Evaluate Model

↳ Train_test_split → x_train, → x_test

$$J(\vec{w},b) = \frac{1}{2m_{train}} \sum_{i=1}^{m_{train}} \left(f_{\vec{w},b}(x^i) - y^i\right)^2 + \underbrace{\frac{\lambda}{2m_{train}} \sum_{j=1}^{n} w_j^2}_{\text{regularization}}$$

test error ↓

$$J_{test}(\vec{w},b) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(f_{\vec{w},b}(\vec{x}_{test}) - y_{test}\right)^2$$

× no regularization

Training error; ↓

$$\frac{1}{2m_{train}} \sum_{i=1}^{m_{train}} \left(f_{\vec{w},b}(\vec{x}^{(i)}_{train}) - y_{train}\right)^2$$

classification → Instead of cost func → count

fraction of test set and fraction of train set model has missclasified.

for polynomial;

Training / cross validation / test-set

60% - train set

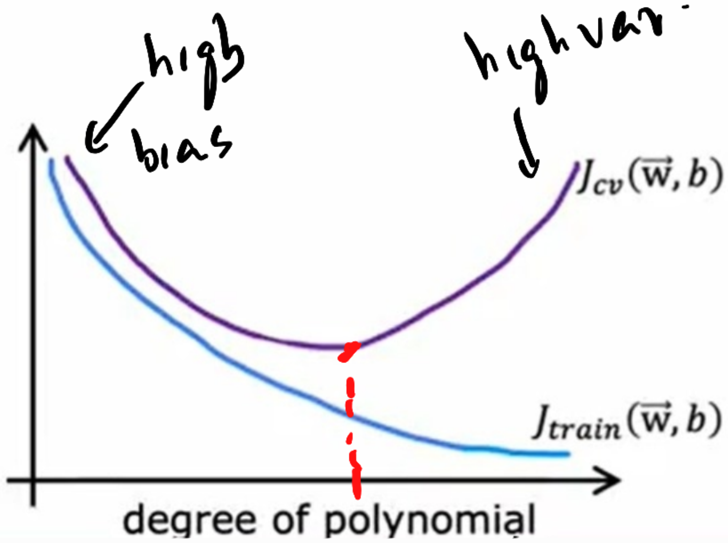
20% cv - cross validation set (dev set) ↳ accuracy check of models

20% - test set

test is used when evaluating

Diagnosing bias and variance

How do you tell if your algorithm has a bias or variance problem?



High bias (underfit)

→ J_{train} will be high not good in train set

($J_{train} \approx J_{cv}$)

High variance (overfit)

→ $J_{cv} \gg J_{train}$

(J_{train} may be low)

worse on cv set

High bias and high variance

J_{train} will be high

and $J_{cv} \gg J_{train}$

$J_{train} \uparrow$ high bias

$J_{train} \downarrow$ $J_{cv} \uparrow$ → if fixed to the data of the train there will be high variance wh Jcv

Some cases;

high bias, high var. is possible

$J_{train} \uparrow$ and $\ll J_{cv} \uparrow$ also

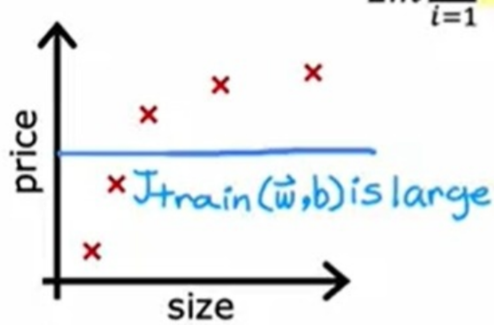
(not common in lin reg.)

Linear regression with regularization

Linear regression with regularization

Model: $f_{\vec{w},b}(x) = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$

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



Large λ

High bias (underfit)

$\lambda = 10{,}000$ $w_1 \approx 0, w_2 \approx 0$

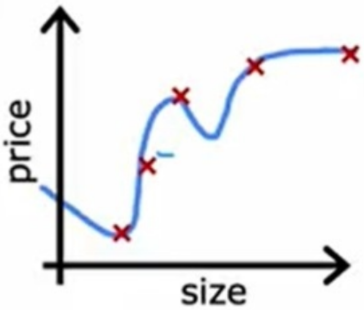
$f_{\vec{w},b}(\vec{x}) \approx b$

$\lambda \uparrow$ to make $J \downarrow$

$w \downarrow$

makes a line

(high bias _ underfit)



Small λ

$\lambda = 0$ $w \uparrow$

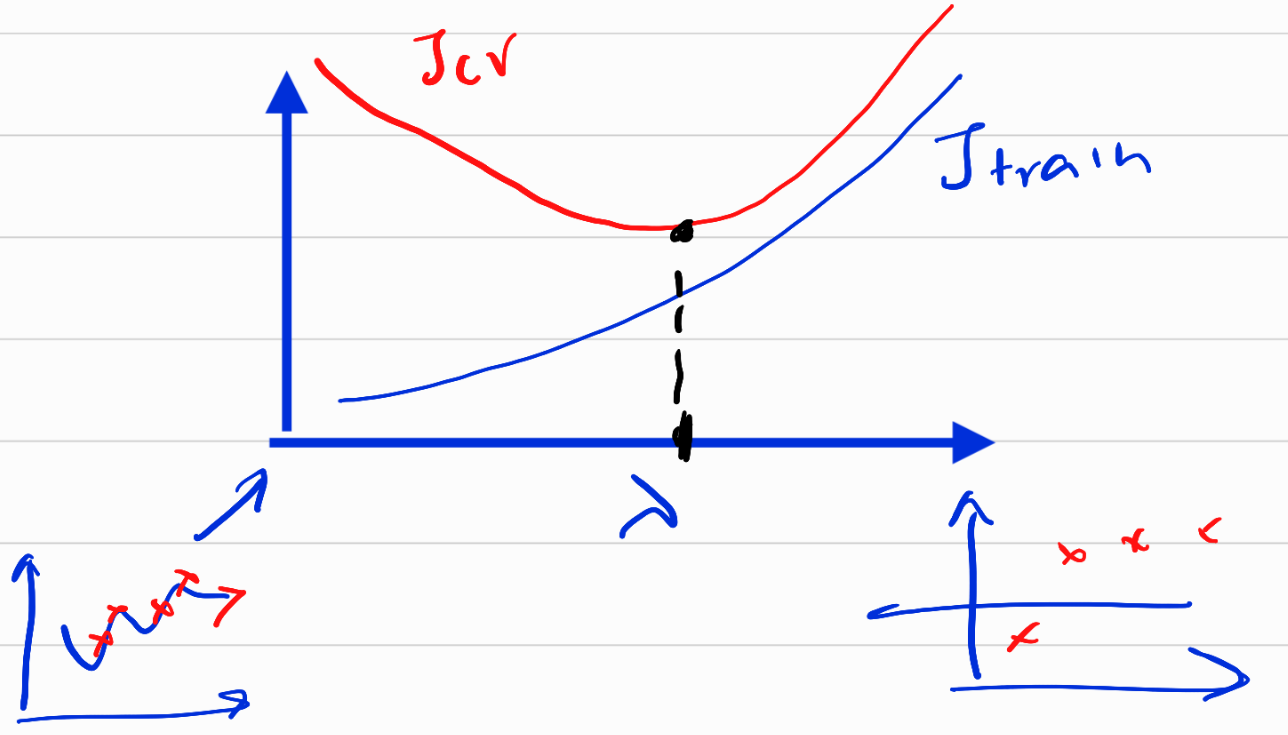
overfits

$J_{train} \ll J_{cv}$

intermediate λ – ✓

for $\lambda \longrightarrow J_{cv}$ lowest

$w^{<x>}, b^{<x>}$



$\lambda = 0$
high var
$J_{cv} \gg J_{t..}$

$\lambda = 10000$
high bias

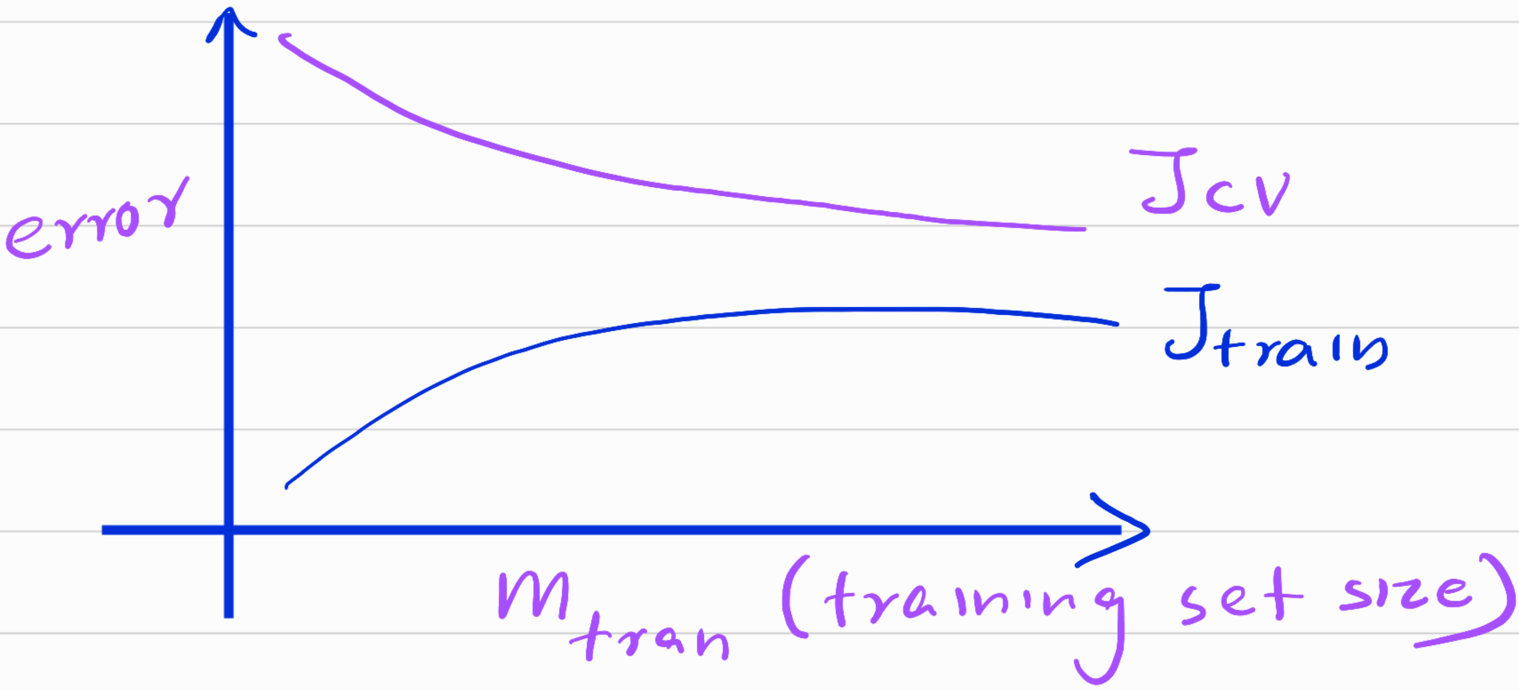
Baseline performance

Baseline perf. (such as human level) — high bias?
(If noise ↳ > 0)

Training error (J_{train}) — high variance?

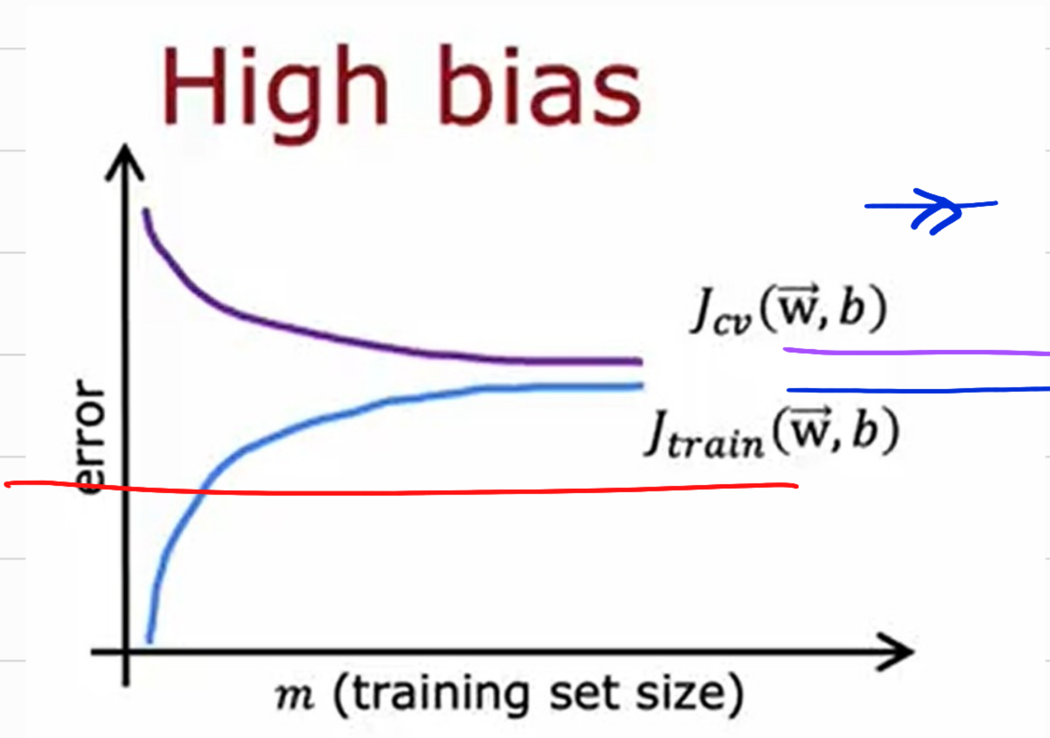
Cross vali error (J_{cv})

Learning Curves;



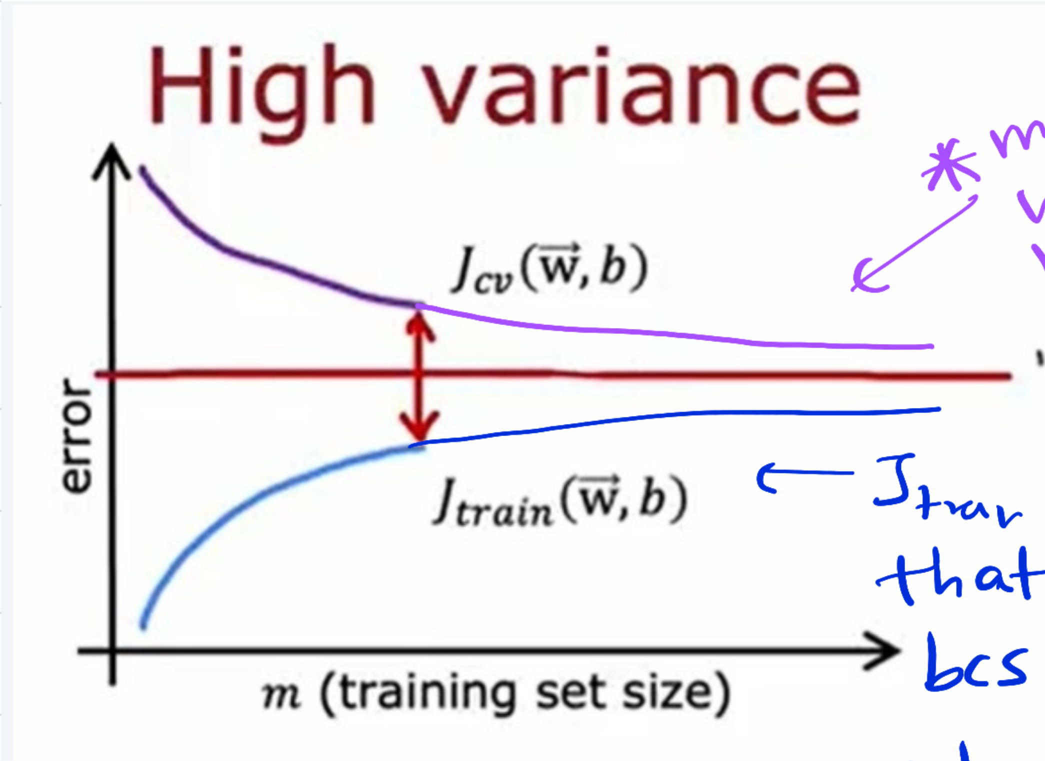
when train examples ↑

hard to fit train error ↑



never goes to human level performance

If high bias more train data doesn't help



* more train data will likely to help

← J_{train} good but that doesn't reflect bcs J_{cv} is what we check

fix

Get more train ex :- high variance ↓

Small sets of feat :- $x, x^2, x^3, x^4, \ldots$ high variance ↓

get additional feat :- high bias ↓

add polynomial. high bias ↓

decrease λ $w \uparrow$ $x + x^4$ high bias ↓

increase λ high var. ↓

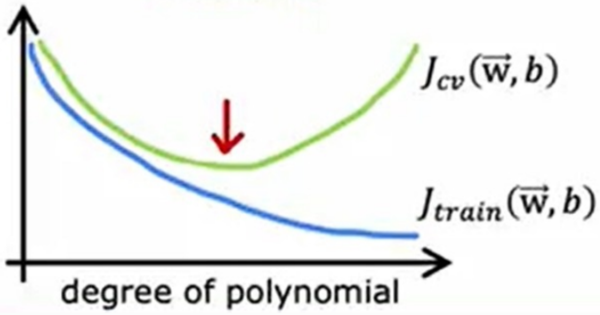
high Var → more data / simplify model

bias → make more complex

* reduce train set ↓ - X for high bias → Worsens J_{cv}

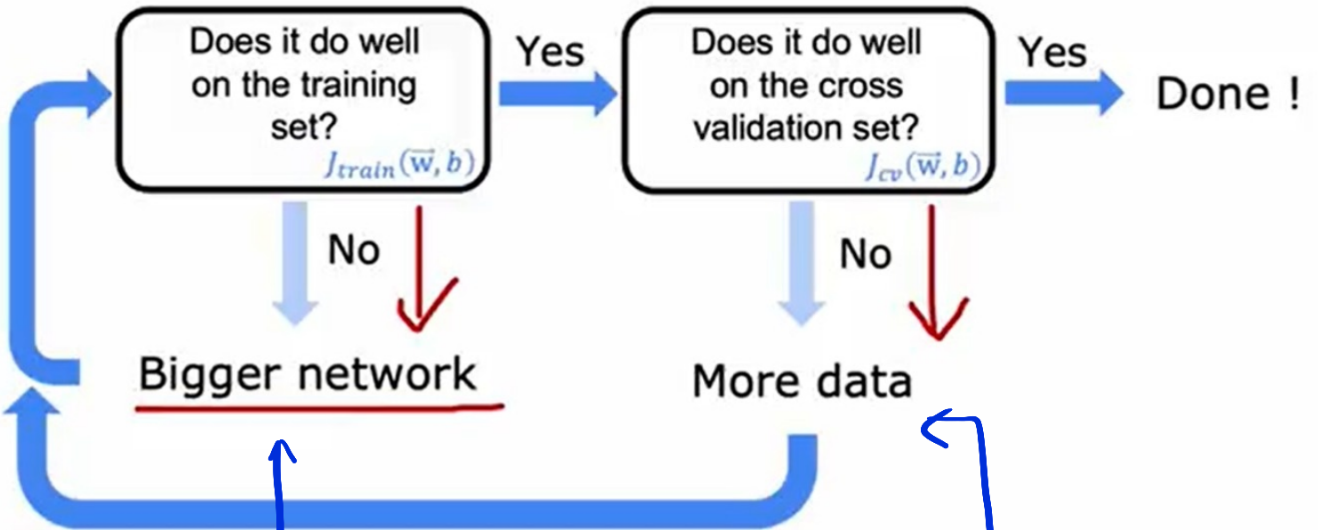
The bias variance tradeoff

$f_{\vec{w},b}(x) = w_1x + b$	$f_{\vec{w},b}(x) = w_1x + w_2x^2 + b$	$f_{\vec{w},b}(x) = w_1x + w_2x^2 + w_3x^3 + w_4x^4 + b$
Simple model		Complex model
High bias	tradeoff	High variance



Neural networks and bias variance

Large neural networks are low bias machines



computational expense

hard to collect

* A large neural network usually do well better than smaller

↳ as long as regulariz. chosen

but comput. expense ↑