

1. 线性回归。

符号表示： m = 训练样本数量 x 's = 输入变量/特征量 y 's = 输出变量/目标变量

(x, y) - 一个训练样本 $(x(i), y(i))$ - 第 i 个训练样本

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
...	...

$m = 47$

Notation:

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

(x, y) - one training example

$(x^{(i)}, y^{(i)})$ - training example

请使用鼠标来选择你认为正确的答案

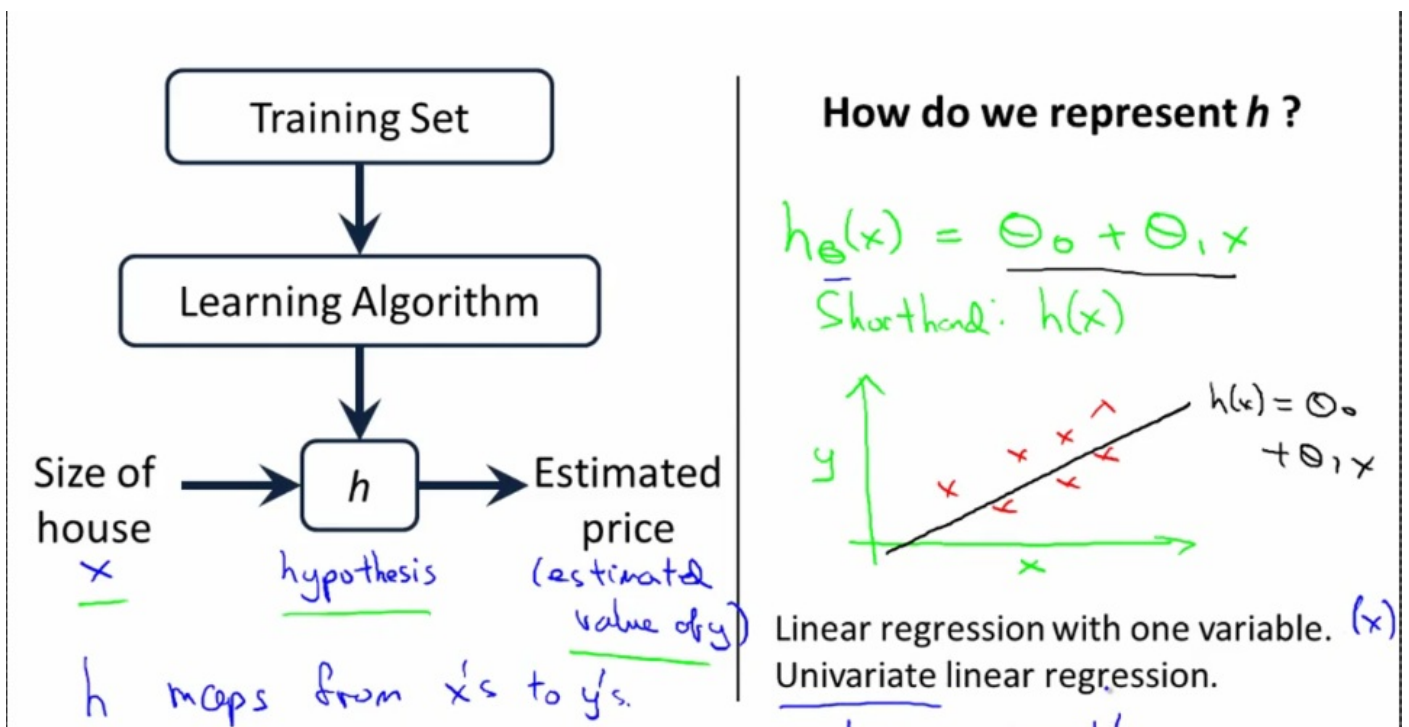
$$x^{(1)} = 2104$$

$$x^{(2)} = 1416$$

$$y^{(1)} = 460$$

Andrew

监督学习算法工作方式：学习算法在训练集学习，然后给定输入预测输出。预测函数因为历史原因命名为hypothesis。线性回归 $h = \theta_0 + \theta_1 x$



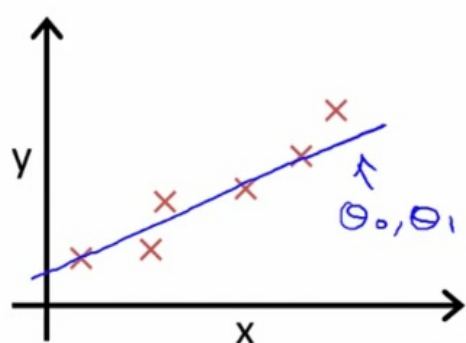
2. 代价函数。选择 θ_0 和 θ_1 使得预测值最接近真实值。

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

Parameters: θ_0, θ_1

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

Goal: minimize $J(\theta_0, \theta_1)$
 θ_0, θ_1



$(x^{(i)}, y^{(i)})$

minimize θ_0, θ_1 $\frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$

#training examples

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

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

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

minimize θ_0, θ_1 $J(\theta_0, \theta_1)$

Cost function

Squared error function

Andrew Ng

对代价函数的直观化理解：首先观察简化版本的代价函数： $h(x) = \theta_1 x$

Simplified

Hypothesis:

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

Parameters:

$$\theta_0, \theta_1$$



Cost Function:

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

Goal: minimize $J(\theta_0, \theta_1)$

$$\theta_0, \theta_1$$

$$h_{\theta}(x) = \theta_1 x$$

$$\theta_0 = 0$$

$$\theta_1$$



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

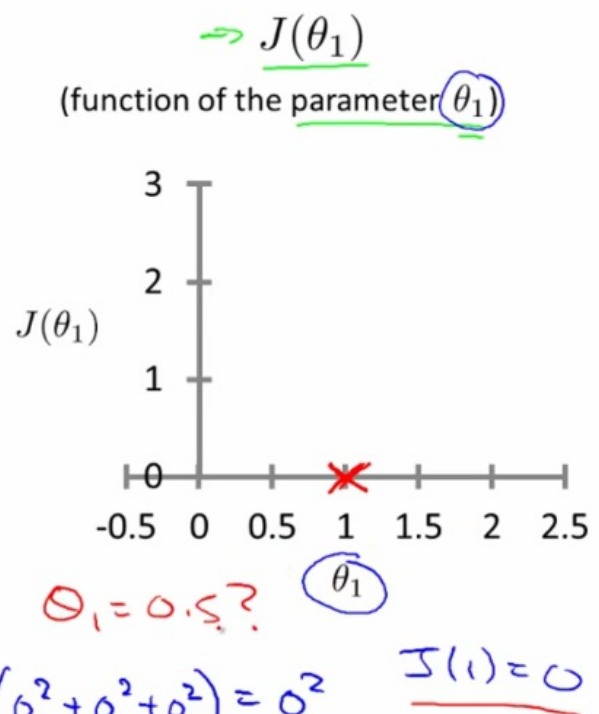
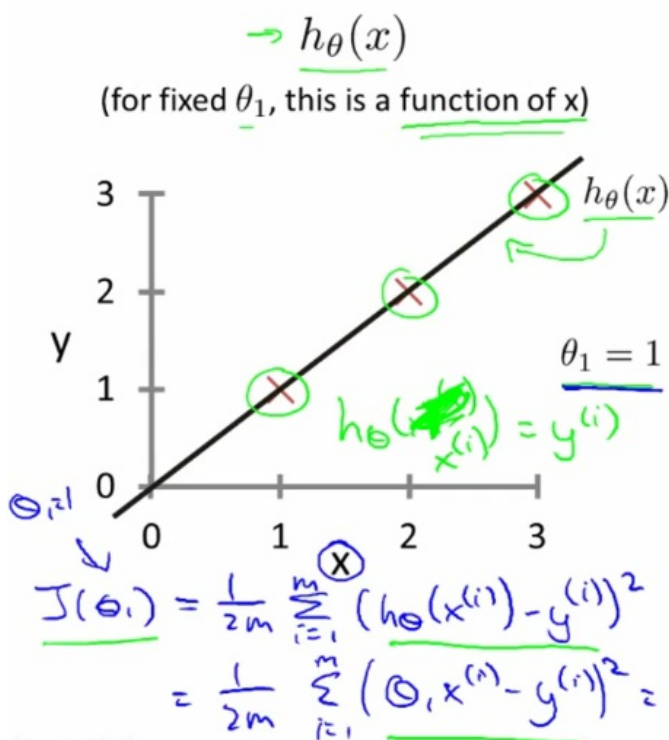
minimize $J(\theta_1)$

$$\theta_1, x^{(i)}$$

代价函数这个概念 我们要理解的是这两个重要的函数

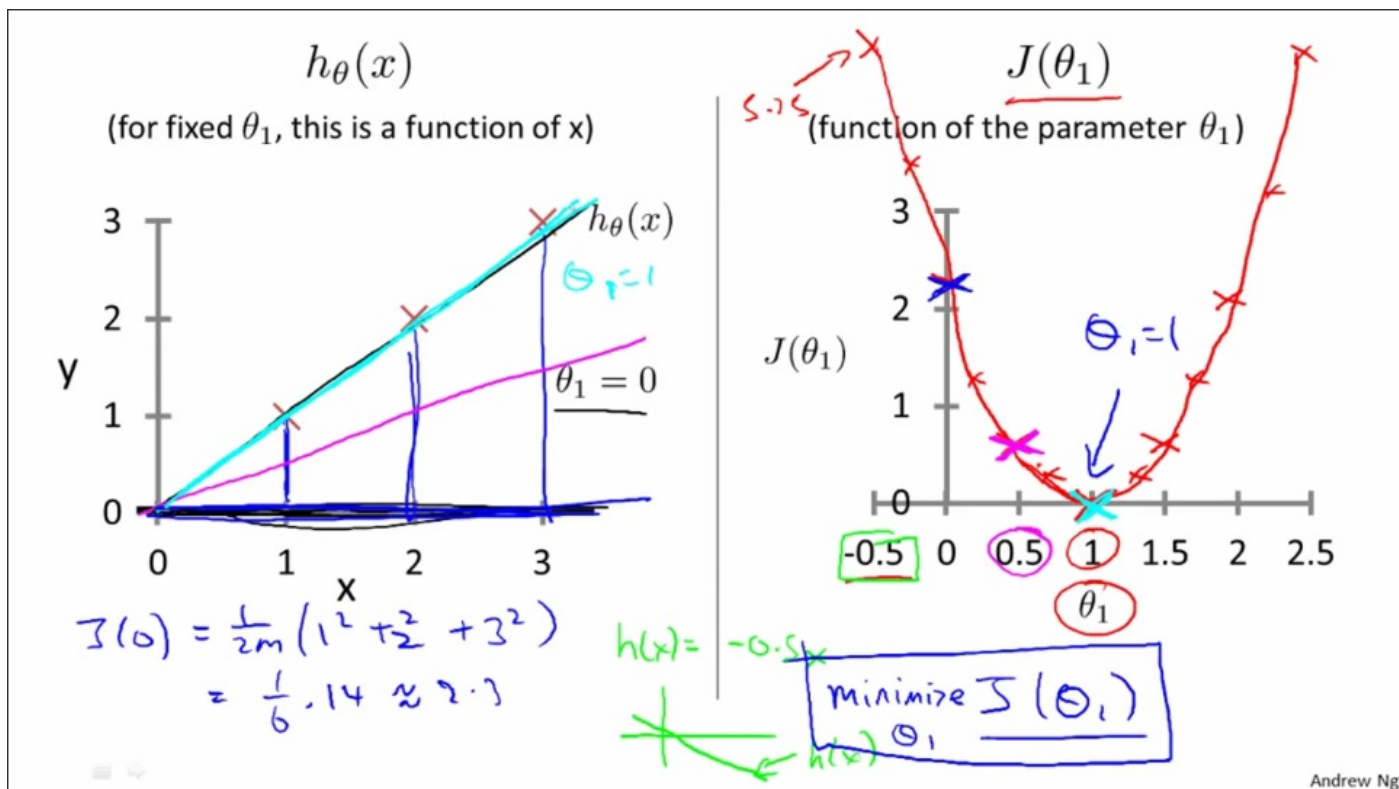
Andrew Ng

由于选择不同的theta，假设函数会有不同的预测效果，也就是说预测值和真实值的差距就不同，所以代价函数是theta的函数。

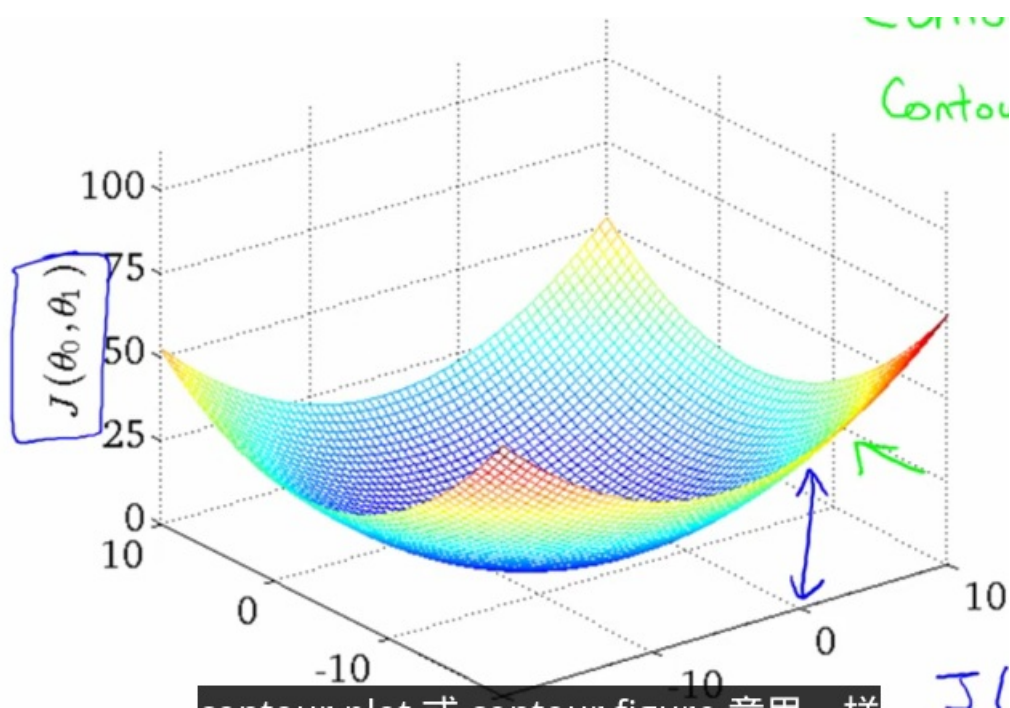


Andrew Ng

所以，我们的目标是最小化代价函数，从而达到最佳的预测效果。



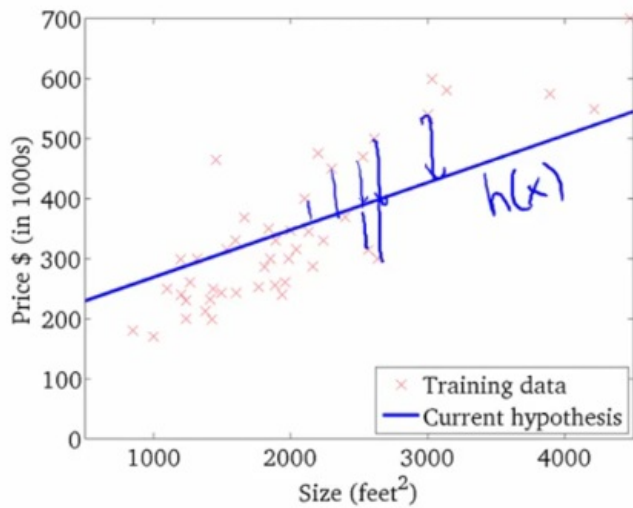
现在不对代价函数进行简化，即：代价函数为标准的 $J(\theta_0, \theta_1)$ 形式。完整代价函数的图形为：



用轮廓图表示为：

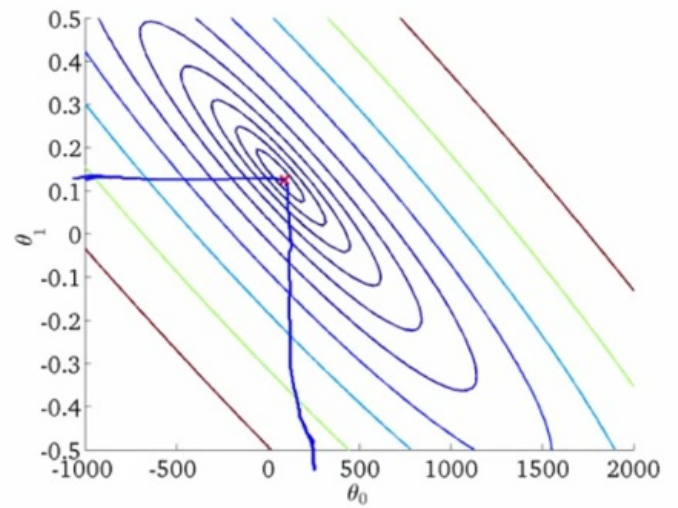
$$h_{\theta}(x)$$

(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)



同一椭圆上的代价函数值相同，中心点是代价函数最小值点。对曲线拟合效果最好，其在代价函数轮廓图中的点越接近中心点。

我们的任务是编程学习出 θ_0, θ_1 的值从而使得代价函数最小。