

# Regularization

---

GA DAT5

# Agenda

The Problem Of Overfitting

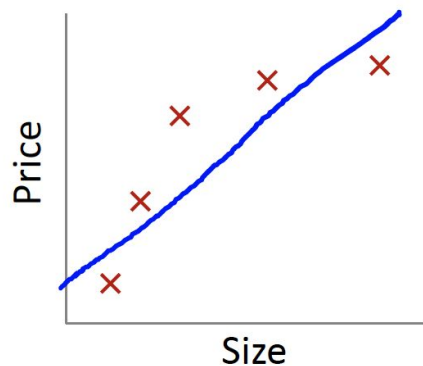
Cost Function

Regularized Regressions

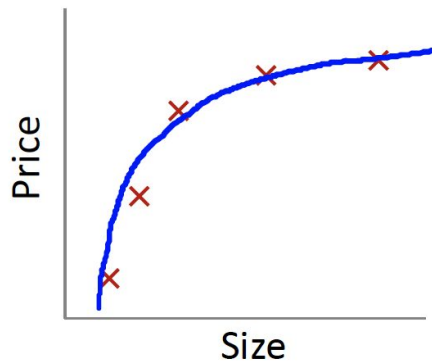
# The Problem of Overfitting

---

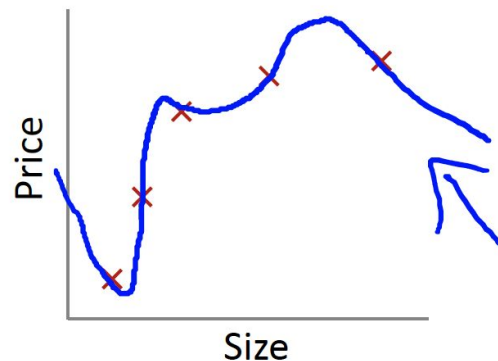
## Example: Linear regression (housing prices)



$\rightarrow \theta_0 + \theta_1 x$   
"Underfit" "High bias"



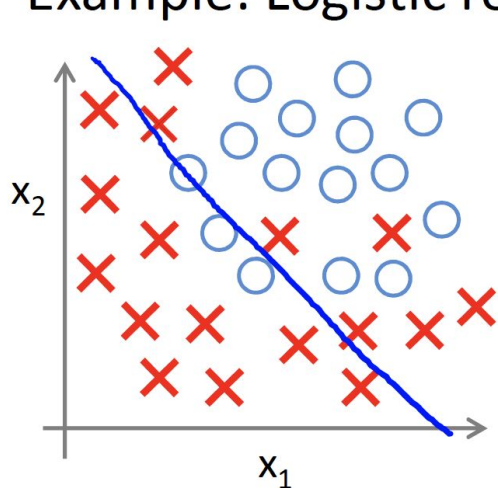
$\rightarrow \theta_0 + \theta_1 x + \theta_2 x^2$   
"Just right"



$\rightarrow \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$   
"Overfit" "High variance"

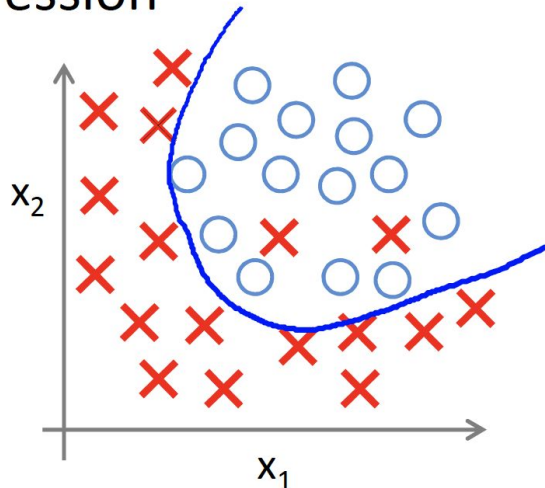
**Overfitting:** If we have too many features, the learned hypothesis may fit the training set very well ( $J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \approx 0$ ), but fail to generalize to new examples (predict prices on new examples).

## Example: Logistic regression

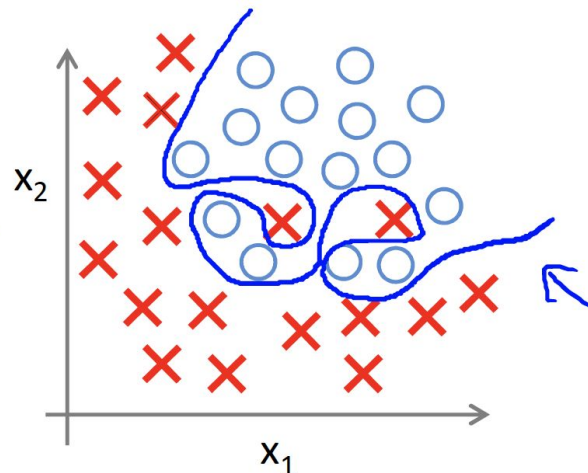


$\rightarrow h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$   
( $g$  = sigmoid function)

"Underfit"



$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2$   
 $+ \theta_3 x_1^2 + \theta_4 x_2^2$   
 $+ \theta_5 x_1 x_2)$



$g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2$   
 $+ \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2$   
 $+ \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \dots)$

"Overfit"

## Addressing overfitting:

$x_1$  = size of house

$x_2$  = no. of bedrooms

$x_3$  = no. of floors

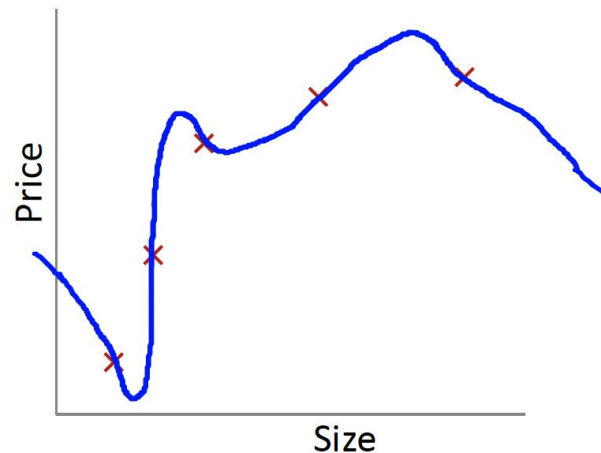
$x_4$  = age of house

$x_5$  = average income in neighborhood

$x_6$  = kitchen size

$\vdots$

$x_{100}$



## Addressing overfitting:

### Options:

1. Reduce number of features.

→ — Manually select which features to keep.

→ — Model selection algorithm (later in course).

2. Regularization.

→ — Keep all the features, but reduce magnitude/values of parameters  $\theta_j$ .

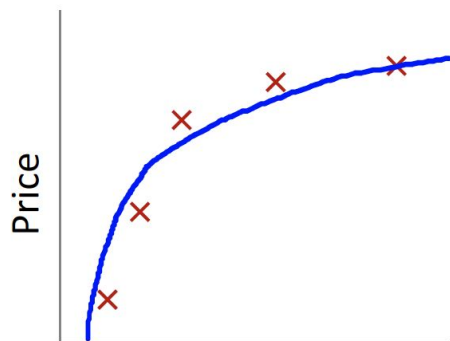
— Works well when we have a lot of features, each of which contributes a bit to predicting  $y$ .

# Cost Function

---

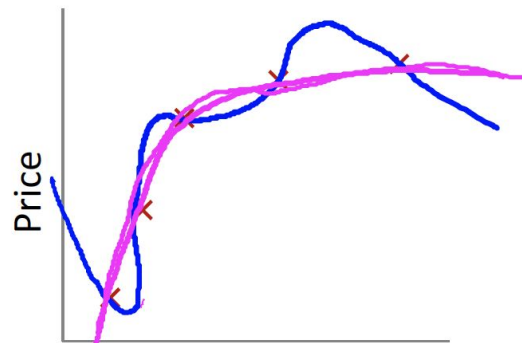


# Intuition



Size of house

$$\theta_0 + \theta_1 x + \theta_2 x^2$$



Size of house

$$\theta_0 + \theta_1 x + \theta_2 x^2 + \cancel{\theta_3 x^3} + \cancel{\theta_4 x^4}$$

Two pink arrows point from the  $\theta_3 x^3$  and  $\theta_4 x^4$  terms in the equation above to the blue curve in the graph to the left, indicating that these terms are being penalized to achieve a smoother fit.

Suppose we penalize and make  $\theta_3, \theta_4$  really small.

$$\rightarrow \min_{\theta} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \underbrace{1000 \theta_3^2}_{\theta_3 \approx 0} + \underbrace{1000 \theta_4^2}_{\theta_4 \approx 0}$$

## Regularization.

Small values for parameters  $\theta_0, \theta_1, \dots, \theta_n$

- “Simpler” hypothesis
- Less prone to overfitting

$$\theta_3, \theta_4 \approx 0$$

Housing:

- Features:  $x_1, x_2, \dots, x_{100}$
- Parameters:  $\theta_0, \theta_1, \theta_2, \dots, \theta_{100}$

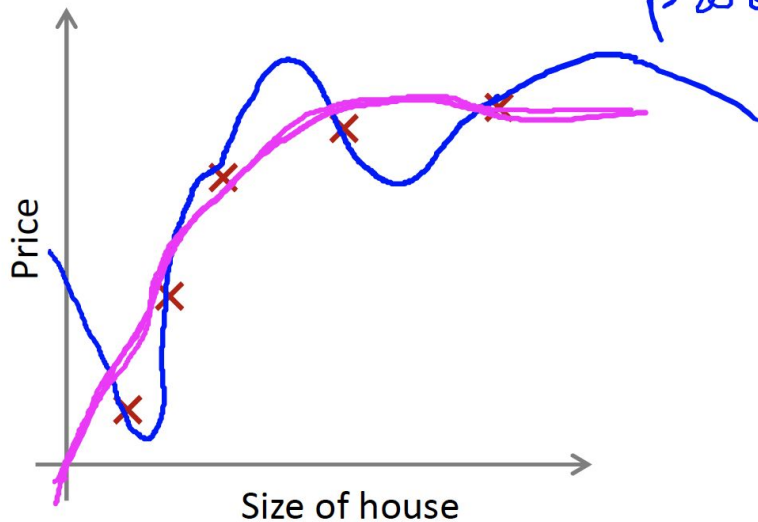
$$J(\theta) = \frac{1}{2m} \left[ \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2 \right]$$

~~$\theta_1, \theta_2, \theta_3, \dots, \theta_{100}$~~

## Regularization.

$$\rightarrow J(\theta) = \frac{1}{2m} \left[ \underbrace{\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2}_{\text{data fit}} + \underbrace{\lambda \sum_{j=1}^n \theta_j^2}_{\text{regularization parameter}} \right]$$

$\min_{\theta} J(\theta)$



In regularized linear regression, we choose  $\theta$  to minimize

$$J(\theta) = \frac{1}{2m} \left[ \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2 \right]$$

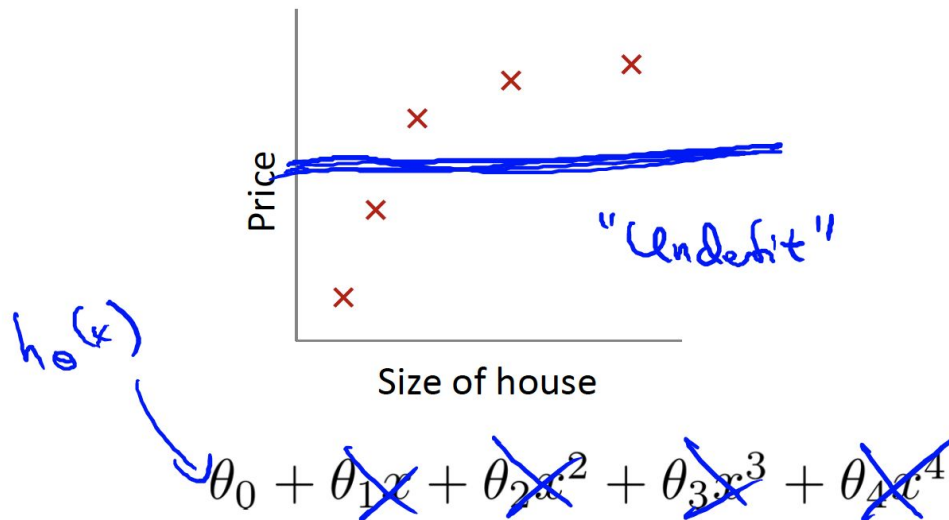
What if  $\lambda$  is set to an extremely large value (perhaps far too large for our problem, say  $\lambda = 10^{10}$ )?

- Algorithm works fine; setting  $\lambda$  to be very large can't hurt it
- Algorithm fails to eliminate overfitting.
- Algorithm results in underfitting. (Fails to fit even training data well).
- Gradient descent will fail to converge.

In regularized linear regression, we choose  $\theta$  to minimize

$$J(\theta) = \frac{1}{2m} \left[ \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2 \right]$$

What if  $\lambda$  is set to an extremely large value (perhaps far too large for our problem, say  $\lambda = 10^{10}$ )?



$$\begin{aligned} \theta_1, \theta_2, \theta_3, \theta_4 \\ \theta_1 \approx 0, \theta_2 \approx 0 \\ \theta_3 \approx 0, \theta_4 \approx 0 \\ \boxed{h_{\theta}(x) = \theta_0} \end{aligned}$$

# Regularized Regressions

---

## Regularized linear regression

$$J(\theta) = \frac{1}{2m} \left[ \underbrace{\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2}_{\text{data fit}} + \underbrace{\lambda \sum_{j=1}^n \theta_j^2}_{\text{regularization}} \right]$$

$$\min_{\theta} \underline{J(\theta)}$$

# Gradient descent

$$\theta_0$$

$$\theta_1, \theta_2, \dots, \theta_n$$

Repeat {

$$\frac{\partial}{\partial \theta_0} J(\theta)$$

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_0^{(i)}$$

$$\theta_j := \theta_j - \alpha \left[ \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} + \frac{\lambda}{m} \theta_j \right]$$

(j = ~~0~~, 1, 2, 3, ..., n)

$$\rightarrow J(\theta)$$

$$\theta_j := \theta_j \left(1 - \alpha \frac{\lambda}{m}\right) - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

$$1 - \alpha \frac{\lambda}{m} < 1$$

$$0.99$$

$$\theta_j \times 0.99$$

$$\theta_j^2$$



## Normal equation

$$\underline{X} = \begin{bmatrix} (x^{(1)})^T \\ \vdots \\ (x^{(m)})^T \end{bmatrix}$$

$$\underset{\uparrow}{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix} \in \mathbb{R}^m$$

$$\Rightarrow \min_{\theta} \underline{J(\theta)}$$

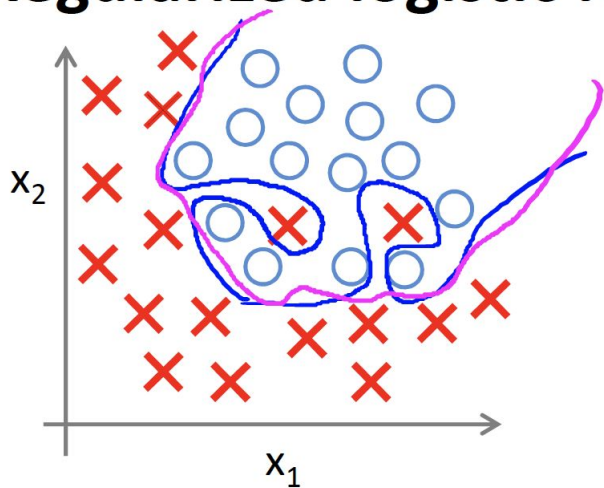
$$\rightarrow \min_{\theta} J(\theta)$$

$$\frac{\partial}{\partial \theta_j} J(\theta) \stackrel{\text{set}}{=} 0$$

$$\Rightarrow \Theta = (X^T X + \lambda \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}}_{(n+1) \times (n+1)})^{-1} X^T y$$

$\in \mathbb{R}^{n \times (n+1)}$      $n=2$      $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

# Regularized logistic regression.



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \dots)$$

Cost function:

$$\rightarrow J(\theta) = - \left[ \frac{1}{m} \sum_{i=1}^m y^{(i)} \log h_{\theta}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$

$\theta_1, \theta_2, \dots, \theta_n$

# Gradient descent

Repeat {

$$\rightarrow \theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_0^{(i)}$$

$$\rightarrow \theta_j := \theta_j - \alpha \left[ \underbrace{\frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)}}_{\substack{(j = \text{red X}, 1, 2, 3, \dots, n) \\ \theta_1, \dots, \theta_n}} + \frac{\lambda}{m} \theta_j \right] \leftarrow$$

}

$$\frac{\partial}{\partial \theta_j} J(\theta)$$

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

Q??

---