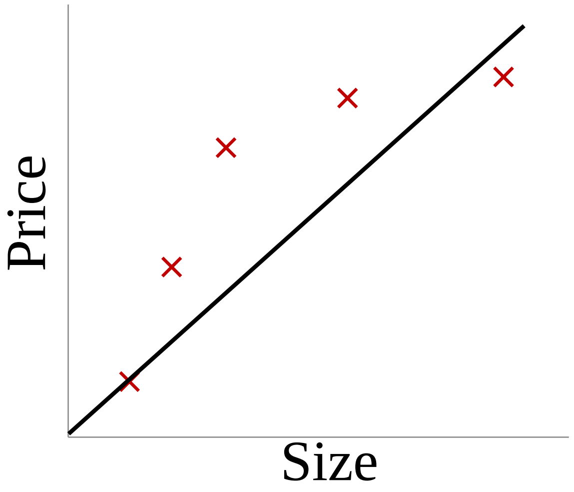
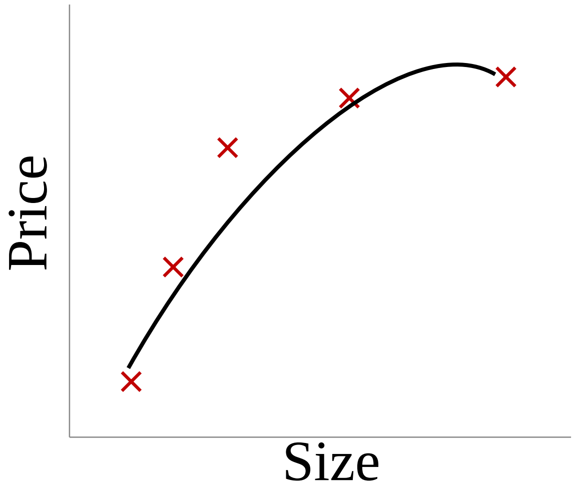
Example: Linear regression (housing prices)



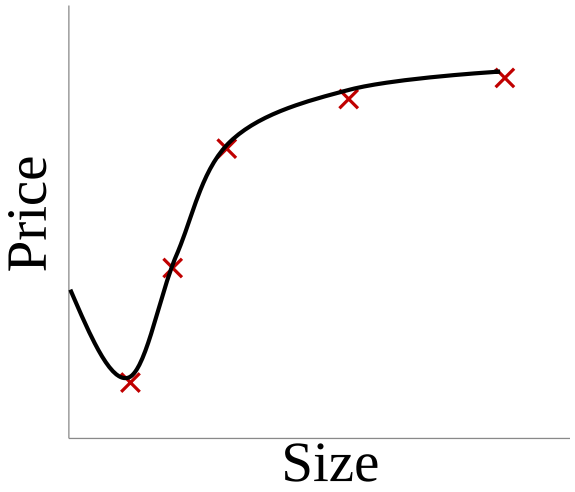
$$\theta_0 + \theta_1 x$$

Underfitting: "High Bias"



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

Correct Fit

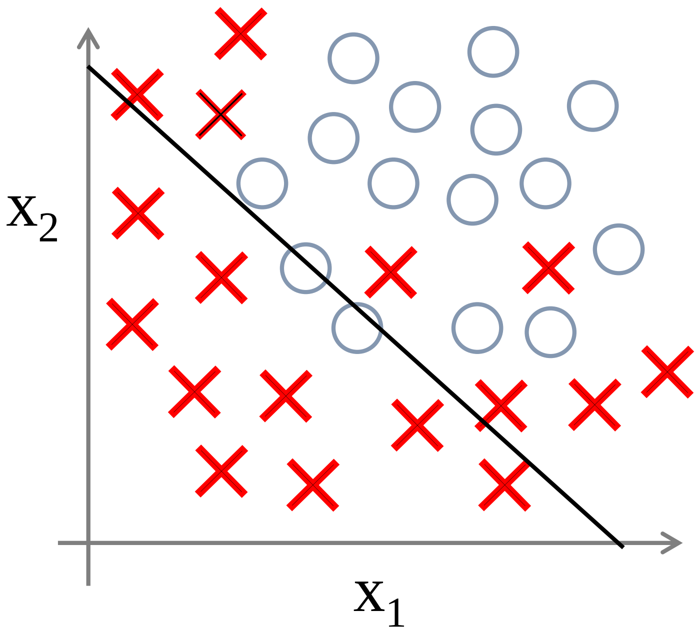


$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Overfitting: "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).

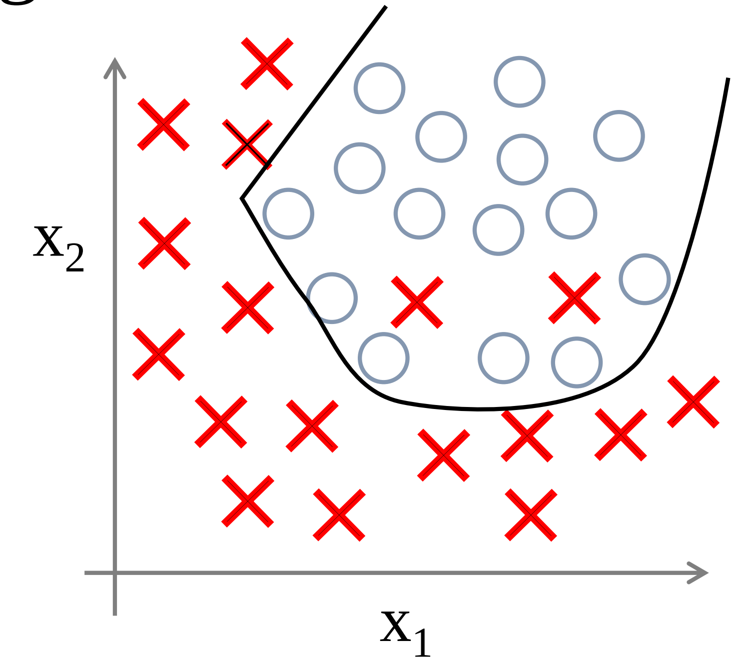
Example: Logistic regression



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

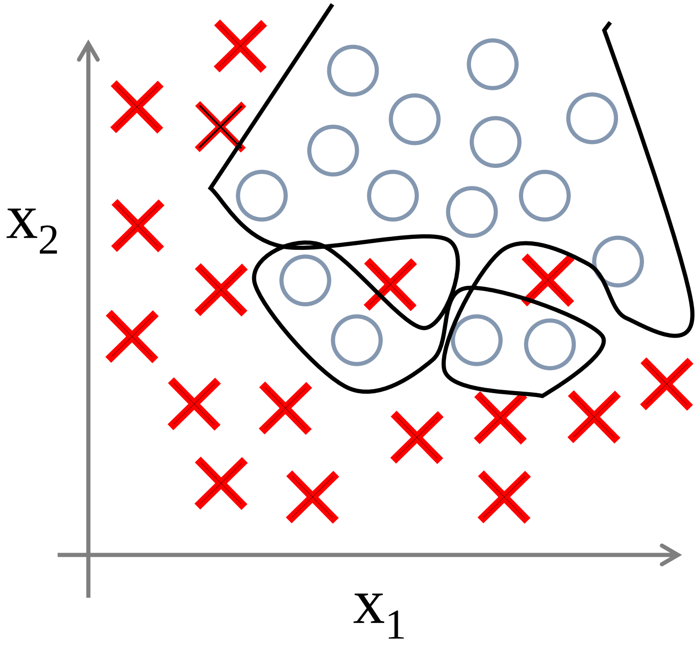
(g = sigmoid function)

Underfitting: “High Bias”



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

Correct Fit



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

Overfitting: “High Variance”

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.

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj

assume lamda = 1000

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

theta 3 & theta 4 is going to decrease

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj^2

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

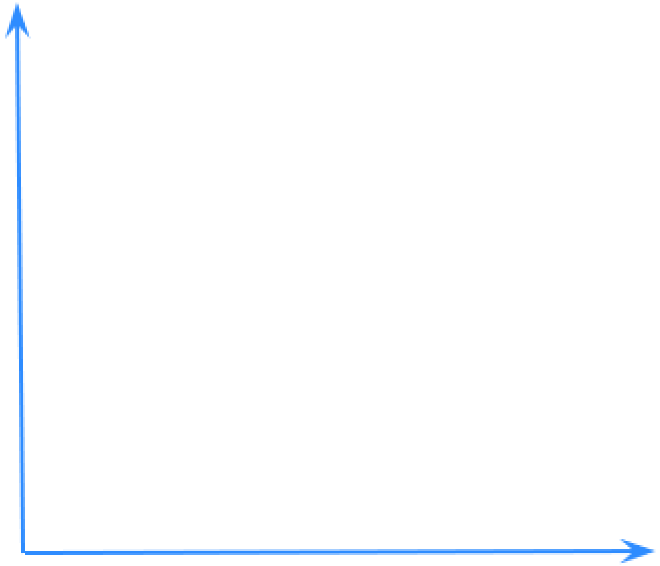
theta 3 & theta 4 is going to decrease

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

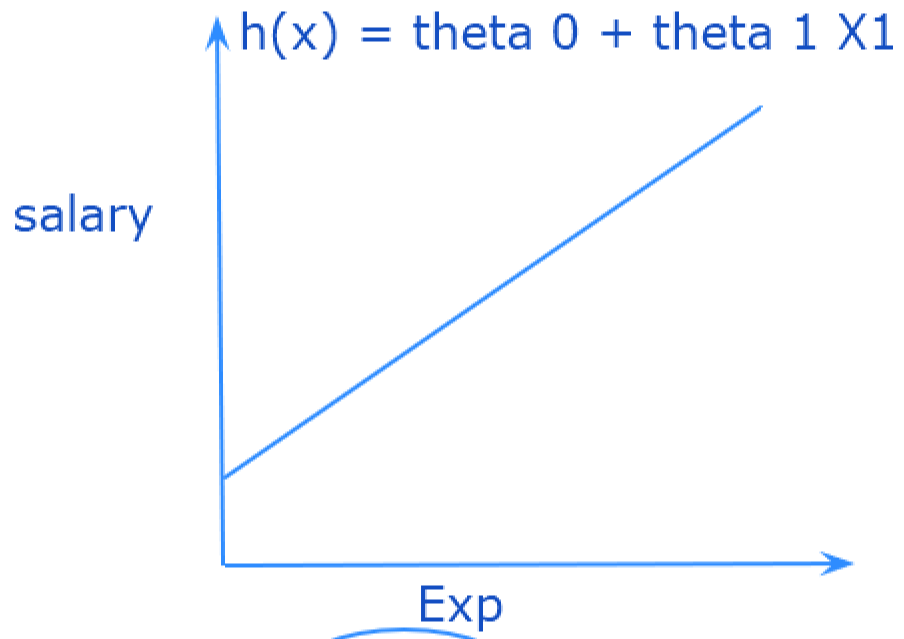


Min = above terms are same + lamda/2m summation j = 1 to n thetaj^{2}

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

theta 3 & theta 4 is going to decrease



Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter

Min = above terms are same + lamda/2m summation j = 1 to n thetaj^{2}

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

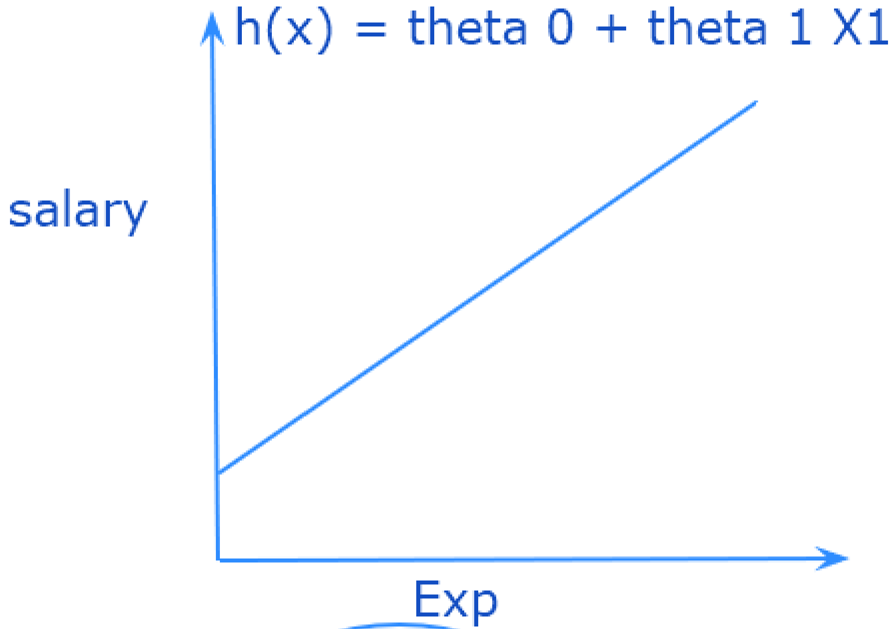
theta 3 & theta 4 is going to decrease

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter



Min = above terms are same + lamda/2m summation j = 1 to n thetaj^2

lamda(slope)^2

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

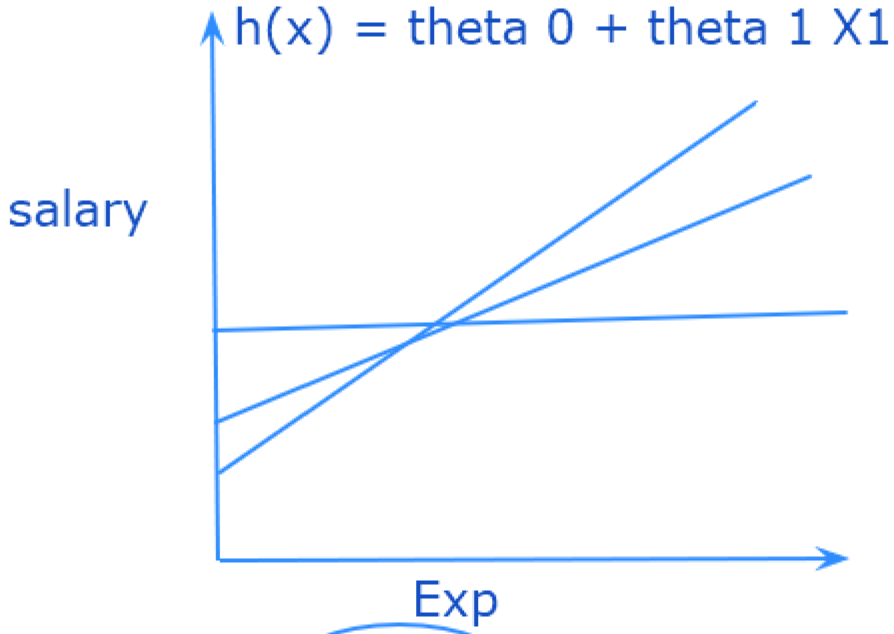
theta 3 & theta 4 is going to decrease

Regularization

Min = 1/2m summation i=1 to m (h(X)^i - y^i)2

parameters= thetaj

lamda = regularization parameter



Min = above terms are same + lamda/2m summation j = 1 to n thetaj^2

lamda(slope)^2

assume lamda = 1000

Min = terms + 1000 (theta 3)^2 + 1000 (theta 4)^2

theta 3 & theta 4 is going to decrease

Regularized Linear Regression

J(theta) = 1/2m [summation i= 1 to m (h(x)^i - yi)^2 + lamda summation j=1 to n theta j^2]

Regularized Linear Regression

J(theta) = 1/2m [summation i= 1 to m (h(x)^i - yi)^2 + lamda summation j=1 to n theta j^2]

Regularized Logistic Regression

J(theta) = -[1/m summation i = 1 to m yi log (h(x)^i + (1-Yi) log (1-h(x)^i)] + lamda /2m summation j= 1 to n (thetaj)^2

Gradient Descent Algorithm

{

Theta 0:= theta 0 - alpha (1/m) summation i = 1 to m (h(x)^i - y i) X0^i
Theta j := theta j - alpha[1/m summation i = 1 to m (h(x)i - yi)Xj^i - lamda/m thetaj

(for j = 1, 2, 3, ...n)

}

Gradient Descent Algorithm

{

Theta 0:= theta 0 - alpha (1/m) summation i = 1 to m (h(x)^i - y i) X0^i
Theta j := theta j - alpha[1/m summation i = 1 to m (h(x)i - yi)Xj^i - lamda/m thetaj

(for j = 1, 2, 3, ...n)

}

penalize the features that has higher value of slope
lamda = 0 to +ve values

Gradient Descent Algorithm

{

Theta 0:= theta 0 - alpha (1/m) summation i = 1 to m (h(x)^i - y i) X0^i
Theta j := theta j - alpha[1/m summation i = 1 to m (h(x)i - yi)Xj^i - lamda/m thetaj

(for j = 1, 2, 3, ...n)

}

penalize the features that has higher value of slope
lamda = 0 to +ve values

Ridge regression & lasso regression

ridge regression

lamda(slope)^2

Lasso regression

lamda(absolute value of the slope)

ridge regression

lamda(slope)^2

Lasso regression

lamda(absolute value of the slope)

ridge regression

lamda(slope)^2 closely equal to zero

Lasso regression

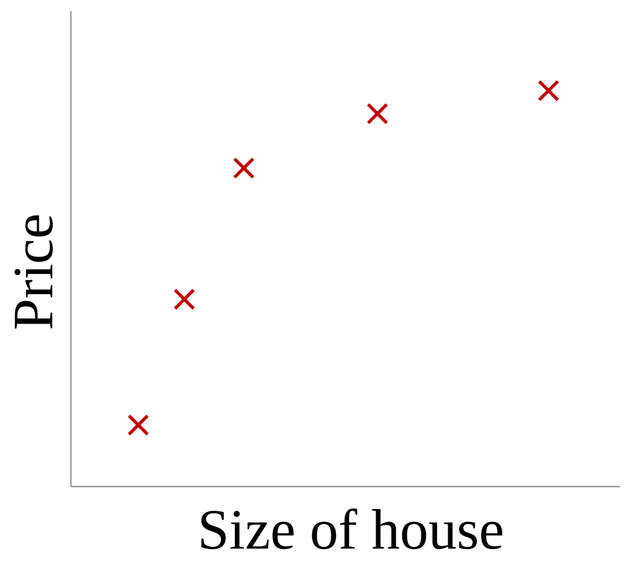
lamda(absolute value of the slope) = zero ridge regression   when most predictors impact the response

lamda(slope)^2   closely equal to zero

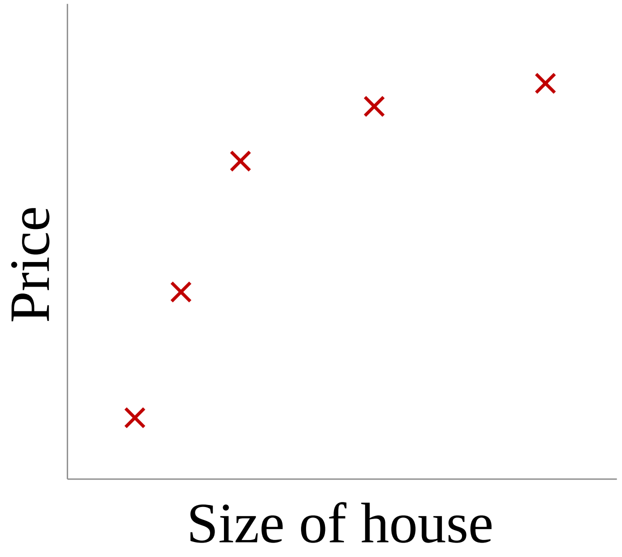
Lasso regression   when only a few predictors actually influence the response

lamda(absolute value of the slope)   = zero

Intuition



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



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

Suppose we penalize and make θ_3, θ_4 really small.

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

Regularization.

Small values for parameters $\theta_0, \theta_1, \ldots, \theta_n$

- "Simpler" hypothesis
- Less prone to overfitting

Housing:

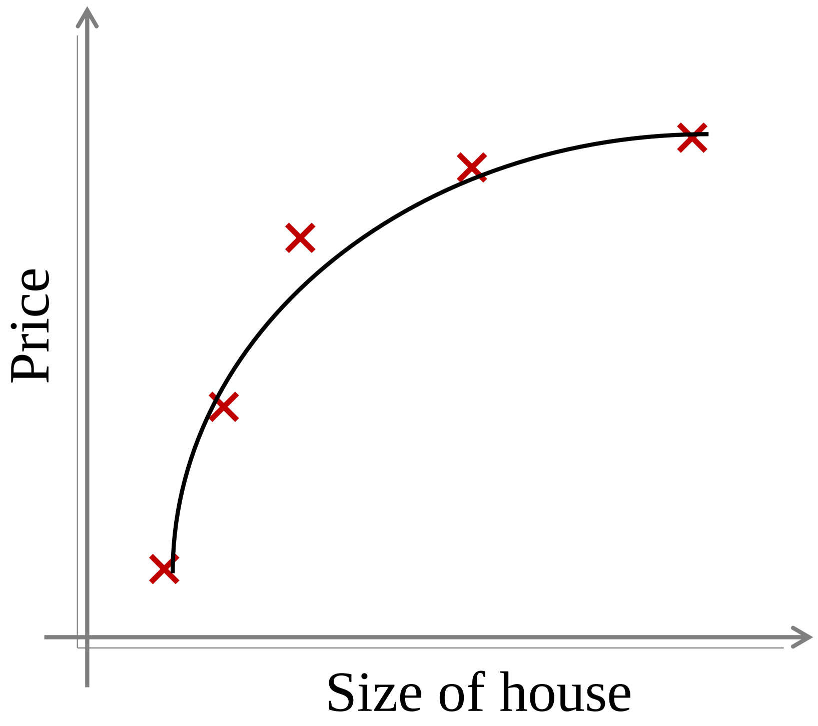
- Features: $x_1, x_2, \ldots, x_{100}$
- Parameters: $\theta_0, \theta_1, \theta_2, \ldots, \theta_{100}$

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

Regularization.

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

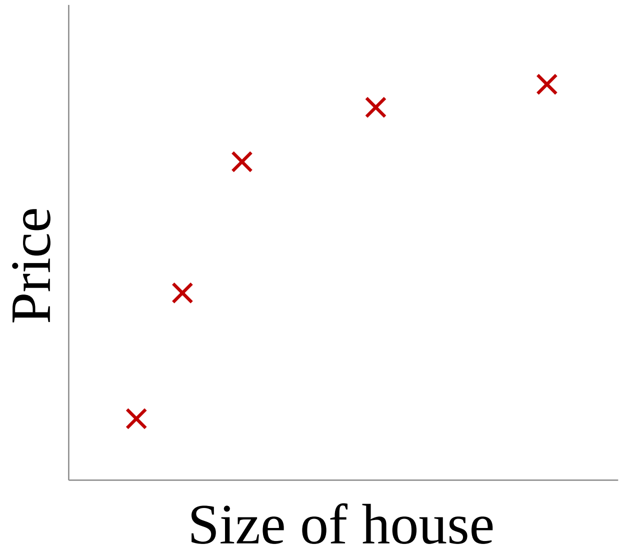
$$\min_\theta J(\theta)$$



In regularized linear regression, we choose θ to minimize

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

What if λ is set to an extremely large value (perhaps for too large for our problem, say $\lambda = 10^{10}$)?



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

Regularized linear regression

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

$$\min_\theta J(\theta)$$

Gradient descent

Repeat $\{$

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

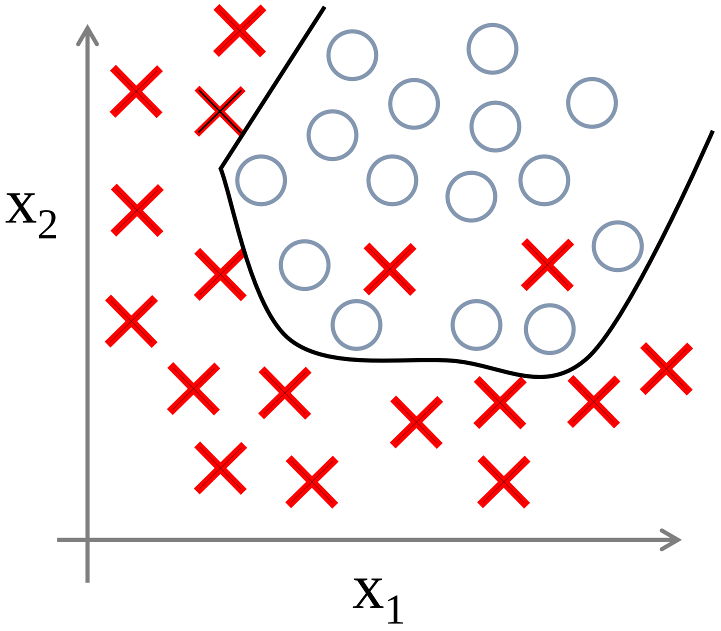
$$\theta_j := \theta_j - \alpha \qquad \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

$$(j = \cancel{0}, 1, 2, 3, \ldots, n)$$

$\}$

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

Regularized logistic regression.



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

Cost function:

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

$$+ \frac{lambda}{2m} summation\ i = 1\ to\ n\ thetaj2$$

Gradient descent

Repeat $\{$

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

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

$$(j = \cancel{0}, 1, 2, 3, \ldots, n)$$

$\}$