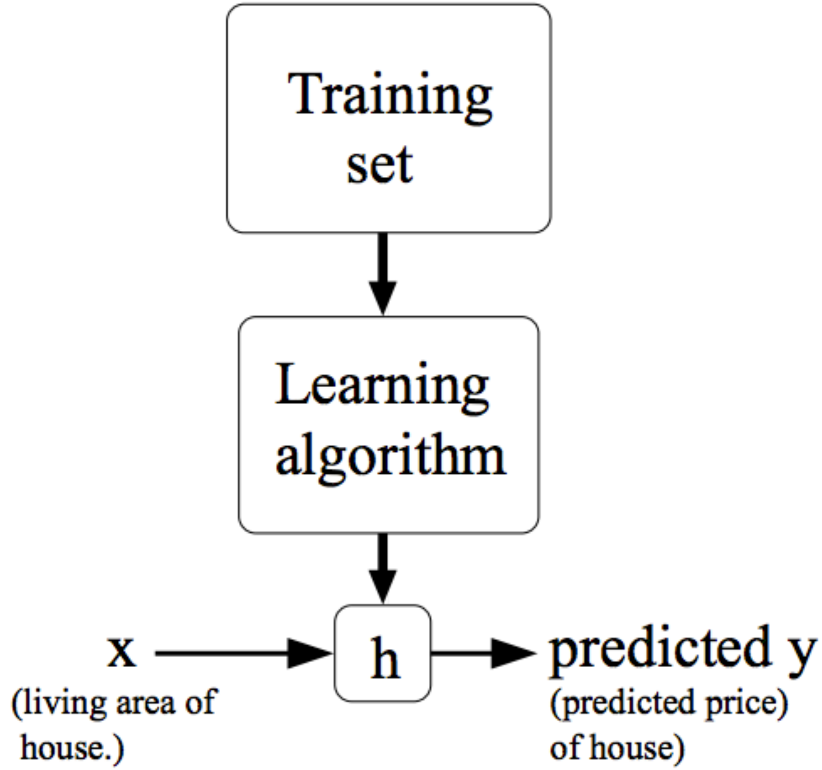
Linear Regression

By Mandel Oats
Based on notes by Andrew Ng

TEXAS A&M UNIVERSITY

Machine Learning Model

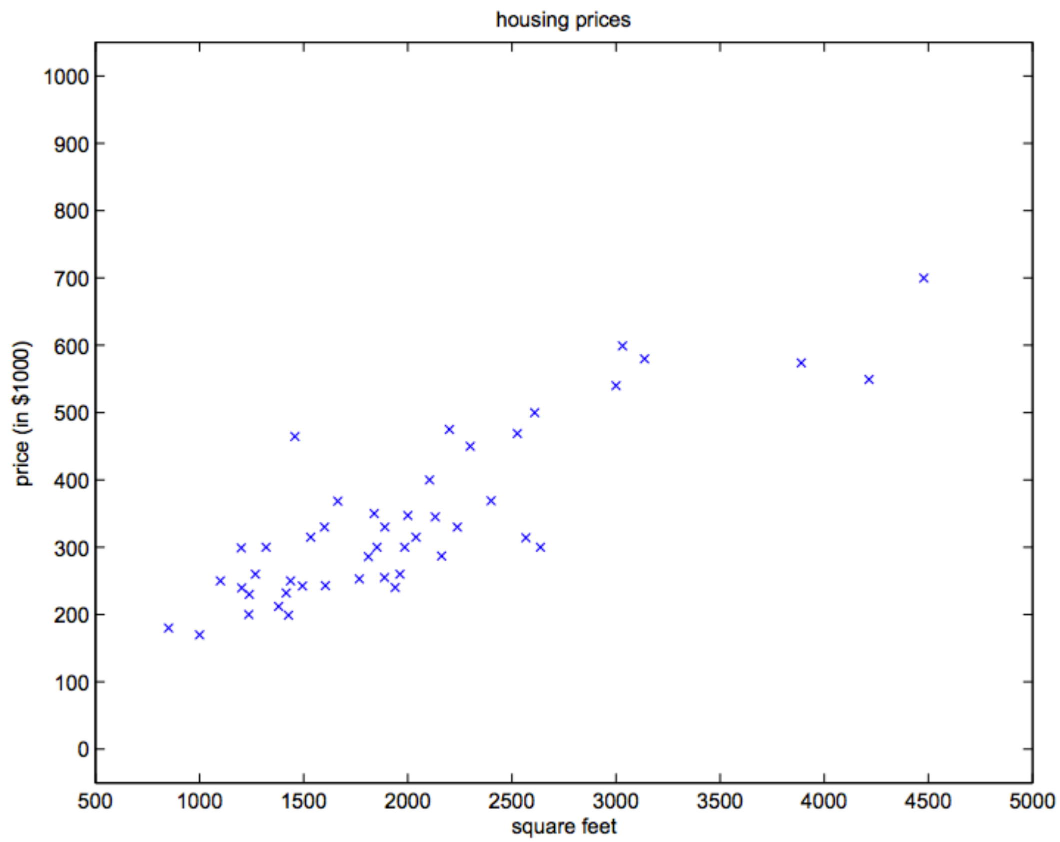


Training Set

Living area ($feet^2$)	#bedrooms	Price (1000$s)
2104	3	400
1600	3	330
2400	3	369
1416	2	232
3000	4	540
$\vdots$	$\vdots$	$\vdots$

x_1 x_2 y

Plotting Data



Hypothesis

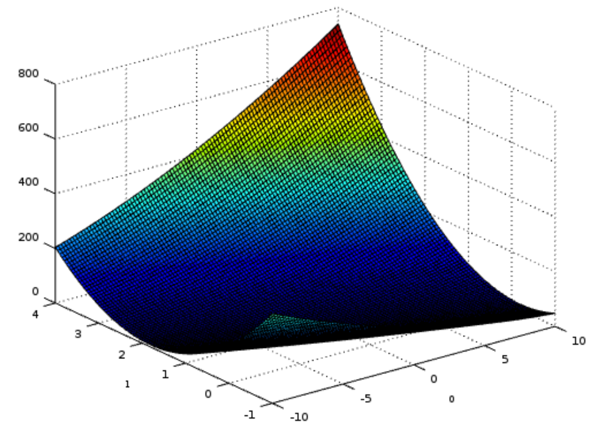
$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

Ordinary Least Squares

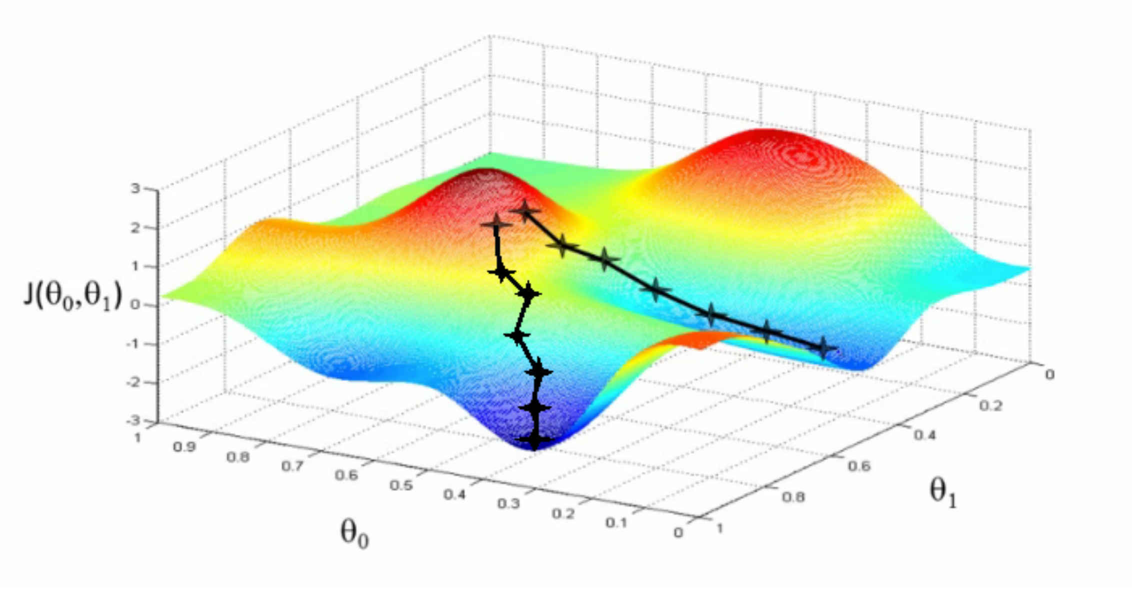
- Cost Function

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



Gradient Descent

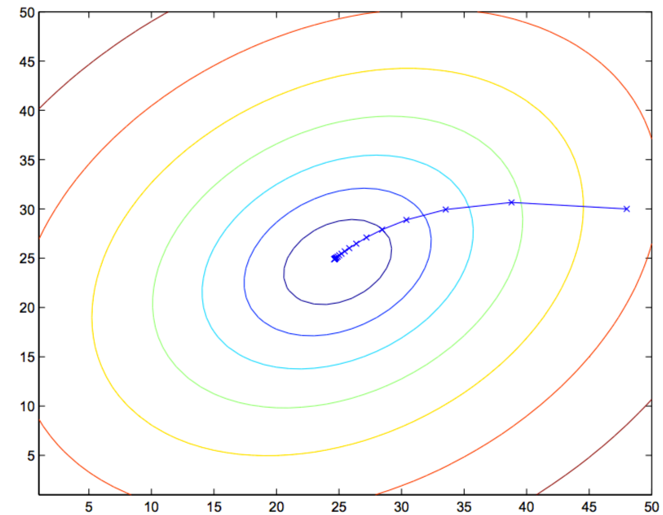
- Update $$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$



Batch Update Equation

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

- Repeat until convergence
- Guaranteed to converge to global minimum (Cost function is convex)



Alternative Minimization

- Stochastic Gradient Descent
  - Not guaranteed to converge exactly
  - Better for large training set
- Normal Equation

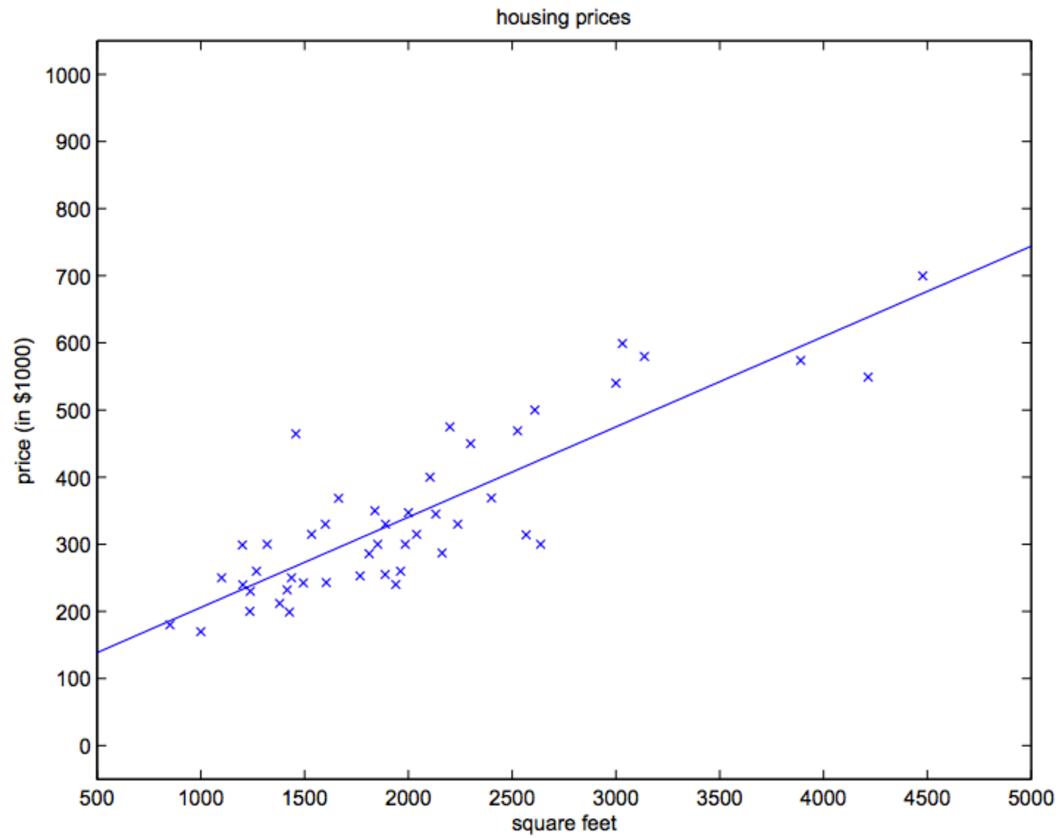
Normal Equation

$$\nabla_\theta J(\theta) = X^T X\theta - X^T\vec{y}$$

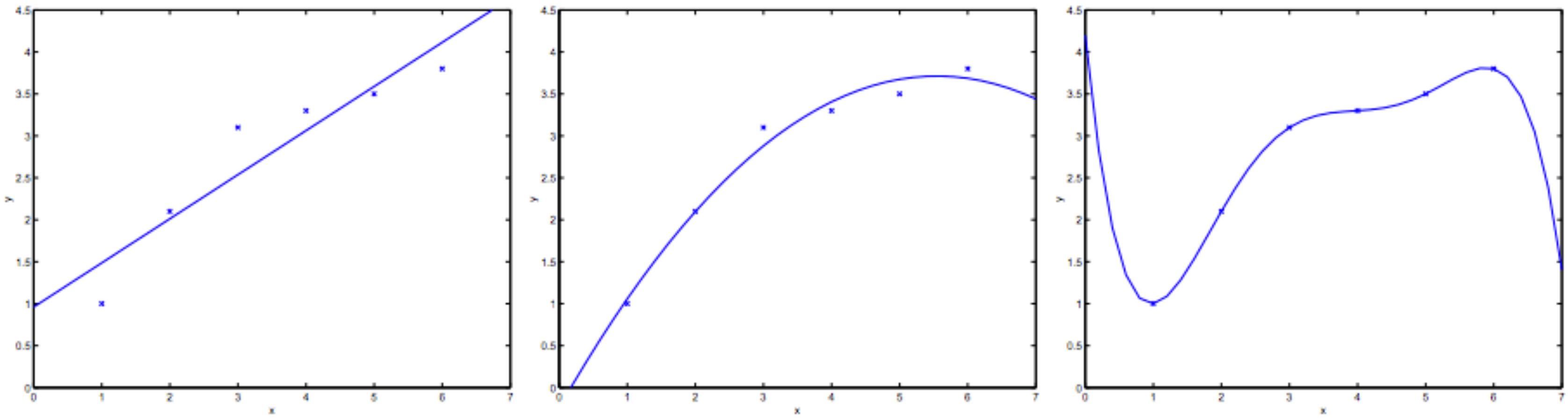
$$X^T X\theta = X^T\vec{y}$$

$$\theta = (X^T X)^{-1} X^T\vec{y}$$

Linear Regression



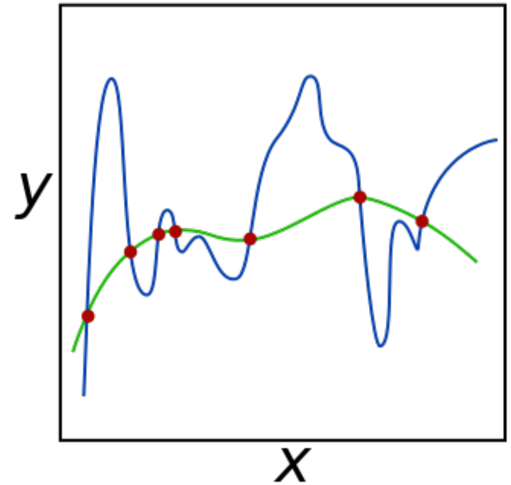
Polynomial Regression



Regularization

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

$$\theta = \left(X^TX + \lambda\begin{bmatrix} 0 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1\end{bmatrix}\right)^{-1} X^Ty$$



Example

TEXAS A&M
UNIVERSITY