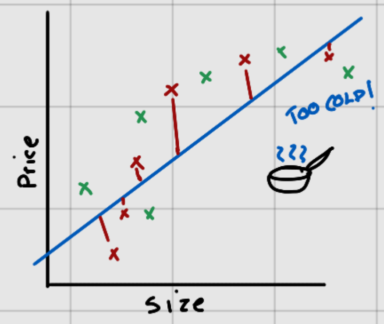
Bias / Variance



$w_1 x + b$

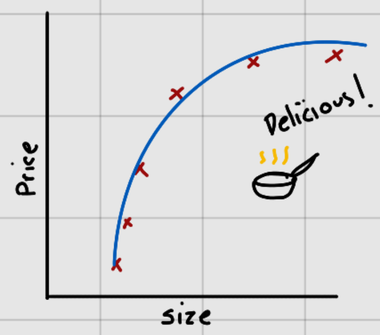
under fit

- Does not fit the training set well

high bias

$d = 1$ J_{train} is high

J_{cv} is high



$w_1 x + w_2 x^2 + b$

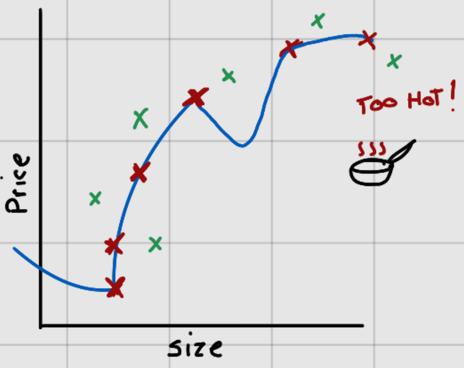
Just right

- Fits training set pretty well

generalization

$d = 2$ J_{train} is low

J_{cv} is low



$w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$

over fit

- Fits the training set extremely well

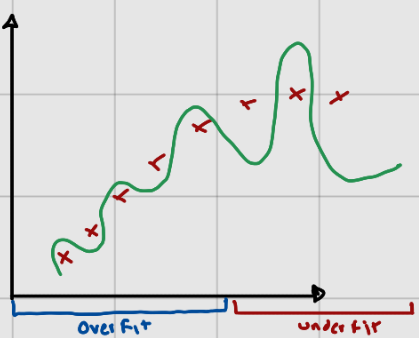
high variance

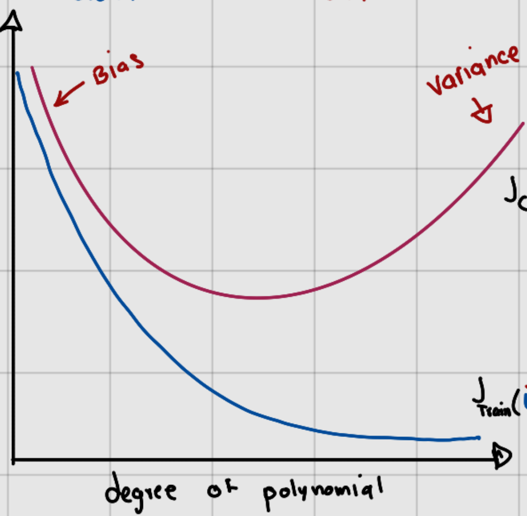
$d = 2$ J_{train} is low

J_{cv} is high

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

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

High variance (overfit)

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

(J_{train} may be low)

High bias and high variance

J_{train} will be high

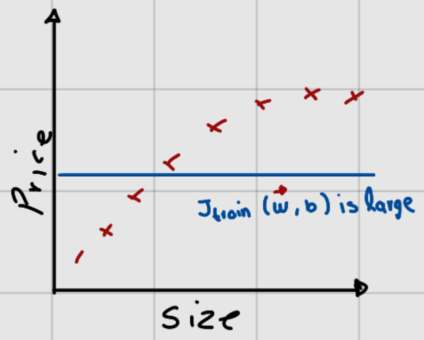
and $J_{cv} \gg J_{train}$

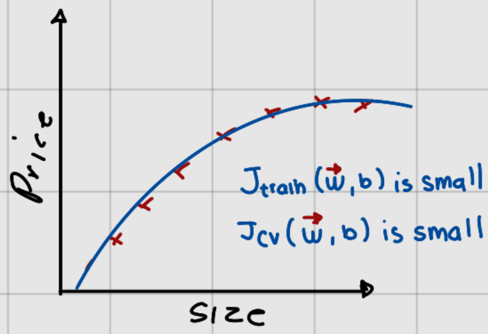
Regularization and bias/variance

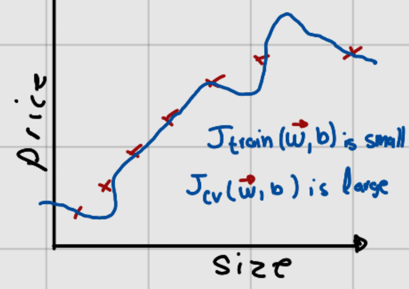
Linear regression with Regularization

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 | Intermediate λ | small λ

λ

High bias (under fit)

High variance (overfit)

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

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

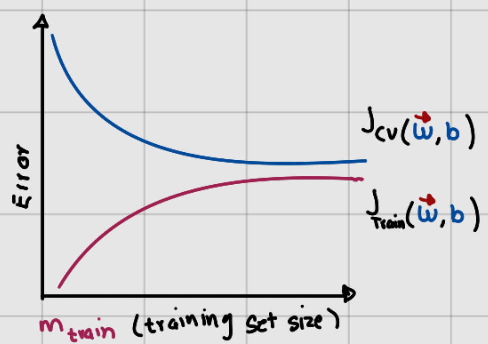
$\lambda = 0$

Learning Curves

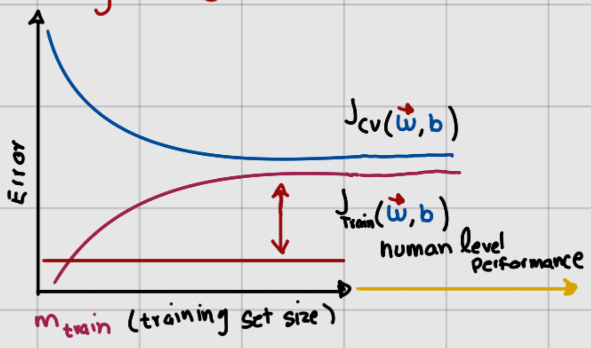
J_{train} = training error

J_{cv} = cross validation error

$f_{\vec{w},b} = w_1 x + w_2 x^2 + b$

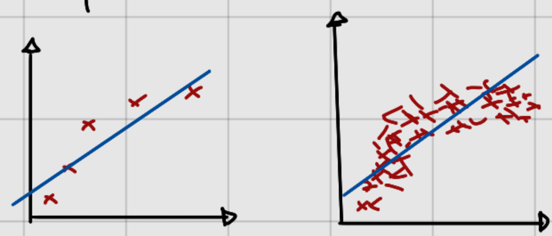


High bias

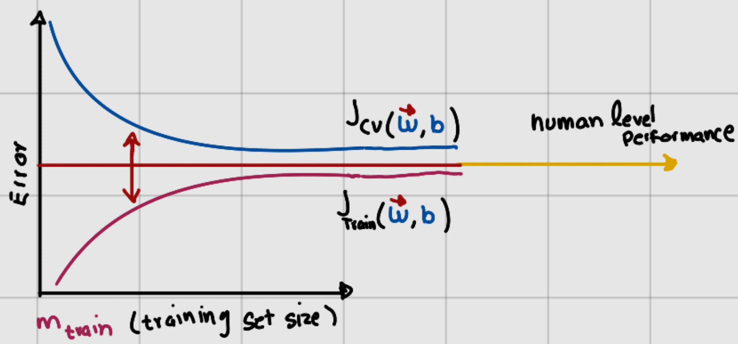


If a learning algorithm suffers from high bias, getting more training data will not help much.

$f_{\vec{w},b} = w_1 x + b$



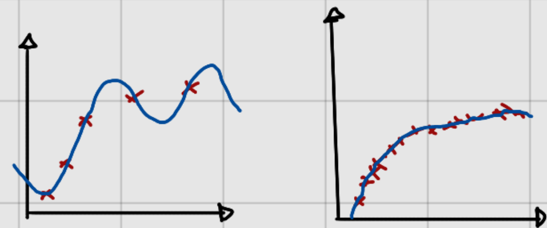
High variance



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

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

(with small λ)



Bias / variance and neural networks

$$J(\mathbf{W},\mathbf{B}) = \frac{1}{m}\sum_{i=1}^{m} L(f(\vec{x}^{(i)}), y^{(i)}) + \frac{\lambda}{2m}\sum_{\textit{all weights } \mathbf{W}} (w^2)$$

b

Unregularized MNIST model

```
layer_1 = Dense(units=25, activation="relu")
layer_2 = Dense(units=15, activation="relu")
layer_3 = Dense(units=1, activation="sigmoid")
model = Sequential([layer_1, layer_2, layer_3])
```

λ

Regularized MNIST model

```
layer_1 = Dense(units=25, activation="relu", kernel_regularizer=L2(0.01))
layer_2 = Dense(units=15, activation="relu", kernel_regularizer=L2(0.01))
layer_3 = Dense(units=1, activation="sigmoid", kernel_regularizer=L2(0.01))
model = Sequential([layer_1, layer_2, layer_3])
```