Linear regression with one variable

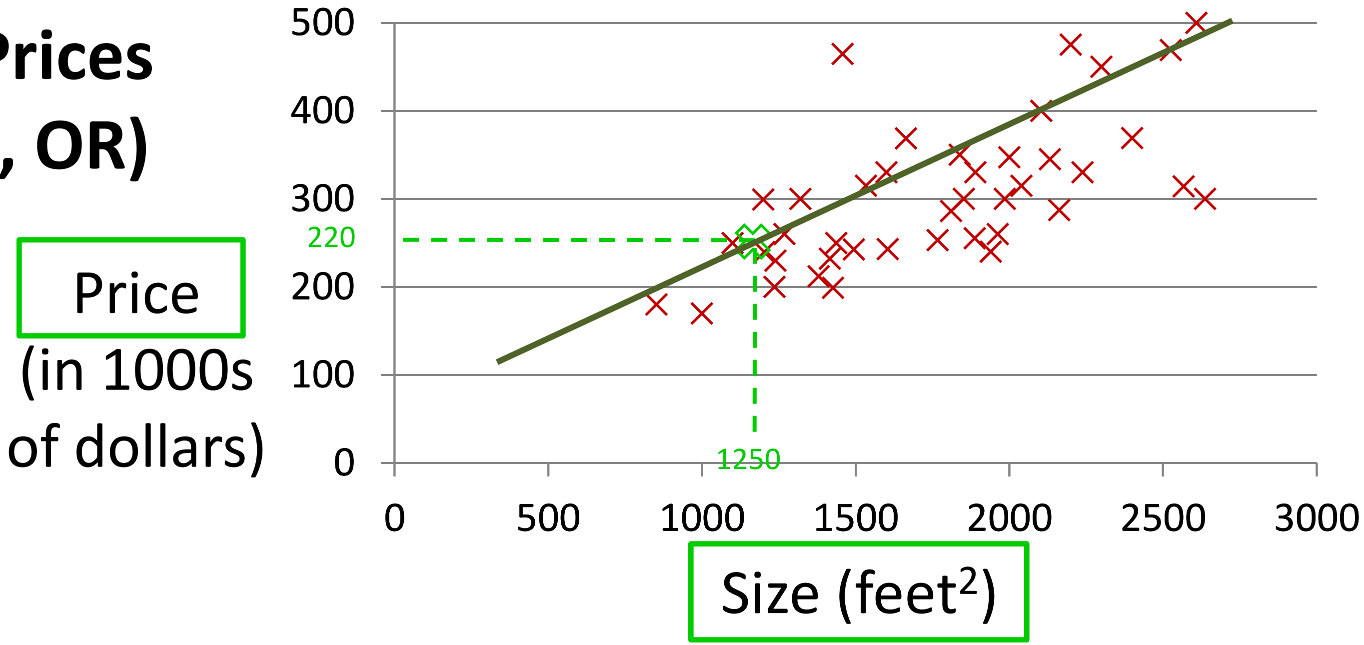
Model representation

Fundamentals of Machine Learning

Model intuition

- Products – Features (Input)
- Prices – model parameters
- Total cost - Output
- Linear dependency between products and the total cost.

Housing Prices (Portland, OR)



Supervised Learning

Given the "right answer" for each example in the data.

Regression Problem

Predict real-valued output

Classification Problem

Predict discrete-valued output

Training set of housing prices (Portland, OR)

Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

Notation:

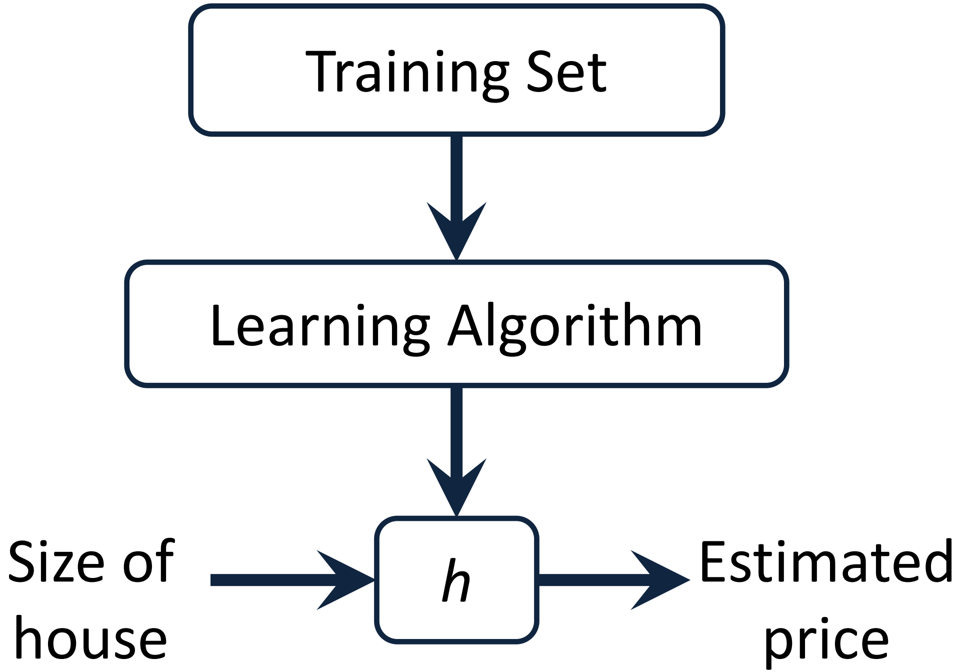
m = Number of training examples
x's = "input" variable / features
y's = "output" variable / "target" variable

$x^{(1)}=2104$
$x^{(2)}=1416$
$y^{(1)}=460$

(x,y) – one training example

(x^i,y^i) – i^{th} training example

Andrew Ng



x

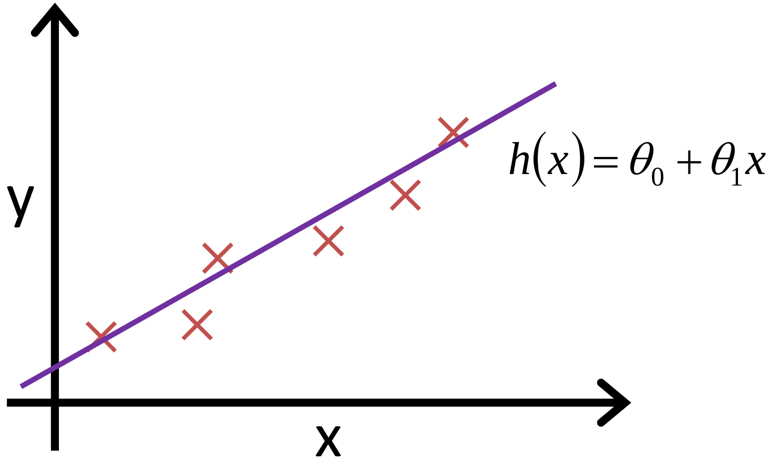
hypothesis

Estimated value of y

h maps from x's to y's

How do we represent *h* ?

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



Linear regression with one variable.
Univariate linear regression.

Linear regression with one variable

Cost function

Fundamentals of Machine Learning

Training Set

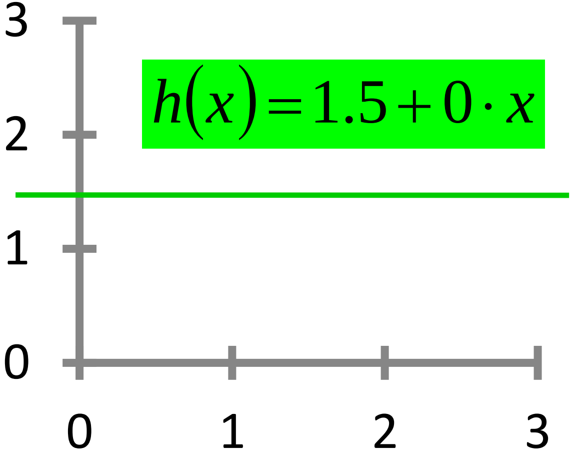
Size in feet^2 (x)	**Price ($) in 1000's (y)**
2104	460
1416	232
1534	315
852	178
…	…

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

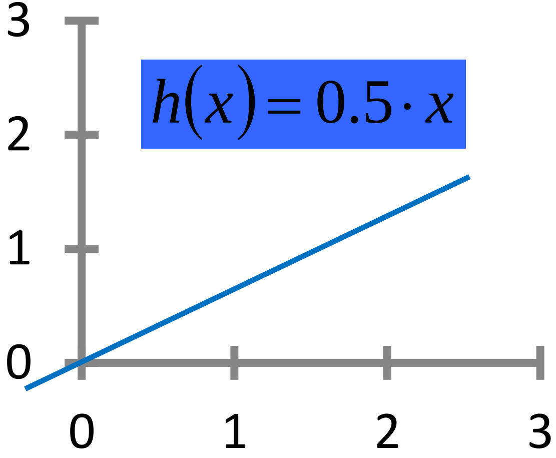
θ_i's: Parameters

How to choose θ_i's ?

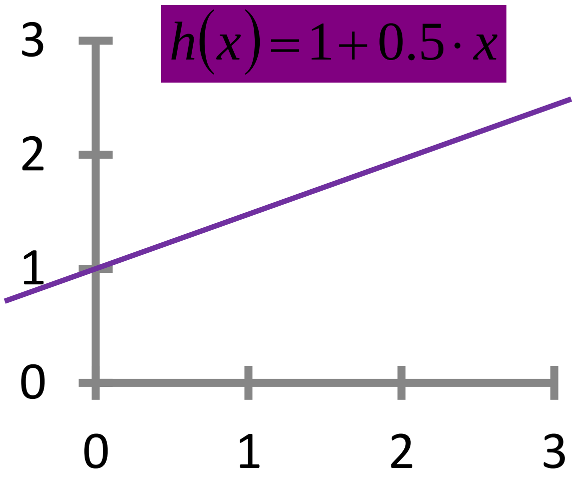
$$h_\theta(x) = \theta_0 + \theta_1 x$$



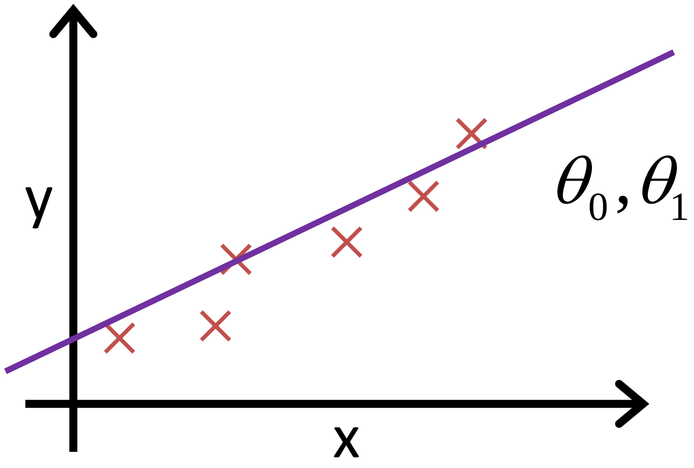
$$\theta_0 = 1.5$$
$$\theta_1 = 0$$



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



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



$$\underset{\theta_0 \ \theta_1}{\text{minimize}} \ \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)^2$$

m - # of training examples

$$h_\theta\left(x^{(i)}\right) = \theta_0 + \theta_1 x^{(i)}$$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)^2$$

Cost function
or
Squared error function

Idea: Choose θ_0, θ_1 so that
$h_\theta(x)$ is close to y for our training examples (x, y)

Linear regression with one variable

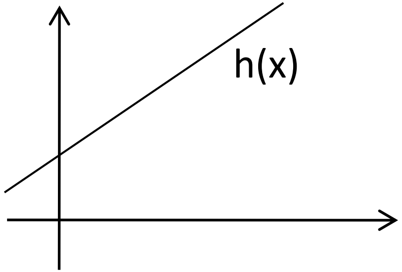
Cost function intuition I

Fundamentals of Machine Learning

Simplified

Hypothesis:

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

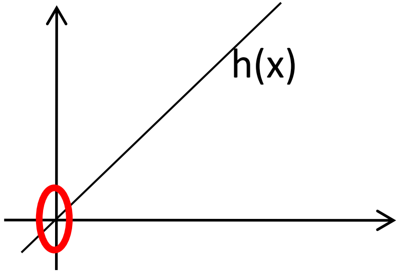


$$h_\theta(x) = \theta_1 x \qquad \theta_0 = 0$$

Parameters:

$$\theta_0, \theta_1$$

$$\theta_1$$



Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

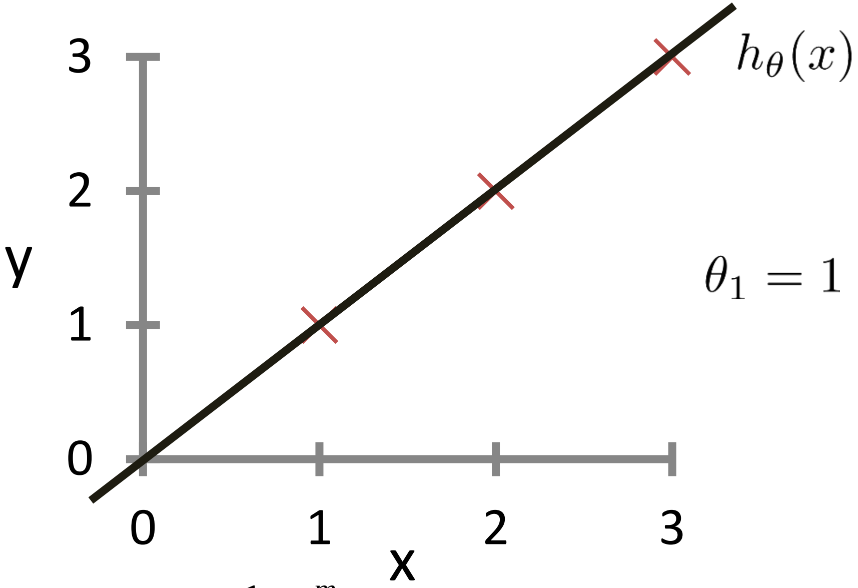
$$J(\theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

$$\underset{\theta_1}{\text{minimize}}\ J(\theta_1)$$

$h_\theta(x)$

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

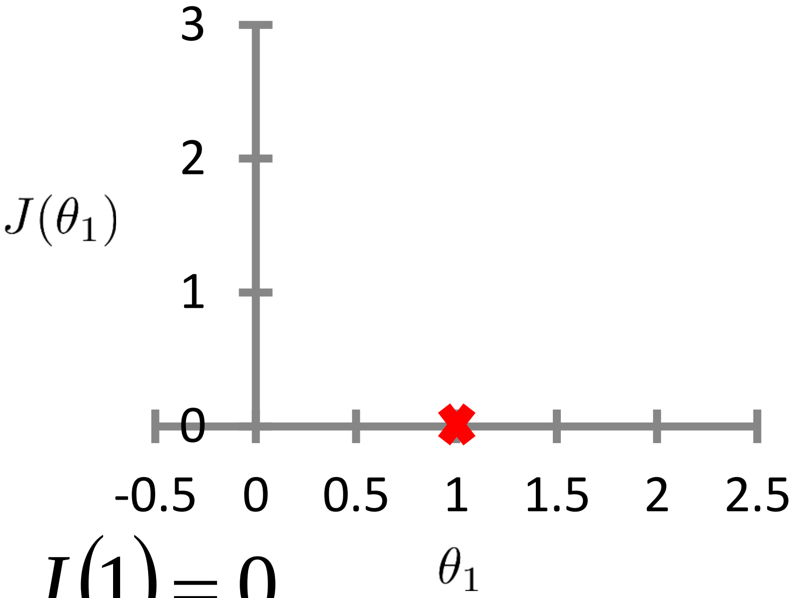


$$J(\theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

$$= \frac{1}{2m}\sum_{i=1}^{m}\left(\theta_1 x^{(i)} - y^{(i)}\right)^2 = \frac{1}{2m}\cdot 0 = 0$$

$J(\theta_1)$

(function of the parameter θ_1)

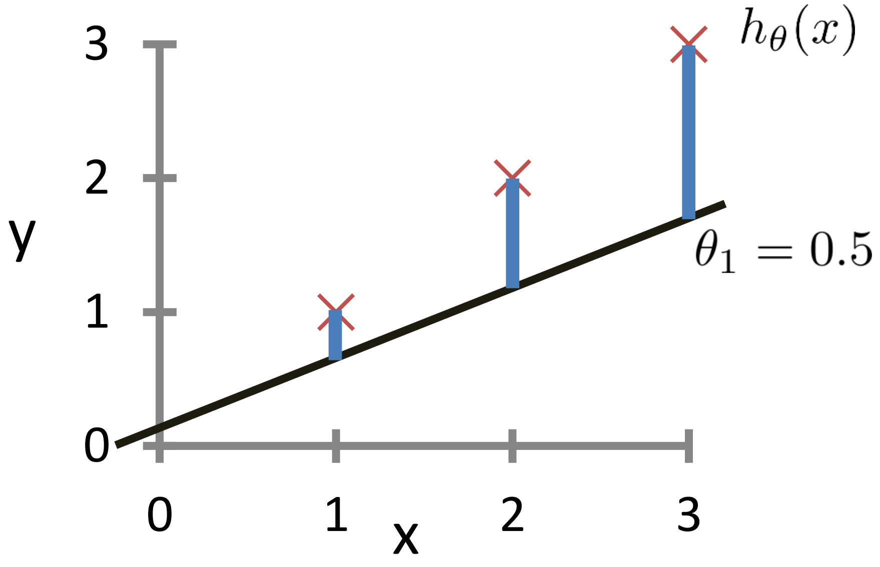


$$J(1) = 0$$

$\theta_1 = 0.5$?

$h_\theta(x)$

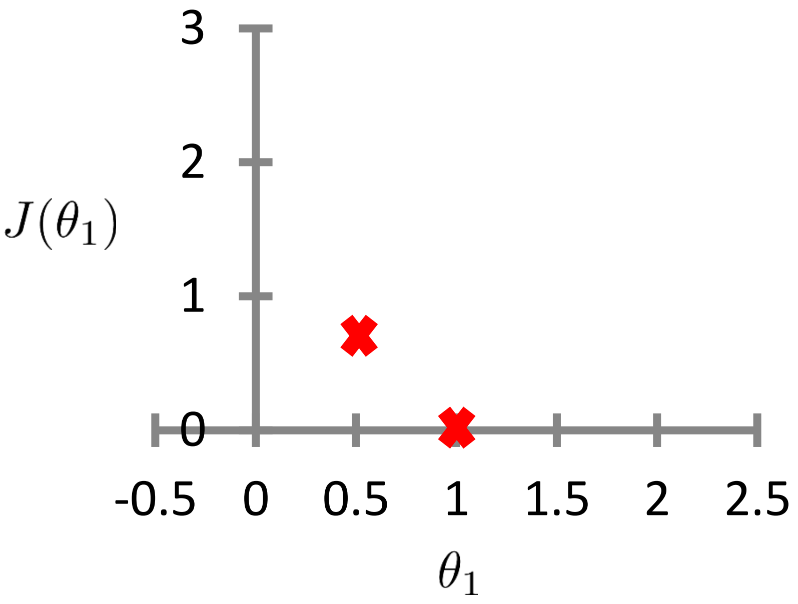
(for fixed θ_1, this is a function of x)



$$J(0.5) = \frac{1}{2m}\left[(0.5-1)^2 + (1-2)^2 + (1.5-3)^2\right] \approx 0.58$$

$J(\theta_1)$

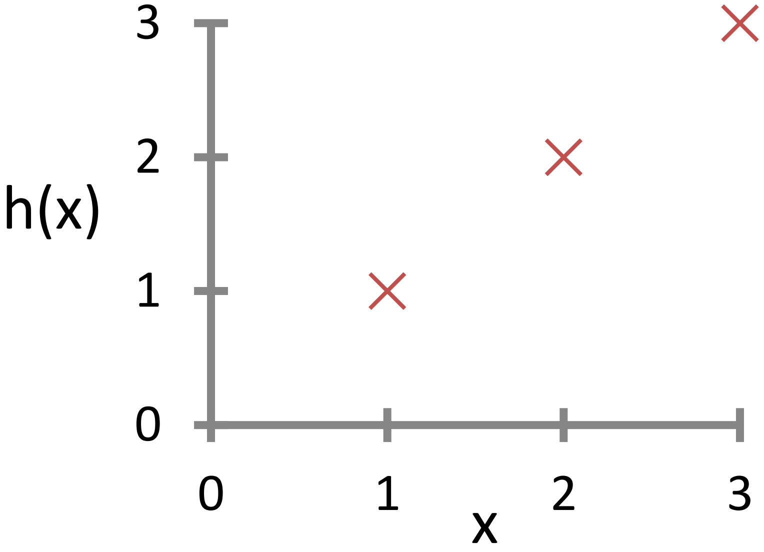
(function of the parameter θ_1)



θ_1=0 ?

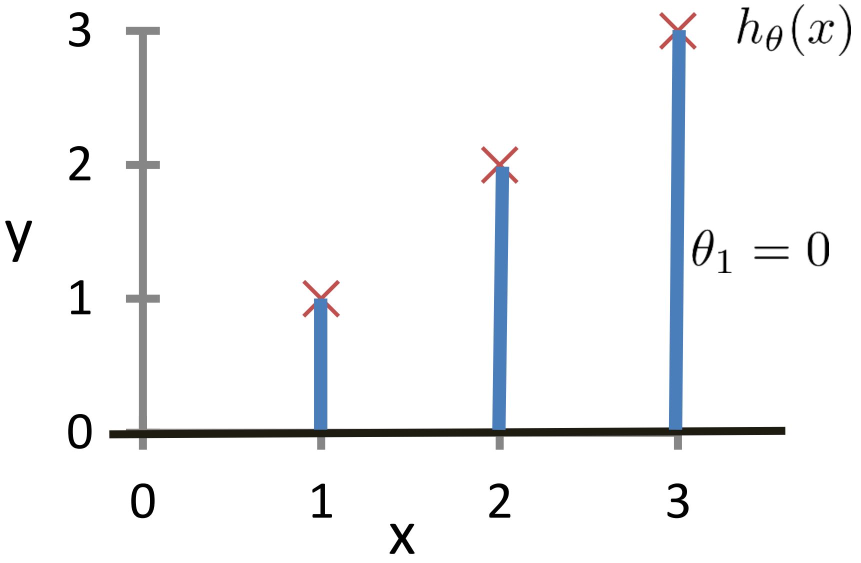
Suppose we have a training set with m=3 examples, plotted below. Our hypothesis representation is $h_\theta(x)=\theta_1 x$, with parameter θ_1. The cost function $J(\theta_1)$ is $J(\theta_1)=\frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{i}\right)^2$ What is $J(0)$?

- 0
- 1/6
- 1
- 14/6



$h_\theta(x)$

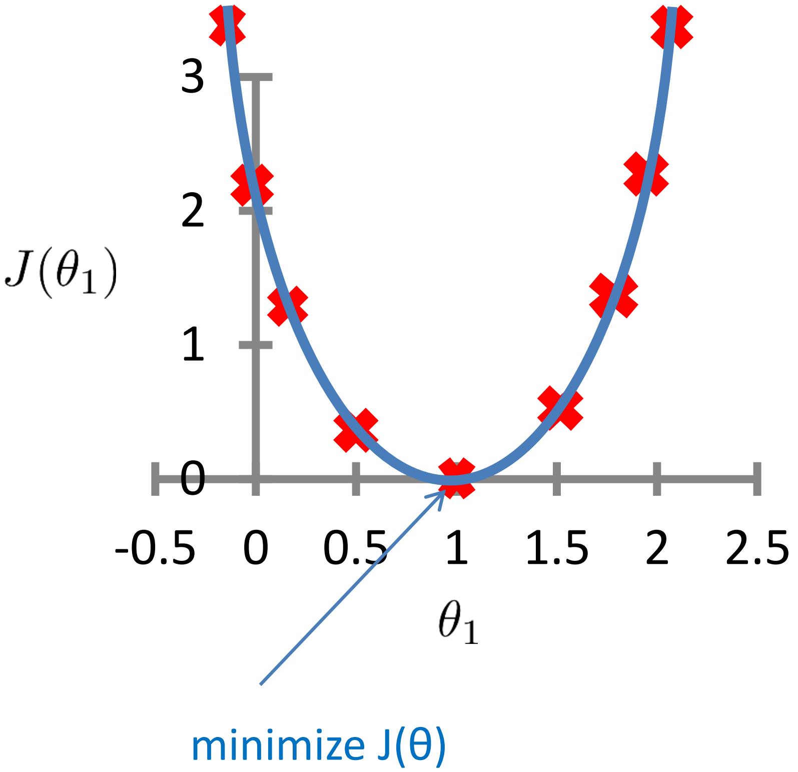
(for fixed θ_1, this is a function of x)



$$J(0) = \frac{1}{2m}\left[(0-1)^2 + (0-2)^2 + (0-3)^2\right] \approx 2.3$$

$J(\theta_1)$

(function of the parameter θ_1)



Andrew Ng

Linear regression with one variable

Cost function intuition II

Fundamentals of Machine Learning

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

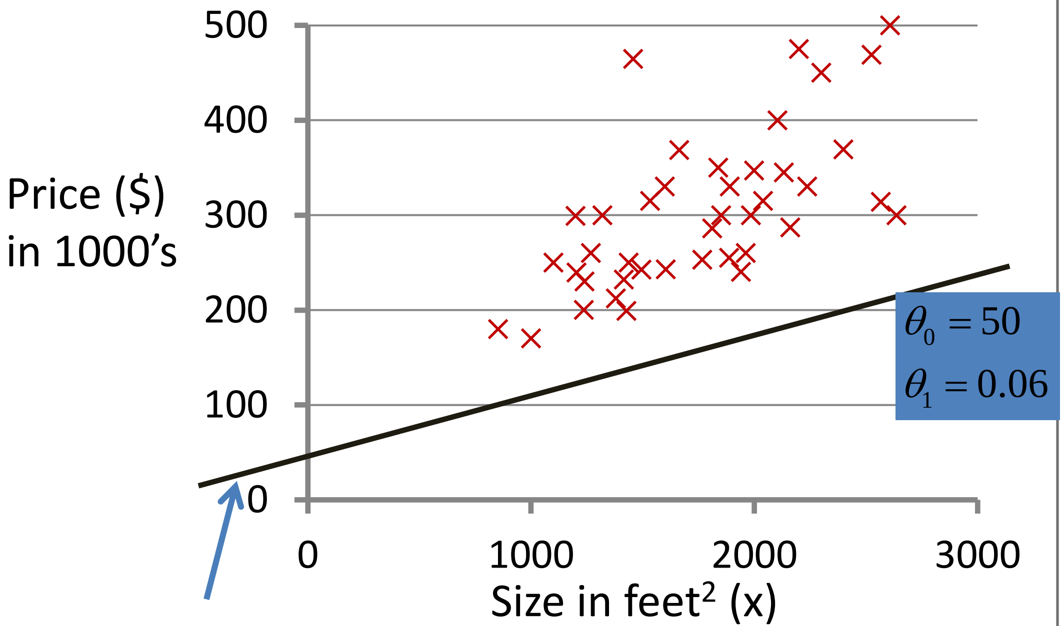
Parameters: θ_0, θ_1

Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

$h_\theta(x)$

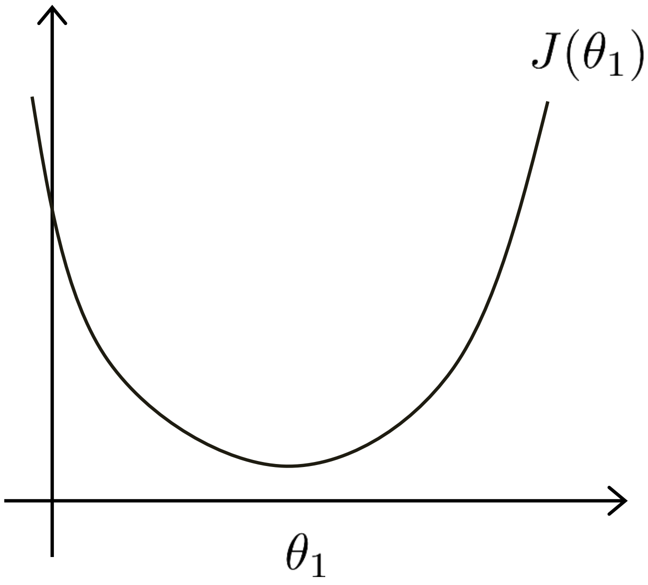
(for fixed θ_0, θ_1, this is a function of x)



$$h_\theta(x) = 50 + 0.06x$$

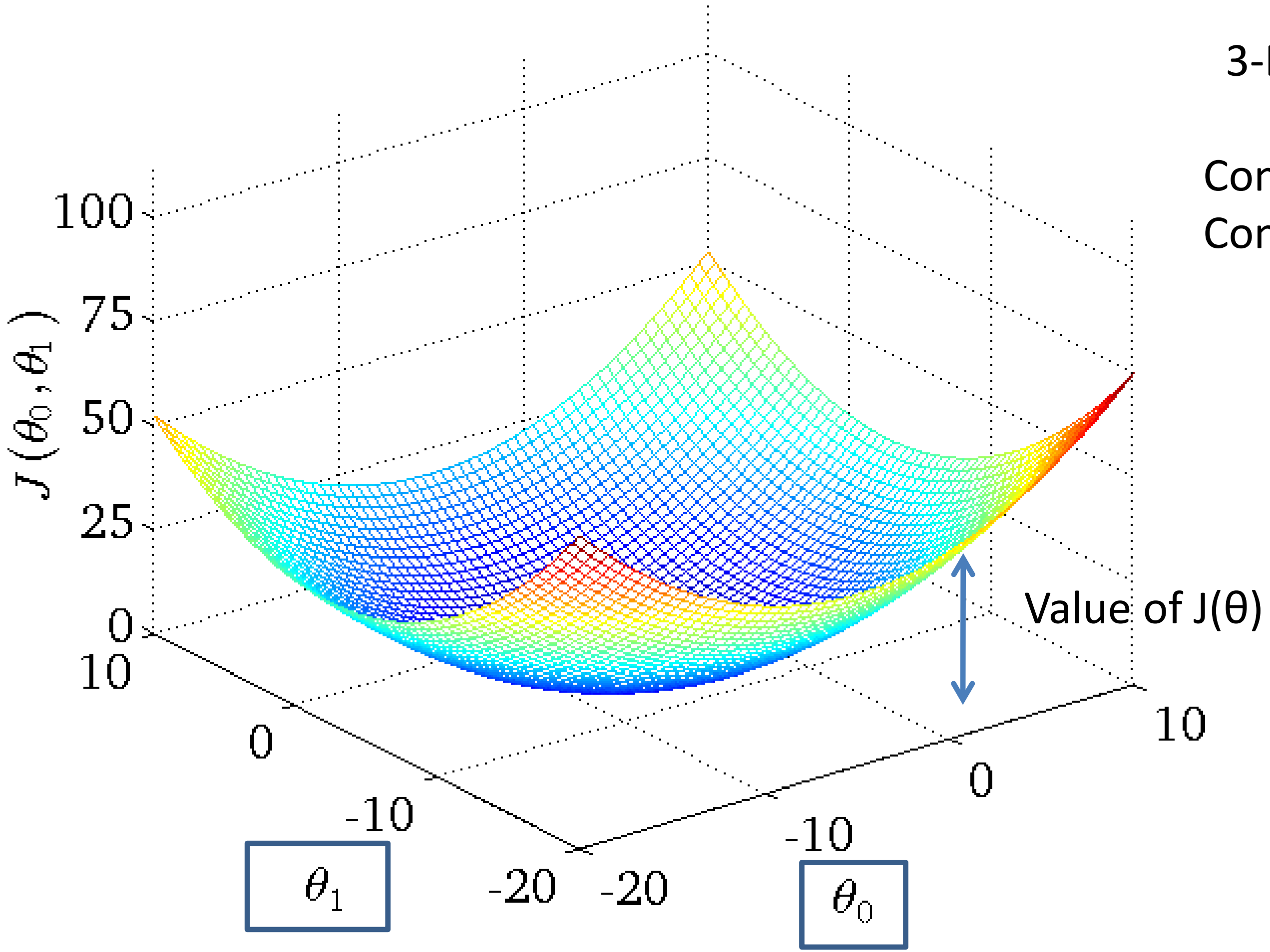
$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



cost function depends of one parameter

Andrew Ng

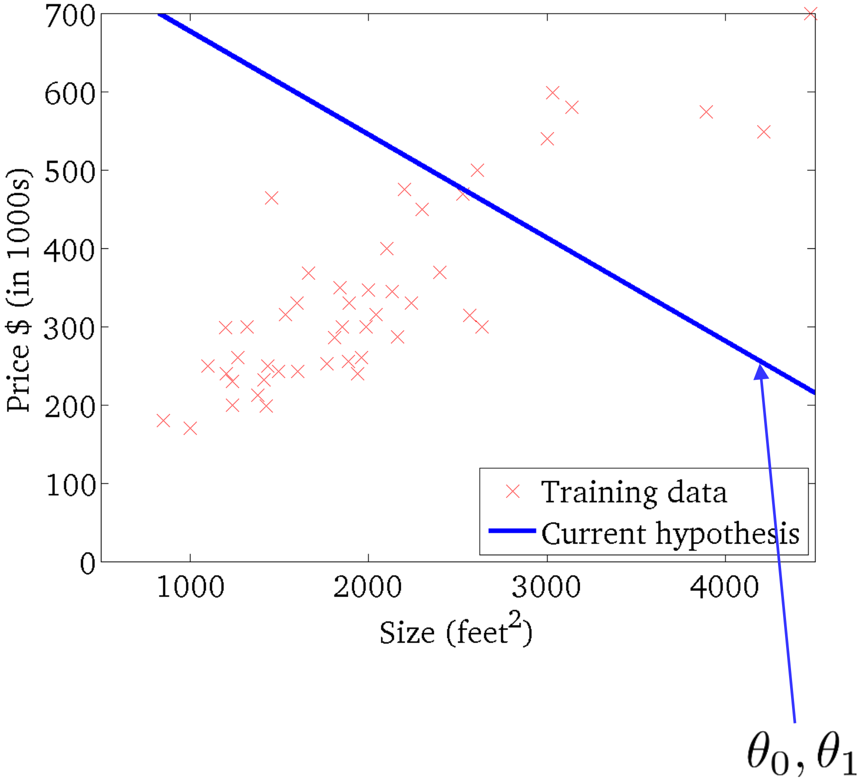


3-D surface plot

Contour plots
Contour figures

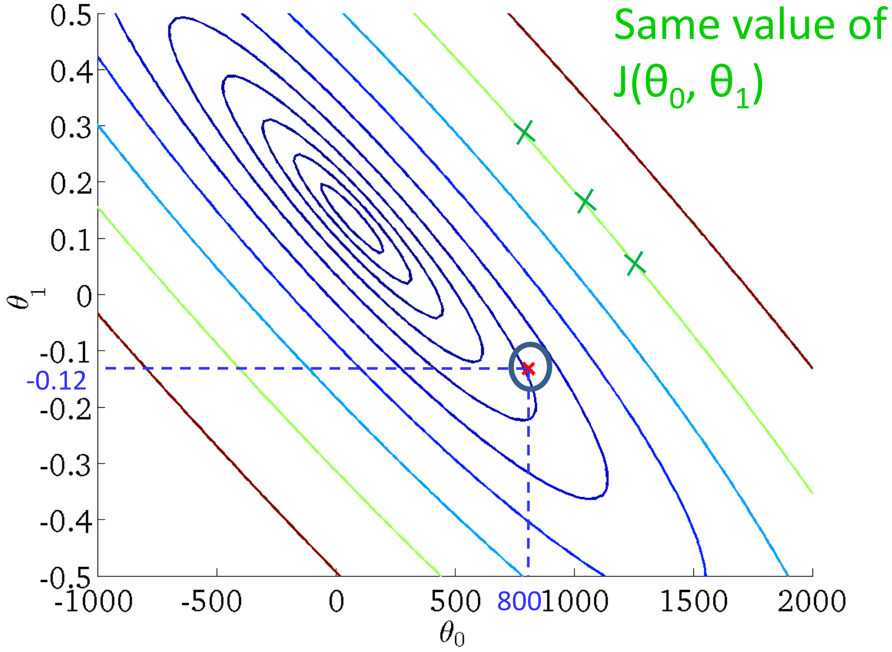
$h_\theta(x)$

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



$J(\theta_0, \theta_1)$

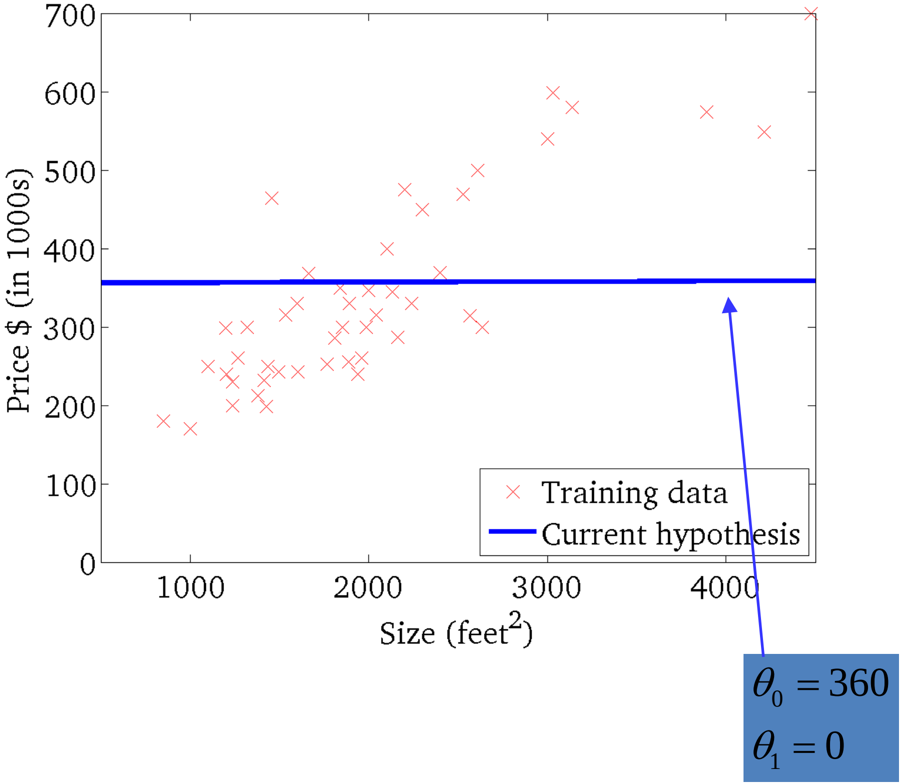
(function of the parameters θ_0, θ_1)



Andrew Ng

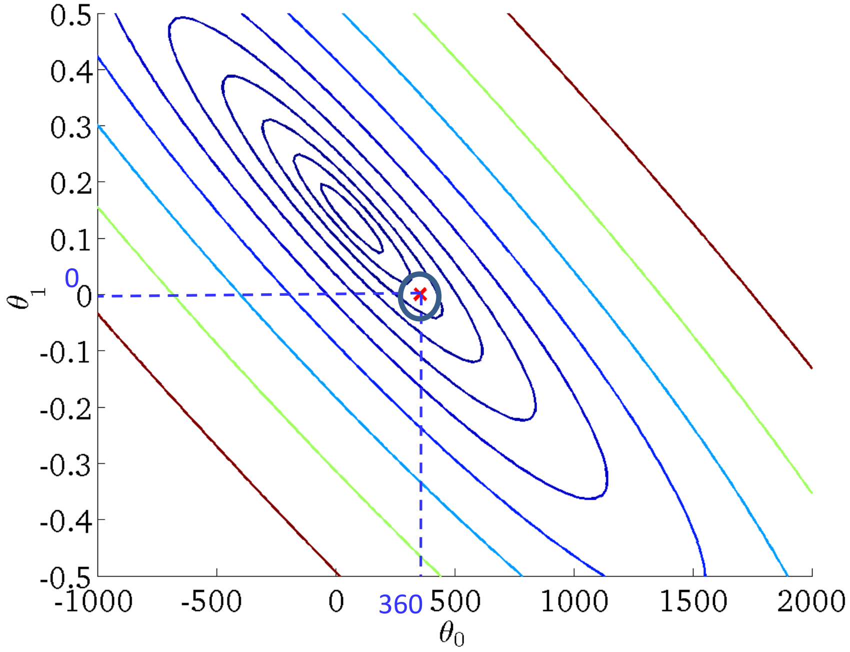
$$h_\theta(x)$$

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



$$J(\theta_0, \theta_1)$$

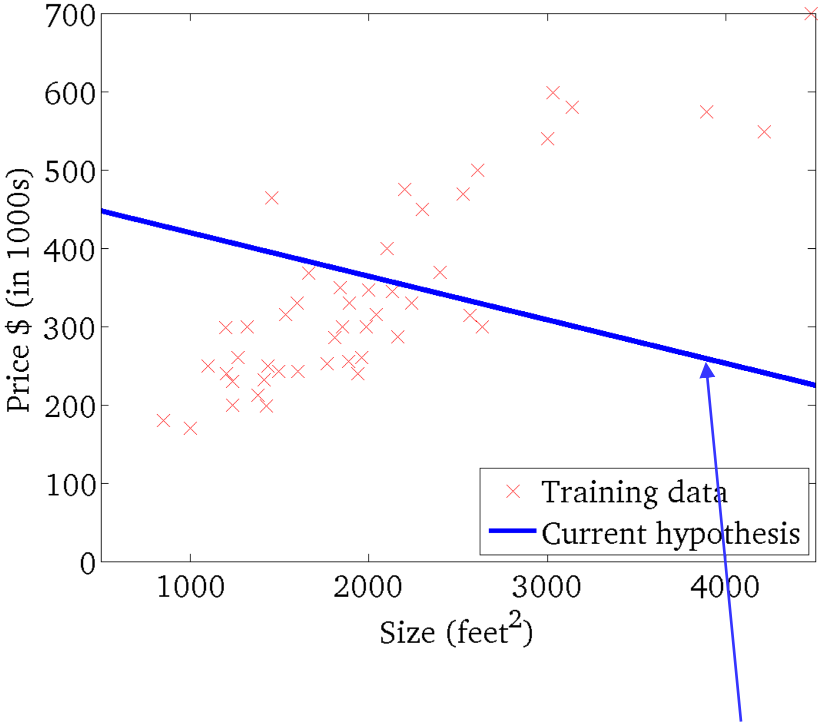
(function of the parameters θ_0, θ_1)



Andrew Ng

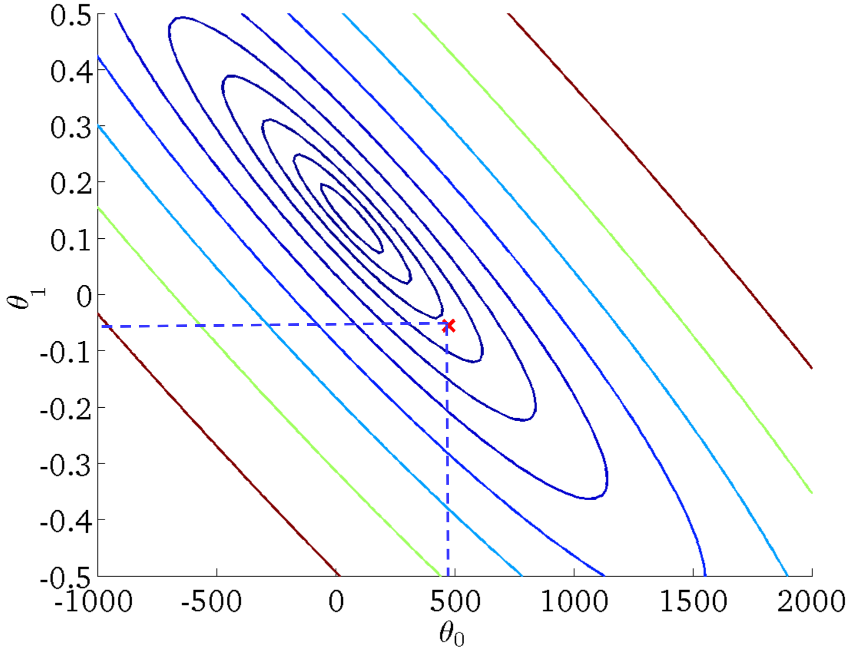
$h_\theta(x)$

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



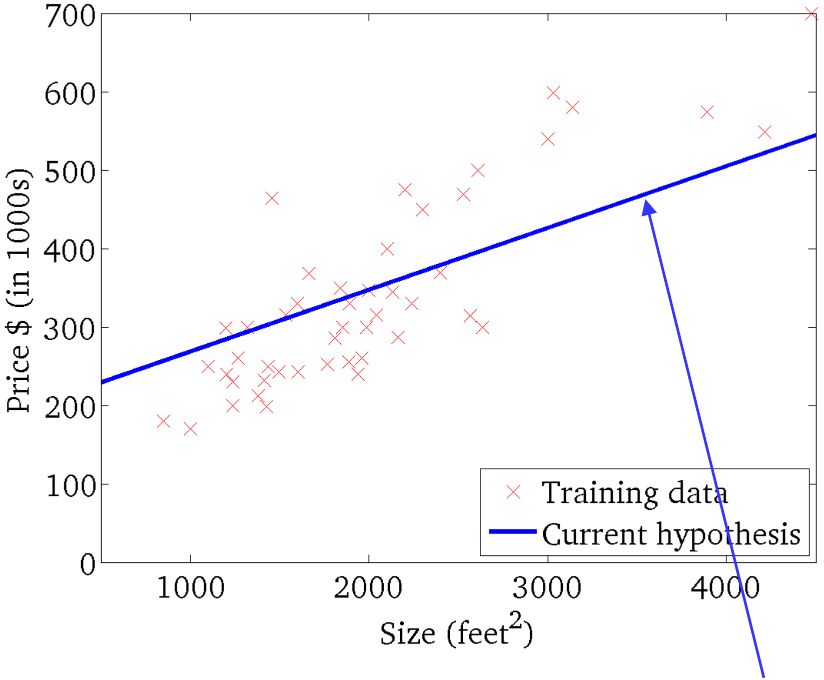
$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



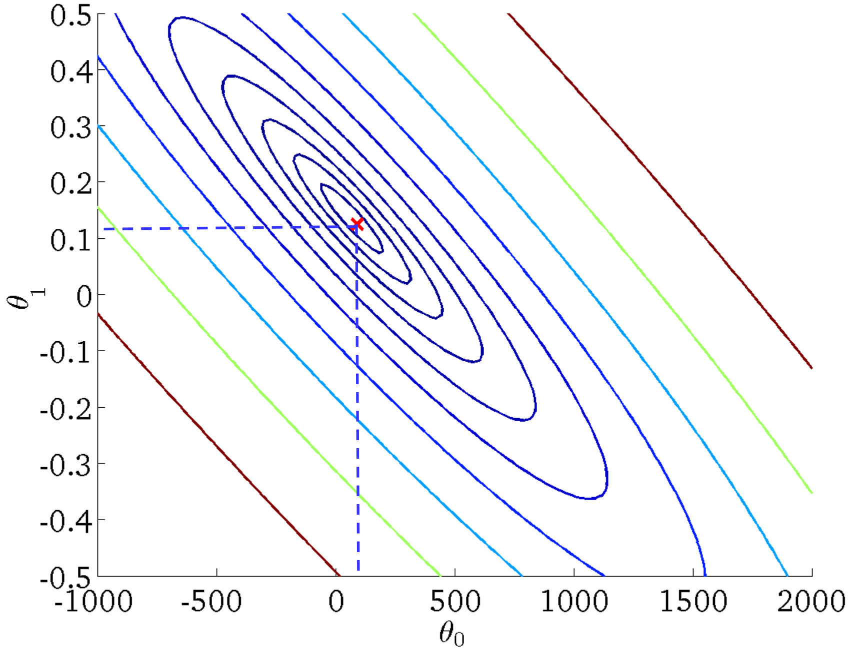
$h_\theta(x)$

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



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



pretty close to the minimum

Linear regression with one variable

Gradient descent

Fundamentals of Machine Learning

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

$J(\theta_0, \theta_1, \theta_2, \ldots, \theta_n)$

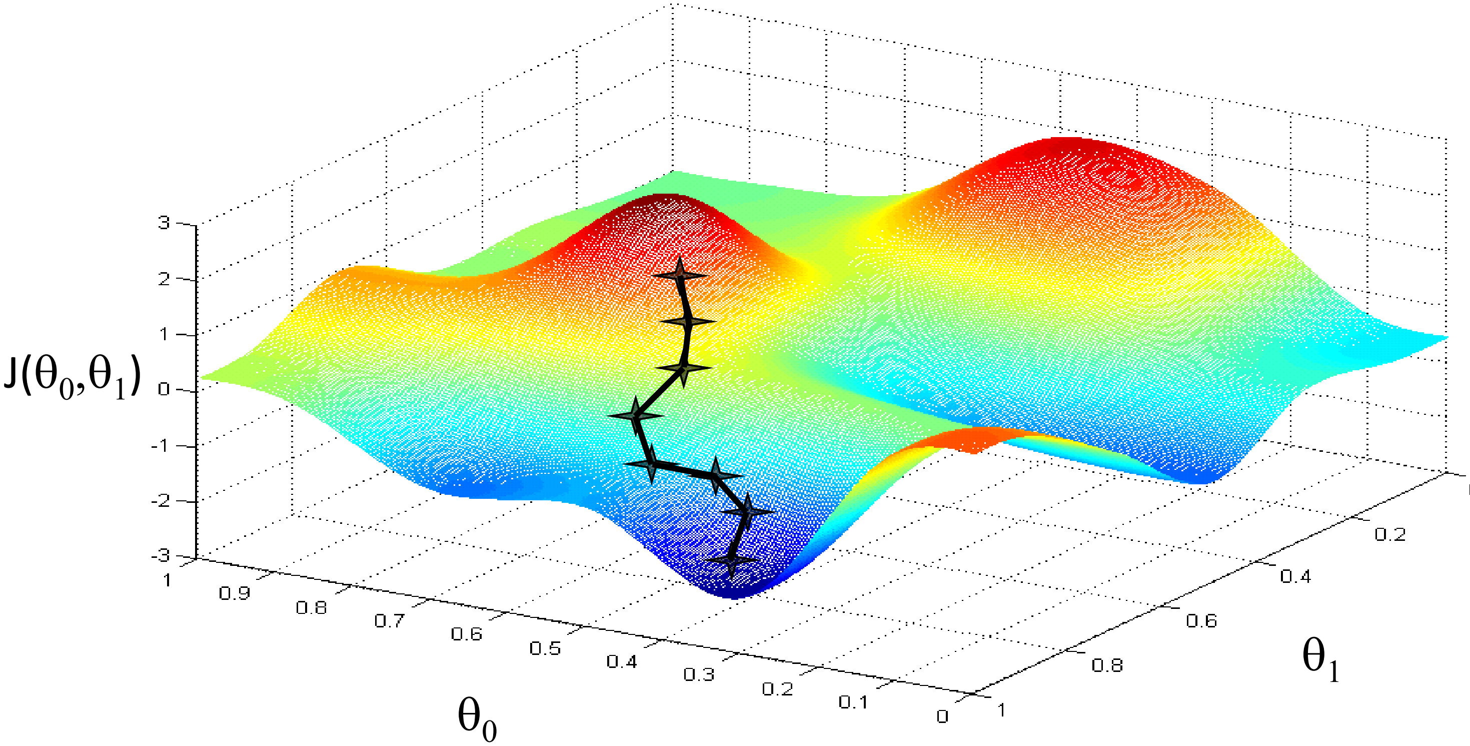
Want $\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$

$\min_{\theta_0 \ldots \theta_n} J(\theta_0, \theta_1, \theta_2, \ldots, \theta_n)$

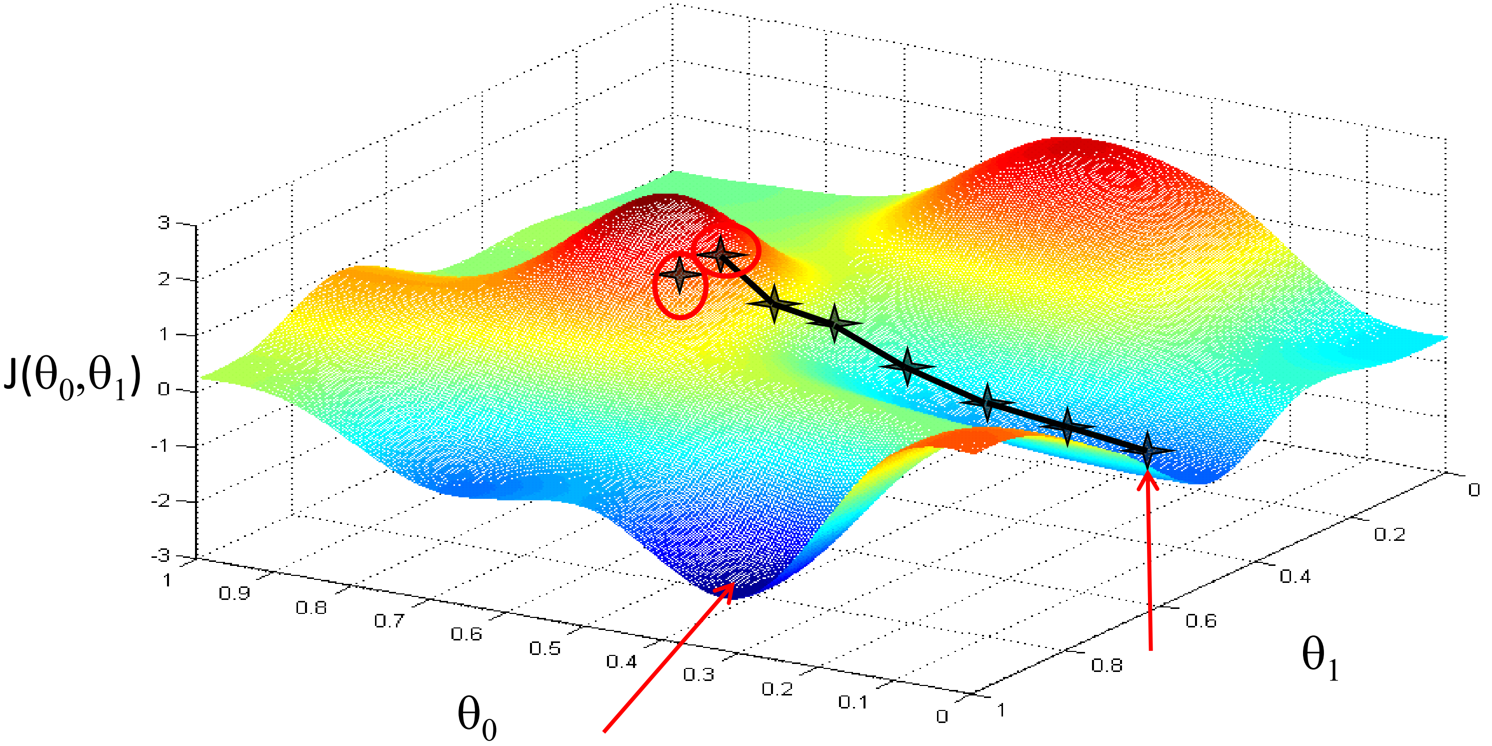
Outline:

- Start with some θ_0, θ_1 say $\theta_0 = 0, \theta_1 = 0$
- Keep changing $\boxed{\theta_0, \theta_1}$ to reduce $\boxed{J(\theta_0, \theta_1)}$

  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



$J(\theta_0,\theta_1)$
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 algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)} \qquad (\text{for } j = 0 \text{ and } j = 1)$$

}

assignment

learning rate

derivative term

Correct: Simultaneous update

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_0 := \text{temp0}$$
$$\theta_1 := \text{temp1}$$

Incorrect:

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\boxed{\theta_0 := \text{temp0}}$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_1 := \text{temp1}$$

Suppose $\theta_0=1, \theta_1=2$, and we simultaneously update θ_0 and θ_1 using the rule: $\theta_j := \theta_j + \sqrt{\theta_0 \theta_1}$ (for j = 0 and j=1)

What are the resulting values of θ_0 and θ_1?

$a)\ \theta_0 = 1, \quad \theta_1 = 2$

$b)\ \theta_0 = 1 + \sqrt{2}, \quad \theta_1 = 2 + \sqrt{2}$

$c)\ \theta_0 = 2 + \sqrt{2}, \quad \theta_1 = 1 + \sqrt{2}$

$d)\ \theta_0 = 1 + \sqrt{2}, \quad \theta_1 = 2 + \sqrt{\left(1 + \sqrt{2}\right) \cdot 2}$

Linear regression with one variable

Gradient descent intuition

Fundamentals of Machine Learning

Gradient descent algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)}$$

(simultaneously update $j = 0$ and $j = 1$)

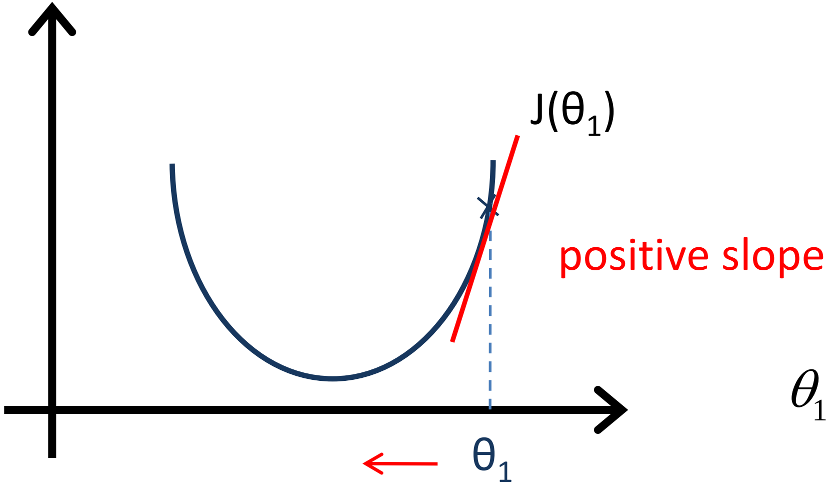
}

learning rate

derivative term

simplier example

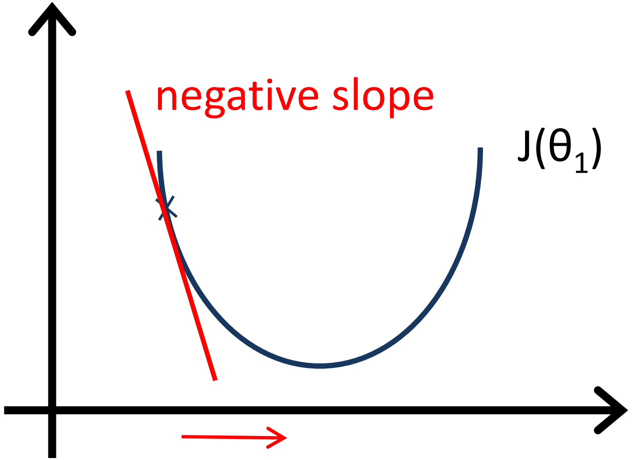
$$\min_{\theta_1} J(\theta_1) \quad \theta_1 \in R$$



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

≥0

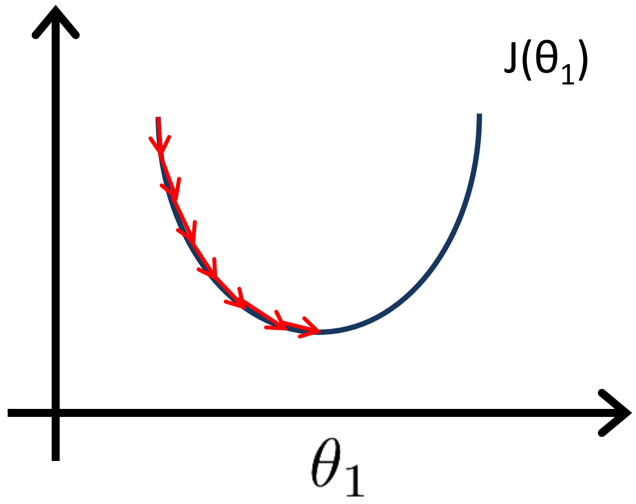
$$\theta_1 := \theta_1 - \alpha(\text{positive number})$$



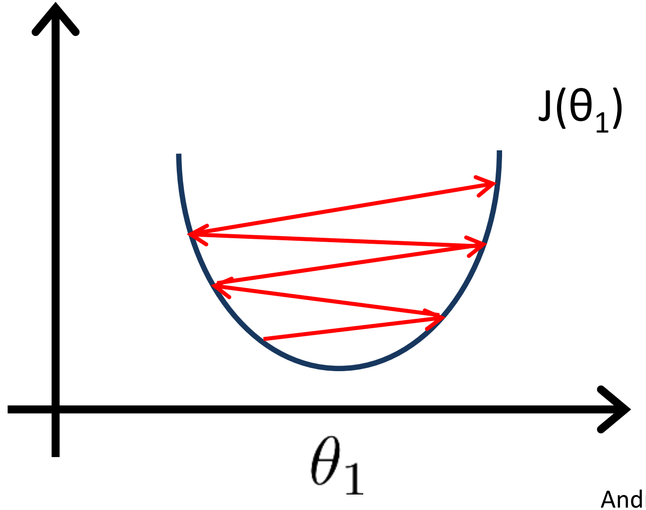
$$\theta_1 := \theta_1 - \alpha(\text{negative number})$$

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

If α is too small, gradient descent can be slow.

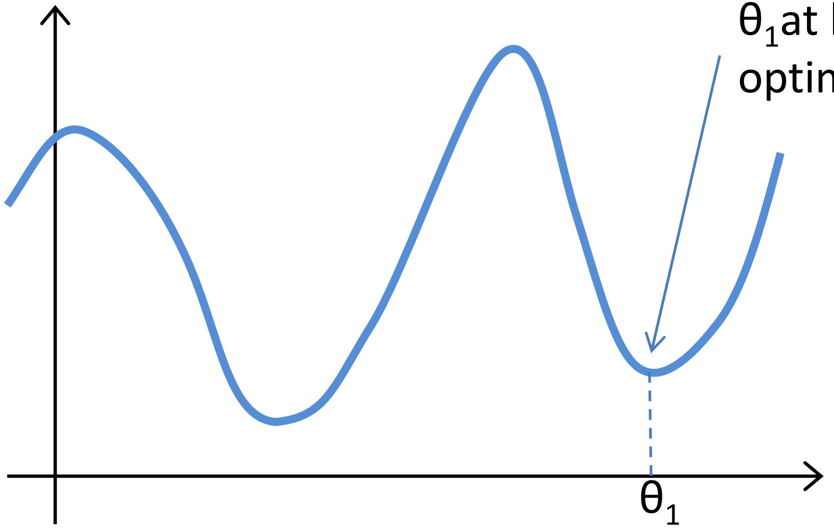


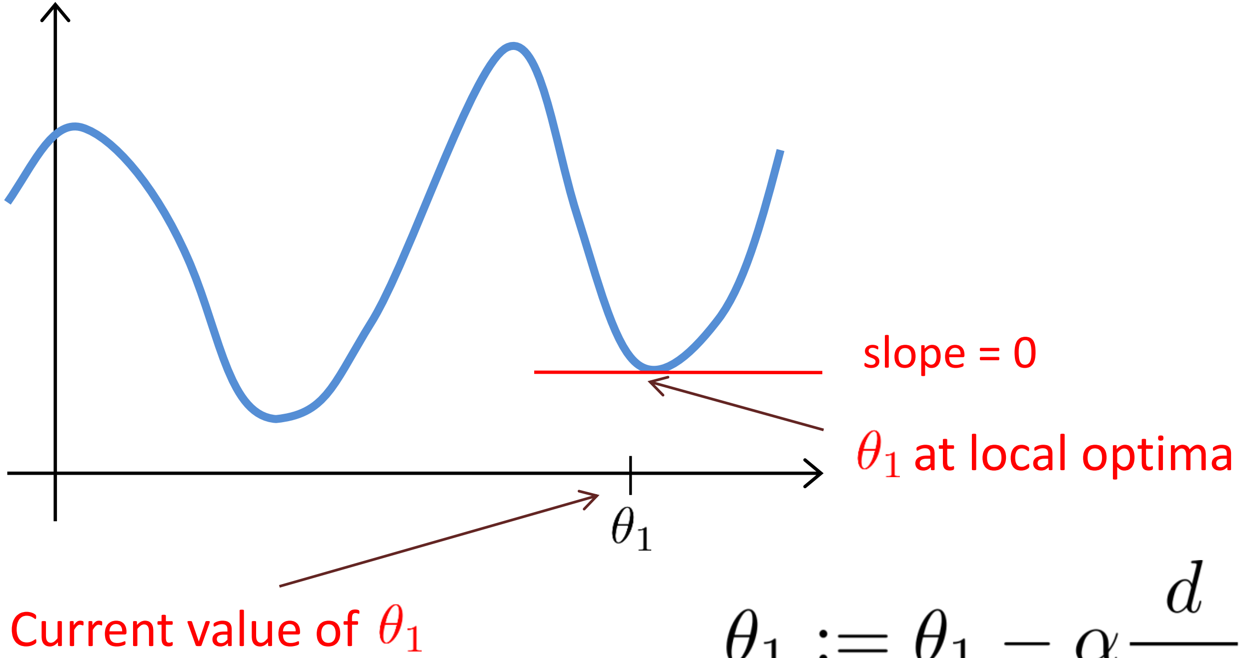
If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.



Suppose θ_1 is at a local optimum of $J(\theta_1)$, such as shown in the figure. What will one step $\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$ of gradient descent do?

a) Leave θ_1 unchanged
b) Change θ_1 in a random direction
c) Move θ_1 in the direction of the Global minimum of $J(\theta_1)$
d) Decrease θ_1



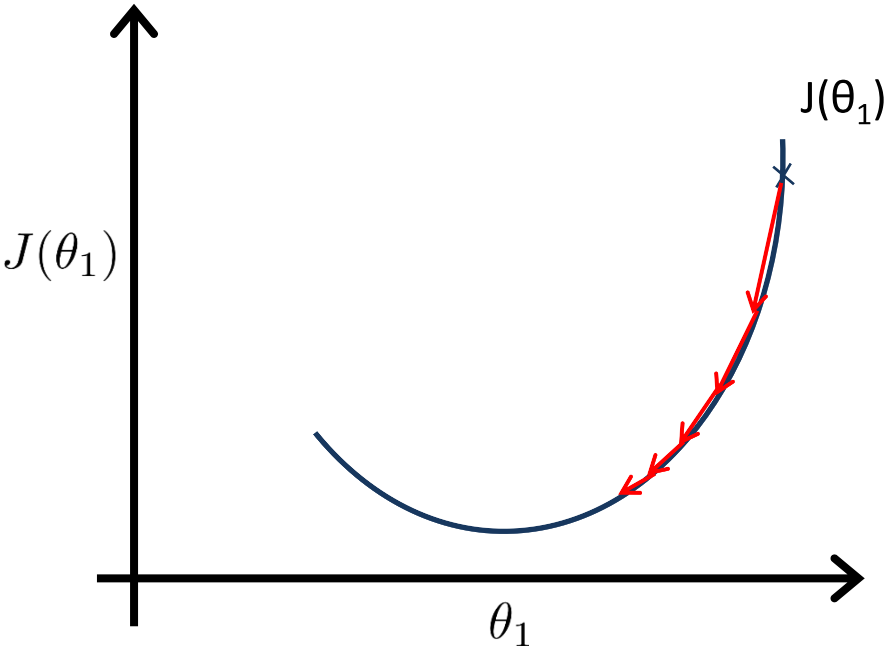


$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

Gradient descent can converge to a local minimum, even with the learning rate α fixed.

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

As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.



Linear regression with one variable

Gradient descent for linear regresion

Fundamentals of Machine Learning

Gradient descent algorithm

$$\text{repeat until convergence } \{$$
$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$
$$(\text{for } j = 1 \text{ and } j = 0)$$
$$\}$$

Linear Regression Model

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

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

$$\frac{\partial}{\partial\theta_j} J(\theta_0, \theta_1) = \frac{\delta}{\delta\theta_j}\frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 = \frac{\delta}{\delta\theta_j}\frac{1}{2m}\sum_{i=1}^{m}\left(\theta_0 + \theta_1 x^{(i)} - y^{(i)}\right)^2$$

$$j = 0 : \frac{\partial}{\partial\theta_0} J(\theta_0, \theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)$$

$$j = 1 : \frac{\partial}{\partial\theta_1} J(\theta_0, \theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x^{(i)}$$

Gradient descent algorithm

$$\frac{\delta}{\delta\theta_0} J(\theta_0, \theta_1)$$

repeat until convergence {

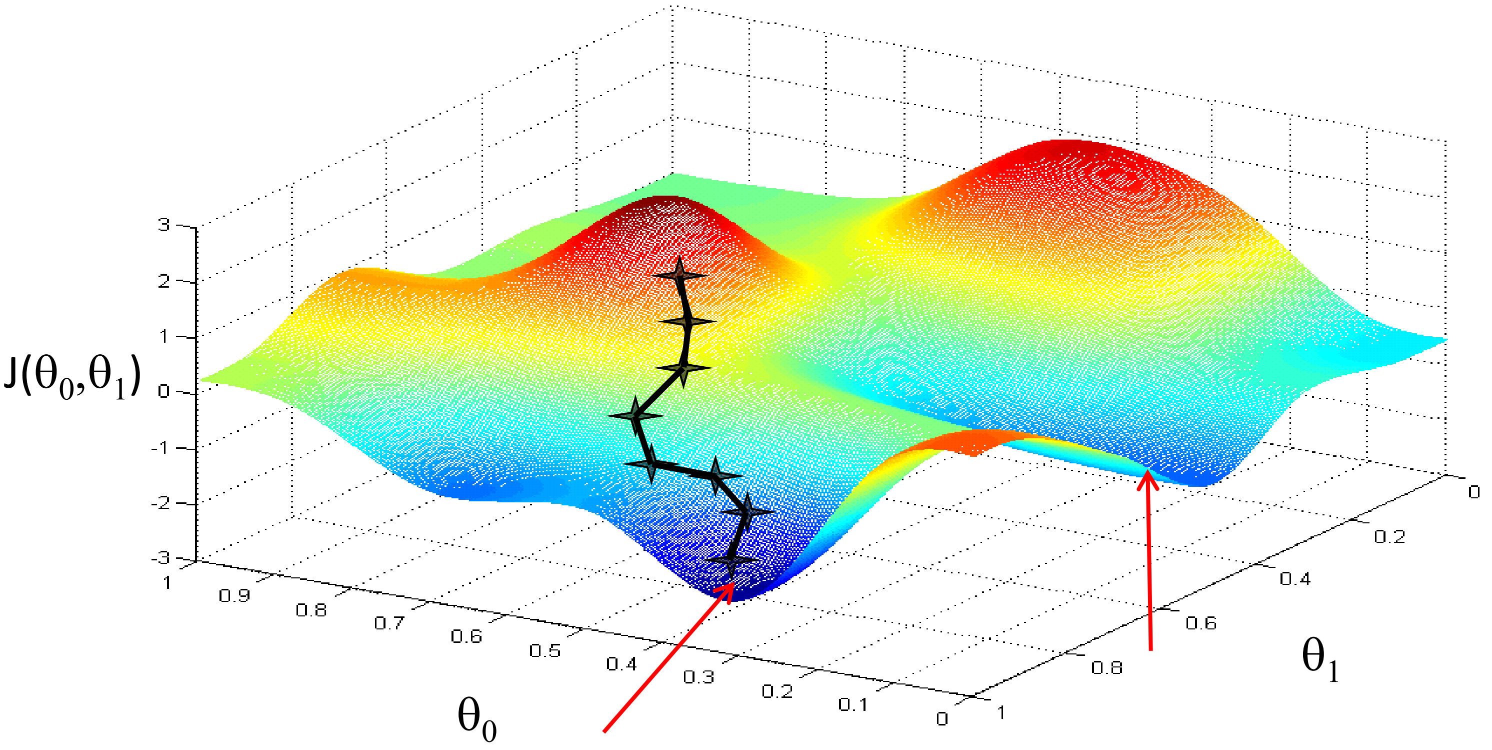
$$\theta_0 := \theta_0 - \alpha \boxed{\frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)}$$

$$\theta_1 := \theta_1 - \alpha \boxed{\frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x^{(i)}}$$

}

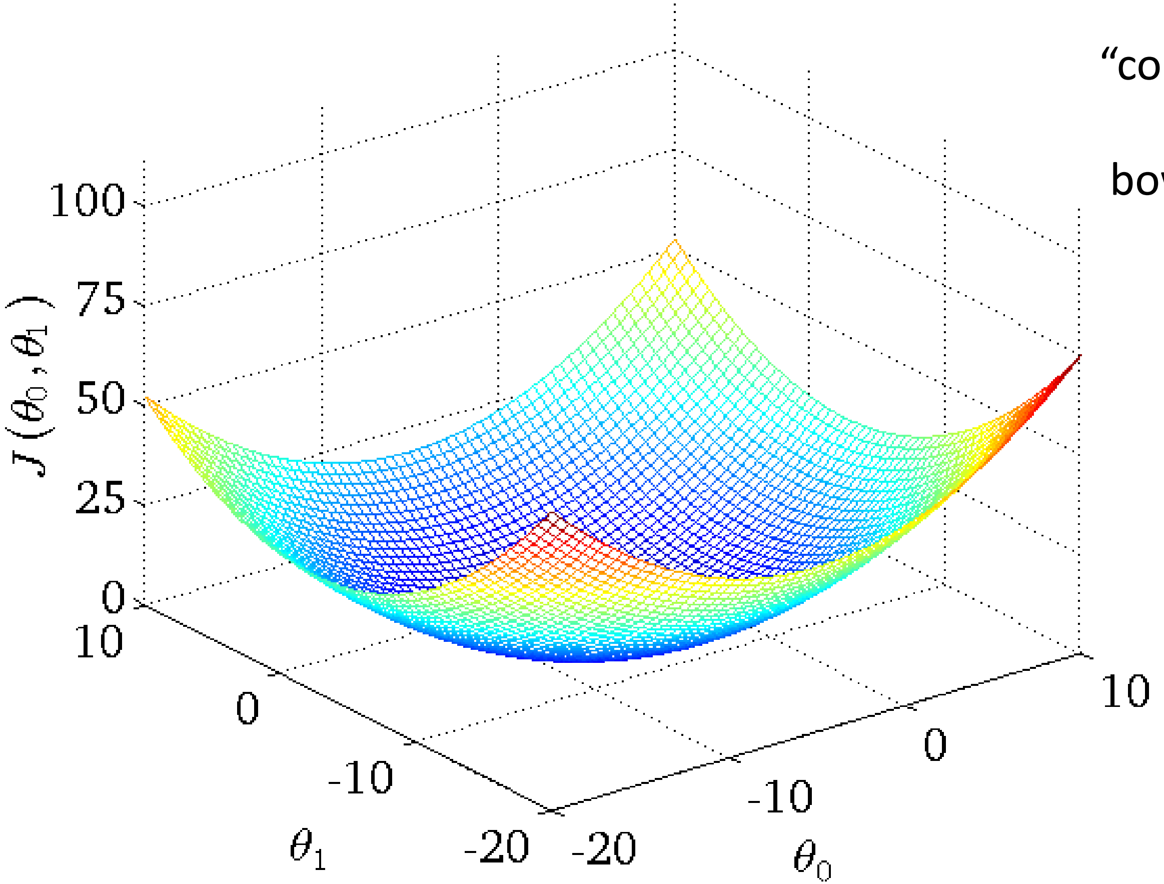
$$\frac{\delta}{\delta\theta_1} J(\theta_0, \theta_1)$$

update θ_0 and θ_1 simultaneously



$J(\theta_0,\theta_1)$
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

Andrew Ng

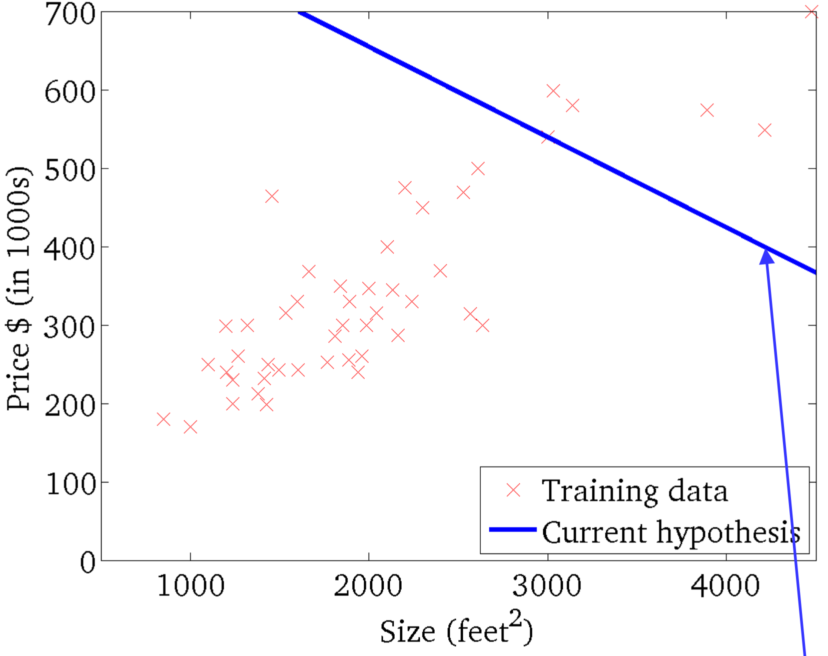


“convex function”

bowl - shaped

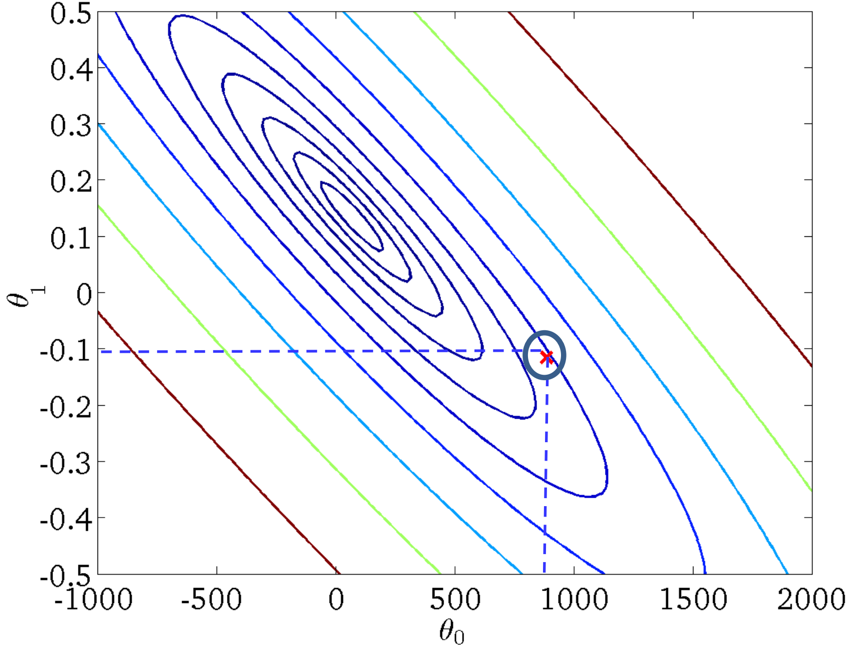
$$h_\theta(x)$$

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



$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)

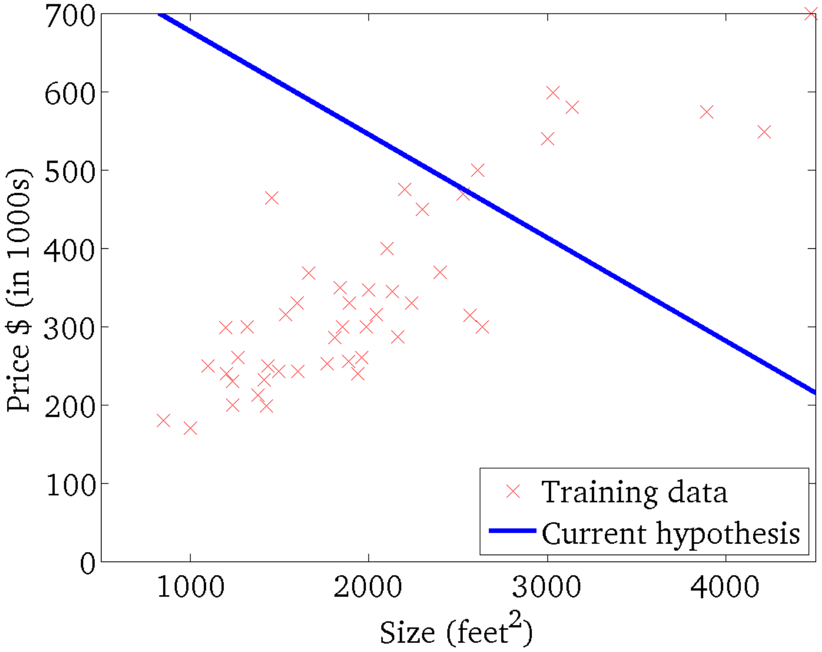


h(x) = 900 -0.1 x

Andrew Ng

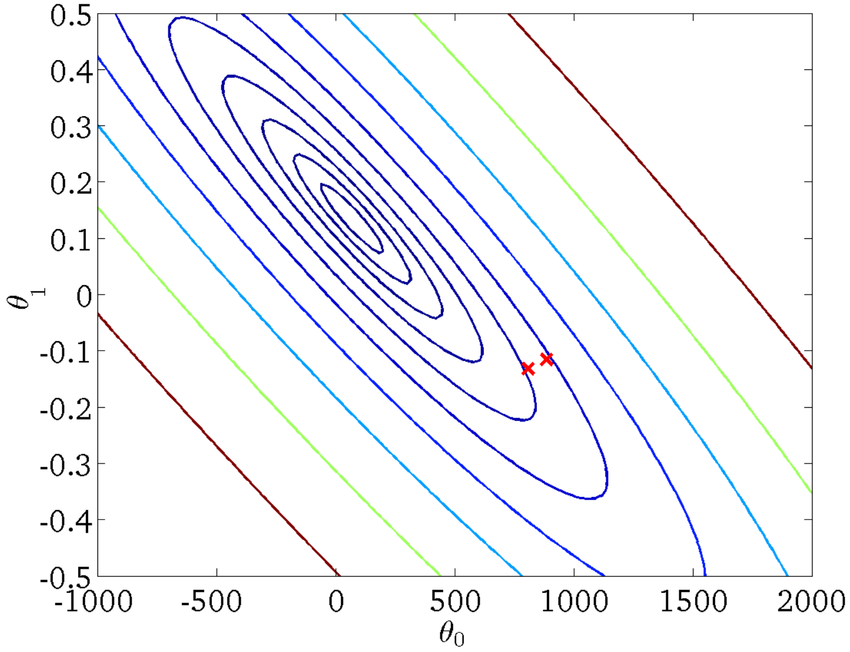
$$h_\theta(x)$$

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



$$J(\theta_0, \theta_1)$$

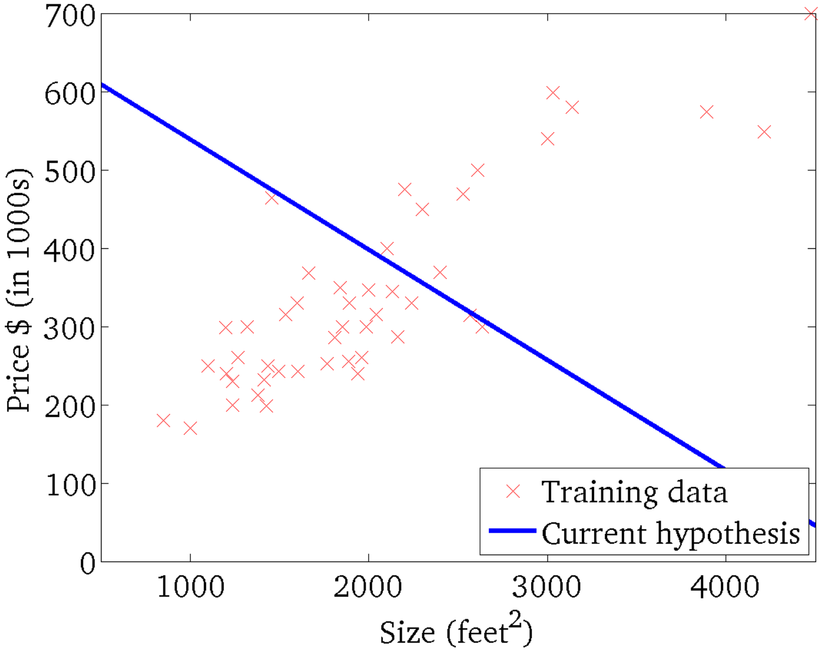
(function of the parameters θ_0, θ_1)



Andrew Ng

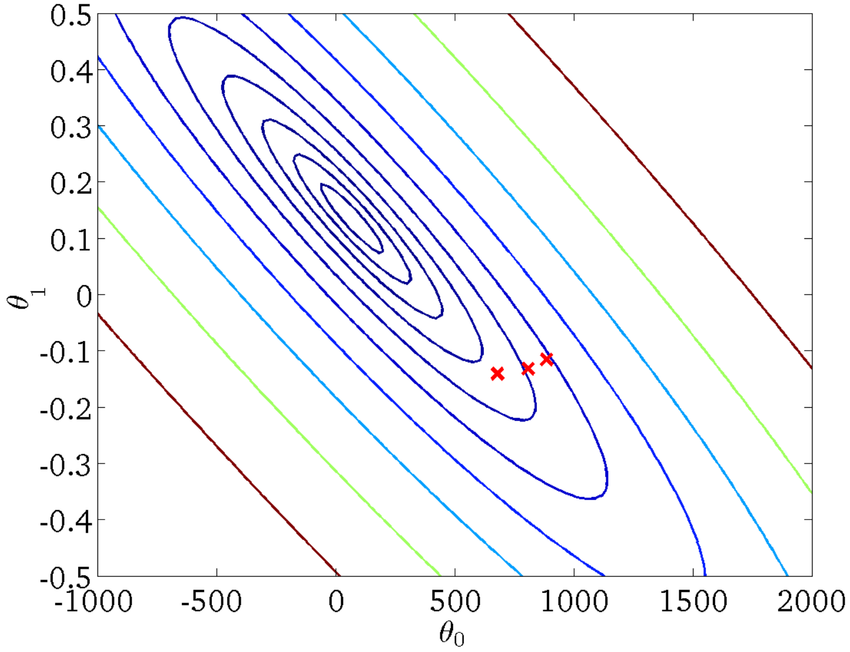
$$h_\theta(x)$$

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



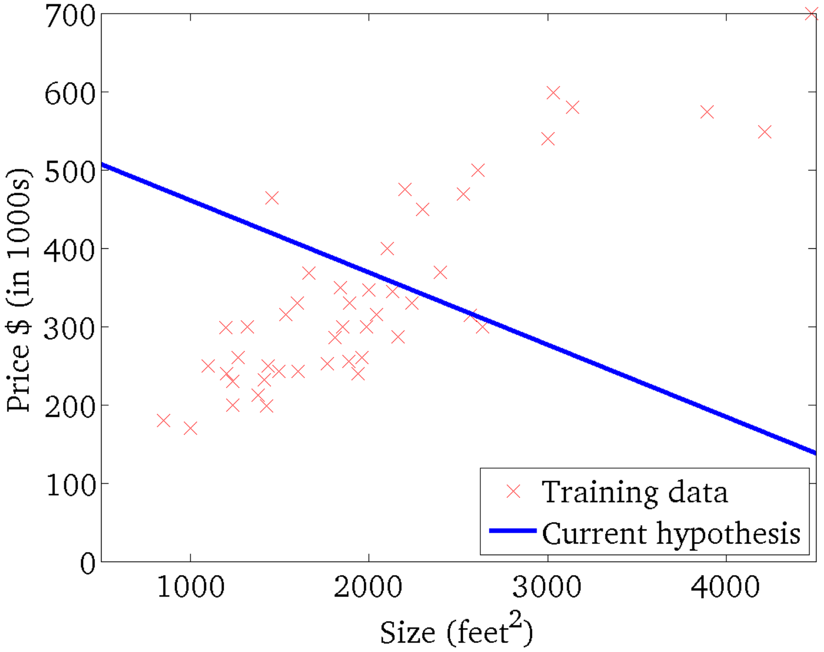
$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)



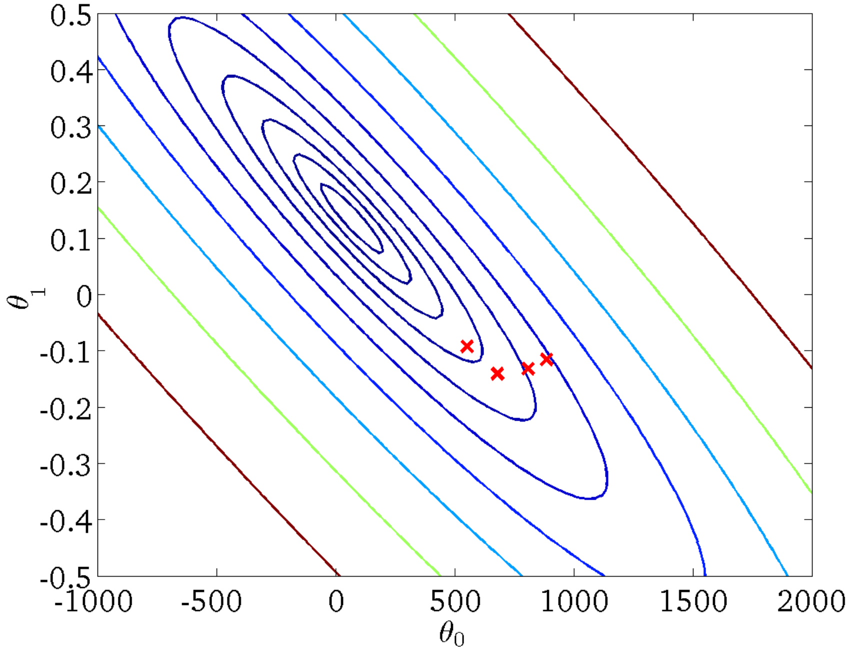
$h_\theta(x)$

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



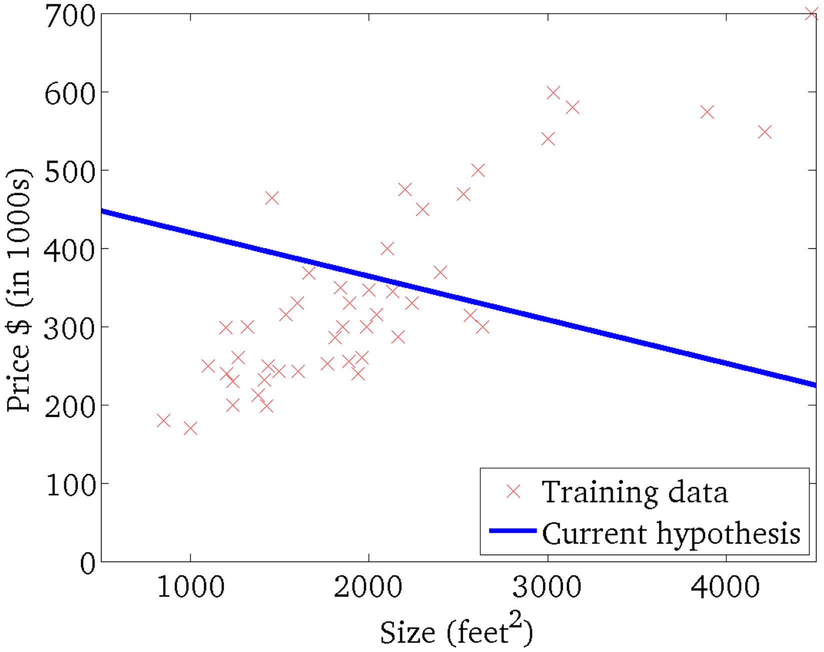
$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



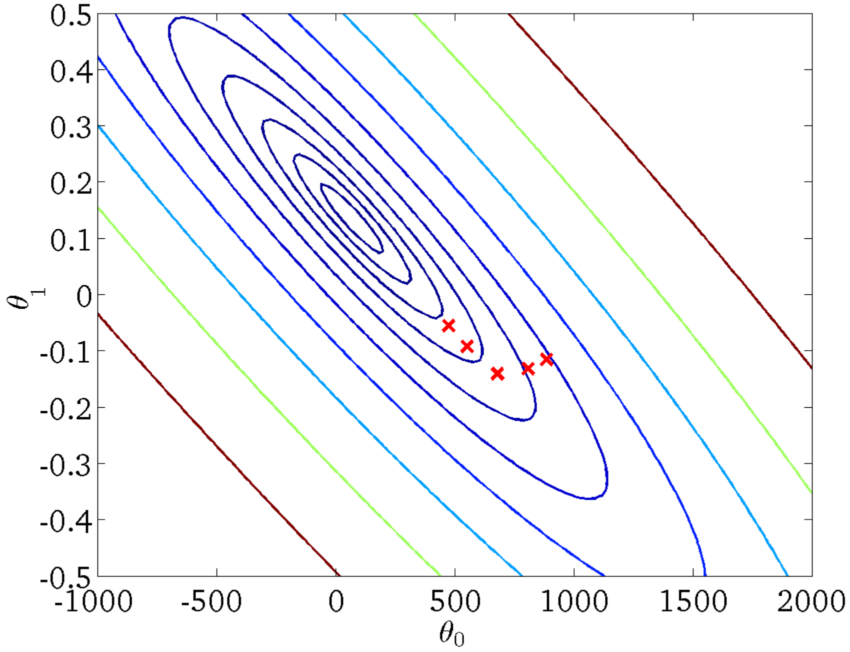
$h_\theta(x)$

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



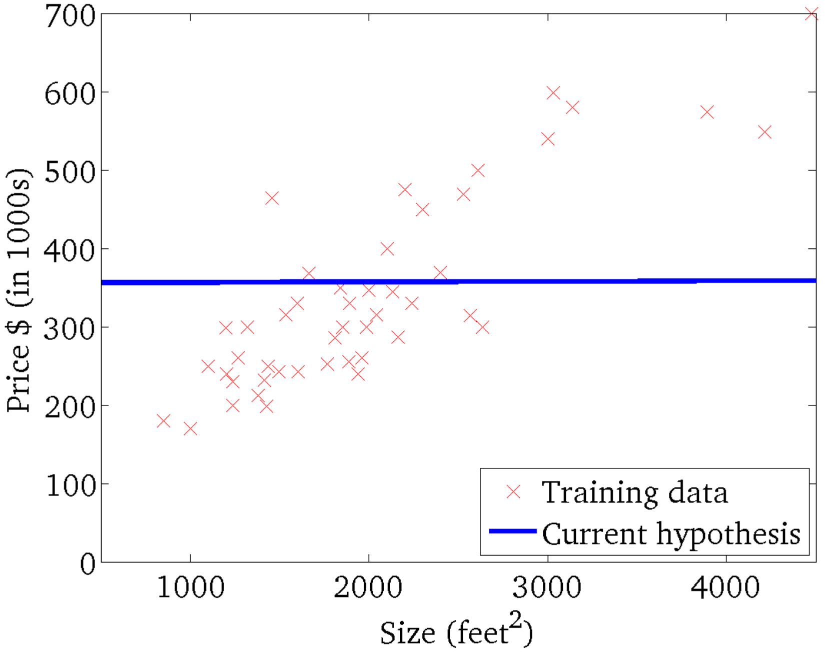
$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



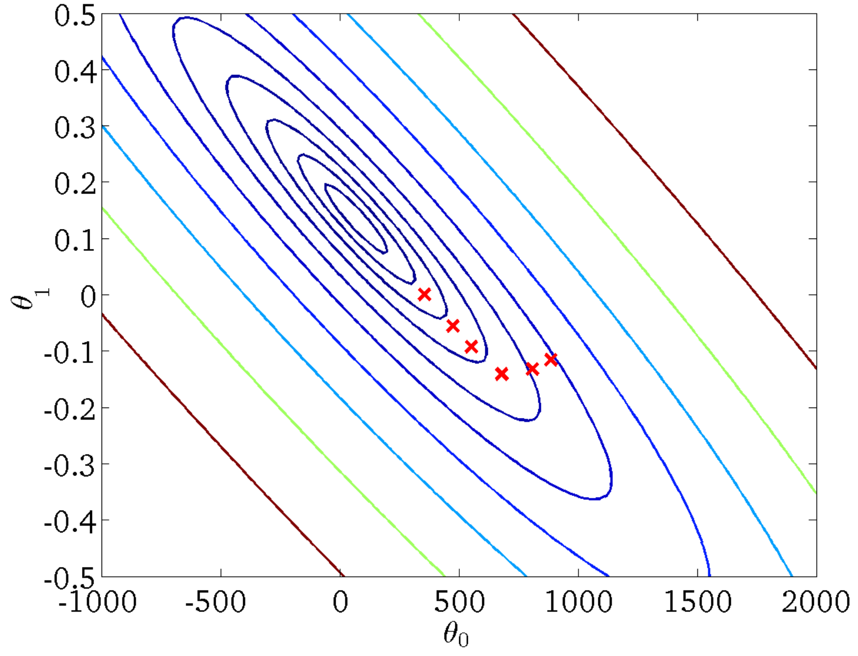
$h_\theta(x)$

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



$J(\theta_0, \theta_1)$

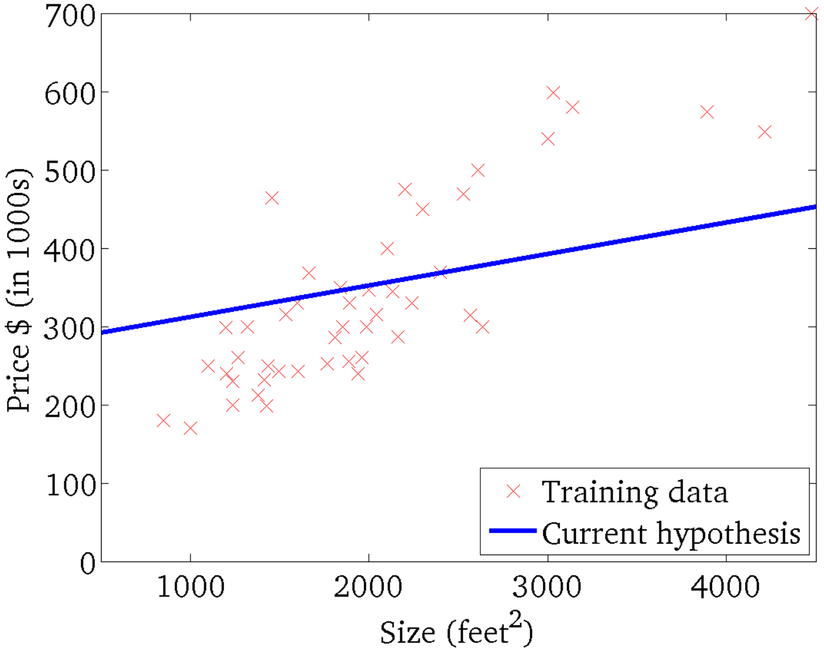
(function of the parameters θ_0, θ_1)



Andrew Ng

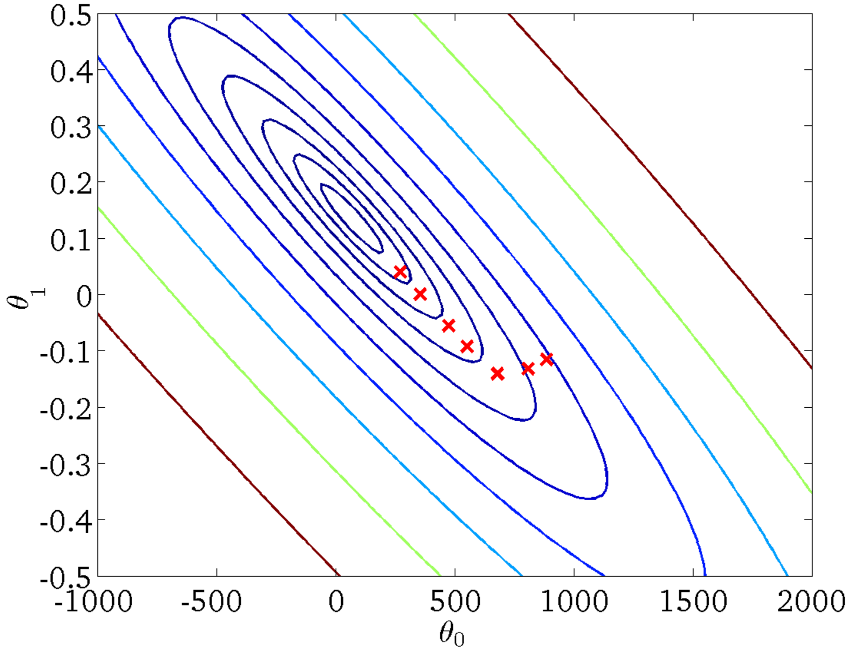
$$h_\theta(x)$$

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



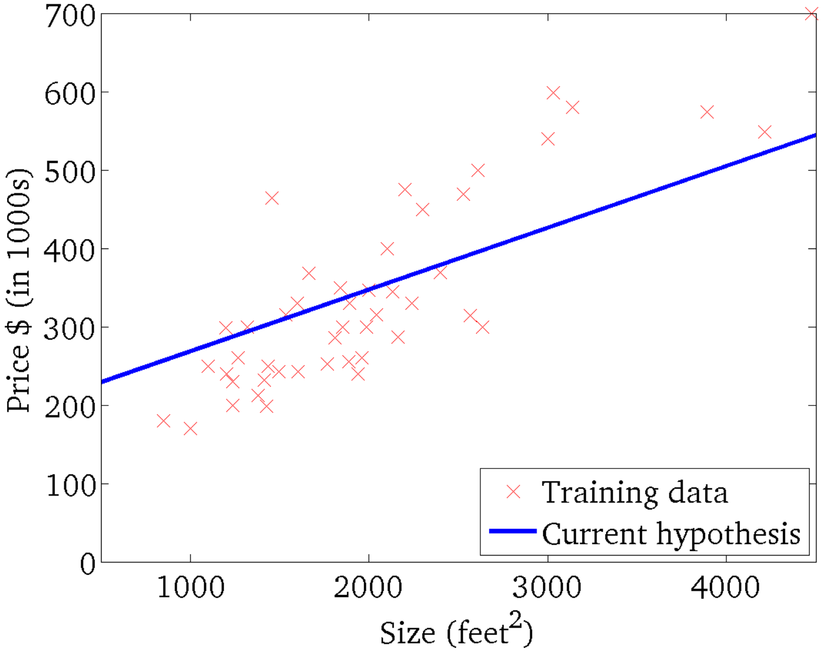
$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)



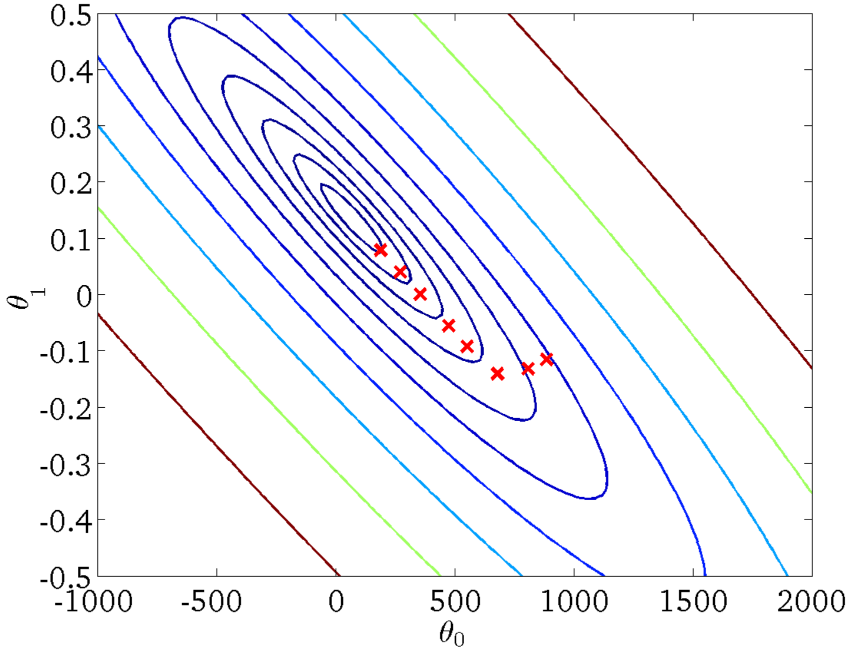
$$h_\theta(x)$$

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



$$J(\theta_0, \theta_1)$$

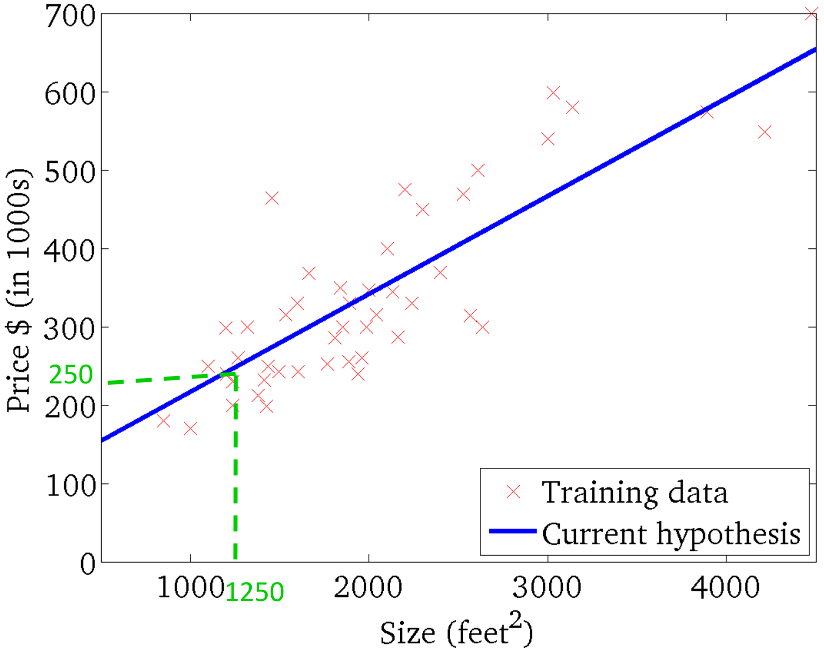
(function of the parameters θ_0, θ_1)



Andrew Ng

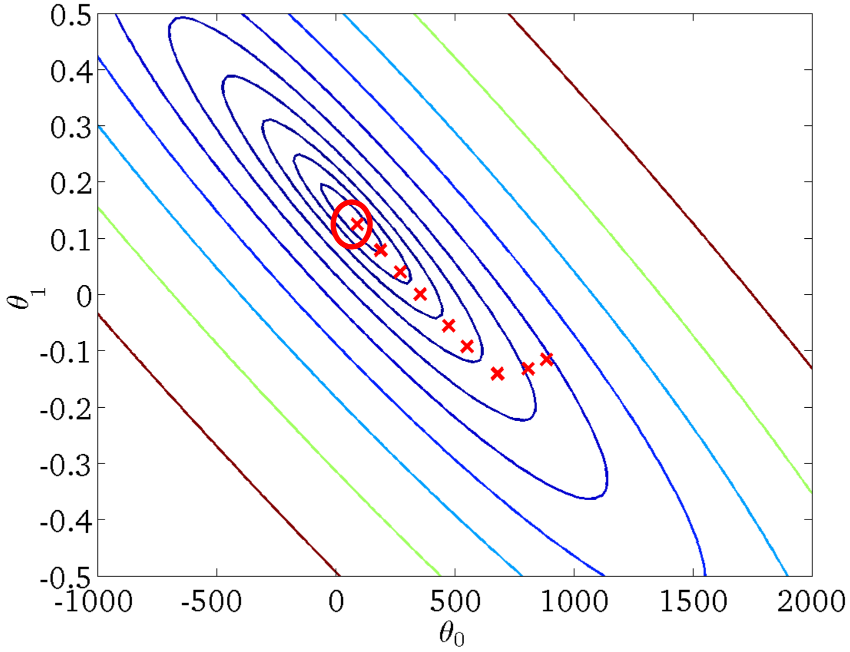
$$h_\theta(x)$$

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



$$J(\theta_0, \theta_1)$$

(function of the parameters θ_0, θ_1)



Andrew Ng

"Batch" Gradient Descent

"Batch": Each step of gradient descent uses all the training examples.

$$\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)$$

Which of the following are true statements? Select all that apply.

a) To make gradient descent converge, we must slowly decrease α over time.
b) Gradient descent is guaranteed to find the global minimum for any function $J(\theta_0,\theta_1)$.
c) Gradient descent can converge even if α is kept fixed. (But α cannot be too large, or else it may fail to converge.)
d) For the specific choice of cost function $J(\theta_0,\theta_1)$ used in linear regression, there are no local optima (other than the global optimum).