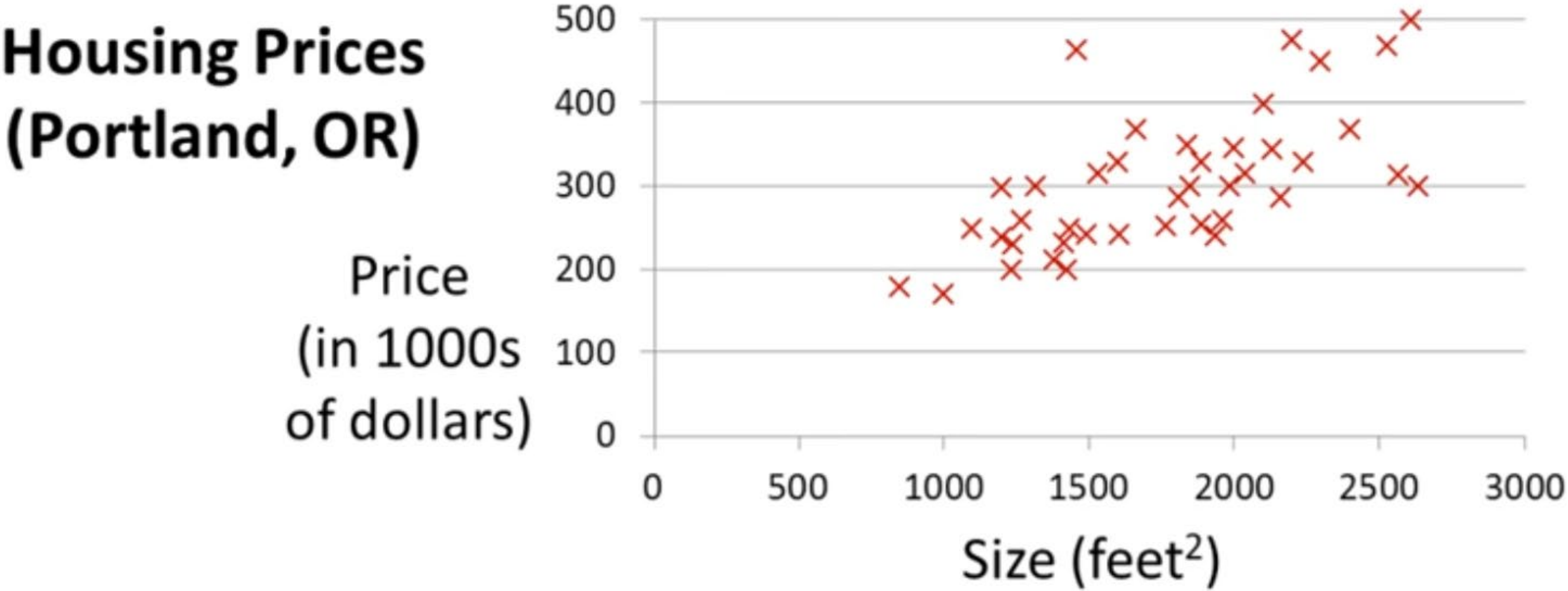
Introduction to Machine Learning

Linear Regression

A characterization of machine learning problems

	Supervised	*Unsupervised*
Discrete	Classification	Clustering
Continuous	Regression	Dimensionality reduction

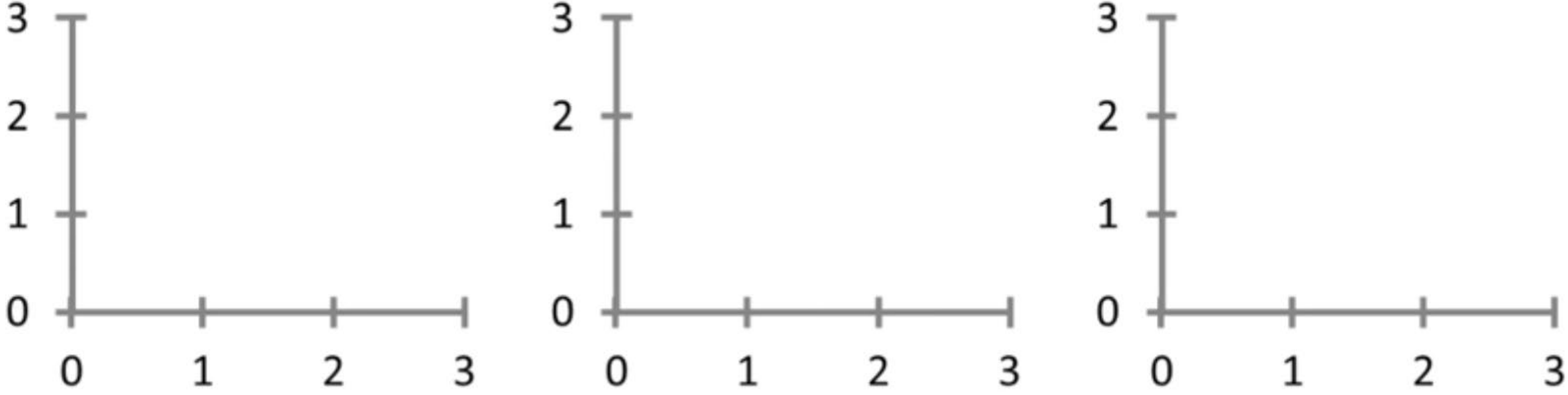
Example - House prices



Goal

- Map data X to real value Y
- Linear regression, mapping is linear function of X

$$h_\theta(x) = \theta_0 + \theta_1 x$$

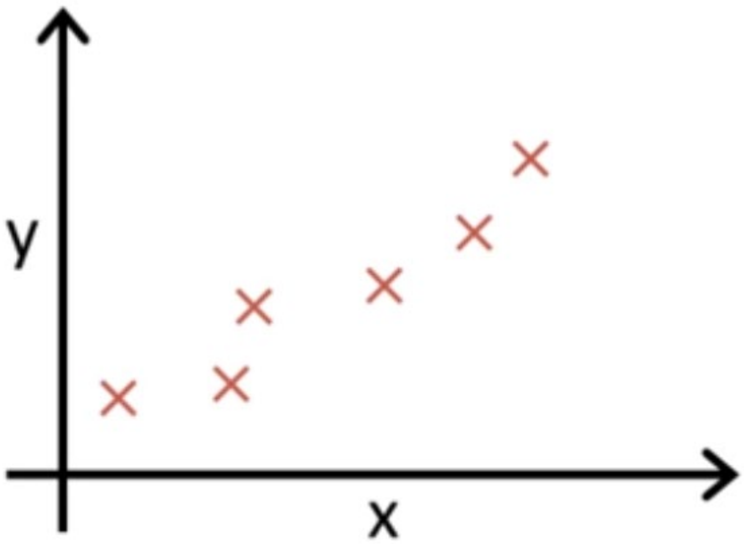


$\theta_0 = 1.5$
$\theta_1 = 0$

$\theta_0 = 0$
$\theta_1 = 0.5$

$\theta_0 = 1$
$\theta_1 = 0.5$

Choose θ so that $h(x)$ is close to y for training examples (x,y)



Loss (cost) function

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

- Simplified

- Hypothesis
  - $\mathrm{h}(x) = \theta_0 + \theta_1 x$

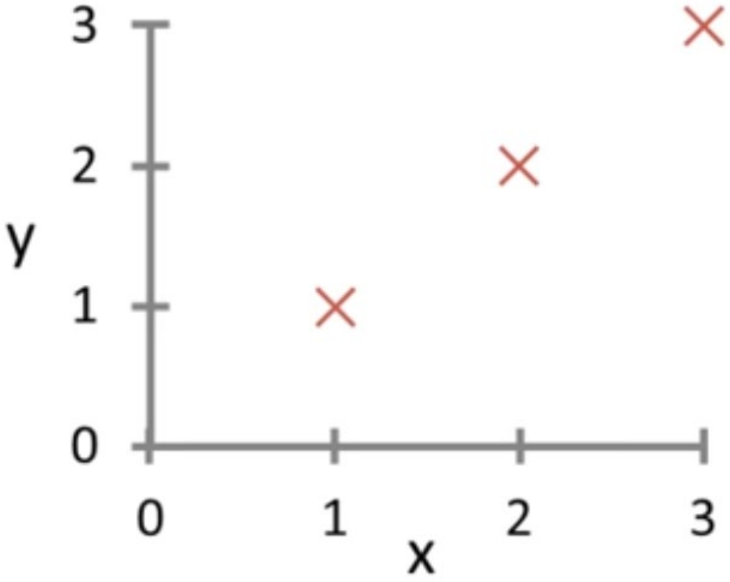
- Parameters
  - θ_0, θ_1

- Cost function
  - $J(\theta_0, \theta_1) = \frac{1}{2n} \sum_{i=1}^{n} (h(x^i) - y^i)^2$

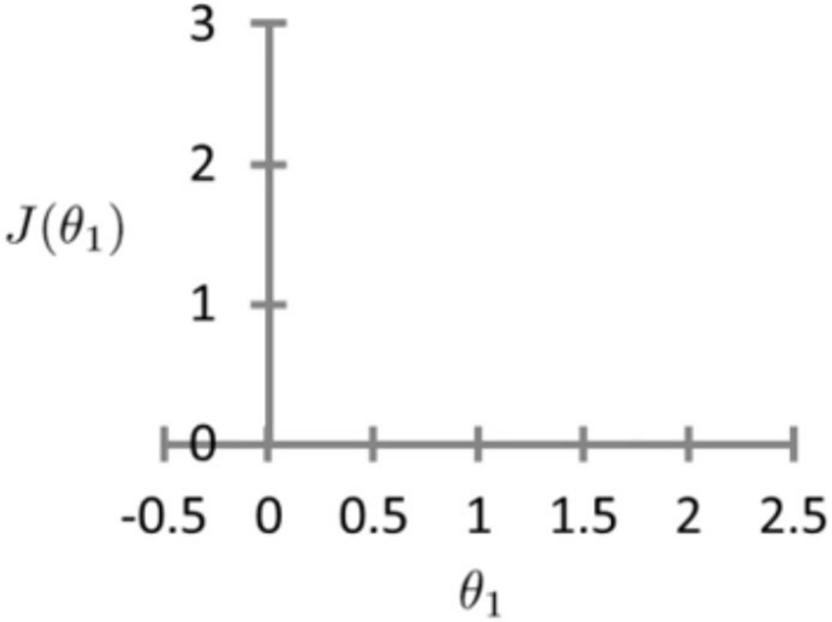
- Goal
  - $min_{\theta_1, \theta_2}\ J(\theta_0, \theta_1)$

Search for θ

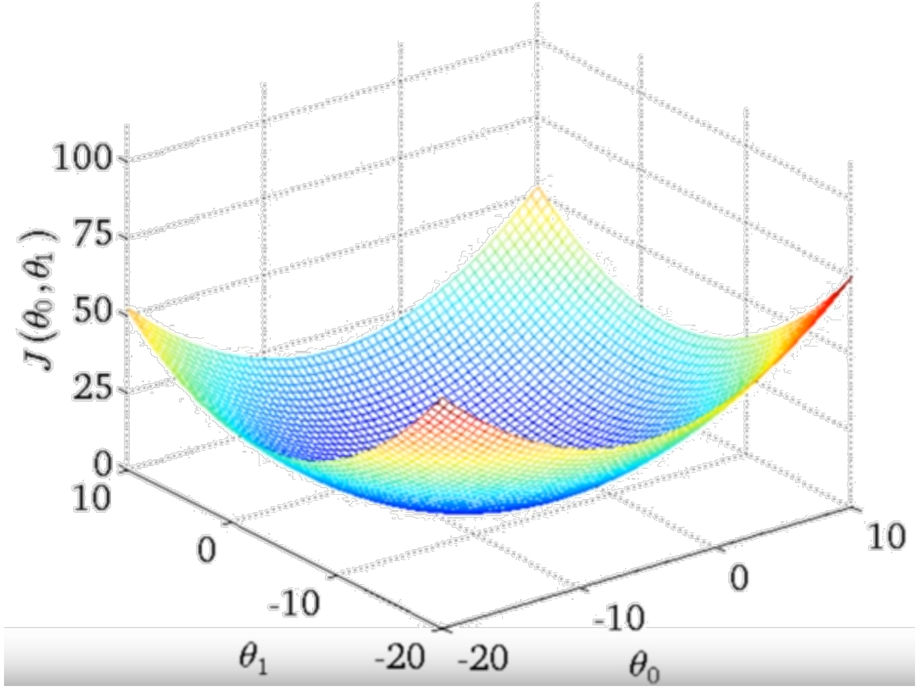
h(x)



J(θ_1)



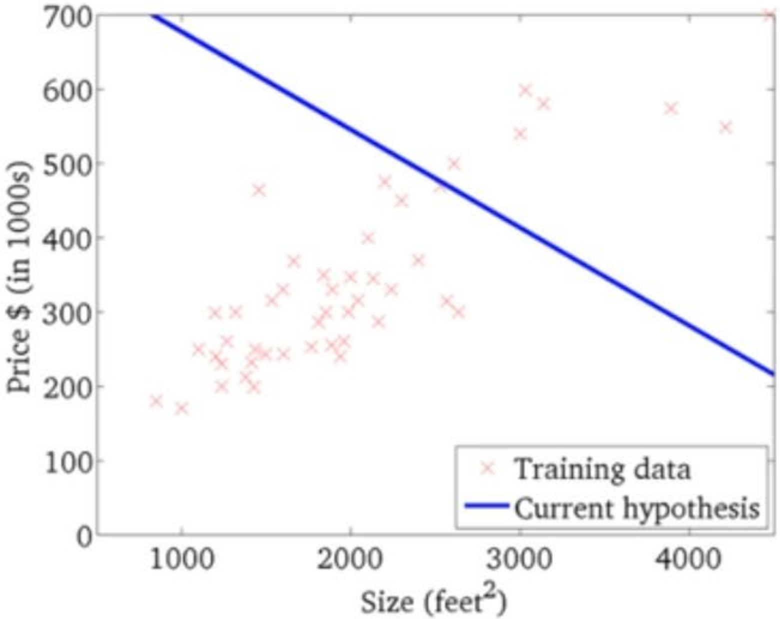
3D surface plot for two dimensions (θ_1, θ_2)



Contour plot for two dimensions (θ_1, θ_2)

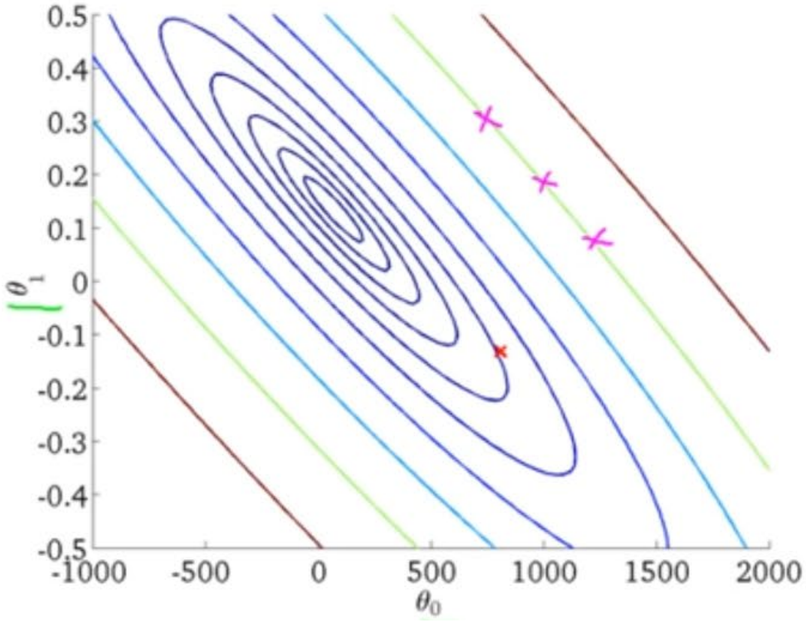
h(x)

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_1, \theta_2)$

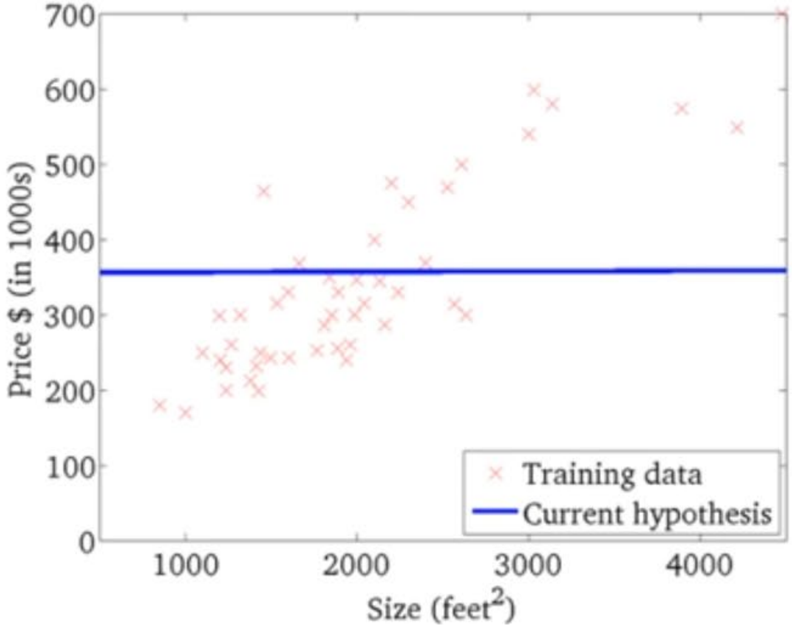
(function of the parameters θ_0, θ_1)



Contour plot for two dimensions (θ_1, θ_2)

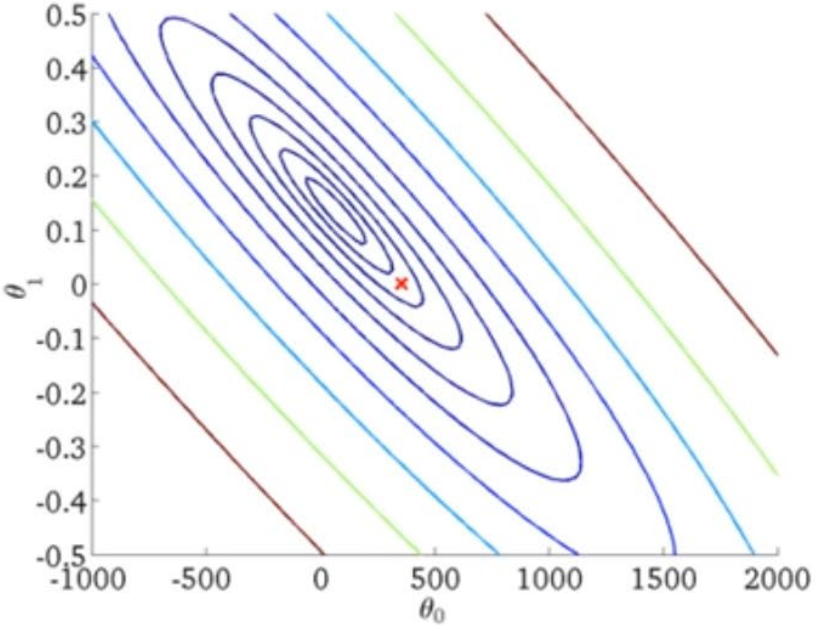
$h(x)$

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_1, \theta_2)$

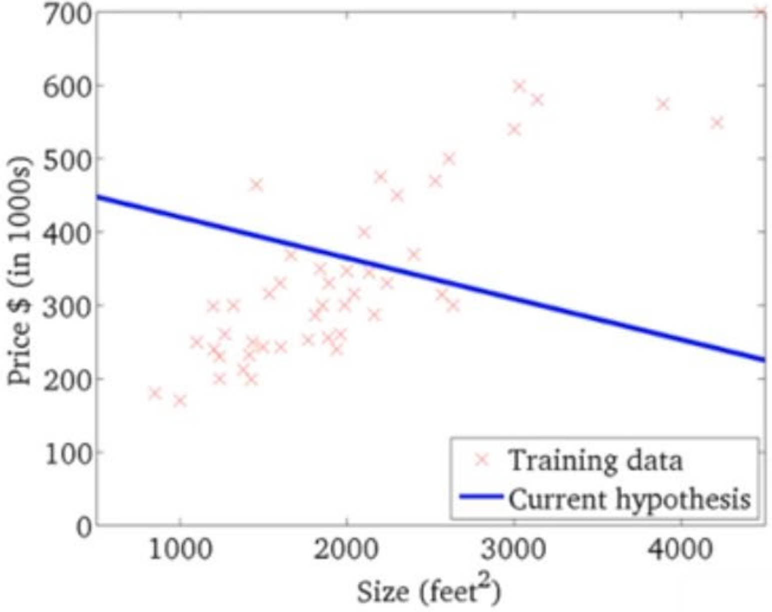
(function of the parameters θ_0, θ_1)



Contour plot for two dimensions (θ_1, θ_2)

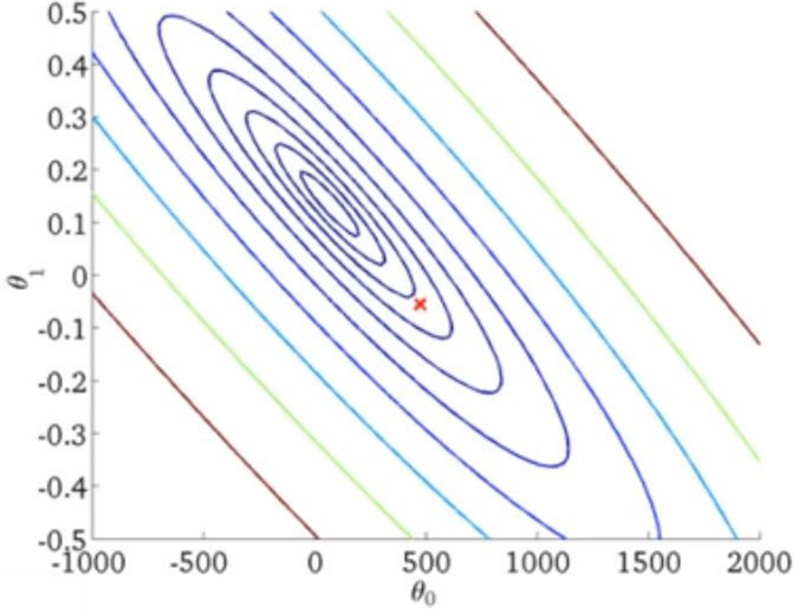
h(x)

(for fixed θ_0, θ_1, this is a function of x)



J(θ_1,θ_2)

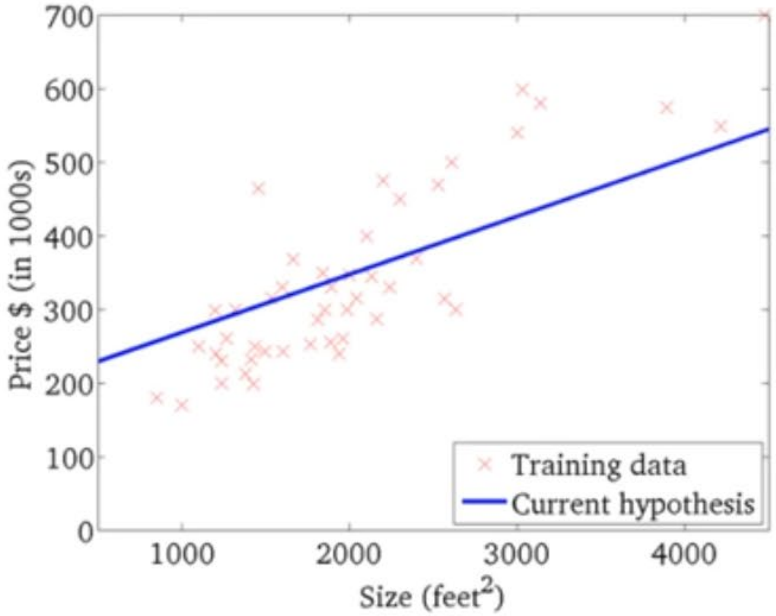
(function of the parameters θ_0, θ_1)



Contour plot for Two Dimensions (θ_1, θ_2)

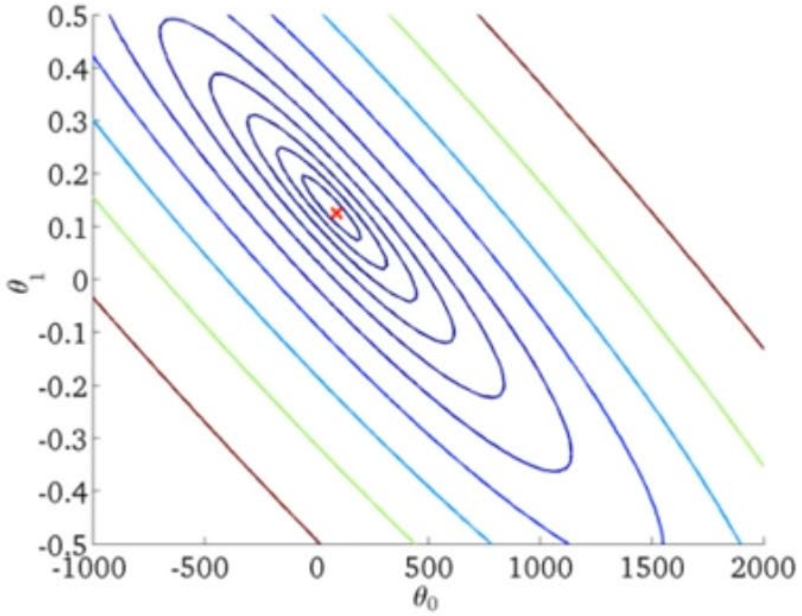
h(x)

(for fixed θ_0, θ_1, this is a function of x)



$J(\theta_{1,}\theta_2)$

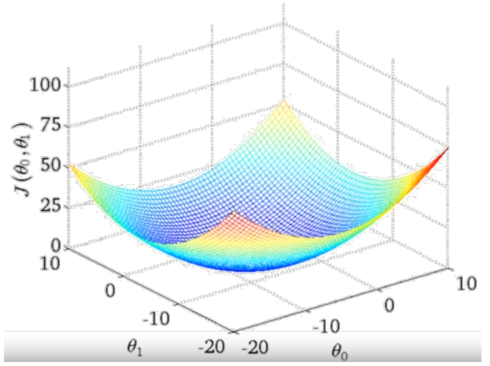
(function of the parameters θ_0, θ_1)



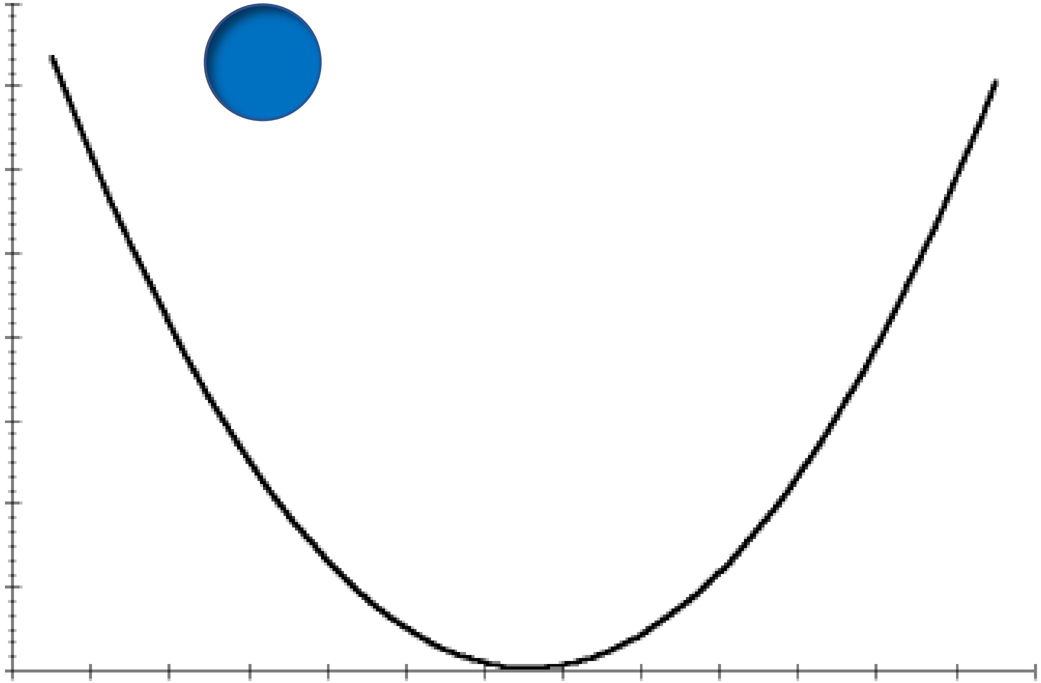
Update θ using gradient descent

- Start with initial guess for θ
- Repeatedly change θ to make J(θ) smaller

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



Intuition



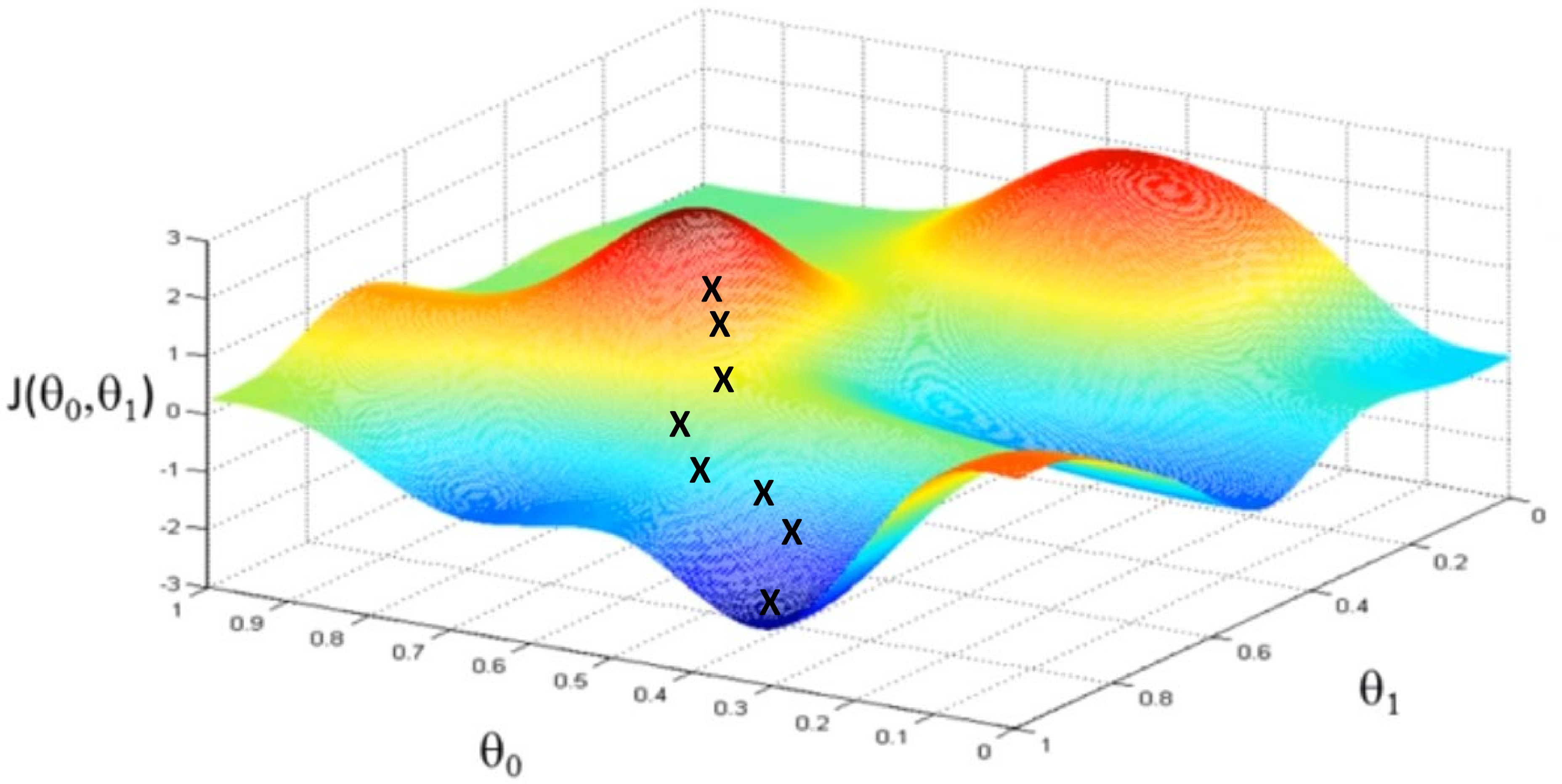
Have some function $J(\theta_0, \theta_1)$

Want $\min\limits_{\theta_0, \theta_1} J(\theta_0, \theta_1)$

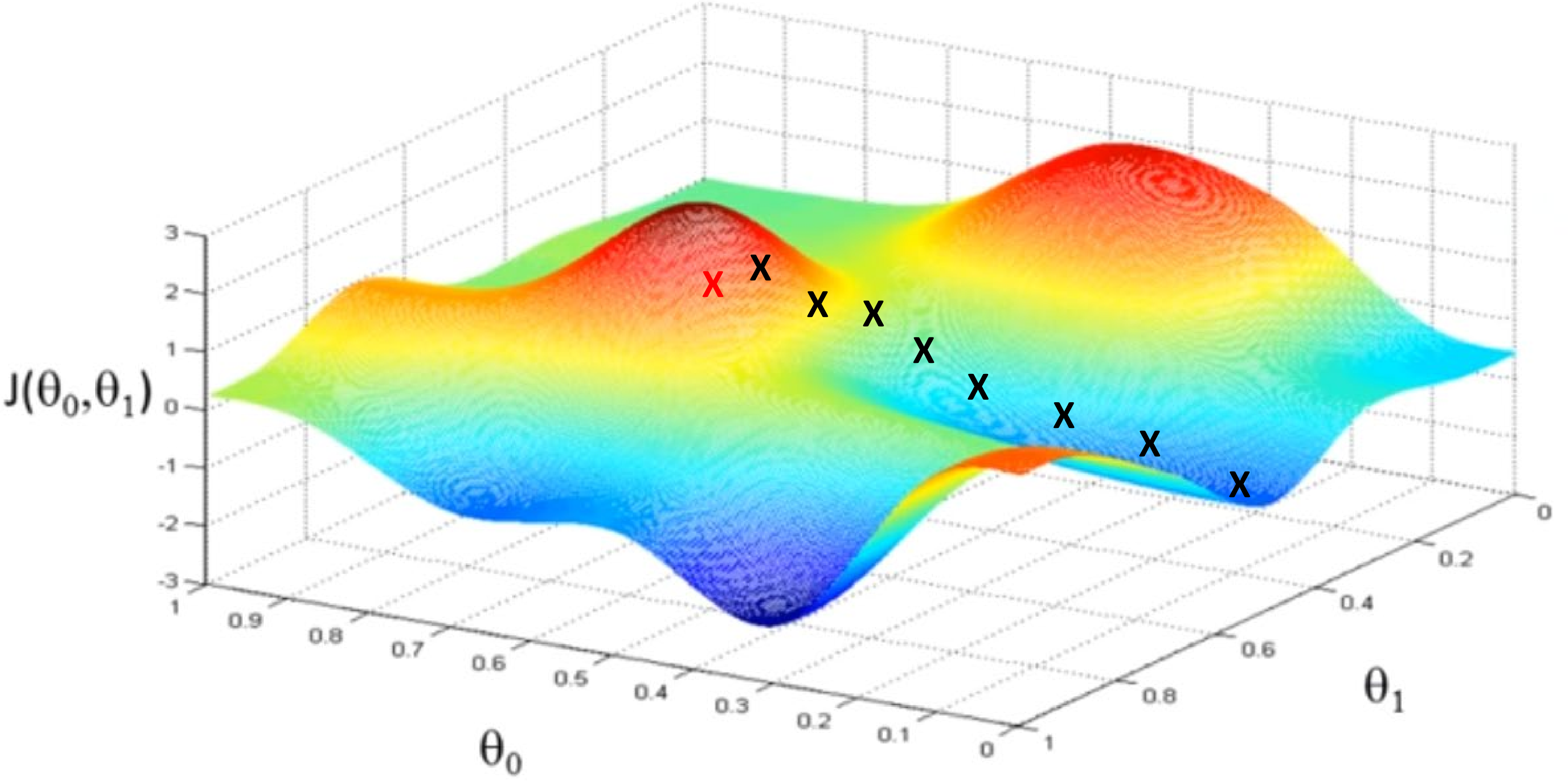
Outline:

- Start with some θ_0, θ_1
- Keep changing θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$

  until we hopefully end up at a minimum



$J(\theta_0,\theta_1)$
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
θ_0
0
0.2
0.4
0.6
0.8
1
θ_1
X
X
X
X
X
X
X
X



$J(\theta_0, \theta_1)$
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
θ_0
0
0.2
0.4
0.6
0.8
1
θ_1

Gradient descent

- Repeat until convergence

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

- Update j=0 and j=1 simultaneously

Gradient descent

- Repeat until convergence

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

- Update j=0 and j=1 simultaneously

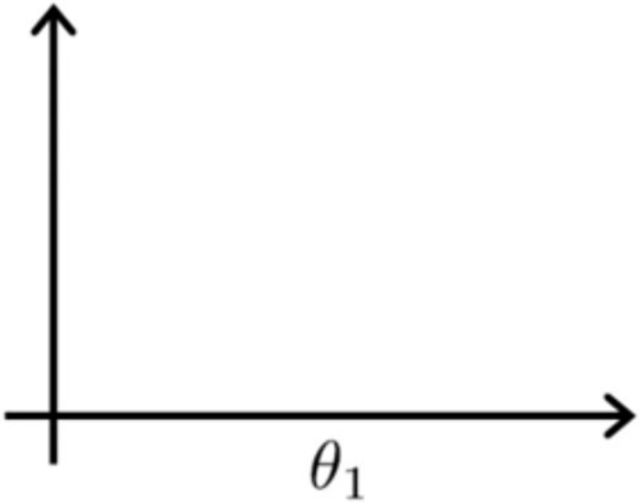
Gradient descent

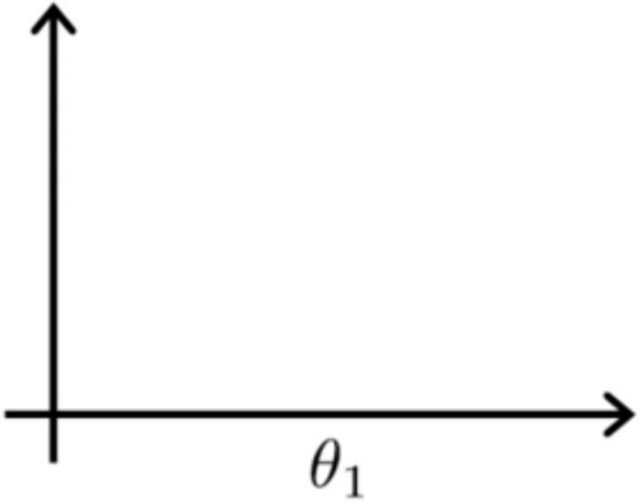
Gradient descent

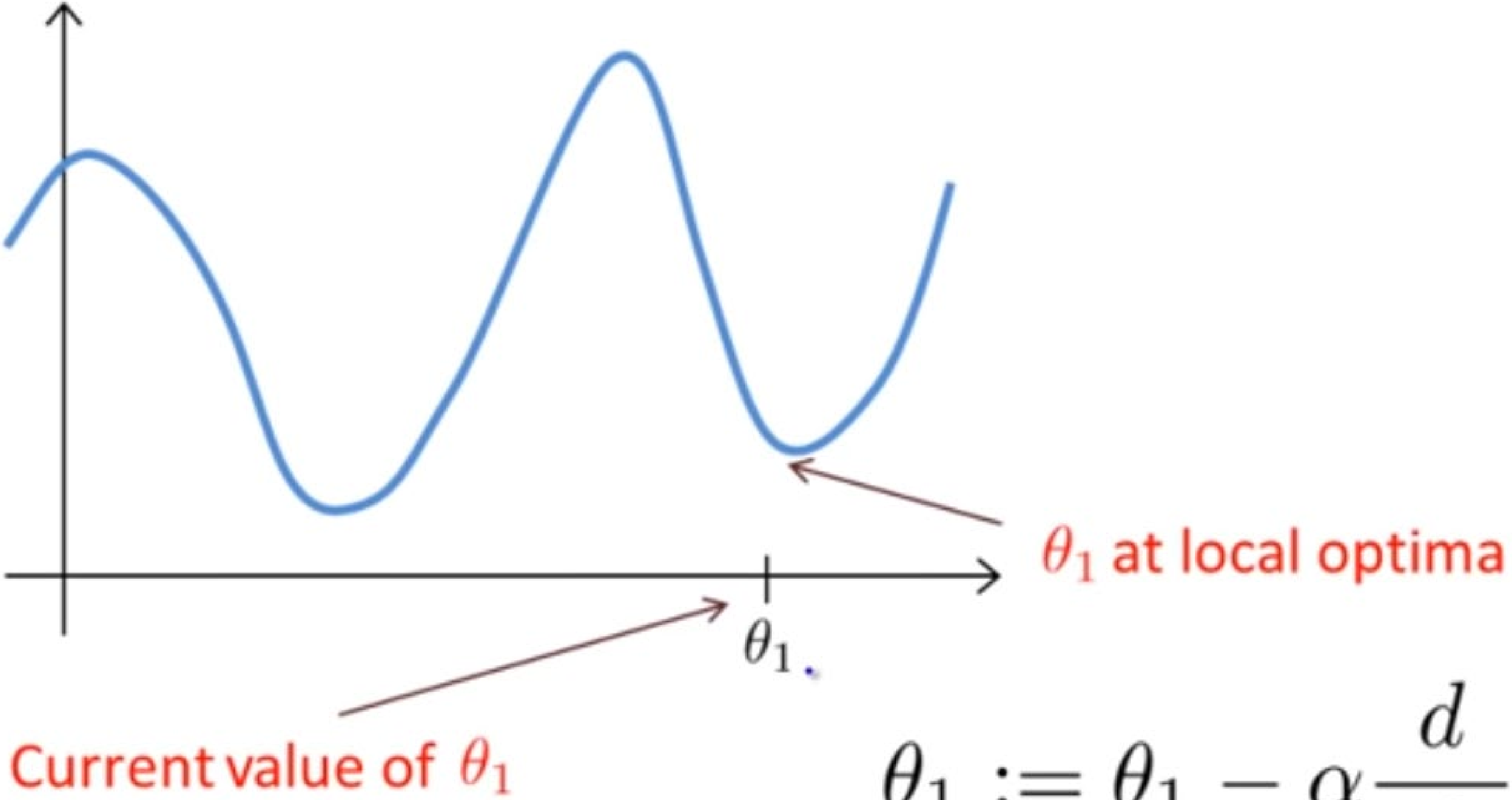
Gradient descent

Learning rate

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

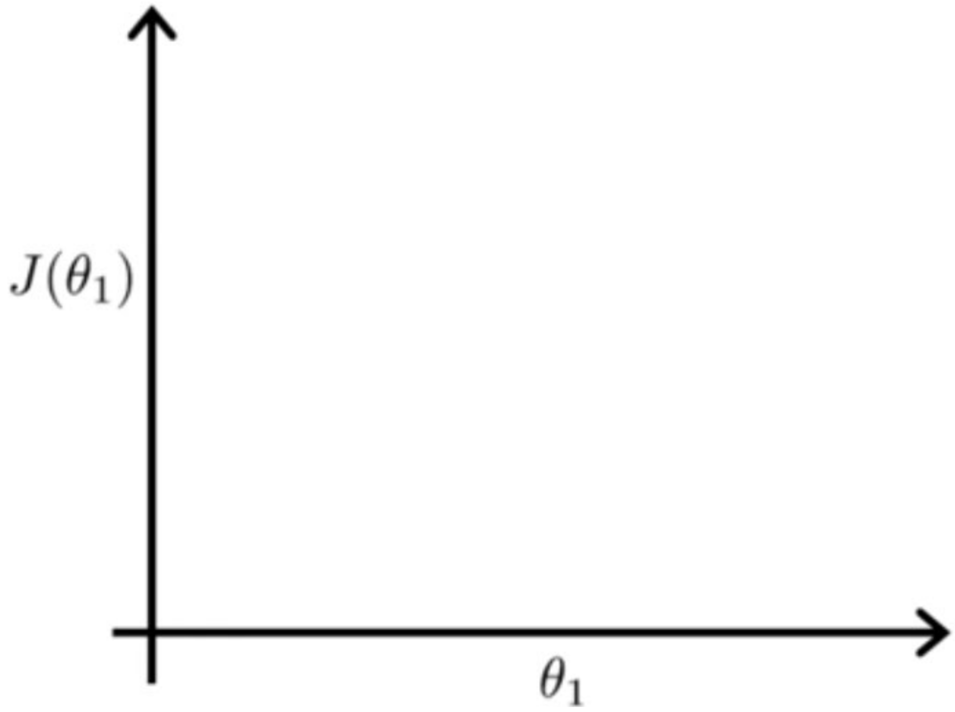






$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

Convergence



Compute the partial derivative

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

Compute the partial derivative

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

Linear regression algorithm

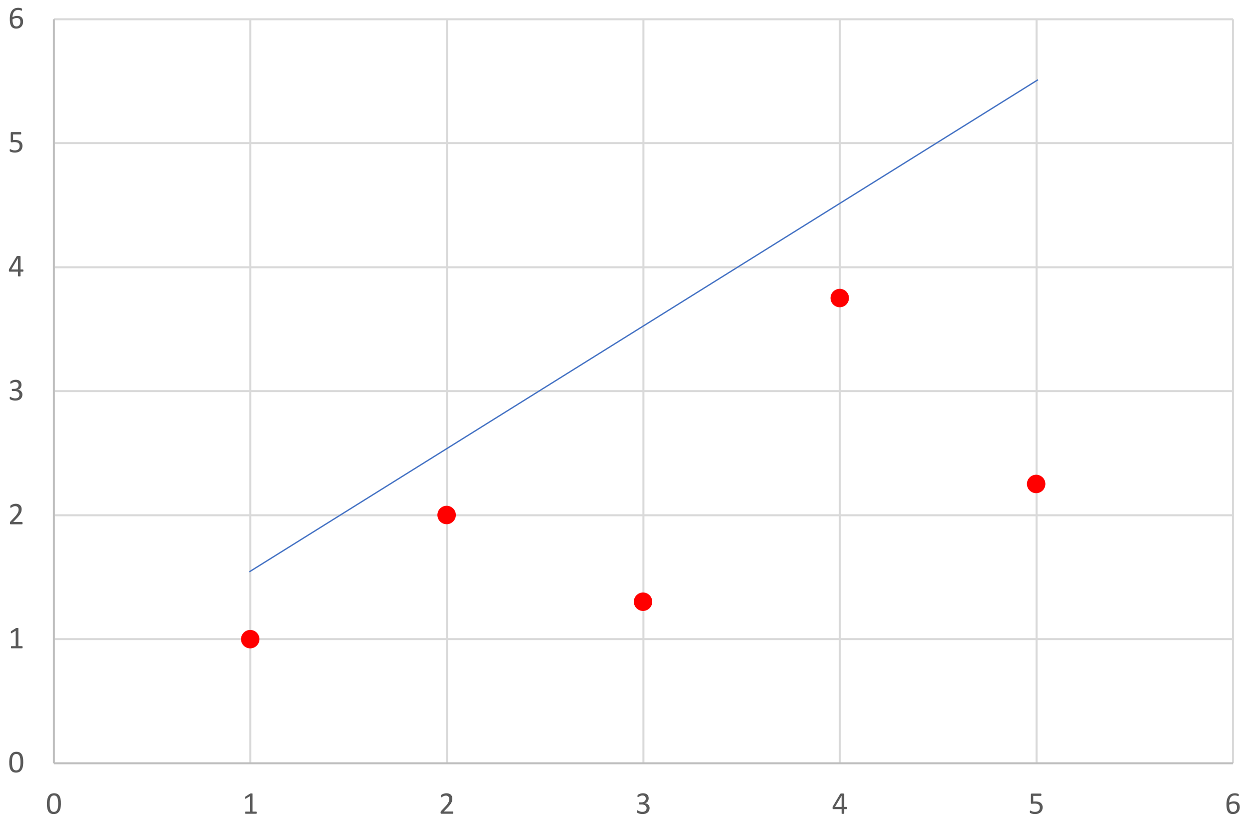
Repeat until convergence {

$$\theta_j = \theta_j + \alpha \sum_{i=1}^{n} (yi - h(xi))\, x_j^i \qquad \text{(for every } j\text{)}$$

}

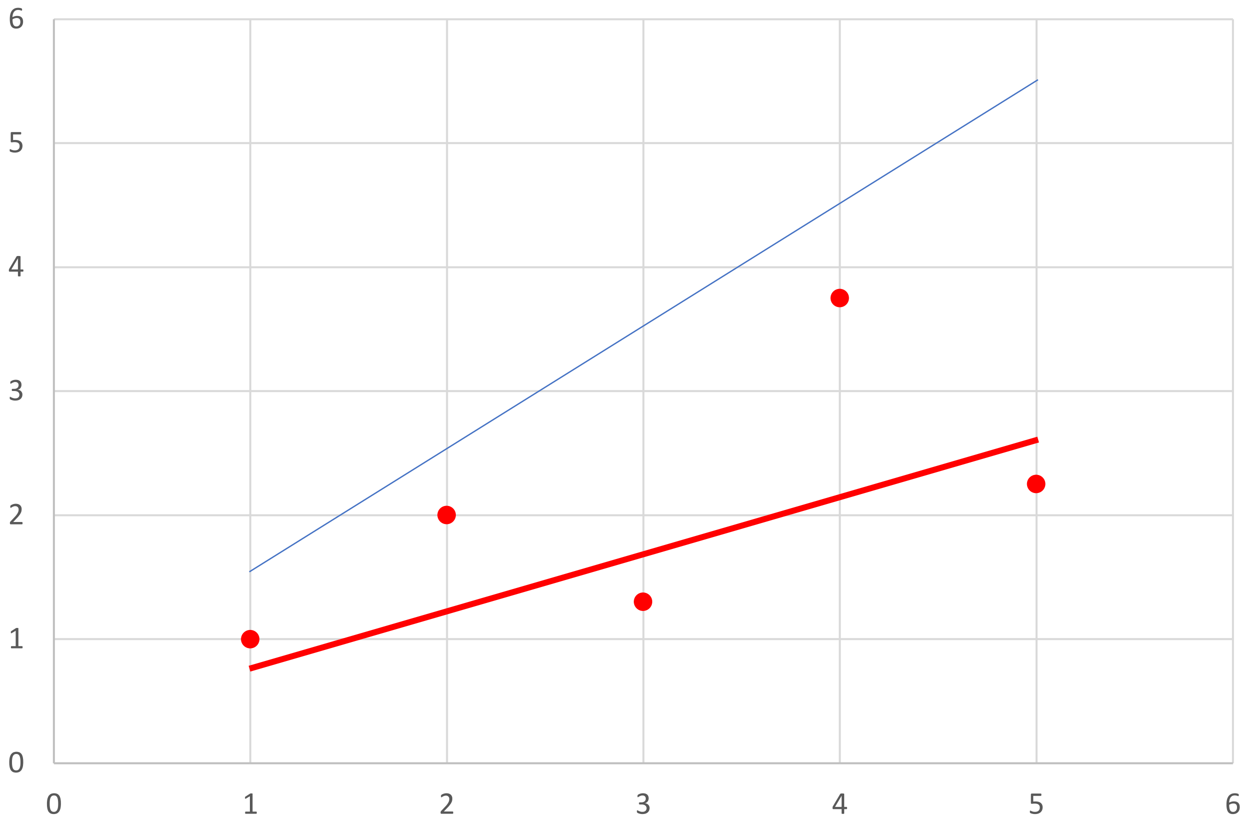
X	Y
1.00	1.00
2.00	2.00
3.00	1.30
4.00	3.75
5.00	2.25

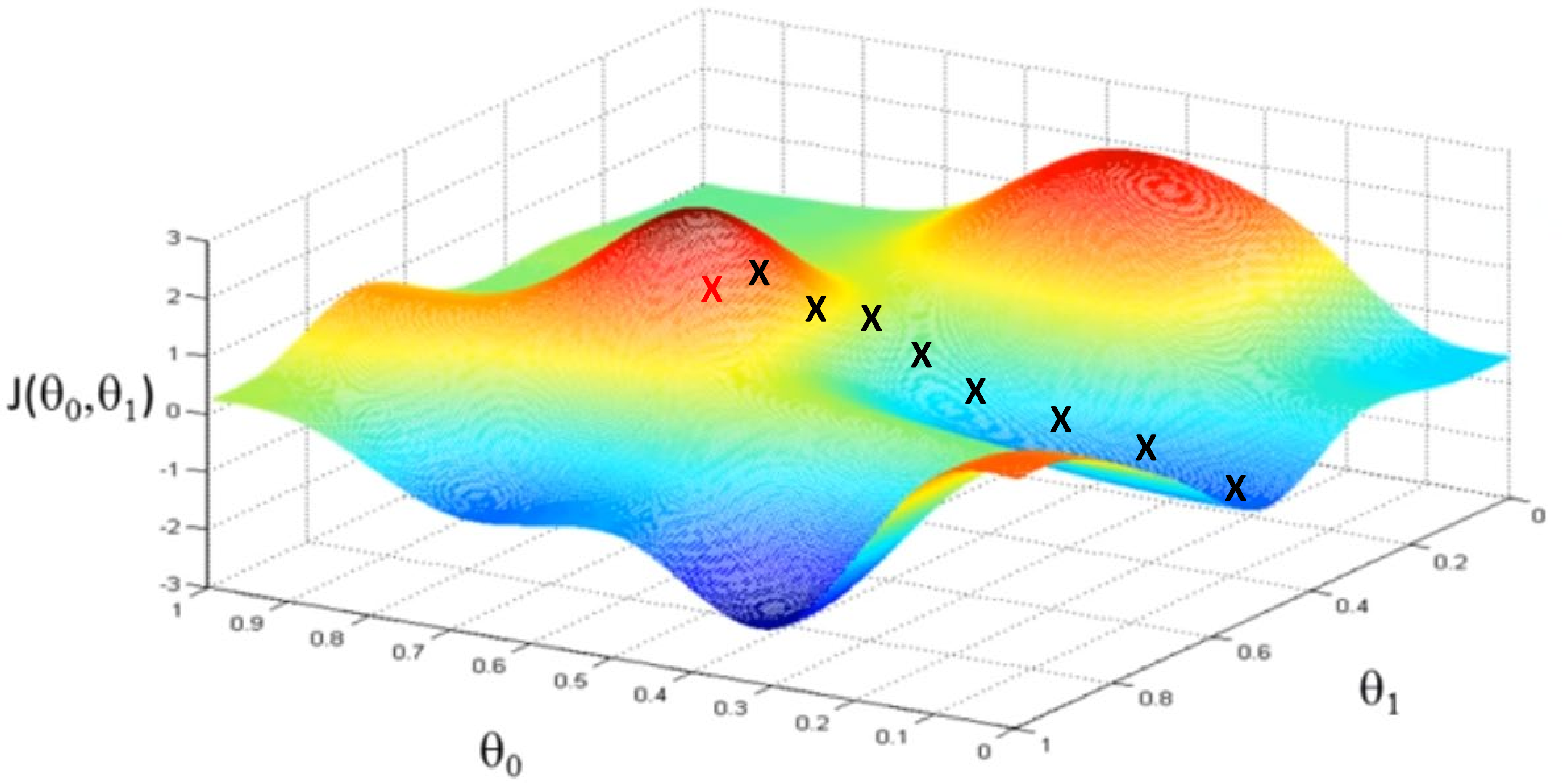
$\theta_1 = 0.50$
$\theta_2 = 0.01$
$\alpha = 0.01$



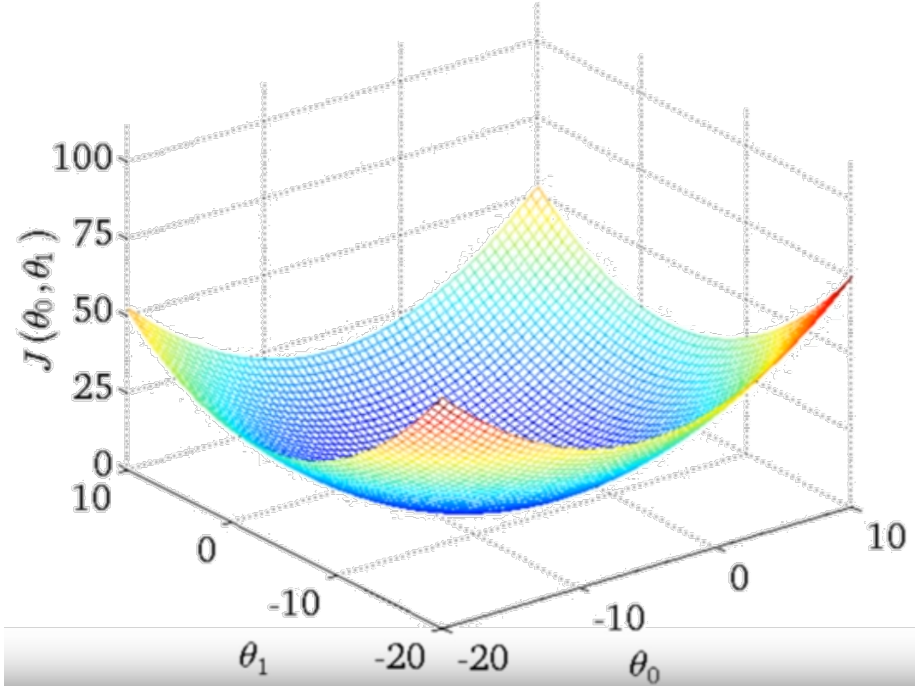
X	Y
1.00	1.00
2.00	2.00
3.00	1.30
4.00	3.75
5.00	2.25

$\theta_1 = 0.50$
$\theta_2 = 0.01$
$\alpha = 0.1$





$J(\theta_0,\theta_1)$
θ_0
θ_1
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
X
X
X
X
X
X
X
X
X



100
75
50
25
0
$J(\theta_0, \theta_1)$
10
0
-10
-20
θ_1
-20
-10
0
10
θ_0

Batch gradient descent

Linear regression with more features

Size ($feet^2$)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178

Linear regression with more features

Matrix:

$$X = \begin{bmatrix} 2104 & 5 & 1 & 45 \\ 1416 & 3 & 2 & 40 \\ 1534 & 3 & 2 & 30 \\ 852 & 2 & 1 & 36 \end{bmatrix}$$

Vector:

$$y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 172 \end{bmatrix}$$

Let's try this out