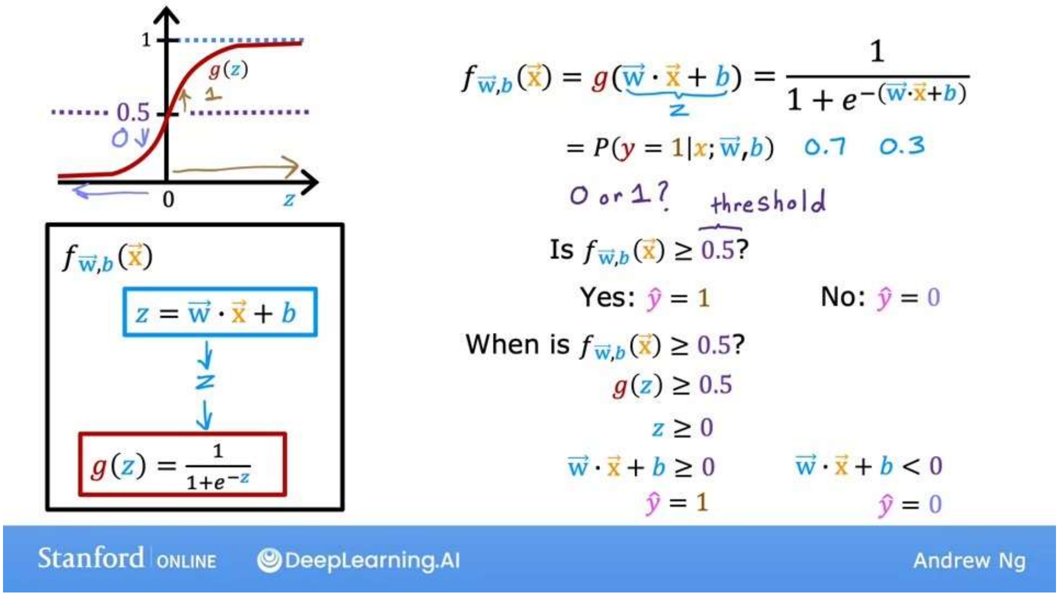
Logistic regression

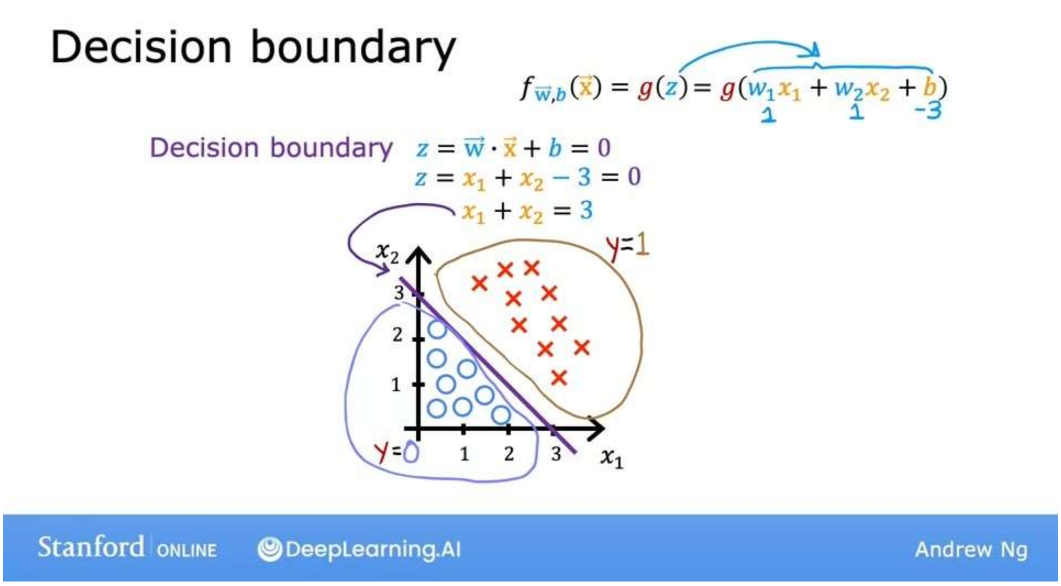


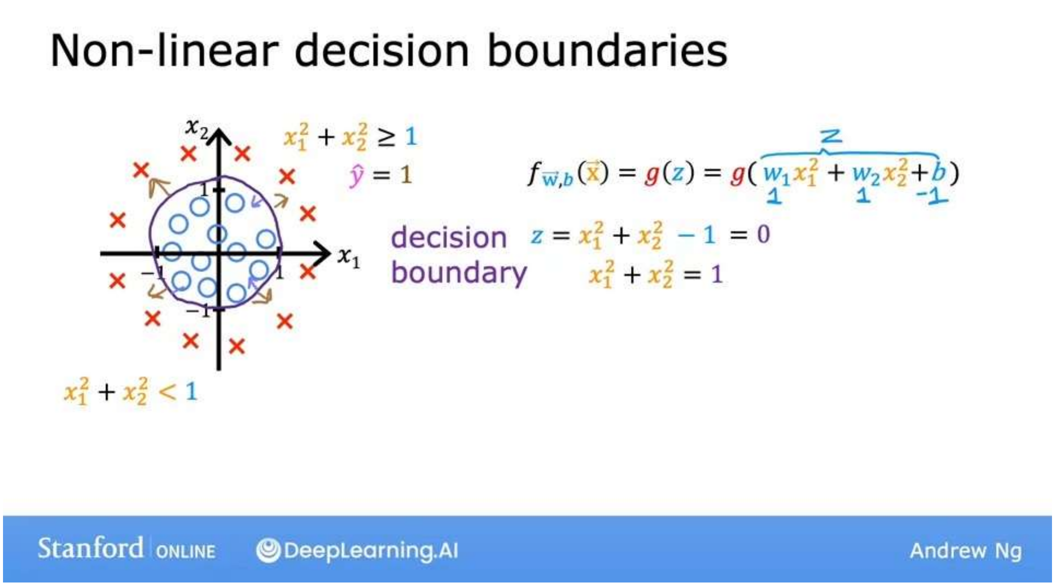




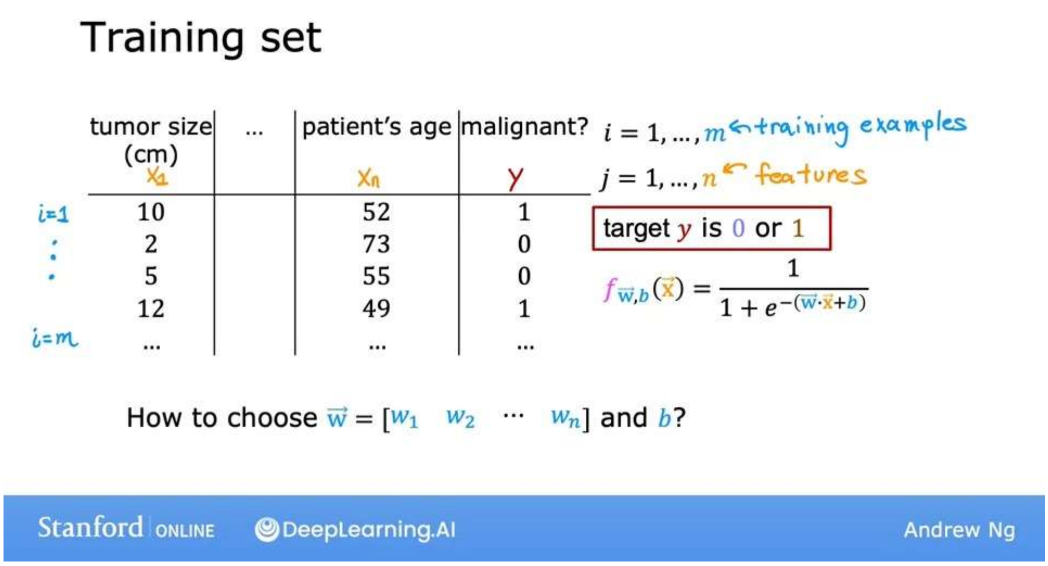
Decision boundary

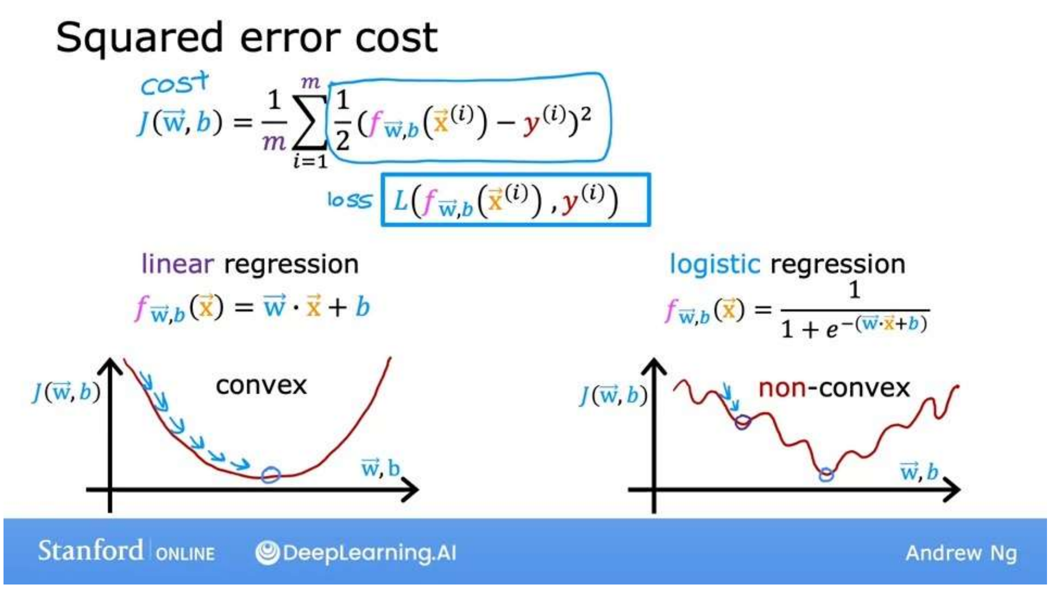


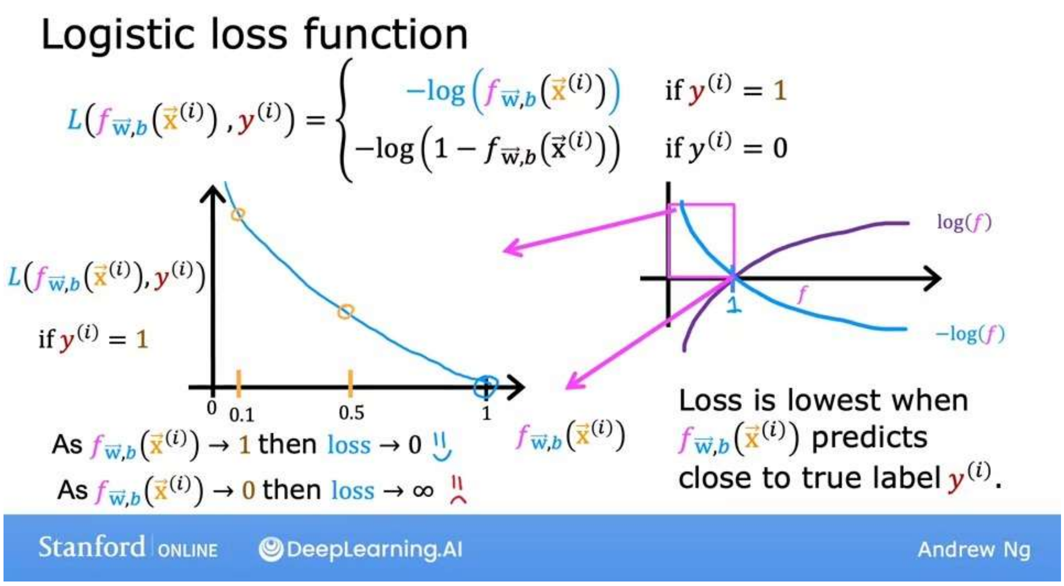




Cost function for logistic regression







Simplified Cost Function for Logistic Regression

Simplified loss function

$$L\left(f_{\vec{w},b}(\vec{x}^{(i)}), y^{(i)}\right) = \begin{cases} -\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) & \text{if } y^{(i)} = 1 \\ \boxed{-\log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right)} & \text{if } y^{(i)} = 0 \end{cases}$$

$$\boxed{L\left(f_{\vec{w},b}(\vec{x}^{(i)}), y^{(i)}\right) = -\underset{0}{\cancel{y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right)}} - \underset{(1-0)}{(1 - y^{(i)})}\log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right)}$$

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

$$L\left(f_{\vec{w},b}(\vec{x}^{(i)}), y^{(i)}\right) = -1 \log\left(f(\vec{x})\right)$$

if $y^{(i)} = 0$:

$$L\left(f_{\vec{w},b}(\vec{x}^{(i)}), y^{(i)}\right) = \underline{-(1-0)\log(1-f(\vec{x}))}$$

Gradient descent for logistic regression

Gradient descent

cost

$$J(\vec{w}, b) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) + (1 - y^{(i)})\log\left(1 - f_{\vec{w},b}(\vec{x}^{(i)})\right)\right]$$

repeat {

$j = 1 \dots n$

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w}, b) \qquad \frac{\partial}{\partial w_j} J(\vec{w}, b) = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)}$$

$$b = b - \alpha \frac{\partial}{\partial b} J(\vec{w}, b) \qquad \frac{\partial}{\partial b} J(\vec{w}, b) = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

} simultaneous updates

Gradient descent for logistic regression

repeat {

looks like linear regression!

$$w_j = w_j - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)}\right]$$

$$b = b - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)\right]$$

} simultaneous updates

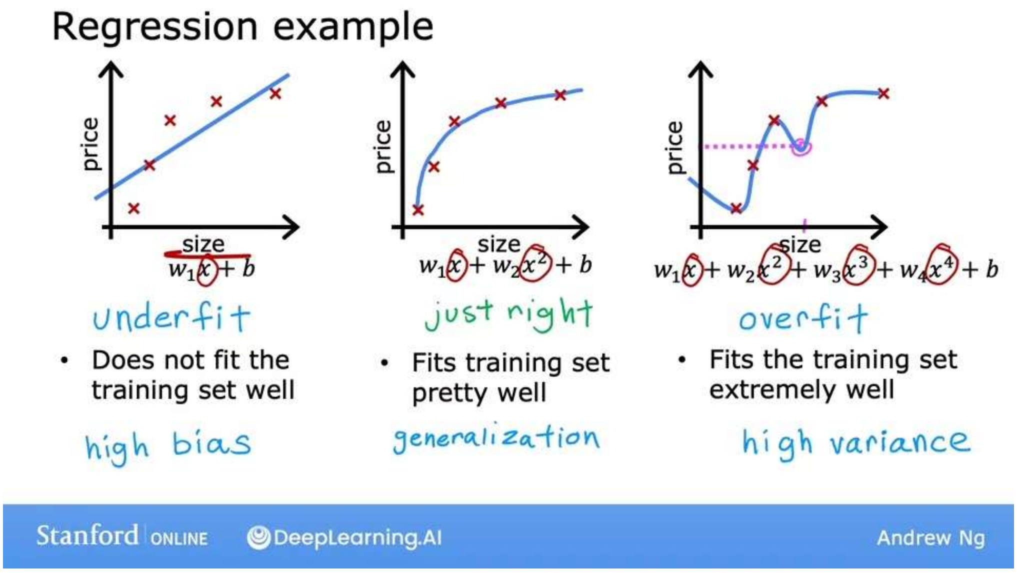
Same concepts:

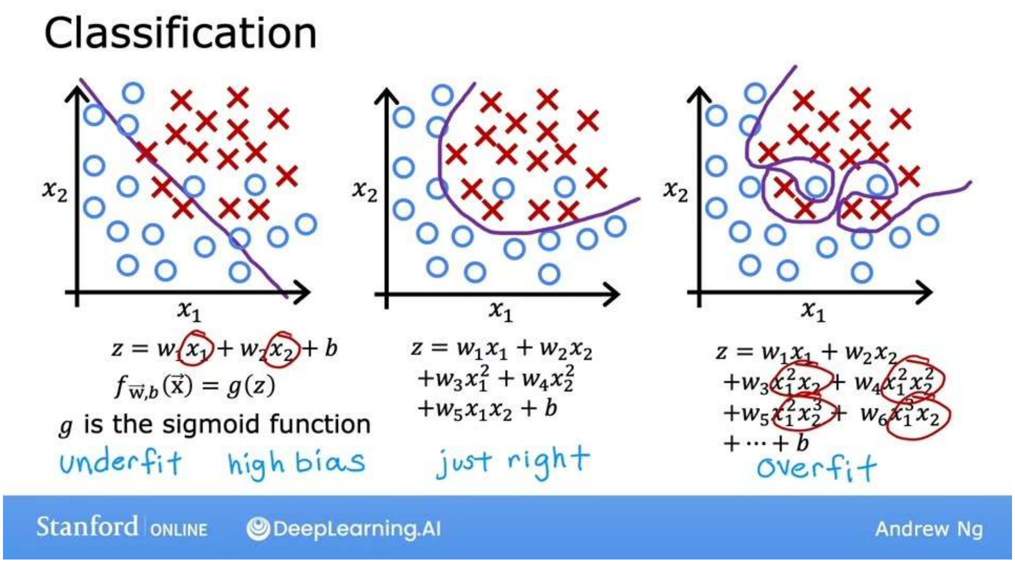
- Monitor gradient descent (learning curve)
- Vectorized implementation
- Feature scaling

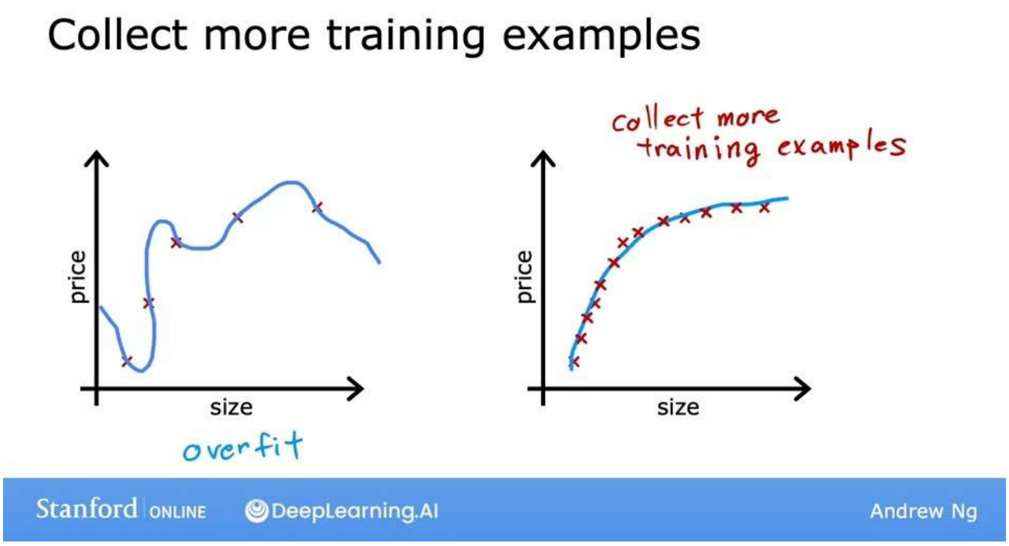
Linear regression $\quad f_{\vec{w},b}(\vec{x}) = \vec{w} \cdot \vec{x} + b$

Logistic regression $\quad f_{\vec{w},b}(\vec{x}) = \dfrac{1}{1 + e^{-(\vec{w} \cdot \vec{x} + b)}}$

Problem of overfitting







Select features to include/exclude

size	bedrooms	floors	age	avg income	...	distance to coffee shop	price
x_1	x_2	x_3	x_4	x_5		x_{100}	y

all features
+
insufficient data
↓
overfit

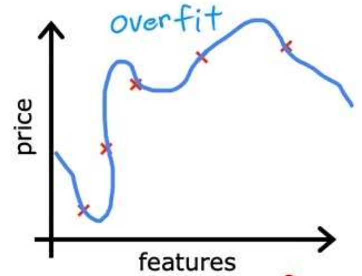
selected features
size
bedrooms
age
just right
feature selection

course 2

disadvantage
↓
useful features could be lost

Regularization

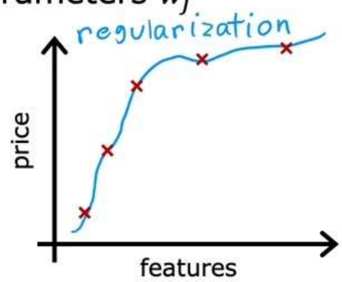
Reduce the size of parameters w_j



$$f(x) = 28x - 385x^2 + 39x^3 - \underbrace{\cancel{174}}_{0}x^4 + 100$$

large values for w_j

← eliminate feature



$$f(x) = 13x - 0.23x^2 + \underline{0.000014}x^3 - \underline{0.0001}\,x^4 + 10$$

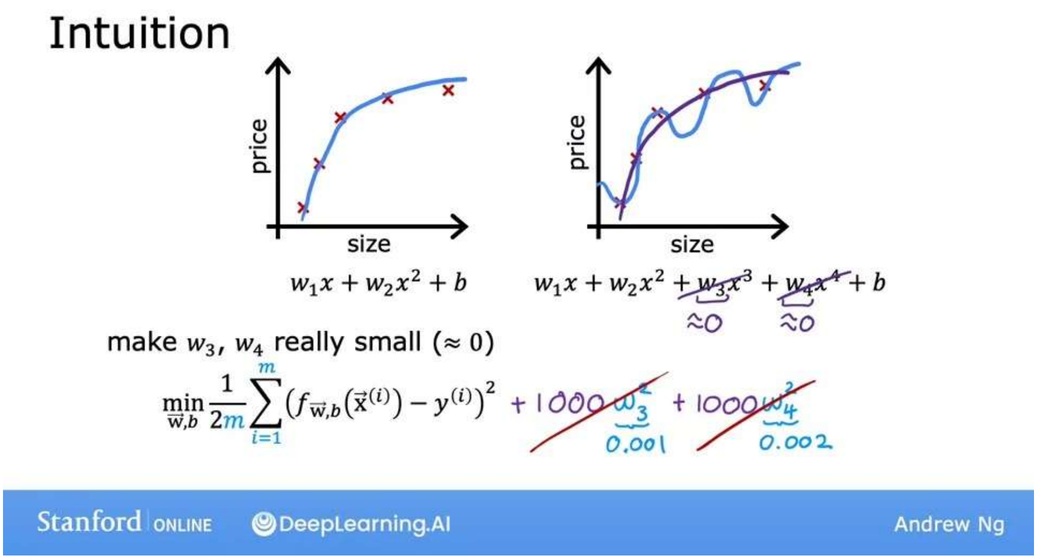
small values for w_j

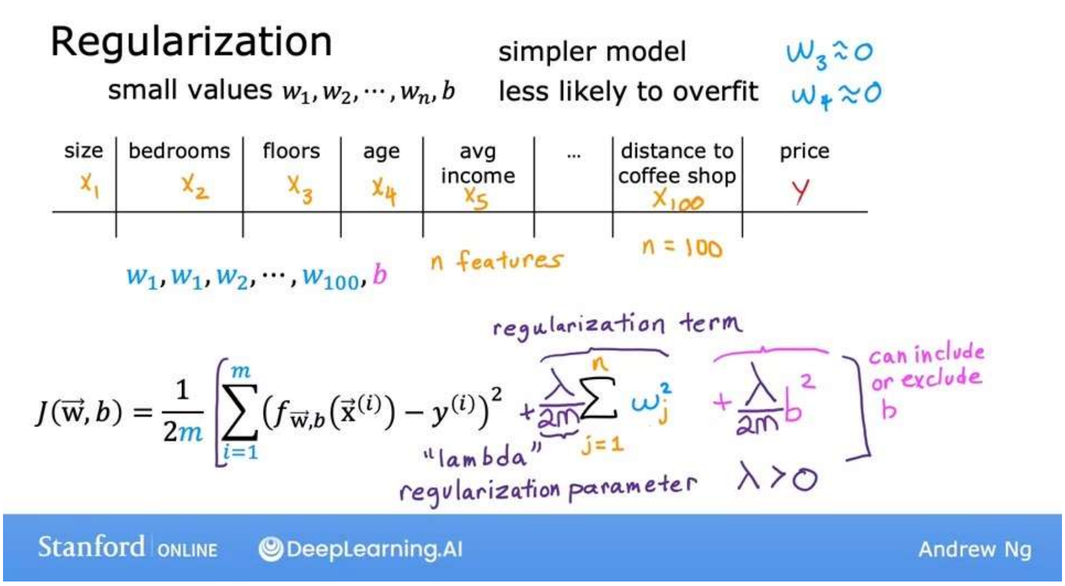
Addressing overfitting

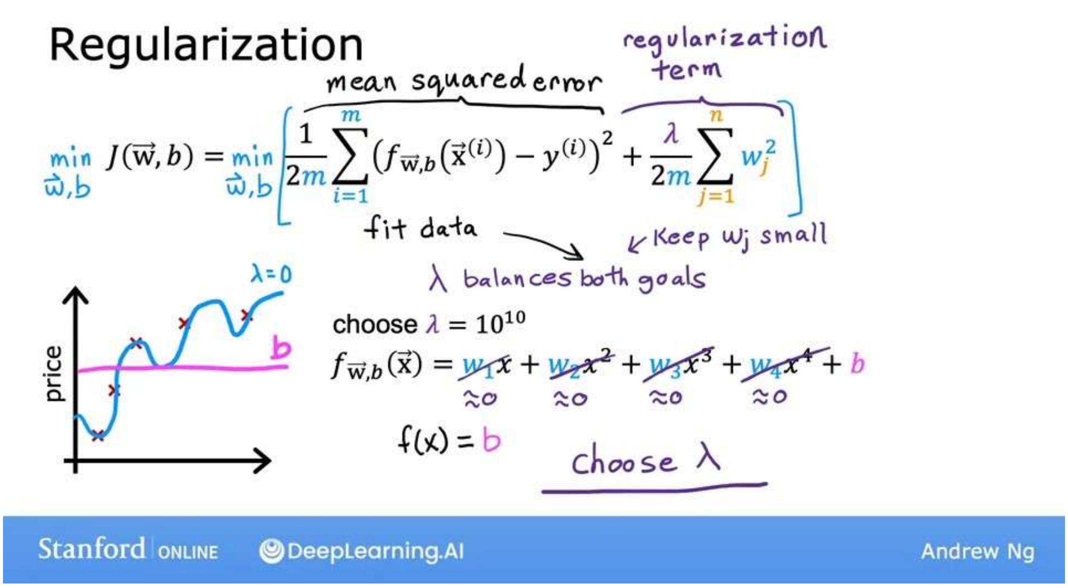
Options

1. Collect more data
2. Select features
   - Feature selection in course 2
3. Reduce size of parameters
   - "Regularization" next videos!

Cost function with regularization







Regularized linear regression

Regularized linear regression

$$\min_{\vec{w},b} J(\vec{w},b) = \min_{\vec{w},b}\left[\frac{1}{2m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2\right]$$

Gradient descent

repeat {

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w},b) \quad = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)} + \frac{\lambda}{m} w_j$$

$j = 1, \ldots, n$

$$b = b - \alpha \frac{\partial}{\partial b} J(\vec{w},b) \quad = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

don't have to regularize b

} simultaneous update

Implementing gradient descent

repeat {

$$w_j = w_j - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}\left[\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)x_j^{(i)}\right] + \frac{\lambda}{m} w_j\right]$$

$$b = b - \alpha\frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

} simultaneous update

Stanford ONLINE DeepLearning.AI Andrew Ng

Regularized logistic regression