Diagnosing ML for Applications

Debugging a learning algorithm:

- A regularized linear regression model has been implemented for predicting house re-sale price. The parameters in the model are obtained from minimizing the following cost:

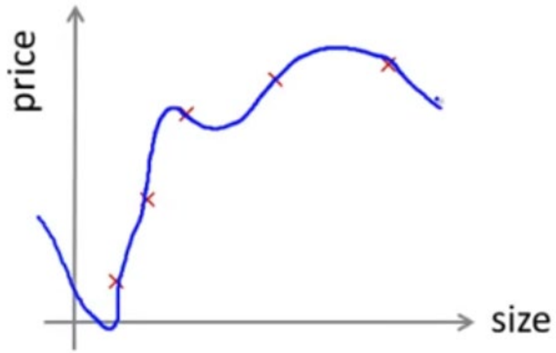
$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda \sum_{i=1}^{n} \theta_j^2\right]$$

- However, the model does not generalize well (i.e., when the realtor uses your model to list the house on the market, the price is way off.)

- How to improve this model?
  - Collecting more data?
  - Using smaller sets of features?
  - Including additional features?
  - Using polynomial features $(x_1^2, x_2^2, x_3^2, x_1x_2, x_1x_3, \ldots)$?
  - Increasing/decreasing λ?

How to address the problem?

- Intuitive idea – plot price against features. Not feasible.
- Run tests to evaluate a hypothesis



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

(just one physical feature)

x_1 = size of house
x_2 = number of bedrooms
x_3 = area safety
x_4 = school rating
x_5 = shopping convenience
x_6 = average income of neighborhood
x_7 = newly upgraded kitchen
.....

Then how to evaluate a hypothesis?
And then how to select a good hypothesis?

Model Selection – prepare datasets

Dataset:

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

$$\left(x^{(1)}, y^{(1)}\right)$$
$$\left(x^{(2)}, y^{(2)}\right)$$
$$\vdots$$
$$\left(x^{(m)}, y^{(m)}\right)$$

Training set
~60% (random shuffle if possible)
m: # of training examples

$$\left(x_{cv}^{(1)}, y_{cv}^{(1)}\right)$$
$$\left(x_{cv}^{(2)}, y_{cv}^{(2)}\right)$$
$$\vdots$$
$$\left(x_{cv}^{(m_{cv})}, y_{cv}^{(m_{cv})}\right)$$

Cross validation (cv) set
~20% (random shuffle if possible)
m_{cv}: # of cv examples

$$\left(x_{test}^{(1)}, y_{test}^{(1)}\right)$$
$$\left(x_{test}^{(2)}, y_{test}^{(2)}\right)$$
$$\vdots$$
$$\left(x_{test}^{(m_{test})}, y_{test}^{(m_{test})}\right)$$

Testing set
~20% (random shuffle if possible)
m_{test}: # of testing examples

Model Selection (through model evaluation)

Training error:

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

⟶ to obtain parameters in a given hypothesis

Cross Validation error:

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

⟶ to evaluate a hypothesis for model selection

Test error:

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

⟶ to test how well the hypothesis generalizes

Model Selection (choose the complexity of a hypothesis)

- First obtain model parameters given a structure using training set
- Then evaluate the hypothesis
- A set of models with different complexity then become available for choice

1. $h_\theta(x) = \theta_0 + \theta_1 x \longrightarrow \theta^{(1)} \longrightarrow J_{\mathrm{CV}}\left(\theta^{(1)}\right)$
2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \longrightarrow \theta^{(2)} \longrightarrow J_{\mathrm{CV}}\left(\theta^{(2)}\right)$
3. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 \longrightarrow \theta^{(3)} \longrightarrow J_{\mathrm{CV}}\left(\theta^{(3)}\right)$

:

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \longrightarrow \theta^{(10)} \longrightarrow J_{\mathrm{CV}}\left(\theta^{(10)}\right)$

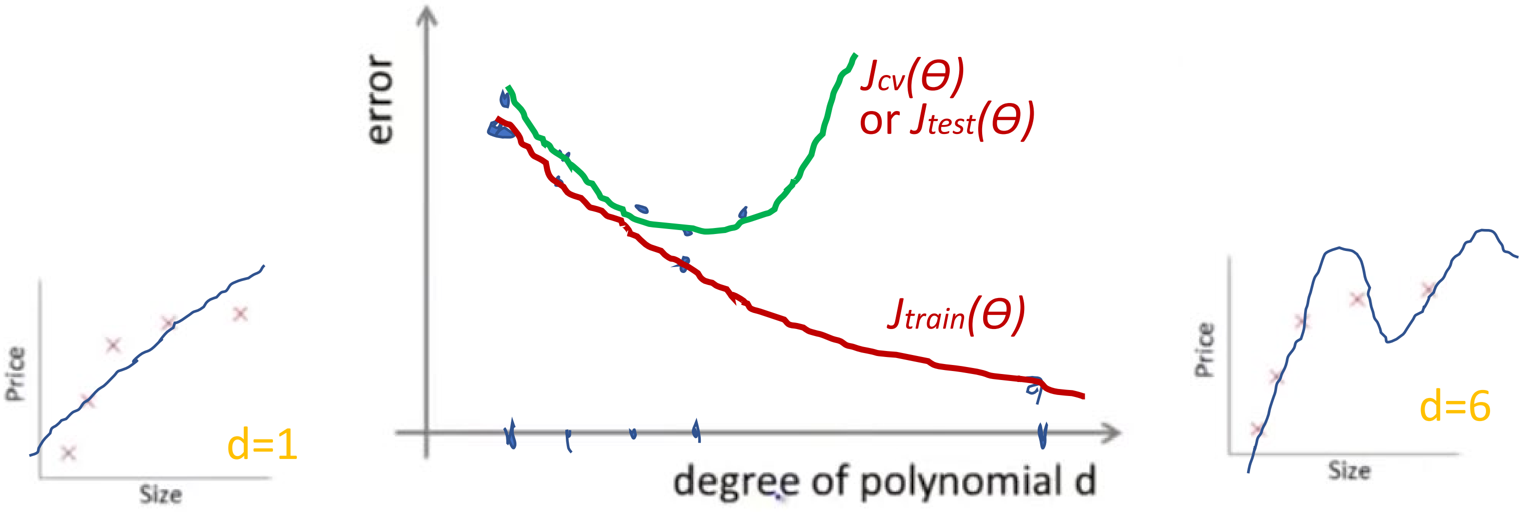
Pick the 3$^{\mathrm{rd}}$ order hypothesis if $J_{\mathrm{CV}}\left(\theta^{(3)}\right)$ has small value

Diagnosing Bias vs Variance problem (classical view)

Bias or Variance (let's make an observation)

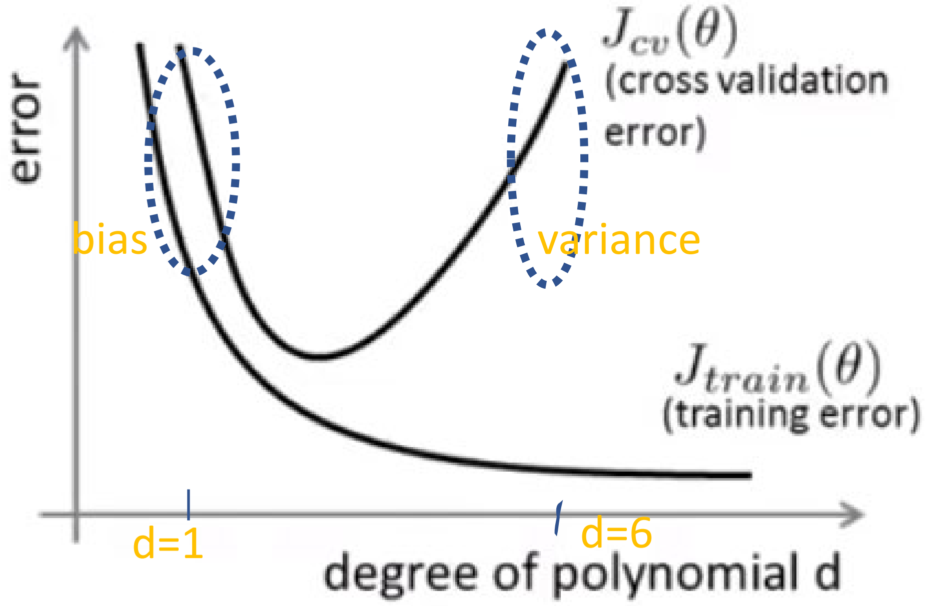
$$J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

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



Diagnosis of a bias or variance problem

- Realizing that there is a problem with the hypothesis/model you generated
- Either $J_{cv}(\Theta)$ or $J_{test}(\Theta)$ is high
- Two possibilities: model too simple (underfit) or model too complex (overfit)



Bias problem (underfit)

$J_{train}(\Theta)$ - *high*

$J_{cv}(\Theta) \sim J_{train}(\Theta)$

Variance problem (overfit)

$J_{train}(\Theta)$ - *low*

$J_{cv}(\Theta) >> J_{train}(\Theta)$

With the bias vs variance diagnosis tool, we can now apply it to the diagnosis of a learning problem (use regularization problem as an example)

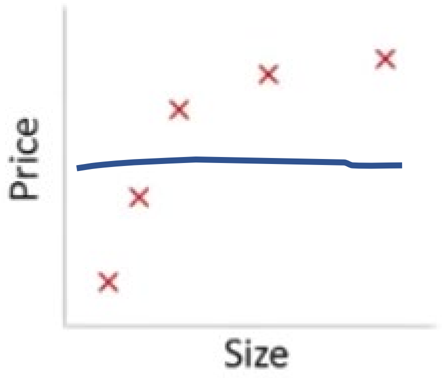
Recall the regularized linear regression problem

$$h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

The parameters θ_0 θ_1 … θ_n are determined from minimizing the following cost function,

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

How to choose λ

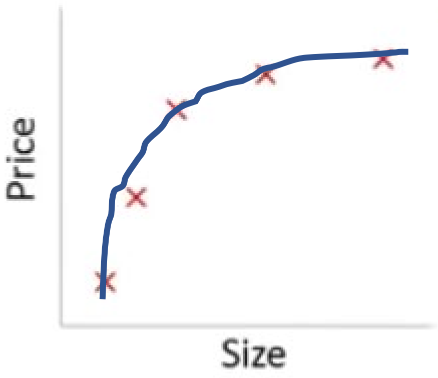


Large λ

High bias (underfit)

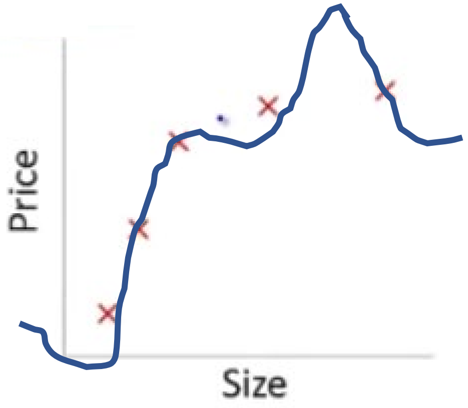
$= 10000.\ \theta_1 \approx 0, \theta_2 \approx 0, \ldots$

$h_\theta(x) \approx \theta_0$



Intermediate λ

"Just right"



Small λ

High variance (overfit)

Given a hypothesis with parameters $\theta_0\ \theta_1 \ldots \theta_n$

$$h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Obtain parameters $\theta_0\ \theta_1 \ldots \theta_n$ from minimizing the following cost function,

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

Evaluation of a Hypothesis

For selecting an appropriate regularization parameter λ, use the following costs:

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

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

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

Choosing the regularization parameter λ

1. Try $\lambda = 0$ $\longrightarrow \theta^{(1)} \longrightarrow J_{CV}\left(\theta^{(1)}\right)$
2. Try $\lambda = 0.01$ $\longrightarrow \theta^{(2)} \longrightarrow J_{CV}\left(\theta^{(2)}\right)$
3. Try $\lambda = 0.02$ $\longrightarrow \theta^{(3)} \longrightarrow J_{CV}\left(\theta^{(3)}\right)$
4. Try $\lambda = 0.04$ $\longrightarrow \theta^{(4)} \longrightarrow J_{CV}\left(\theta^{(4)}\right)$
5. Try $\lambda = 0.08$ $\longrightarrow \theta^{(5)} \longrightarrow J_{CV}\left(\theta^{(5)}\right)$

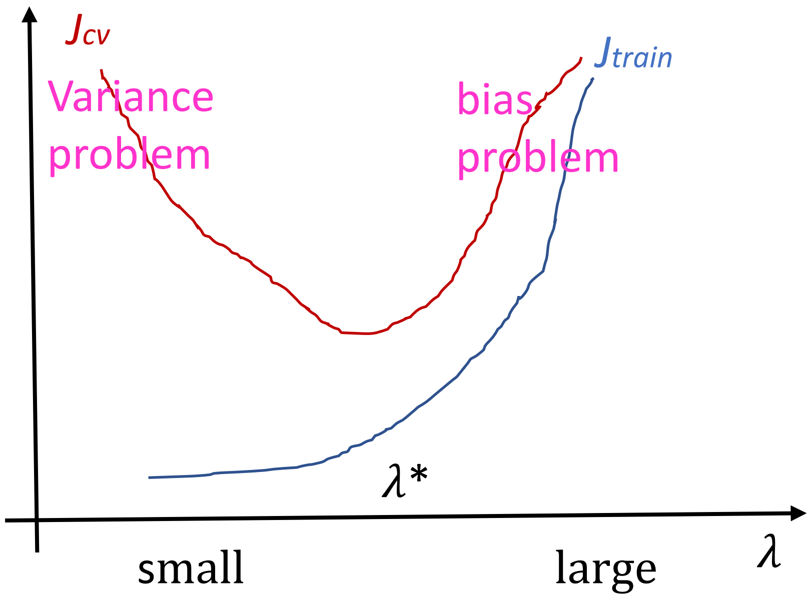
…

20. Try $\lambda = 10$ $\longrightarrow \theta^{(20)} \longrightarrow J_{CV}\left(\theta^{(20)}\right)$

Pick $\lambda = 0.08$ say if $J_{CV}\left(\theta^{(5)}\right)$ has small value

Diagnosis of a bias or variance problem

- Realizing that there is a problem with the hypothesis/model, which is related to choice of λ
- Either $J_{cv}(\Theta)$ or $J_{test}(\Theta)$ is high
- Two possibilities: model too simple/very large λ (underfit) or model too complex/very small λ (overfit)



Bias problem (underfit)

$J_{train}(\Theta)$ - high

$J_{cv}(\Theta) \sim J_{train}(\Theta)$

Variance problem (overfit)

$J_{train}(\Theta)$ - low

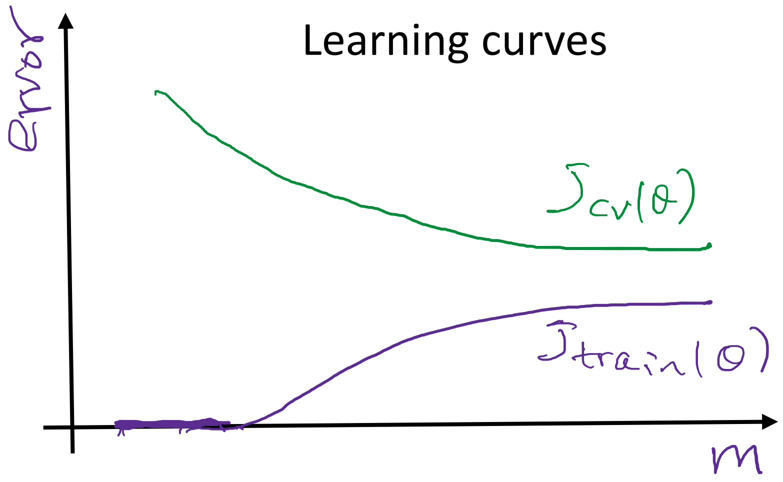
$J_{cv}(\Theta) >> J_{train}(\Theta)$

Impact of # of samples on bias-variance relationship

Consider a hypothesis with given structure/complexity:

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

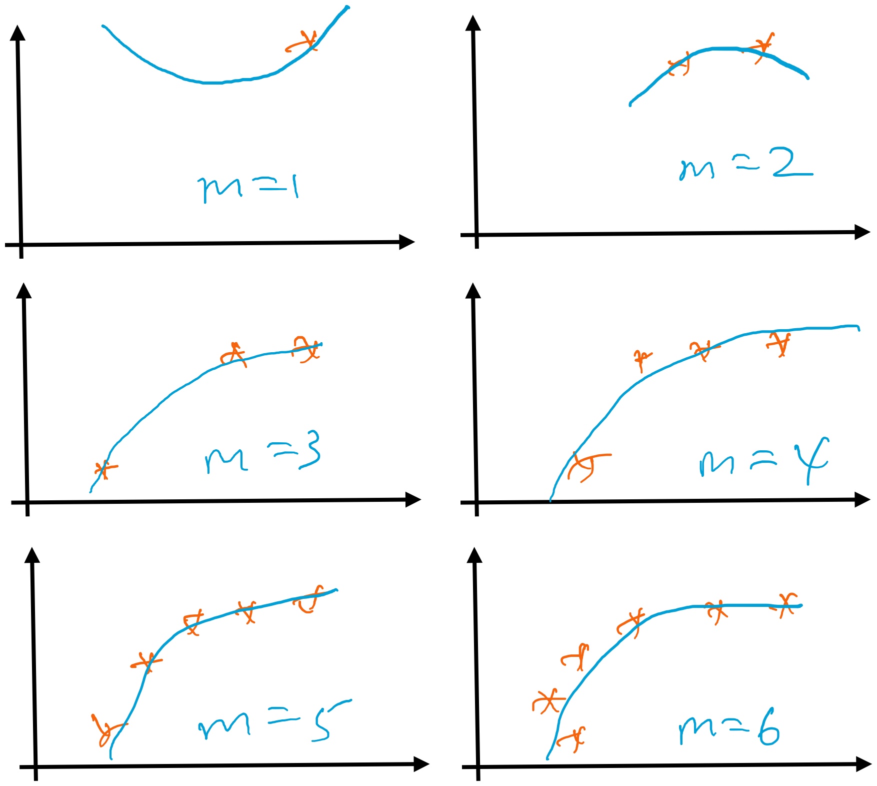
Learning curves



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

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

Illustration of hypothesis vs # of samples (m)



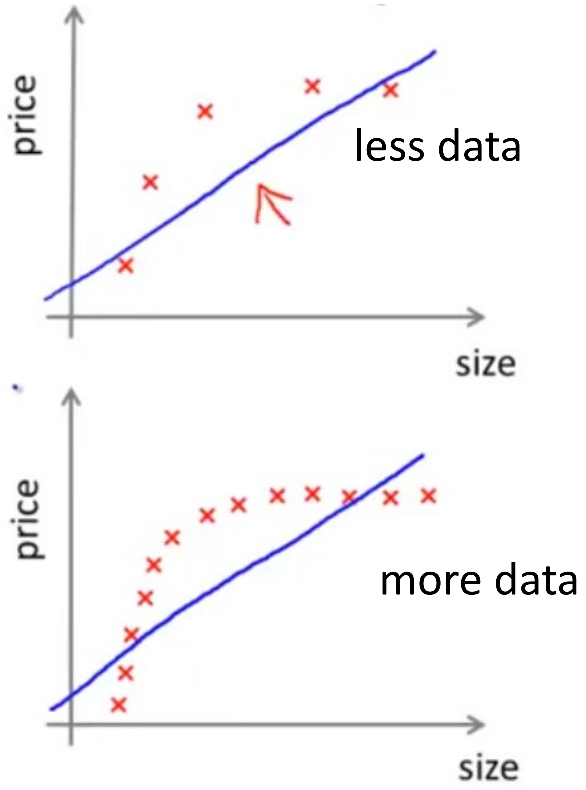
Learning curves associated with high bias / high variance

High bias problems

If learning suffers from a high bias problem, getting more training data by itself is not likely helpful

Given a hypothesis

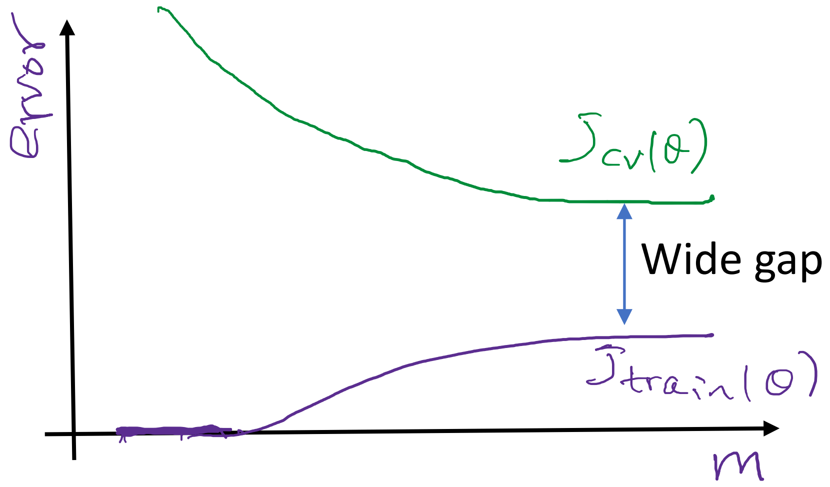
$$h_\theta(x) = \theta_0 + \theta_1 x$$

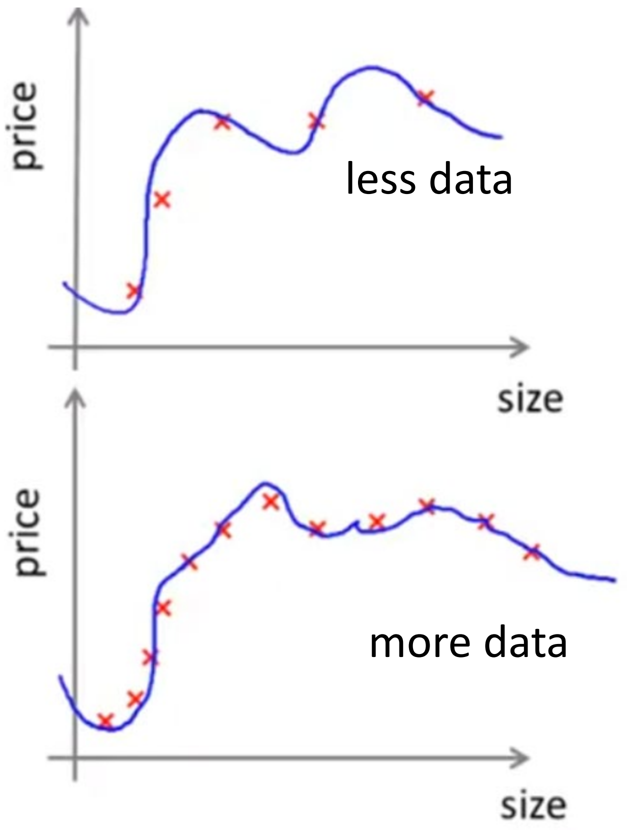


High variance problems

Given a hypothesis (the parameters of which are obtained using a small λ for regulation)

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_{100} x_{100}$$





If learning suffers from a high variance problem, getting more training data is likely helpful

Debugging a learning algorithm:

- A regularized linear regression model has been implemented for predicting house re-sale price. The parameters in the model are obtained from minimizing the following cost:

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

- However, the model does not generalize well (i.e., when the realtor uses your model to list the house on the market, the price is way off.)

- How to improve this model?
  - Collecting more data?   addresses high variance problem
  - Using smaller sets of features? addresses high variance problem
  - Including additional features? addresses high bias
  - Using polynomial features $(x_1^2, x_2^2, x_3^2, x_1x_2, x_1x_3, \ldots)$? addresses high bias
  - Increasing λ?   addresses high variance
  - Decreasing λ?   addresses high bias

Modern development

- Rich models such as deep neural networks can fit or interpolate the data really well
- Current evidence shows that they also are accurate on test data
- The “double-descent” performance curve instead of the U-shaped bias–variance trade-off curve, for beyond the point of interpolation
- On-going research to understand the phenomenon