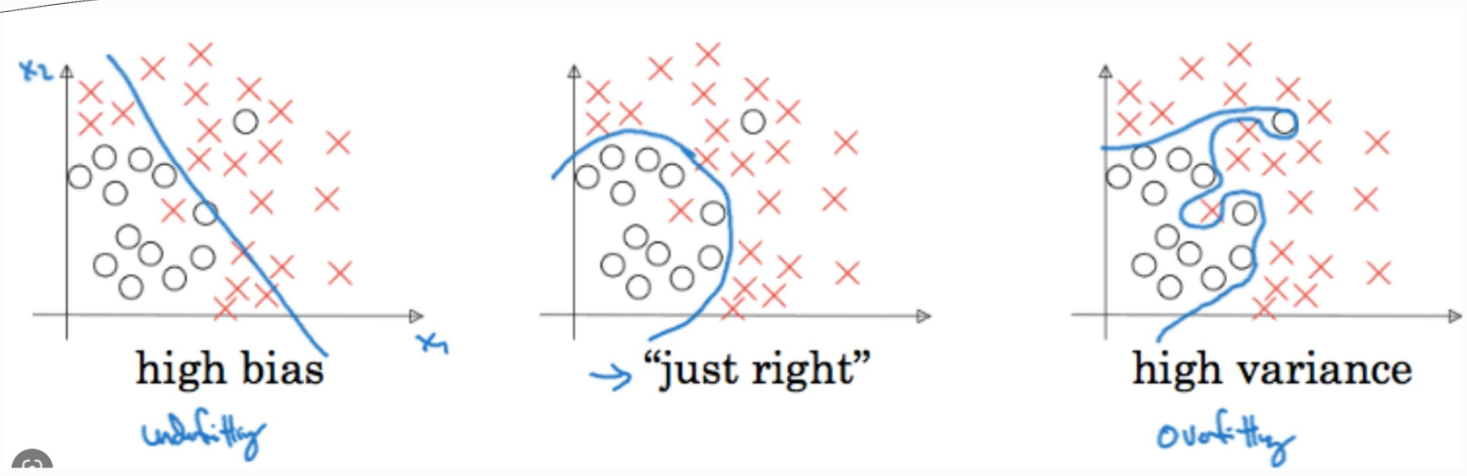
Bias vs Variance



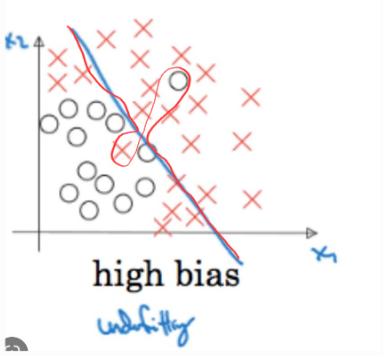
Let's take an example of cat vs dog:-

train error	1%	15%	15%	0.5%
val error	11%	16%	30%	1%
	high variance	high bias	high bias high variance	low bias low variance

Assume human error to be 0.0%

If human error was 15% then this would have been high variance only & this would have been low bias, low variance

High bias & High Variance!-

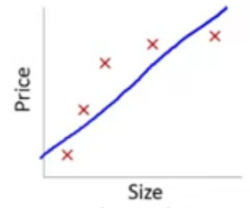


- high bias & high variance

Regularization

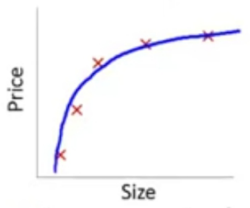
Example

Example: Linear regression (housing prices)



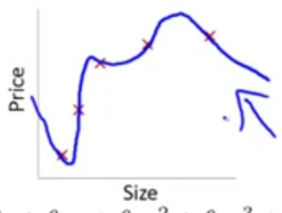
$\rightarrow \theta_0 + \theta_1 x$

"Underfit" "High bias"



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

"Just right"

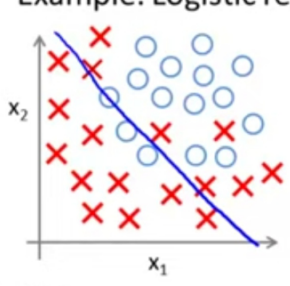


$\rightarrow \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

"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).

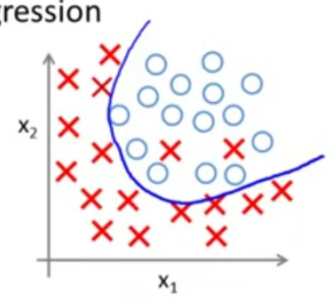
Example: Logistic regression



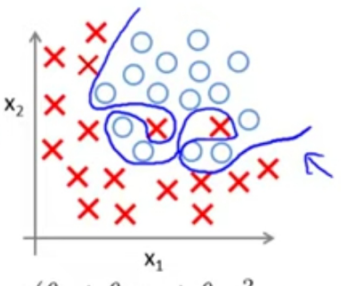
$\rightarrow h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

(g = sigmoid function)

"Underfit"



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



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

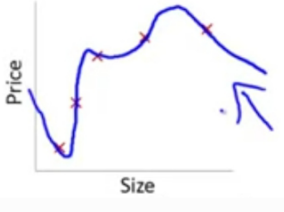
"Overfit"

Addressing overfitting

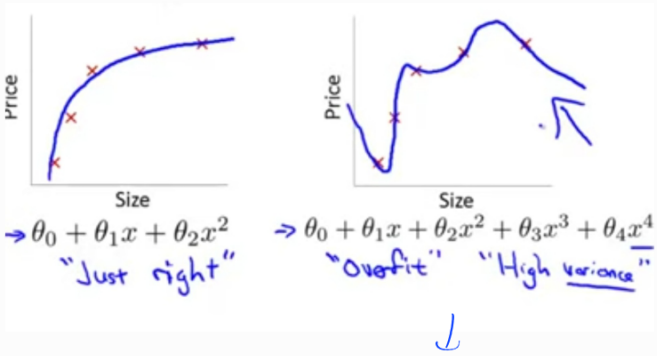
Addressing overfitting:

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



Intuition



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

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

↓

suppose we regularize θ_3, θ_4 & make them really small so,

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

Now when we min. above we'll end up with $\theta_3 \& \theta_4 \approx 0$, which will effectively make cost function to $\theta_0 + \theta_1 x + \theta_2 x^2$, which is good.

Idea

small values for parameters $\theta_0 - - - \theta_n$

- Simpler hypothesis
- Less prone to overfitting

Modified cost function:-

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

$\lambda \sum_{j=1}^{n} \theta_j^2$ → regularization term

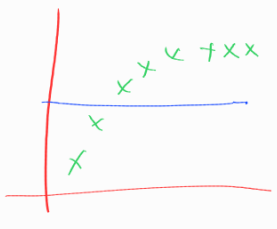
λ → regularization parameter

λ provides us a tradeoff b/w fitting the data & keeping the parameters small.

Q what if λ is very large?

then all θ's will be ≈ 0 & we'll be fitting a line

line parallel to x-axis i.e. $h_\theta(x) = \theta_0$



this is underfitting.

Regularized linear regression

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

let's see gradient descent of linear regression :-

Gradient descent

$\theta_0 \quad \theta_1, \theta_2, \ldots, \theta_n$

Repeat {

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

$$\theta_j := \theta_j - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)} - \frac{\lambda}{m}\theta_j\right] \qquad (j = 1, 2, 3, \ldots, n)$$

}

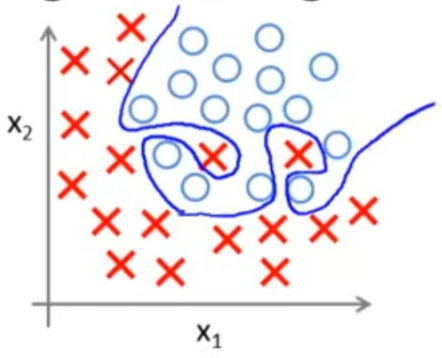
$$\theta_j := \theta_j(1 - \alpha\frac{\lambda}{m}) - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$$

$\approx < 1$ — This same as original

so it becomes $0.9\,\theta_j$ or $0.99\,\theta_j$

Logistic Regression Regularized

Regularized logistic regression.



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

Cost function:

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

to make it regularize we add $+\frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$

Gradient descent

Repeat {

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

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

$(j = 1, 2, 3, \ldots, n)$

}

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

θ_0 $\theta_1, \theta_2, \ldots, \theta_n$

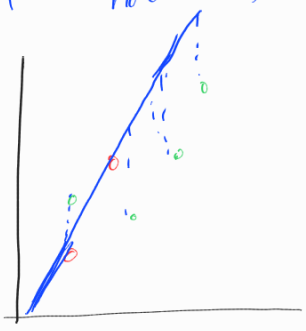
$\frac{\partial}{\partial \theta_0} J(\theta)$

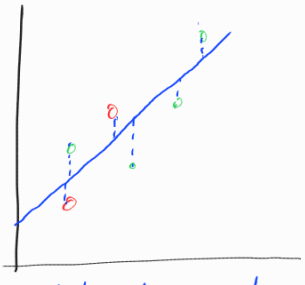
$h_\theta(x) = \frac{1}{1 + e^{-\theta x}}$

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

Ridge Regression

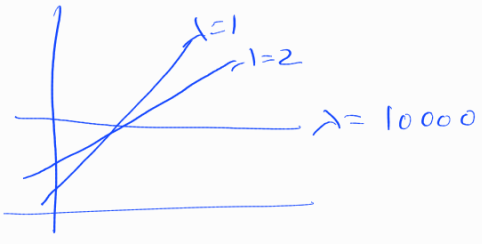
with small bias that penalty creates the least square fit will have large variance.





no bias, high variance little bias, low variance

↑ λ = ↓ slope



In order to find correct λ we just try a bunch of value for λ & use cross-validation, typically 10-fold cross validation, to determine which one results in lower variance.

Lasso vs Ridge

Regularization

Regularization seeks to control variance by adding a tuning parameter, lambda, or alpha:

LASSO (L1 regularization)

- regularization term penalizes absolute value of the coefficients
- sets irrelevant values to 0
- might remove too many features in your model

Ridge regression (L2 regularization)

- penalizes the size (square of the magnitude) of the regression coefficients
- enforces the *B* (slope/partial slope) coefficients to be lower, but not 0
- does not remove irrelevant features, but minimizes their impact