

1.遇到错误怎么解决：

Debugging a learning algorithm:

Suppose you have implemented regularized linear regression to predict housing prices.

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

However, when you test your hypothesis on a new set of houses, you find that it makes unacceptably large errors in its predictions. What should you try next?

- \rightarrow - Get more training examples
- Try smaller sets of features $x_1, x_2, x_3, \dots, x_{100}$
- \rightarrow - Try getting additional features
- Try adding polynomial features ($\underline{x_1^2}, \underline{x_2^2}, \underline{x_1 x_2}$, etc.)
- Try decreasing λ
- Try increasing λ

Machine learning diagnostic:

Diagnostic: A test that you can run to gain insight what is/isn't working with a learning algorithm, and gain guidance as to how best to improve its performance.

Diagnostics can take time to implement, but doing so can be a very good use of your time.

2. 评估假设：

按照7：3分为训练集和测试集。

Evaluating your hypothesis

Dataset:

Size	Price	
2104	400	70% Training set
1600	330	
2400	369	
1416	232	
3000	540	
1985	300	
1534	315	
<hr/>		
1427	199	30% Test set
1380	212	
1494	243	

第一组测试样本

$$\begin{pmatrix} (x^{(1)}, y^{(1)}) \\ (x^{(2)}, y^{(2)}) \\ \vdots \\ (x^{(m)}, y^{(m)}) \end{pmatrix}$$

$$\begin{pmatrix} (x_{test}^{(1)}, y_{test}^{(1)}) \\ (x_{test}^{(2)}, y_{test}^{(2)}) \\ \vdots \\ (x_{test}^{(m_{test})}, y_{test}^{(m_{test})}) \end{pmatrix}$$

$m_{test} = \text{no. of test example}$
 $(x_{test}^{(i)}, y_{test}^{(i)})$

最好随机选择70%作为训练集。

典型的方法：

Training/testing procedure for linear regression

- - Learn parameter θ from training data (minimizing training error $J(\theta)$) 70%

- Compute test set error:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_{\theta}(x_{test}^{(i)}) - y_{test}^{(i)})^2$$

Training/testing procedure for logistic regression

- - Learn parameter θ from training data
- Compute test set error:

$$\rightarrow J_{test}(\theta) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_{\theta}(x_{test}^{(i)}) + (1 - y_{test}^{(i)}) \log h_{\theta}(x_{test}^{(i)})$$

Training/testing procedure for logistic regression

→ - Learn parameter θ from training data

- Compute test set error:

m_{test}

$$\rightarrow J_{test}(\theta) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_{\theta}(x_{test}^{(i)}) + (1 - y_{test}^{(i)}) \log h_{\theta}(x_{test}^{(i)})$$

- Misclassification error (0/1 misclassification error):

$$err(h_{\theta}(x), y) = \begin{cases} 1 & \text{if } h_{\theta}(x) \geq 0.5, y = 0 \\ & \text{or if } h_{\theta}(x) < 0.5, y = 1 \end{cases} \text{ error}$$

$$0 \text{ otherwise}$$

$$\text{Test error} = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err(h_{\theta}(x_{test}^{(i)}), y_{test}^{(i)})$$

那部分测试集中的样本

模型选择：通过分别对不同模型进行训练，得到各个模型测试集误差，现在从中选择误差最小的模型，如果要测试这个模型，就不能在原来的测试集上进行测试，因为这些参数就是在测试集上训练而来的。

Model selection

→ d = degree of polynomial

$$d=1 \quad 1. \quad h_{\theta}(x) = \theta_0 + \theta_1 x \rightarrow \Theta^{(1)} \rightarrow J_{test}(\Theta^{(1)})$$

$$d=2 \quad 2. \quad h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \Theta^{(2)} \rightarrow J_{test}(\Theta^{(2)})$$

$$d=3 \quad 3. \quad h_{\theta}(x) = \theta_0 + \theta_1 x + \dots + \theta_3 x^3 \rightarrow \Theta^{(3)} \rightarrow J_{test}(\Theta^{(3)})$$

\vdots

$$d=10 \quad 10. \quad h_{\theta}(x) = \theta_0 + \theta_1 x + \dots + \theta_{10} x^{10} \rightarrow \Theta^{(10)} \rightarrow J_{test}(\Theta^{(10)})$$

Choose $\theta_0 + \dots + \theta_5 x^5$ ←

How well does the model generalize? Report test set error $J_{test}(\theta^{(5)})$.

$\theta_0, \theta_1, \dots$

Problem: $J_{test}(\theta^{(5)})$ is likely to be an optimistic estimate of generalization error. I.e. our extra parameter (d = degree of polynomial) is fit to test set.

因此采用下面方法：把数据集分为三个部分：训练集，交叉验证集，测试集：

Evaluating your hypothesis

Dataset:

Size	Price	
2104	400	60% Training set
1600	330	
2400	369	
1416	232	
3000	540	
1985	300	
1534	315	20% Cross validation set (cv)
1427	199	
1380	212	20% test set
1494	243	

用m下标test来表示测试样本的总数

$\begin{pmatrix} (x^{(1)}, y^{(1)}) \\ (x^{(2)}, y^{(2)}) \\ \vdots \\ (x^{(m)}, y^{(m)}) \end{pmatrix}$

$\begin{pmatrix} (x_{cv}^{(1)}, y_{cv}^{(1)}) \\ (x_{cv}^{(2)}, y_{cv}^{(2)}) \\ \vdots \\ (x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})}) \end{pmatrix}$

$\begin{pmatrix} (x_{test}^{(1)}, y_{test}^{(1)}) \\ (x_{test}^{(2)}, y_{test}^{(2)}) \\ \vdots \\ (x_{test}^{(m_{test})}, y_{test}^{(m_{test})}) \end{pmatrix}$

$m_{cv} = \text{no. of cv examples}$
 $(x_{cv}^{(i)}, y_{cv}^{(i)})$
 m_{test}

各种误差：

Train/validation/test error

Training error:

$$J_{train}(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Cross Validation error:

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_{\theta}(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$$

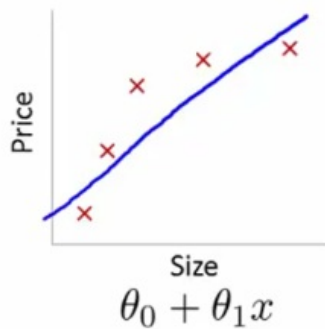
Test error:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_{\theta}(x_{test}^{(i)}) - y_{test}^{(i)})^2$$

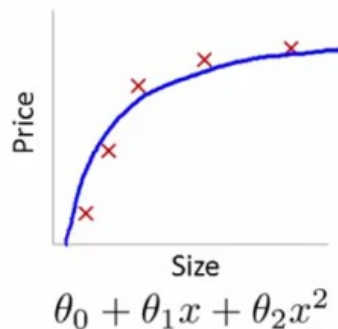
和测试误差

Bias / Variance

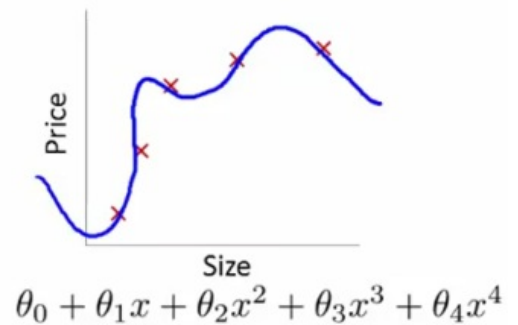
Bias/variance



High bias
(underfit)
 $d=1$



"Just right"
 $d=2$

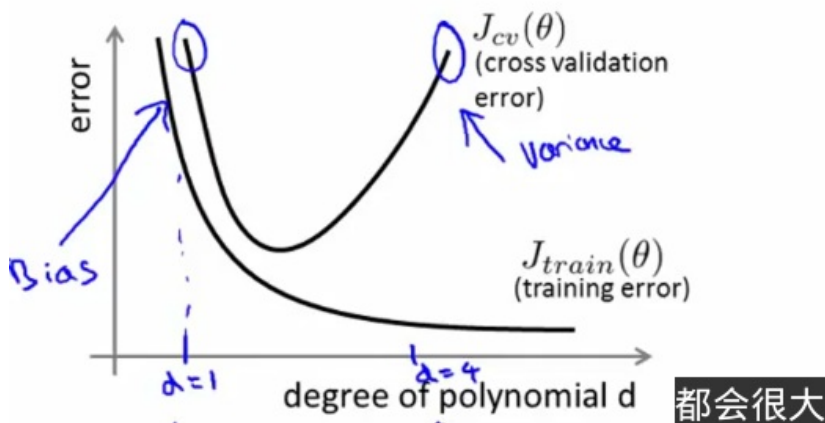


High variance
(overfit)
 $d=4$

训练误差和交叉验证误差：

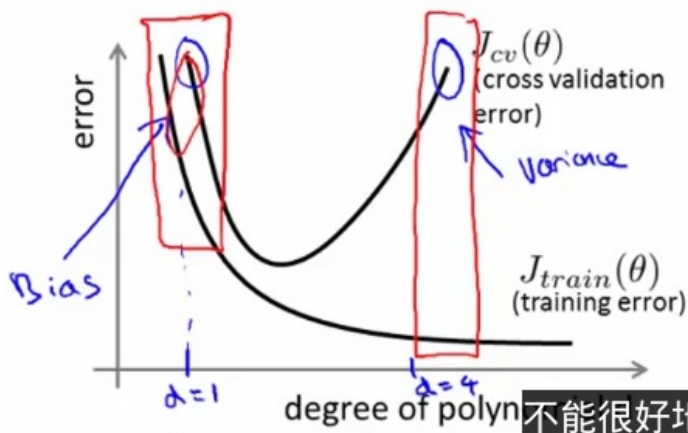
Diagnosing bias vs. variance

Suppose your learning algorithm is performing less well than you were hoping. ($J_{cv}(\theta)$ or $J_{test}(\theta)$ is high.) Is it a bias problem or a variance problem?



Diagnosing bias vs. variance

Suppose your learning algorithm is performing less well than you were hoping. ($J_{cv}(\theta)$ or $J_{test}(\theta)$ is high.) Is it a bias problem or a variance problem?



Bias (underfit):

$$\left. \begin{array}{l} J_{train}(\theta) \text{ will be high} \\ J_{cv}(\theta) \approx J_{train}(\theta) \end{array} \right\}$$

Variance (overfit):

$$\left. \begin{array}{l} J_{train}(\theta) \text{ will be low} \\ J_{cv}(\theta) \gg J_{train}(\theta) \end{array} \right\}$$

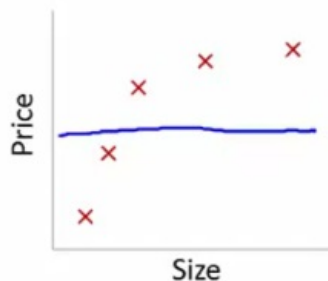
不能很好地拟合训练集数据

诊断一个模型是处于什么状态：

Linear regression with regularization

Model: $h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$ ←

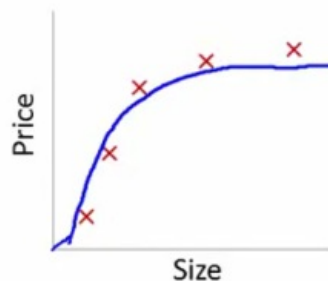
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^m \theta_j^2$$
 ←



Large λ ←

→ High bias (underfit)

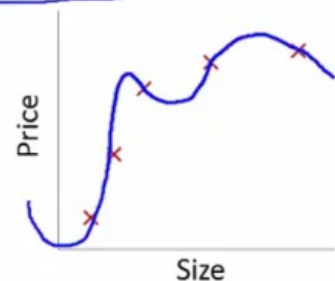
→ $\lambda = 10000$, $\theta_1 \approx 0, \theta_2 \approx 0, \dots$
 $h_{\theta}(x) \approx \theta_0$



Intermediate λ ←

"just right"

但在这里我就不讨论这些情况了



→ Small λ

High variance (overfit)

→ $\lambda = 0$

选择正规化参数lambda：

Choosing the regularization parameter λ

$$h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 \quad \leftarrow$$

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^m \theta_j^2 \quad \leftarrow$$

$$\Rightarrow J_{train}(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \quad J(\theta)$$

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_{\theta}(x_{cv}^{(i)}) - y_{cv}^{(i)})^2 \quad J_{train}$$

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_{\theta}(x_{test}^{(i)}) - y_{test}^{(i)})^2 \quad J_{cv}$$

这就是模型选择在选取正则化参数 λ 时的应用

Choosing the regularization parameter λ

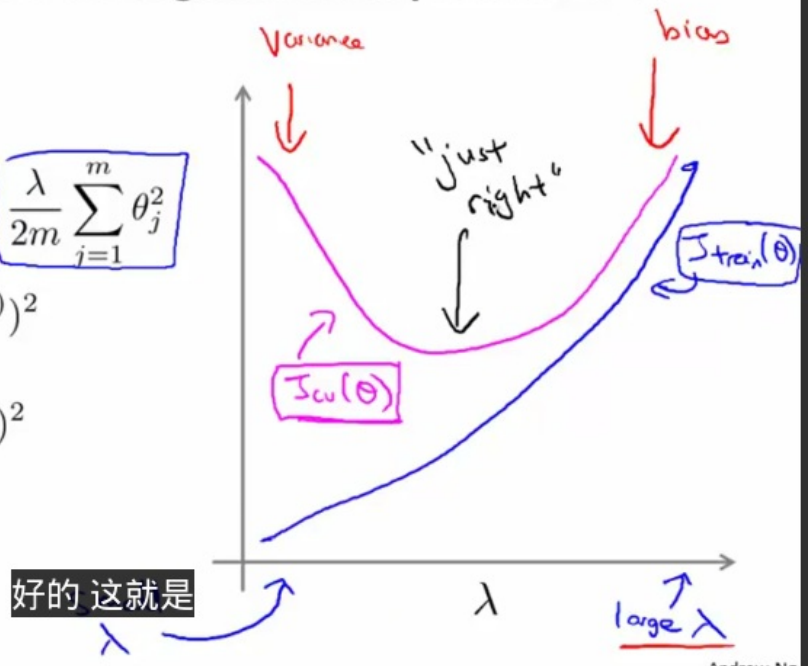
Model: $h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

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

1. Try $\lambda = 0 \quad \leftarrow \quad \rightarrow \min_{\theta} J(\theta) \rightarrow \theta^{(1)} \rightarrow J_{cv}(\theta^{(1)})$
 2. Try $\lambda = 0.01 \quad \rightarrow \min_{\theta} J(\theta) \rightarrow \theta^{(2)} \rightarrow J_{cv}(\theta^{(2)})$
 3. Try $\lambda = 0.02 \quad \rightarrow \theta^{(3)} \rightarrow J_{cv}(\theta^{(3)})$
 4. Try $\lambda = 0.04$
 5. Try $\lambda = 0.08 \quad \rightarrow \theta^{(5)} \quad J_{cv}(\theta^{(5)})$
 - \vdots
 12. Try $\lambda = 10 \quad \rightarrow \theta^{(12)} \rightarrow J_{cv}(\theta^{(12)})$
- ↑ 10.24 来测出它对测试集的 $J_{test}(\theta^{(5)})$
- Pick (say) $\theta^{(5)}$. Test error: $J_{test}(\theta^{(5)})$

Bias/variance as a function of the regularization parameter λ

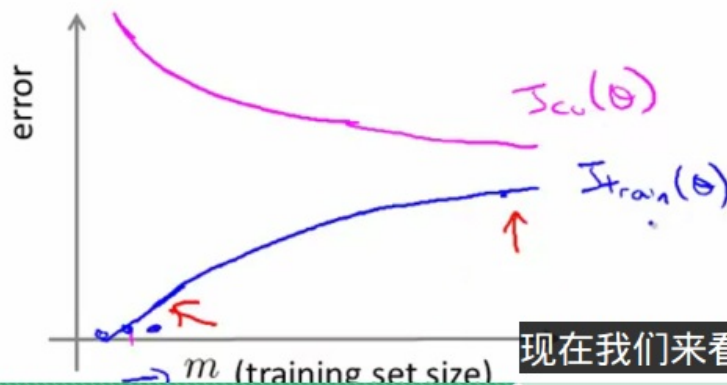
$$\begin{aligned} \rightarrow J(\theta) &= \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \boxed{\frac{\lambda}{2m} \sum_{j=1}^m \theta_j^2} \\ \rightarrow J_{train}(\theta) &= \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \\ \rightarrow \boxed{J_{cv}(\theta)} &= \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_{\theta}(x_{cv}^{(i)}) - y_{cv}^{(i)})^2 \end{aligned}$$



学习曲线：

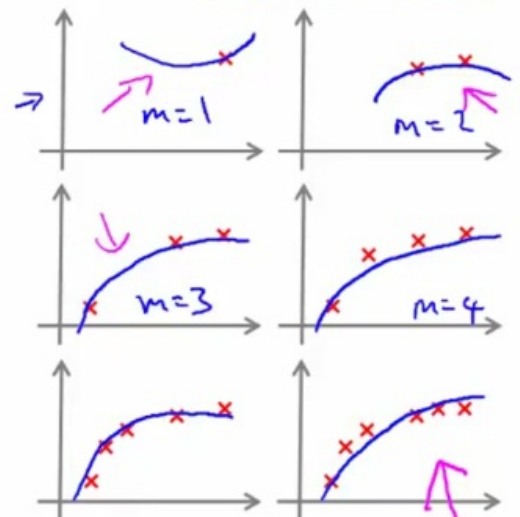
Learning curves

$$\begin{aligned} \rightarrow J_{train}(\theta) &= \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 \\ \rightarrow J_{cv}(\theta) &= \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_{\theta}(x_{cv}^{(i)}) - y_{cv}^{(i)})^2 \end{aligned}$$



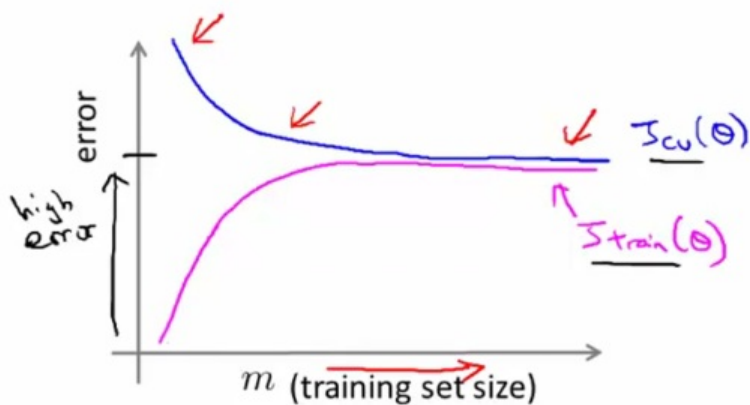
现在我们来看看

$$h_{\theta}(x) = \theta_0 + \theta_1 x + \theta_2 x^2$$



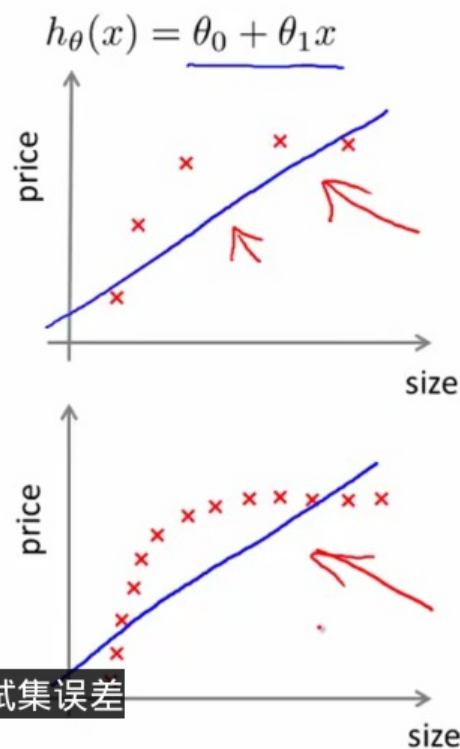
高偏差状态（欠拟合）：

High bias



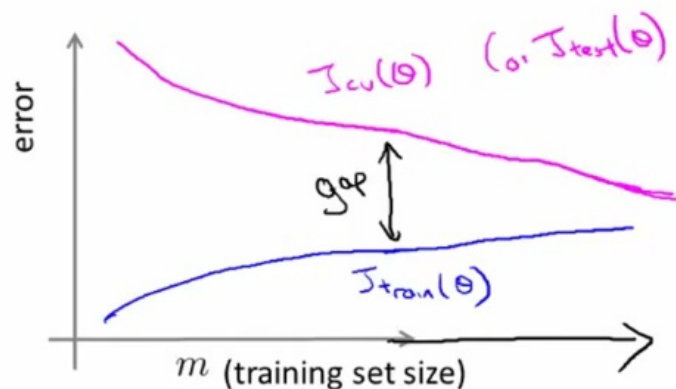
If a learning algorithm is suffering from high bias, getting more training data will not (by itself) help much.

交叉验证集误差或测试集误差



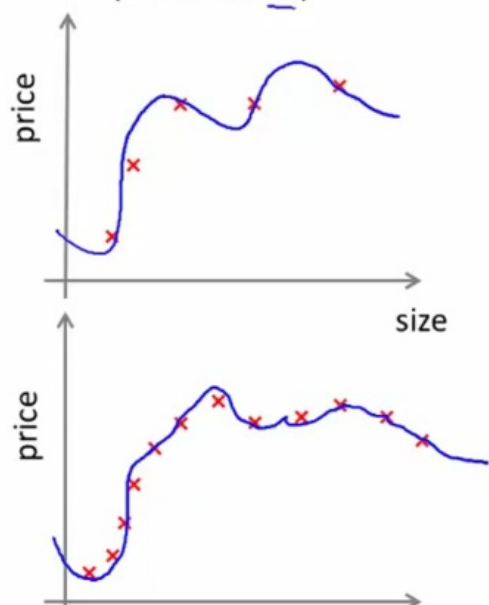
过拟合：

High variance



If a learning algorithm is suffering from high variance, getting more training data is likely to help. ←

$$h_{\theta}(x) = \theta_0 + \theta_1 x + \dots + \theta_{100} x^{100} \quad (\text{and small } \lambda)$$



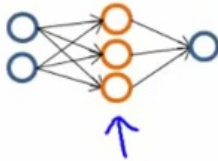
Debugging a learning algorithm:

Suppose you have implemented regularized linear regression to predict housing prices. However, when you test your hypothesis in a new set of houses, you find that it makes unacceptably large errors in its prediction. What should you try next?

- Get more training examples → fixes high variance
- Try smaller sets of features → fixes high variance
- Try getting additional features → fixes high bias
- Try adding polynomial features (x_1^2, x_2^2, x_1x_2 , etc) → fixes high bias.
- Try decreasing λ → fixes high bias
- Try increasing λ → fixes high variance

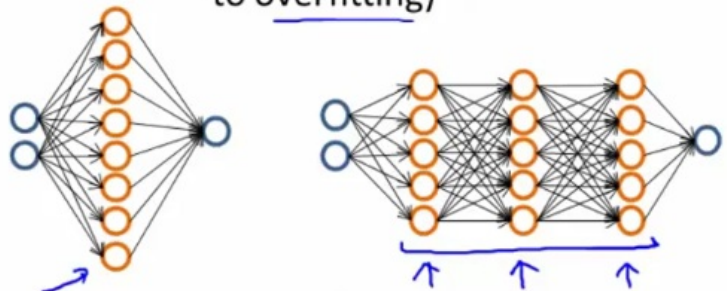
Neural networks and overfitting

→ “Small” neural network
(fewer parameters; more prone to underfitting)



Computationally cheaper

→ “Large” neural network
(more parameters; more prone to overfitting)



Computationally more expensive.

Use regularization (λ) to address overfitting.

$$J_{co}(\theta)$$

