

回归分析

Regression



Size in feet ² (x)	Price (\$) in 1000's (y)
2104	460
1416	232
1534	315
852	178
...	...
特征值(feature)	标签/结果(target)



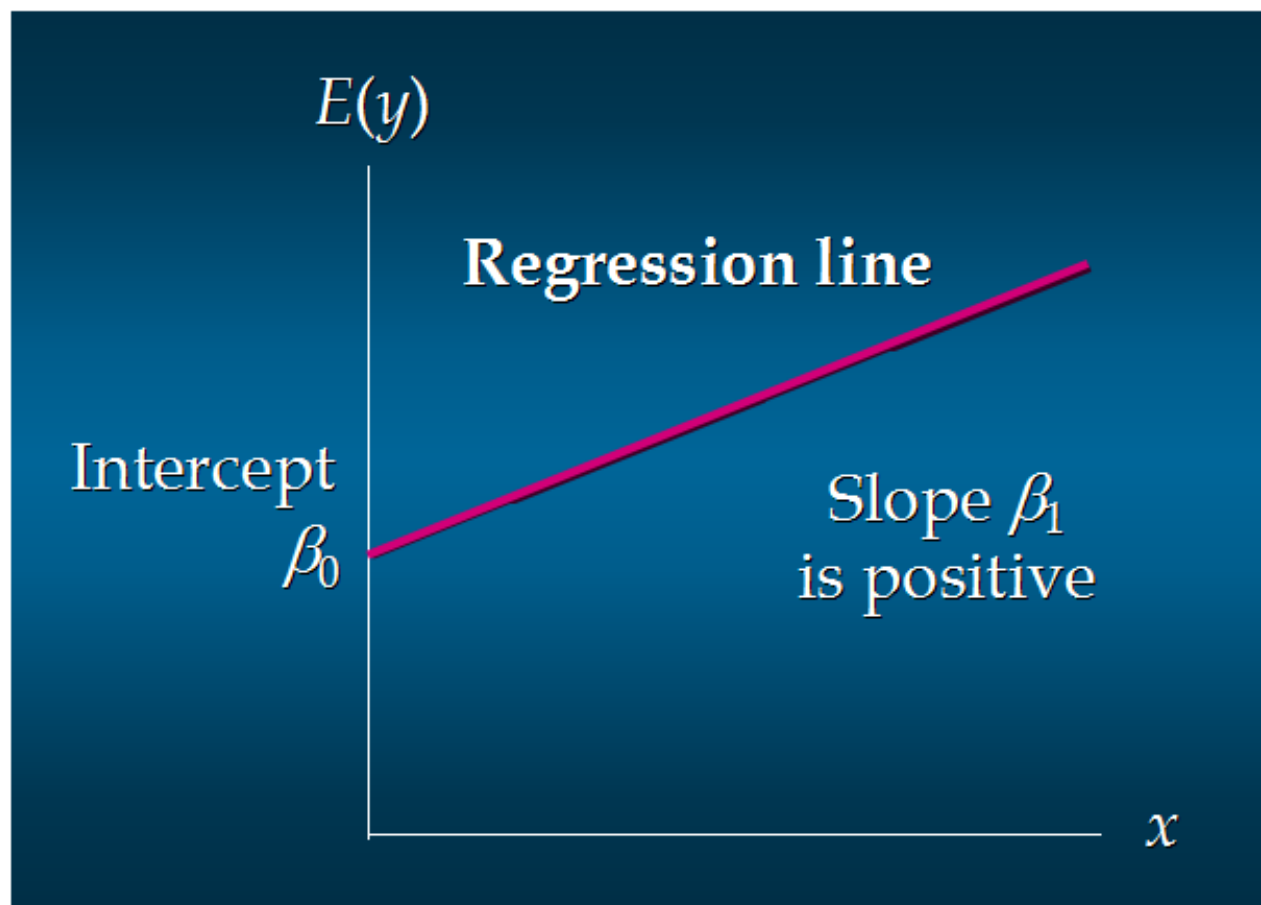
- 回归分析(regression analysis)用来建立方程模拟两个或者多个变量之间如何关联
- 被预测的变量叫做：因变量(dependent variable), 输出(output)
- 被用来进行预测的变量叫做：自变量(independent variable), 输入(input)
- 一元线性回归包含一个自变量和一个因变量
- 以上两个变量的关系用一条直线来模拟
- 如果包含两个以上的自变量，则称作多元回归分析(multiple regression)



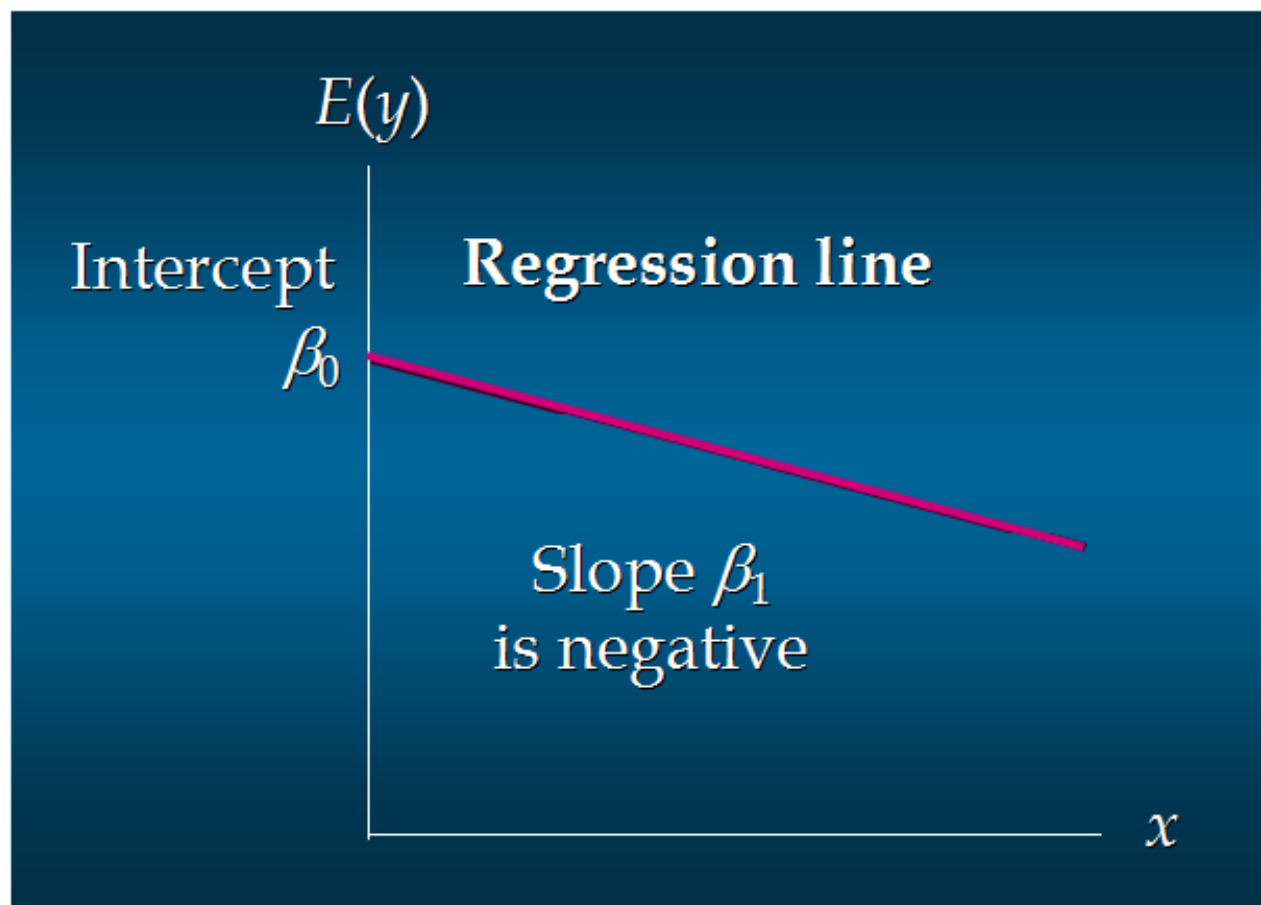
$$h_{\theta}(x) = \theta_0 + \theta_1 x$$

这个方程对应的图像是一条直线，称作回归线。其中， θ_1 为回归线的斜率， θ_0 为回归线的截距。

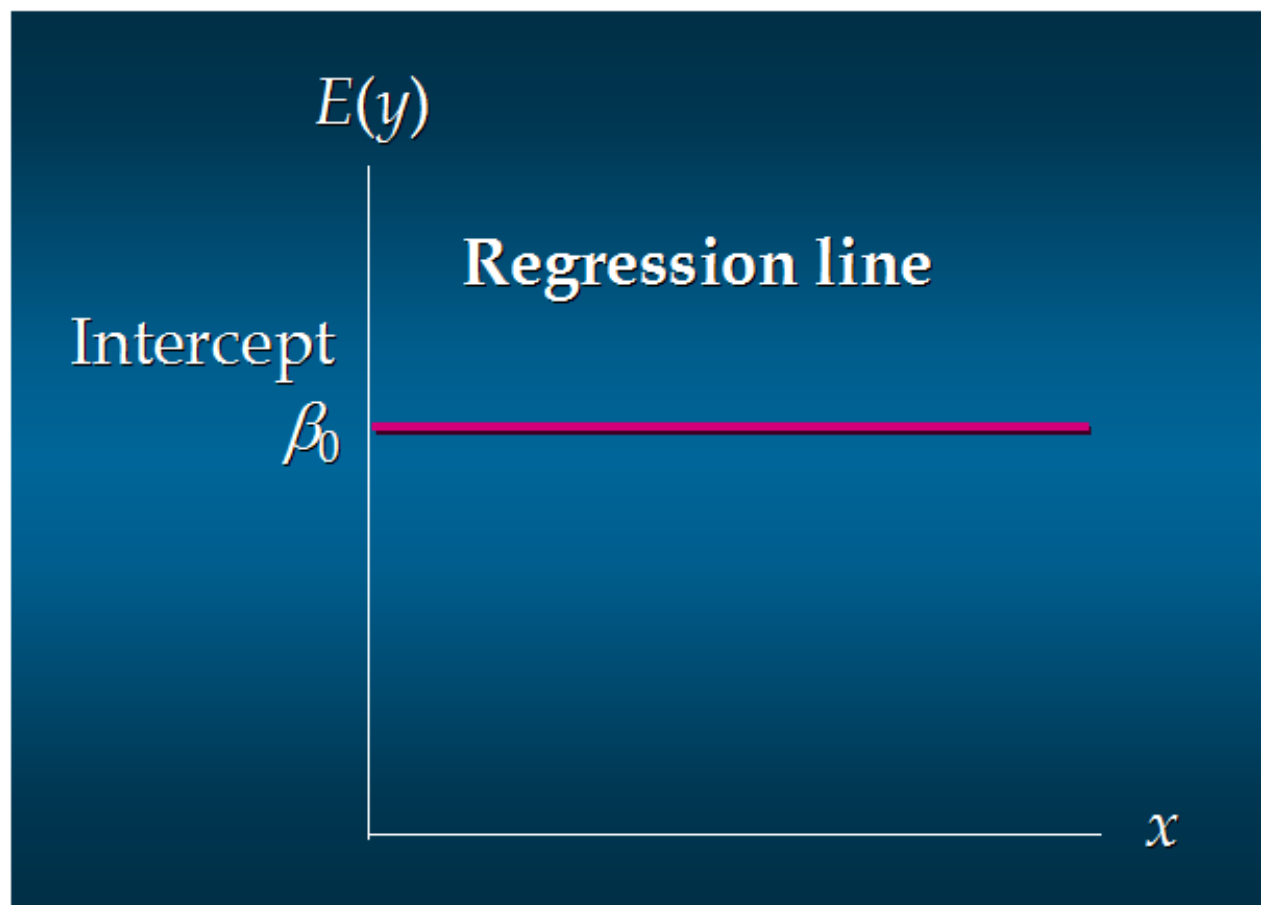
一元线性回归-正相关



一元线性回归-负相关



一元线性回归-不相关



求解方程系数



训练集：

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

需要求解方程：
$$h_{\theta}(x) = \theta_0 + \theta_1 x$$

代价函数

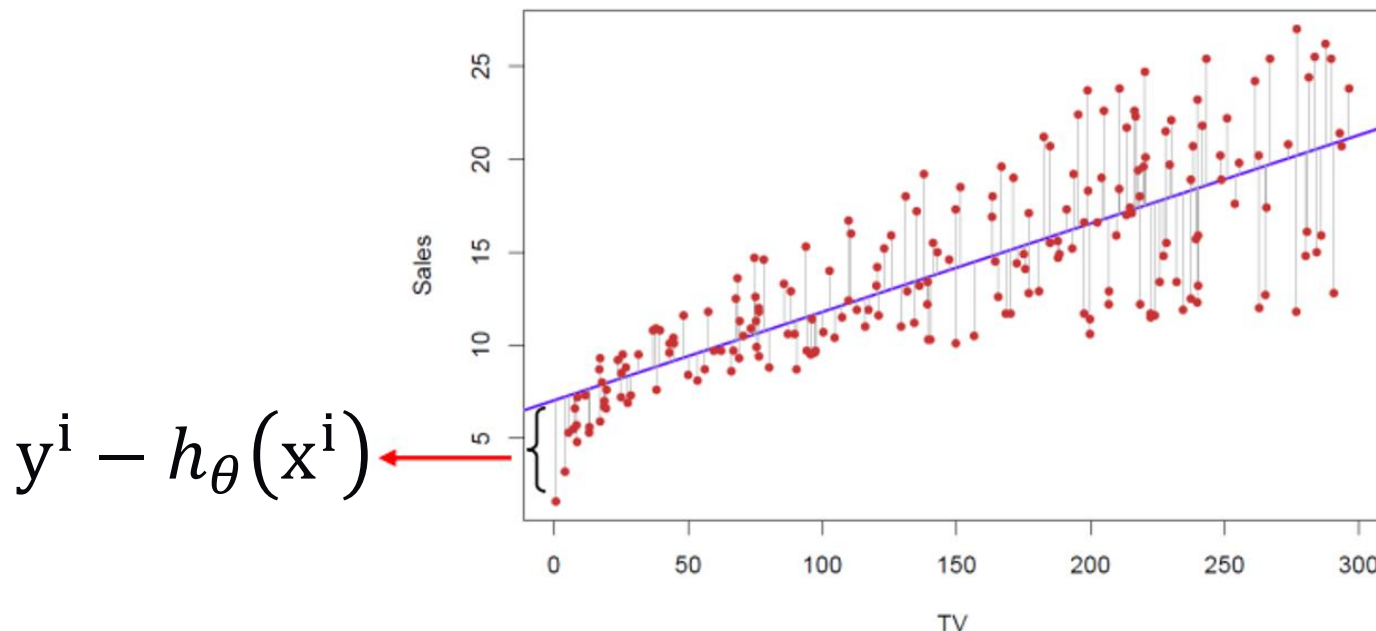
Cost Function

代价函数(Cost Function)



- 最小二乘法
- 真实值 y ，预测值 $h_{\theta}(x)$ ，则误差平方为 $(y - h_{\theta}(x))^2$
- 找到合适的参数，使得误差平方和：

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



代价函数(Cost Function)



假设

Hypothesis:

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

简化

Simplified

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

参数

Parameters:

$$\theta_0, \theta_1$$

$$\theta_1$$

成本函数

Cost Function:

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

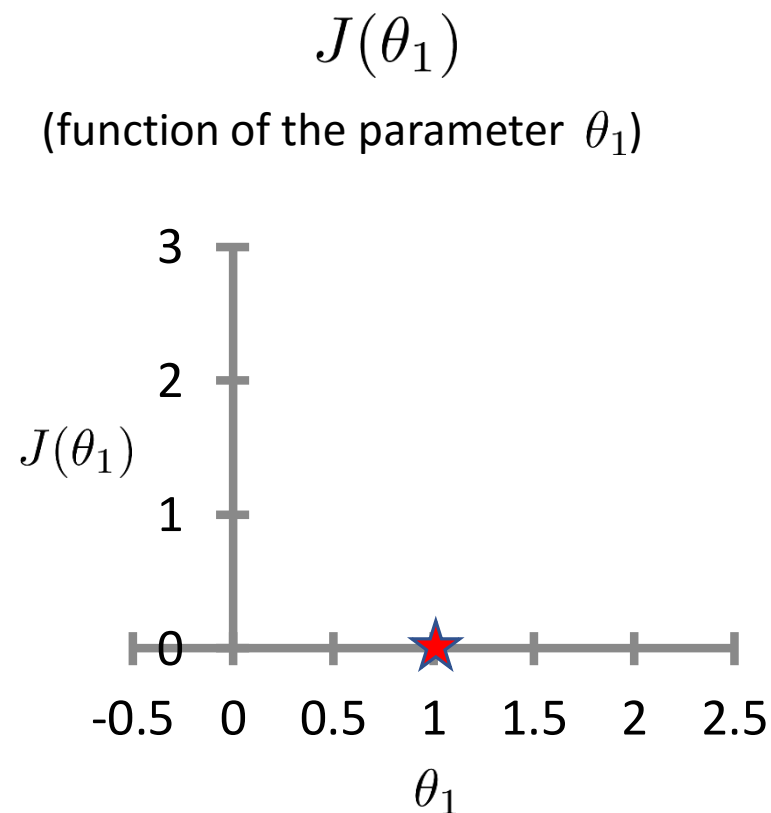
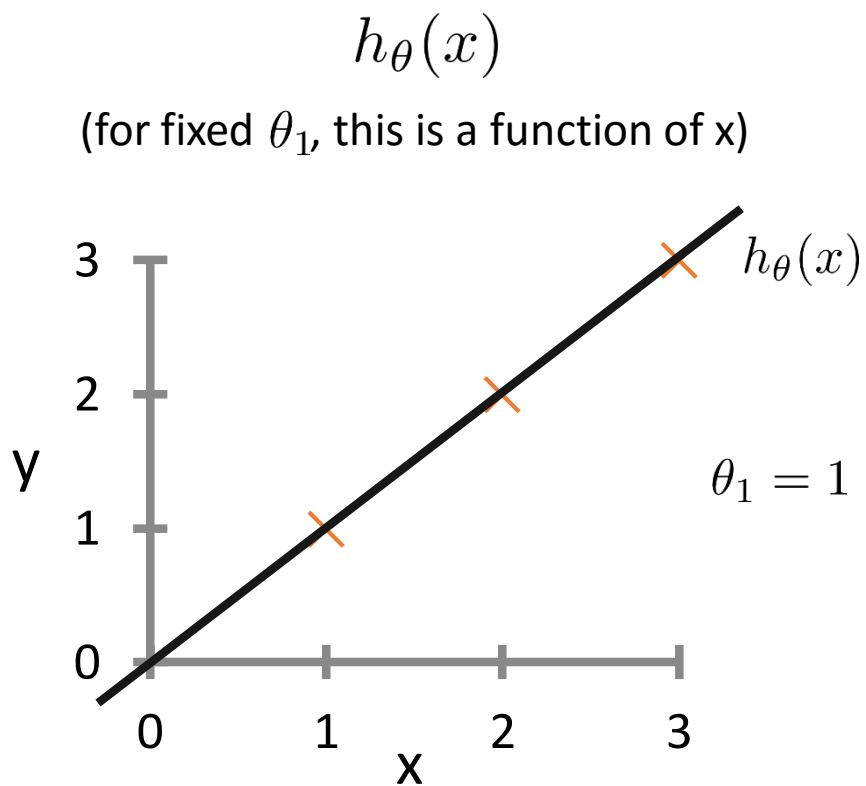
$$J(\theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

目标

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

minimize $J(\theta_1)$
 θ_1

代价函数(Cost Function)

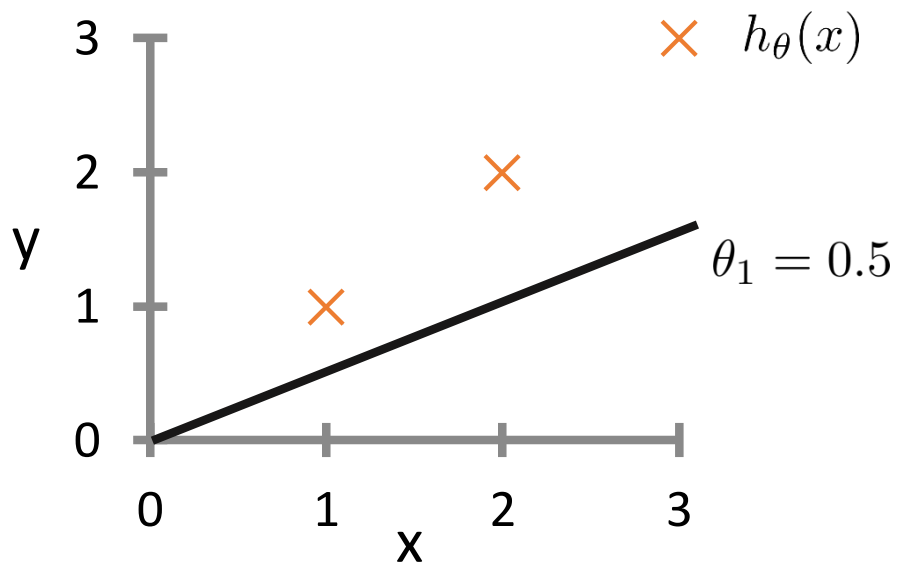


代价函数(Cost Function)



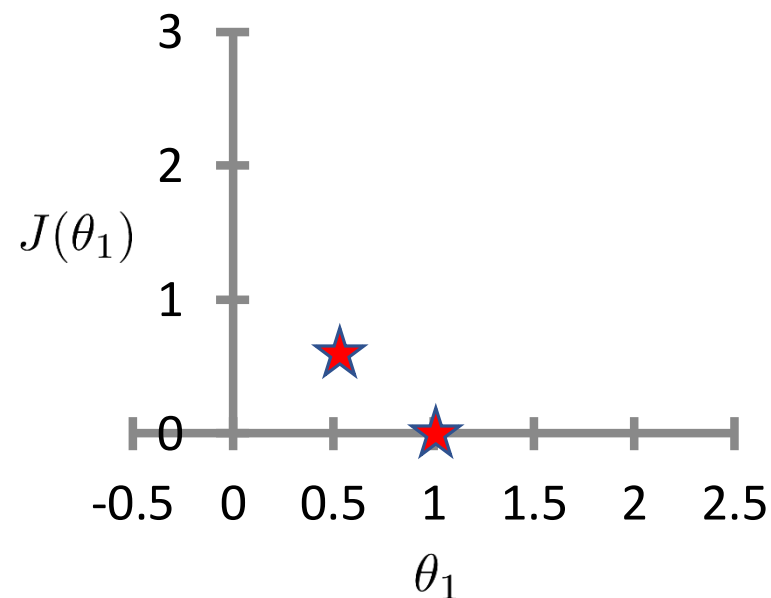
$$h_{\theta}(x)$$

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



$$J(\theta_1)$$

(function of the parameter θ_1)

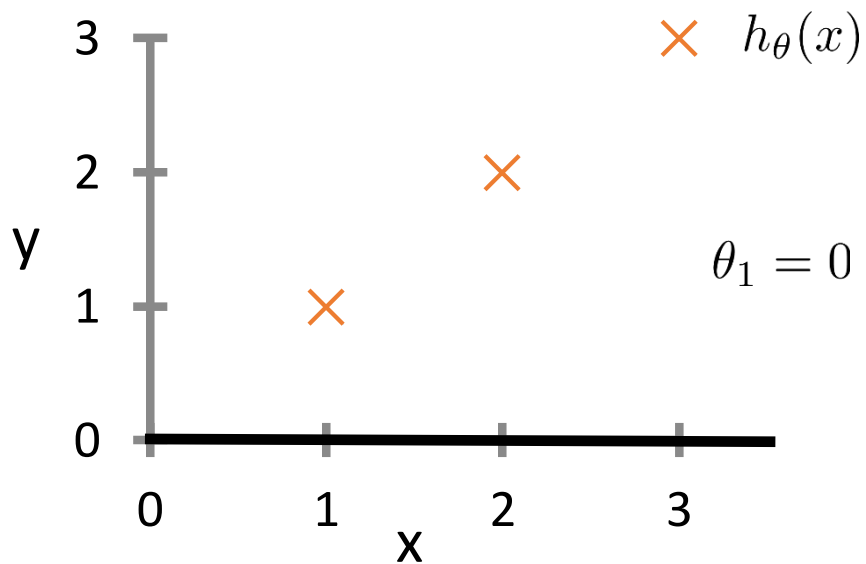


代价函数(Cost Function)



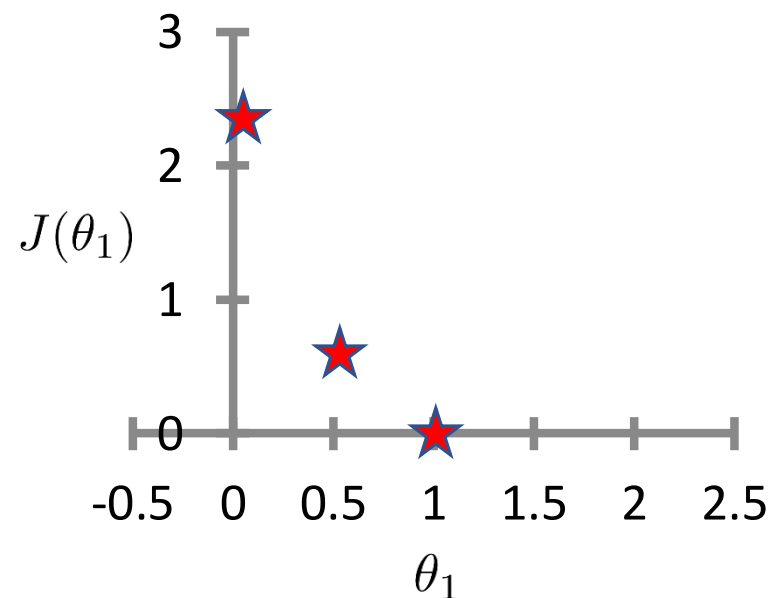
$$h_{\theta}(x)$$

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

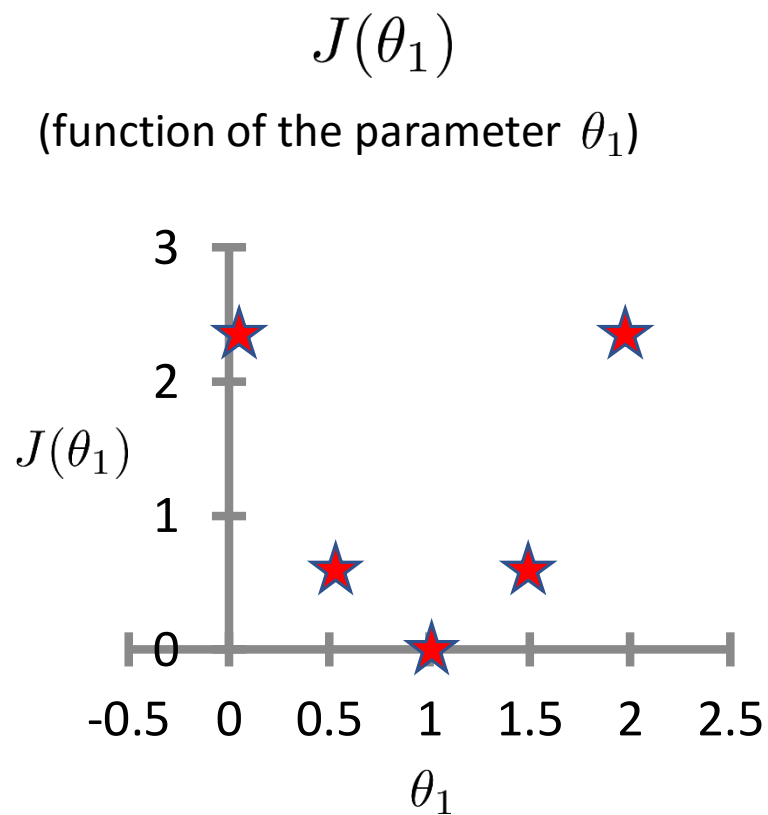


$$J(\theta_1)$$

(function of the parameter θ_1)



代价函数(Cost Function)

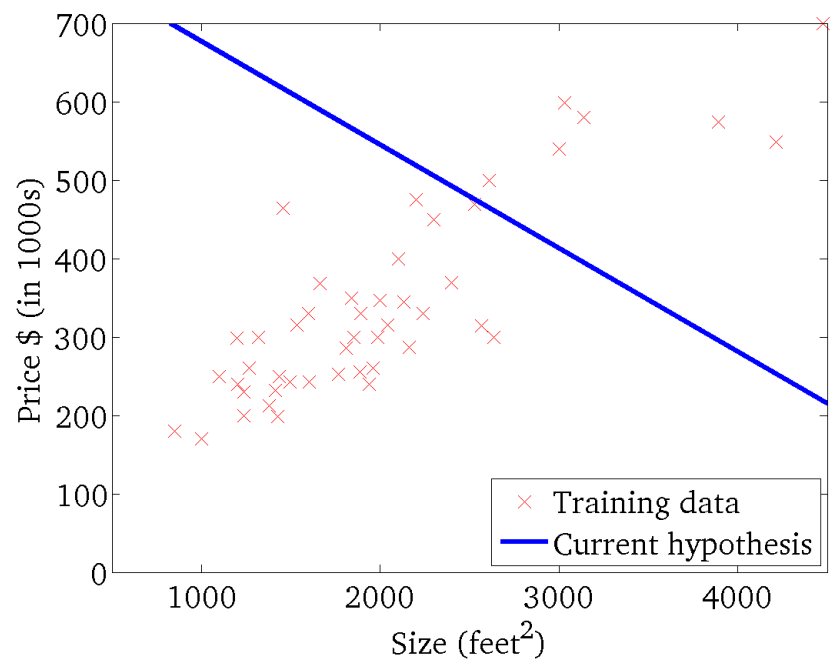


代价函数(Cost Function)



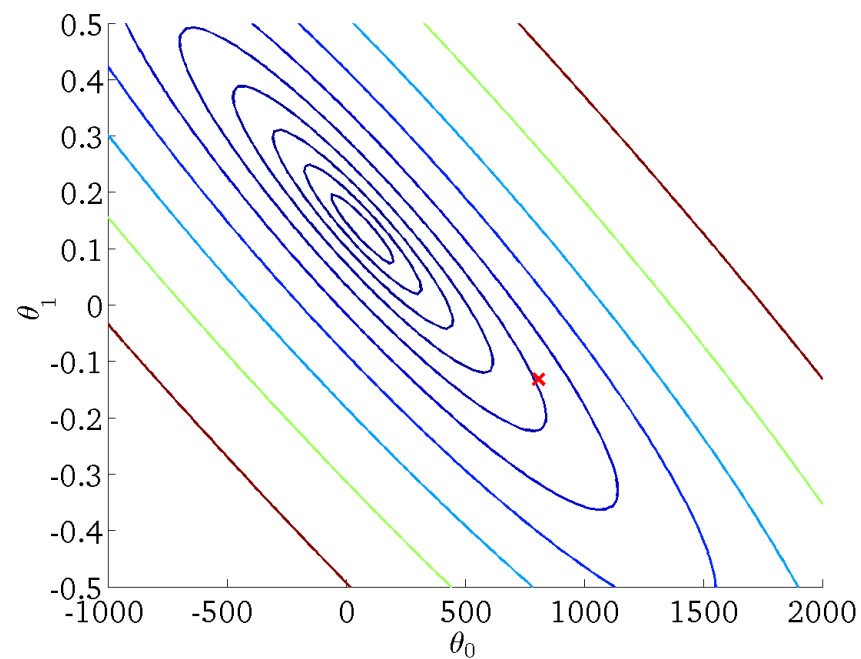
$$h_{\theta}(x)$$

(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)

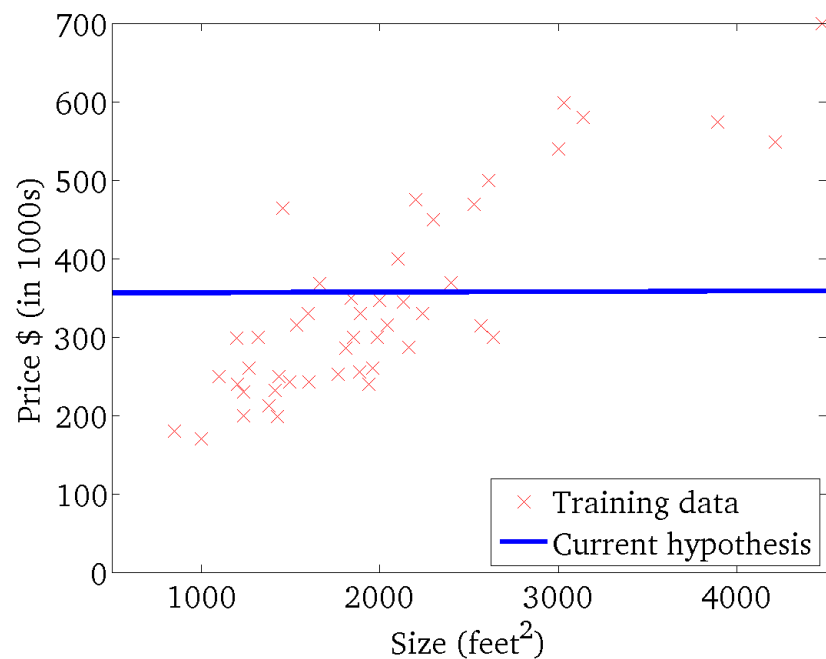


代价函数(Cost Function)



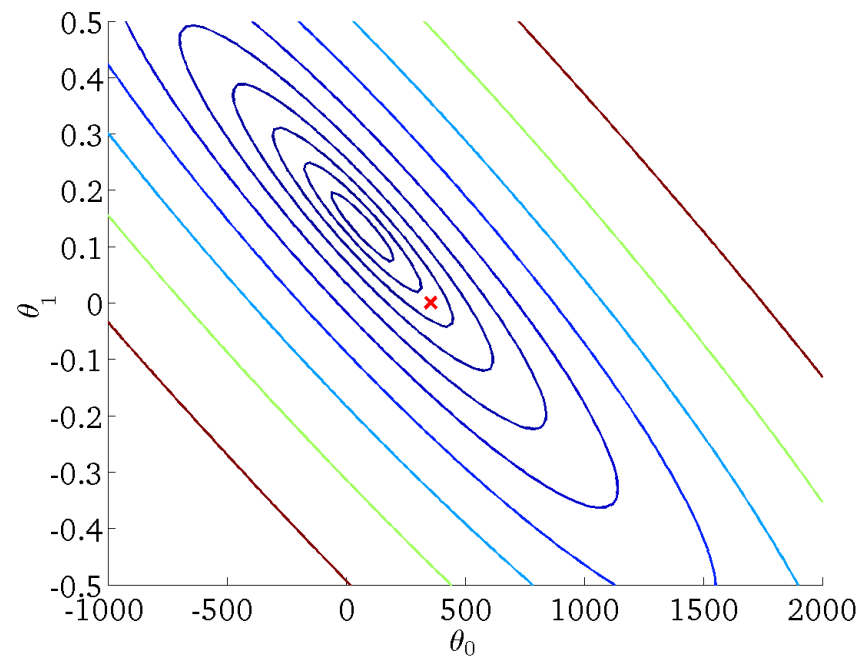
$$h_{\theta}(x)$$

(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)

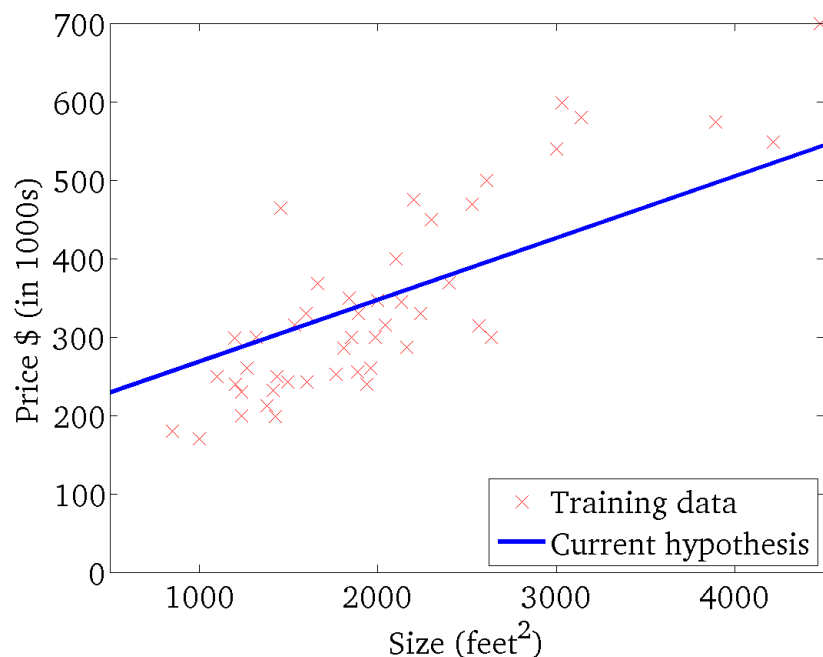


代价函数(Cost Function)



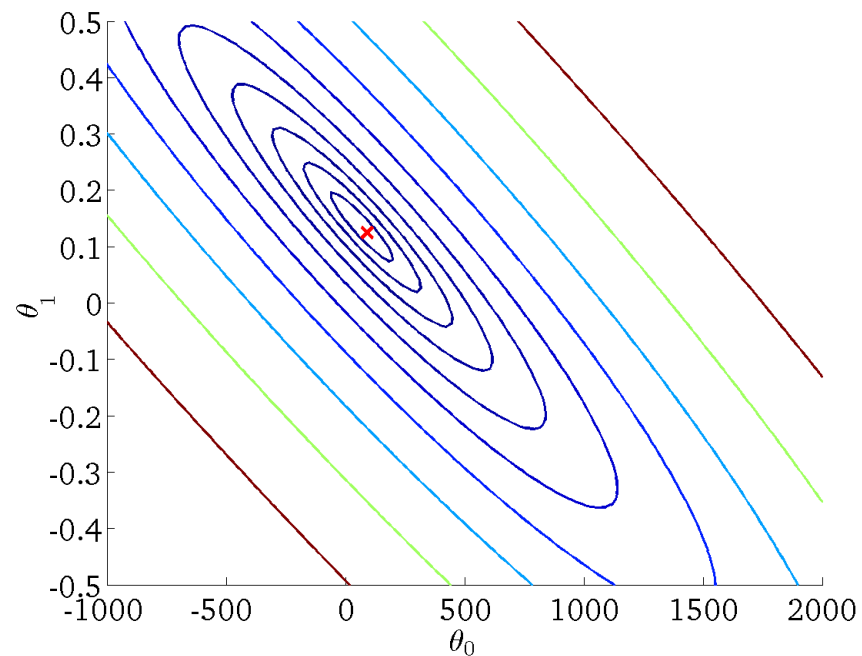
$$h_{\theta}(x)$$

(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)

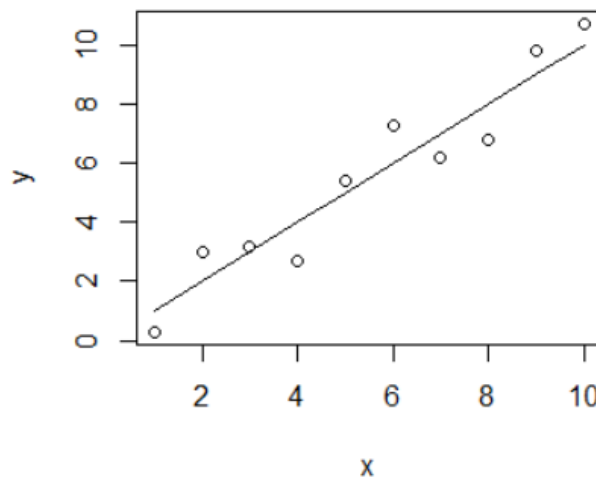
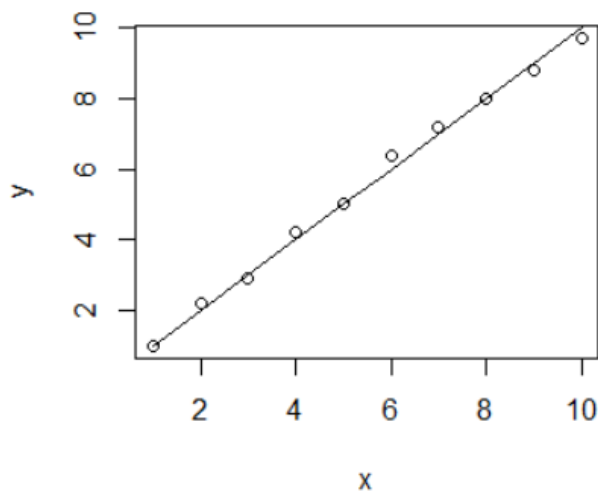


相关系数



我们使用相关系数去衡量线性相关性的强弱

$$r_{XY} = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y_i - \bar{Y})^2}}$$



通过计算，左图的相关系数为0.993，右图的相关系数为0.957



相关系数 R^2 (coefficient of determination)是用来描述两个变量之间的线性关系的，但决定系数的适用范围更广，可以用于描述非线性或者有两个及两个以上自变量的相关关系。它可以用来评价模型的效果。

总平方和(SST)： $\sum_{i=1}^n (y_i - \bar{y})^2$

回归平方和(SSR)： $\sum_{i=1}^n (\hat{y} - \bar{y})^2$

残差平方和(SSE)： $\sum_{i=1}^n (y_i - \hat{y})^2$

它们三者的关系是： $SST = SSR + SSE$

决定系数： $R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$

梯度下降法

Gradient Descent

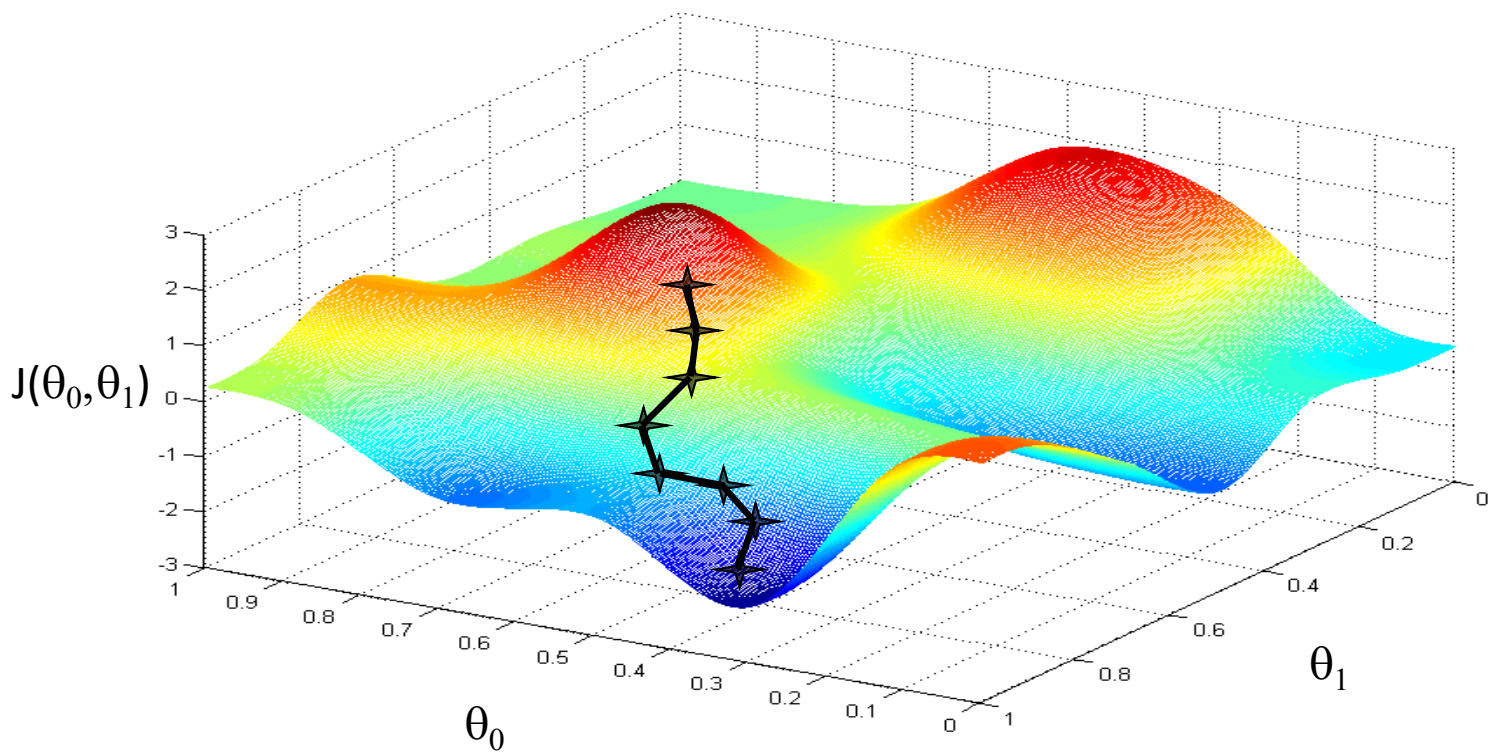


Have some function $J(\theta_0, \theta_1)$

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

- 初始化 θ_0, θ_1
- 不断改变 θ_0, θ_1 , 直到 $J(\theta_0, \theta_1)$ 到达一个全局最小值, 或局部极小值。

梯度下降法



梯度下降法



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

\downarrow
学习率

正确做法：同步更新

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

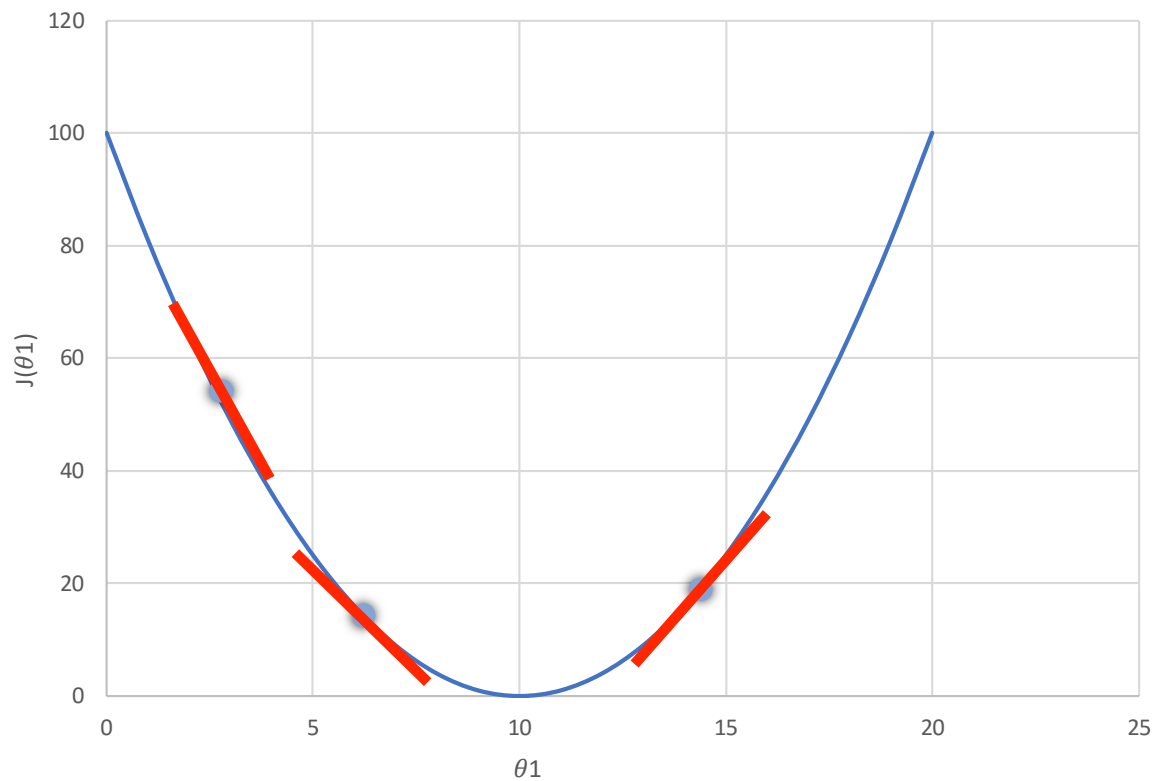
不正确做法

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


梯度下降法



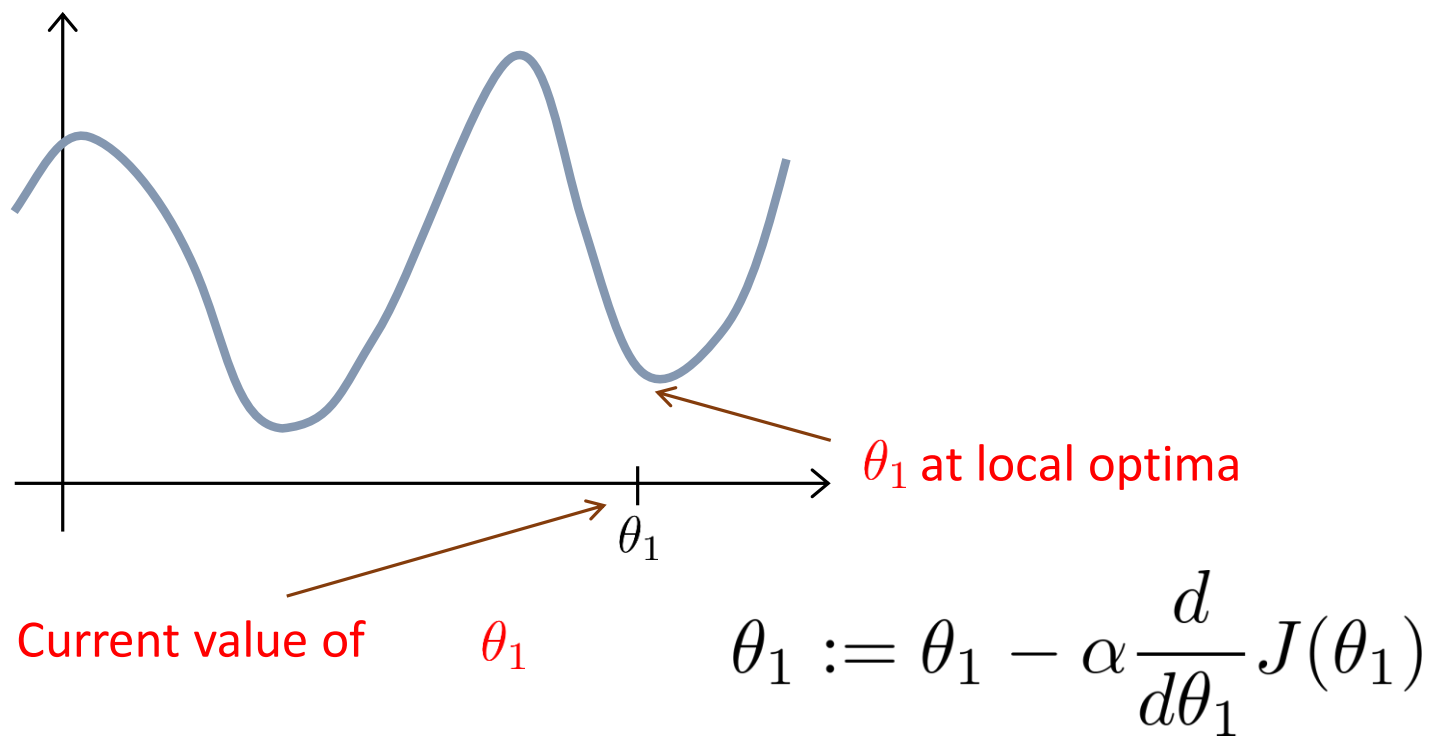
$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$



梯度下降法



有可能会陷入局部极小值



用梯度下降法来求解线性回归



梯度下降法

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

线性回归的模型和代价函数

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

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

用梯度下降法来求解线性回归



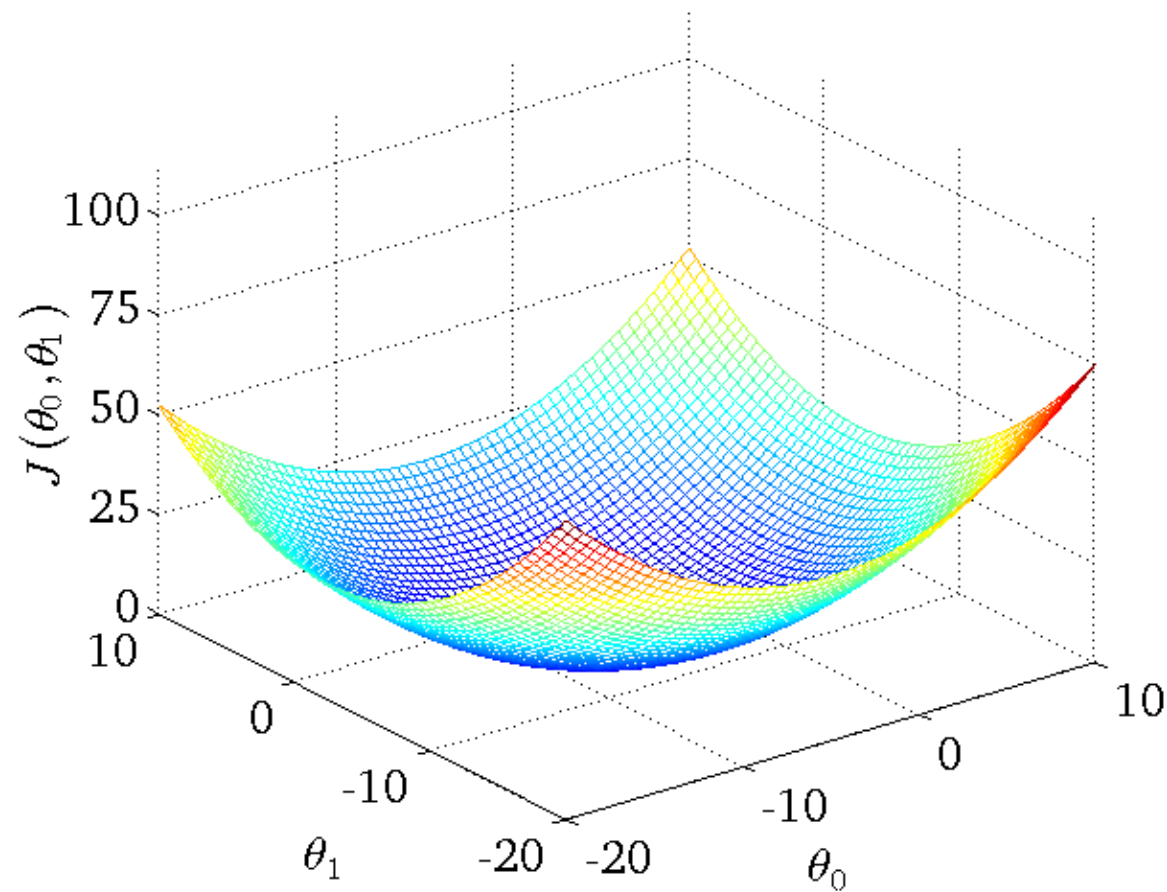
$$\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \begin{aligned} j = 0 : & \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \\ j = 1 : & \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \cdot x^{(i)} \end{aligned}$$

对 $h_{\theta}(x)$ 用 θ_1 求导得到的 x

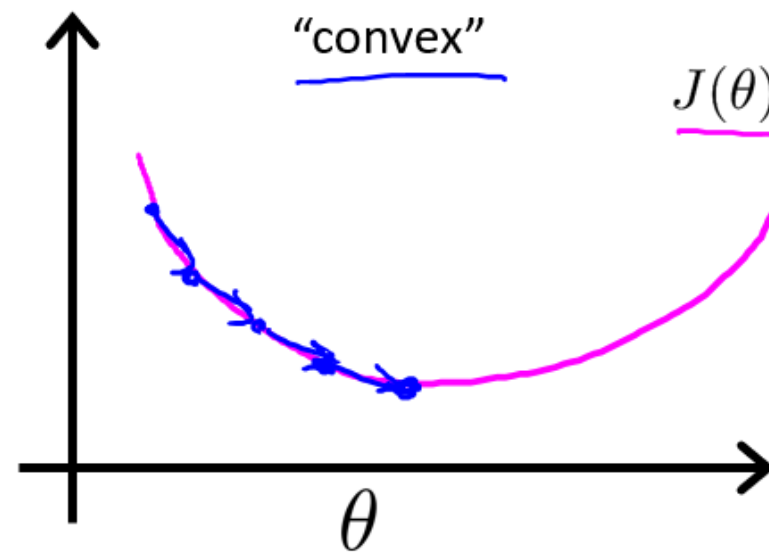
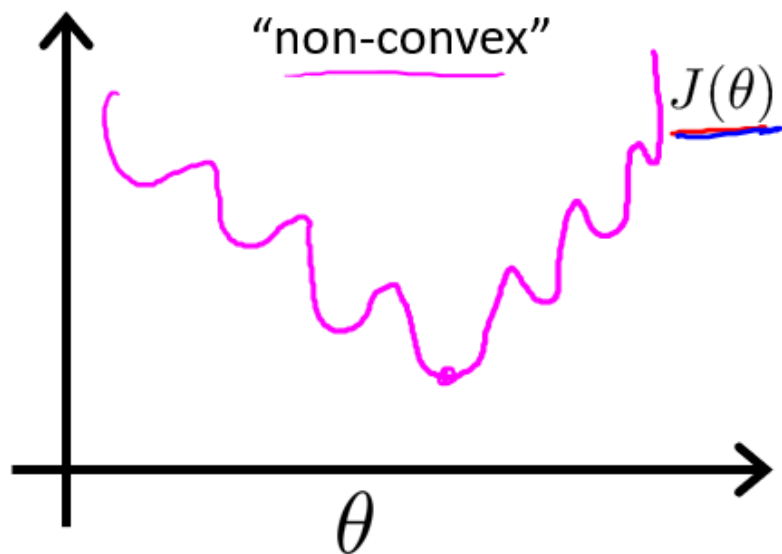
这里是两个参数才是 x
若三个参数应该也是 x
不过
第三个, 也就是 θ_2 , 得到的
应该是 x^2

repeat until convergence {
 $\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})$
 $\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) \cdot x^{(i)}$
}

线性回归的代价函数是凸函数



非凸函数和凸函数

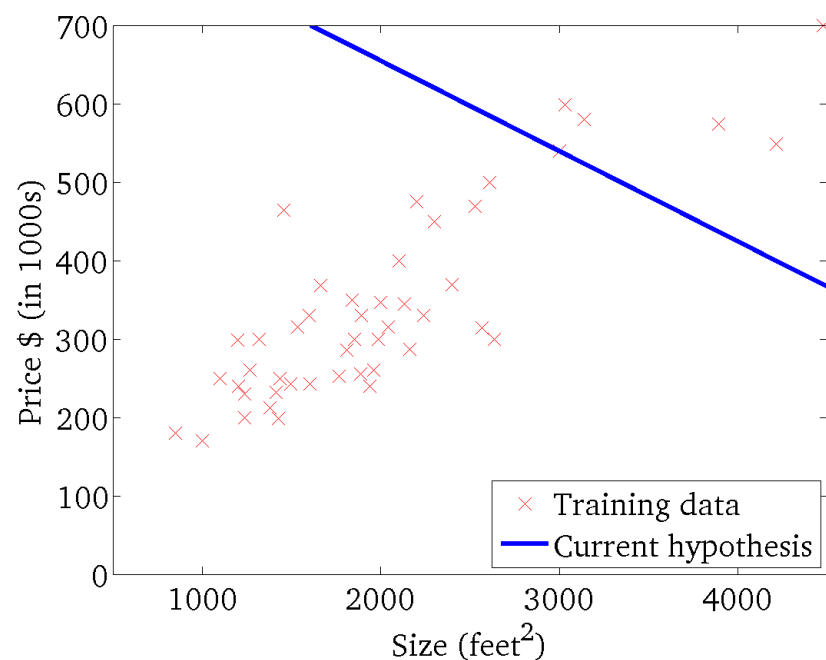


梯度下降法优化过程



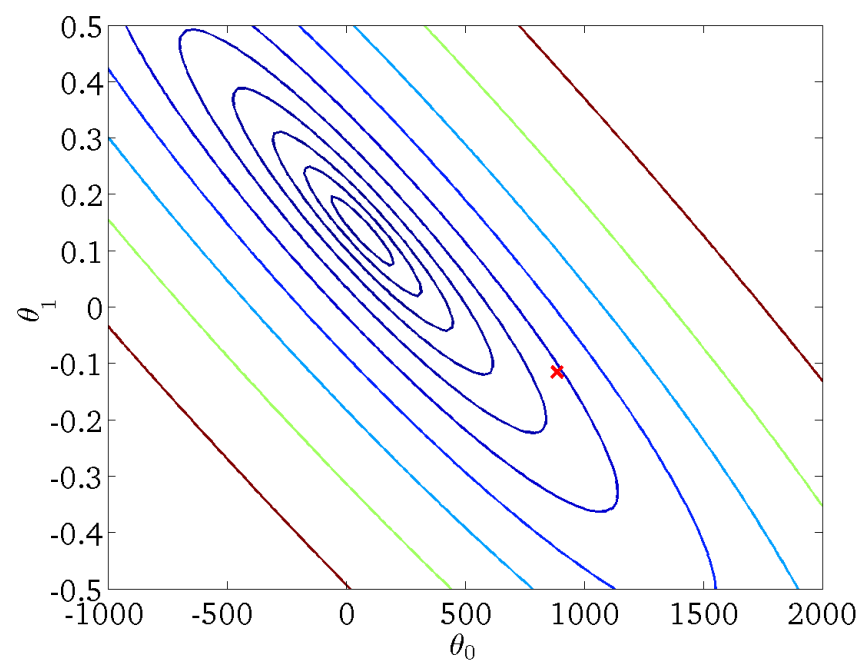
$$h_{\theta}(x)$$

(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)

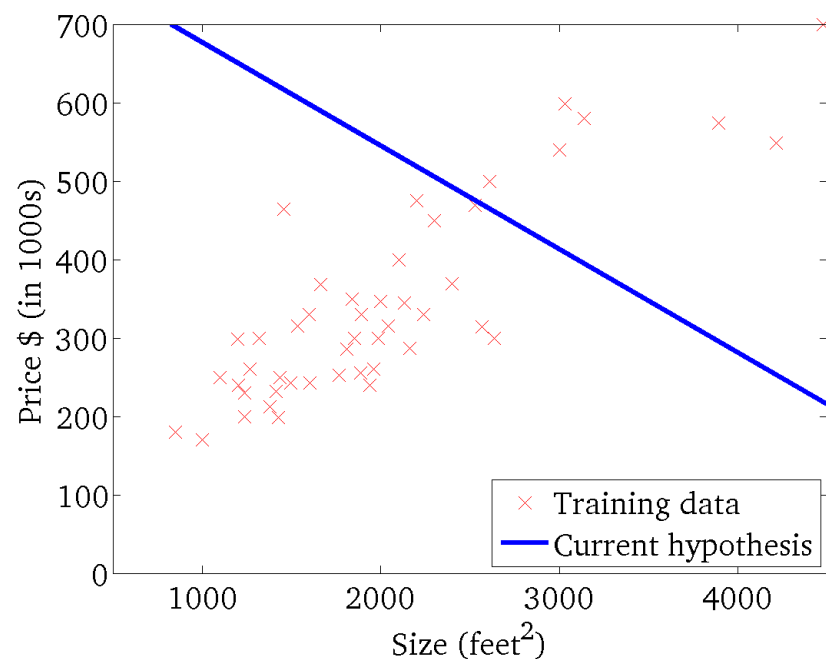


梯度下降法优化过程



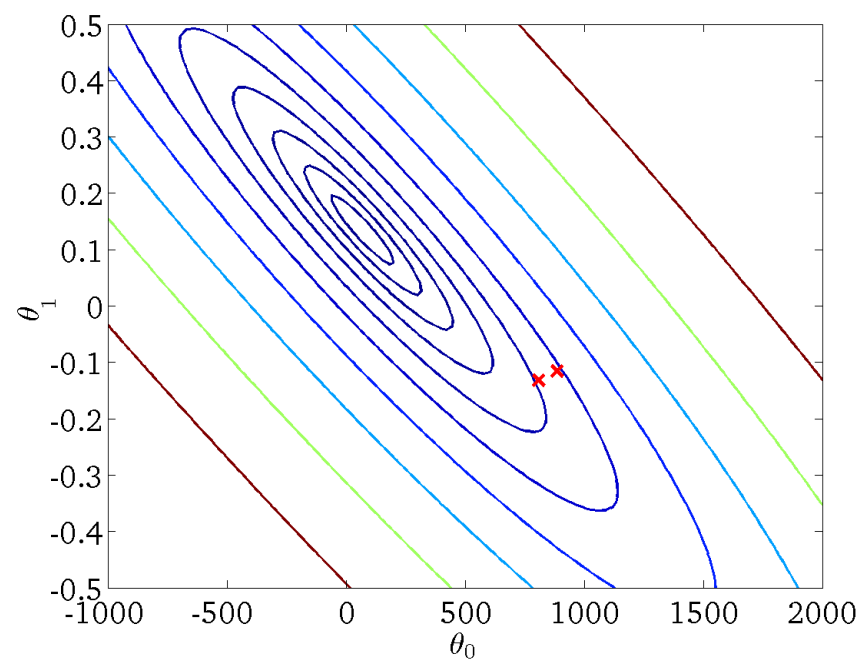
$$h_{\theta}(x)$$

(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)

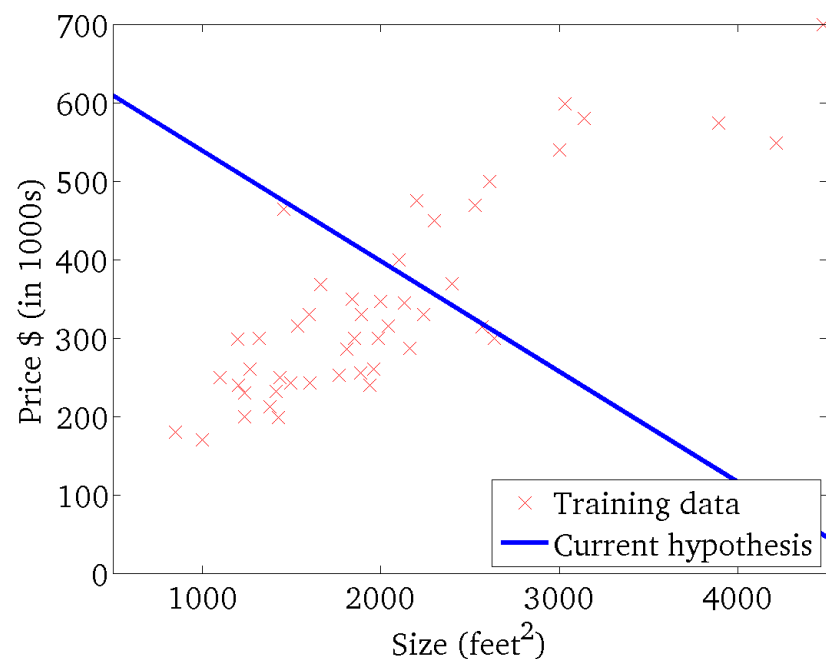


梯度下降法优化过程



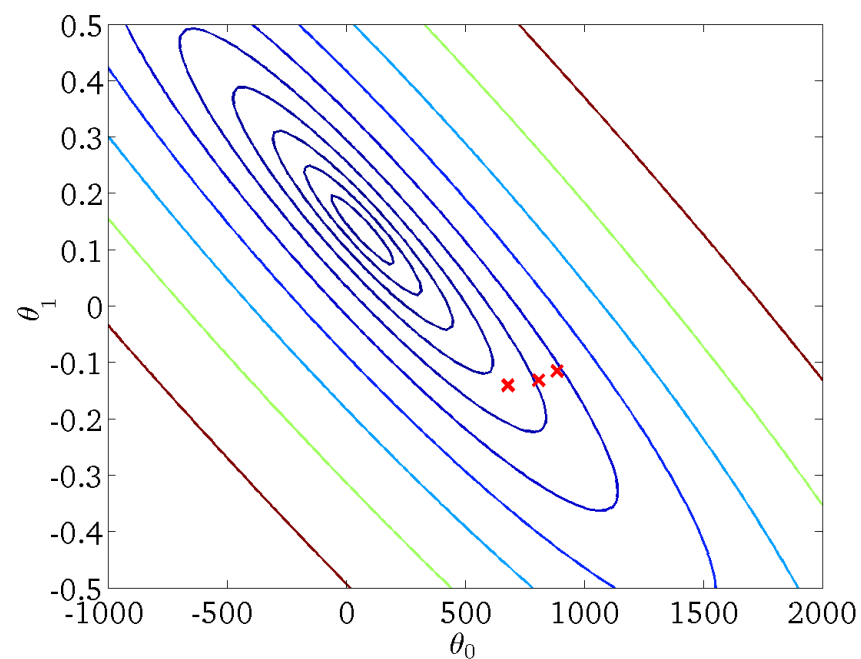
$$h_{\theta}(x)$$

(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)

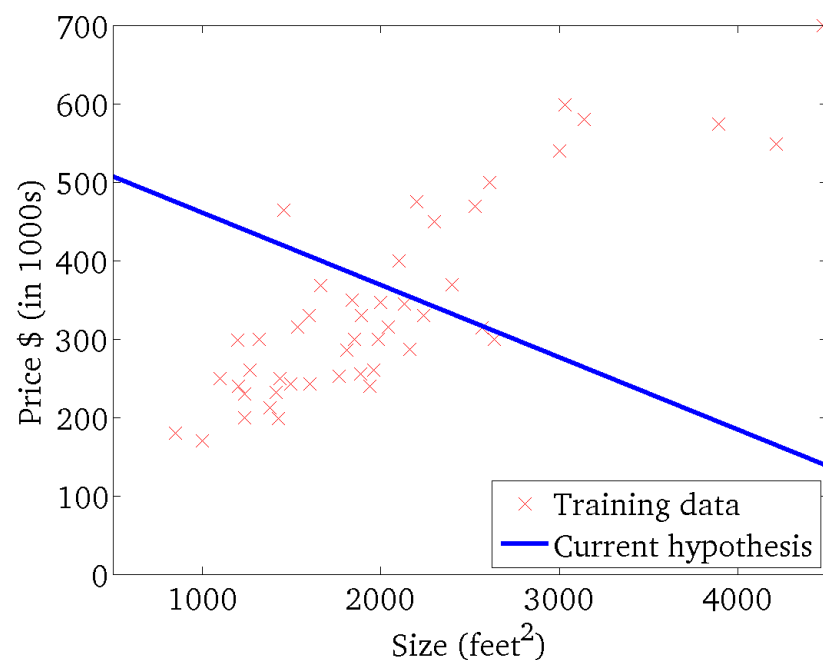


梯度下降法优化过程



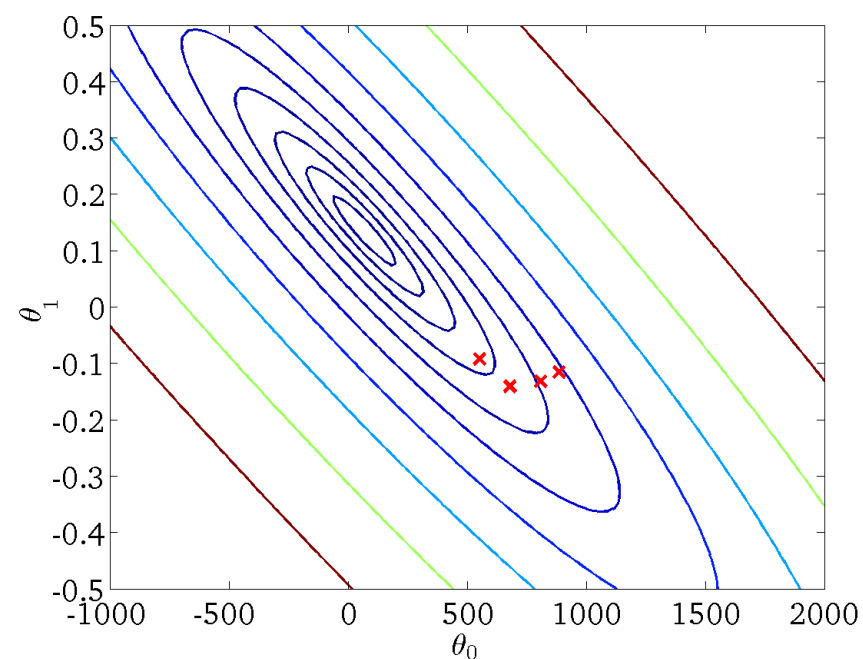
$$h_{\theta}(x)$$

(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)

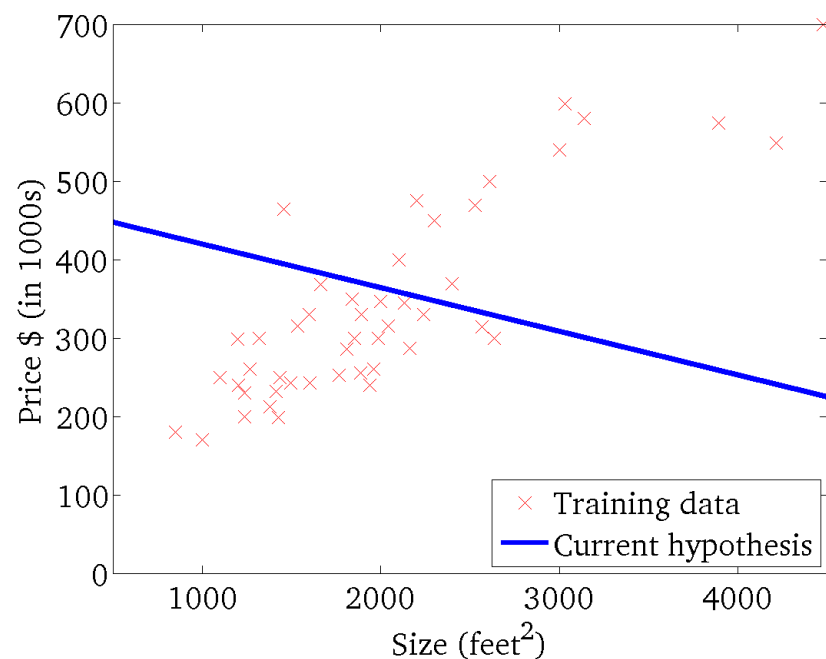


梯度下降法优化过程



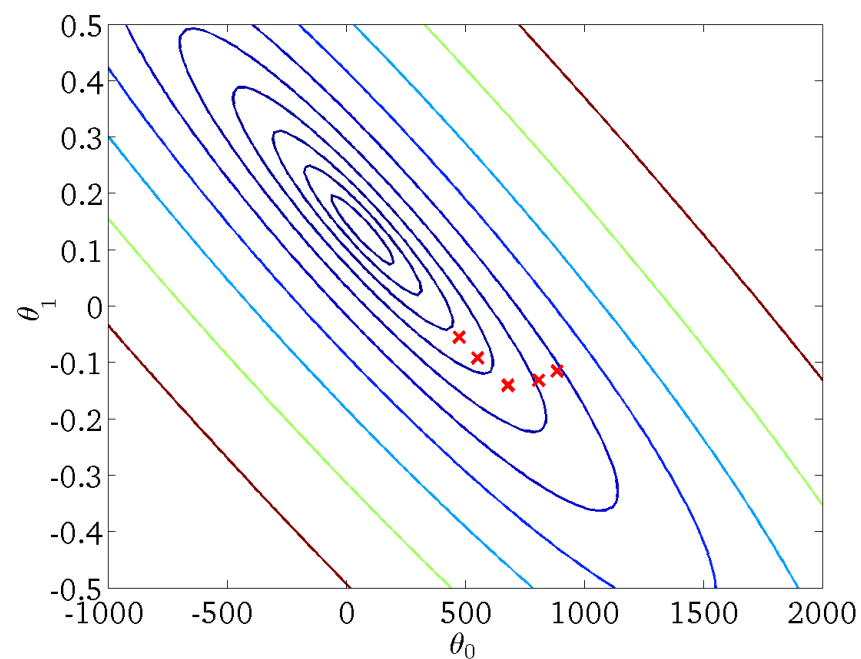
$$h_{\theta}(x)$$

(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)

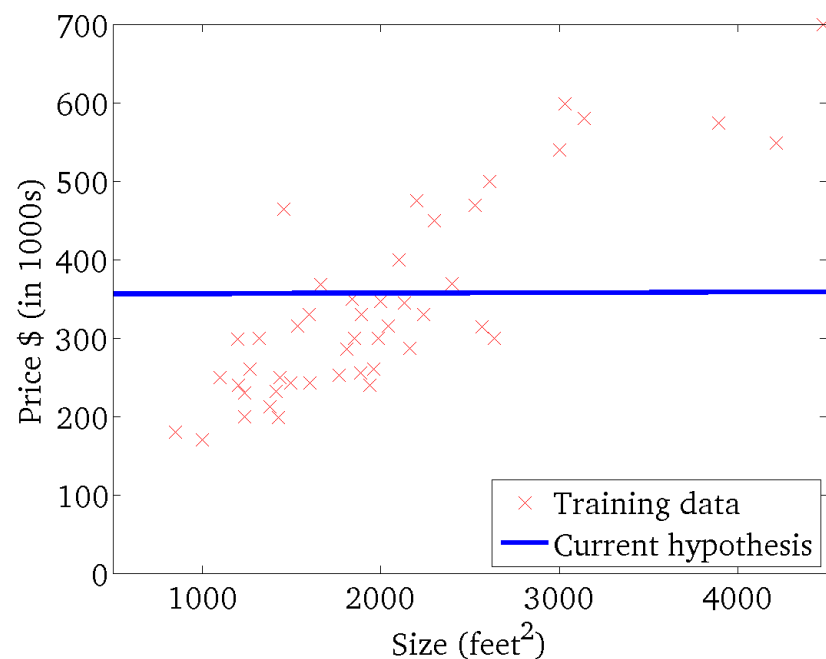


梯度下降法优化过程



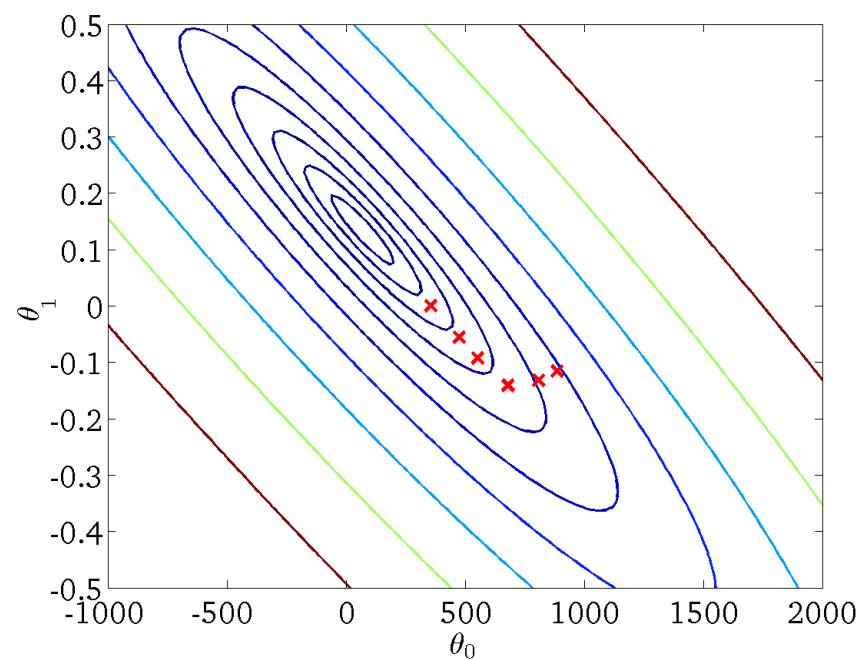
$$h_{\theta}(x)$$

(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)

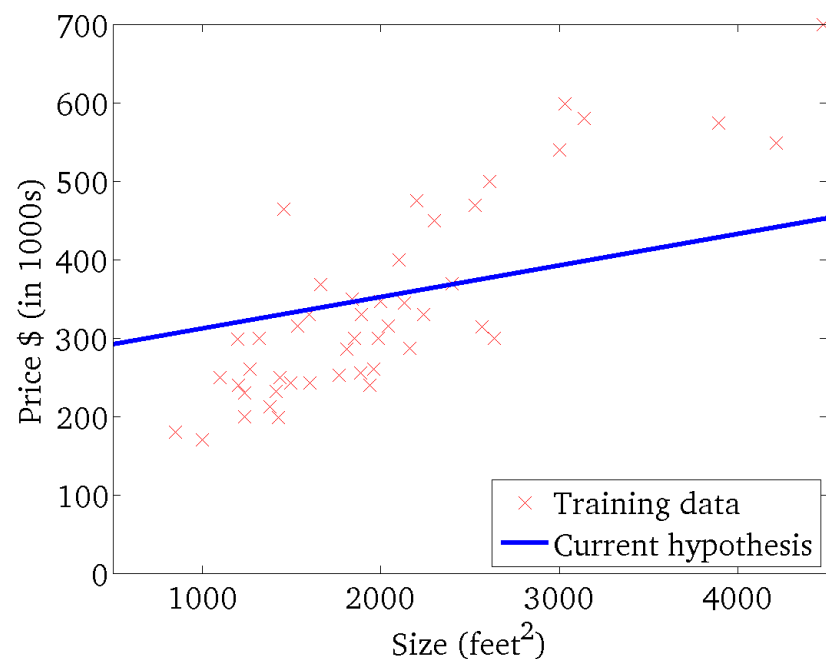


梯度下降法优化过程



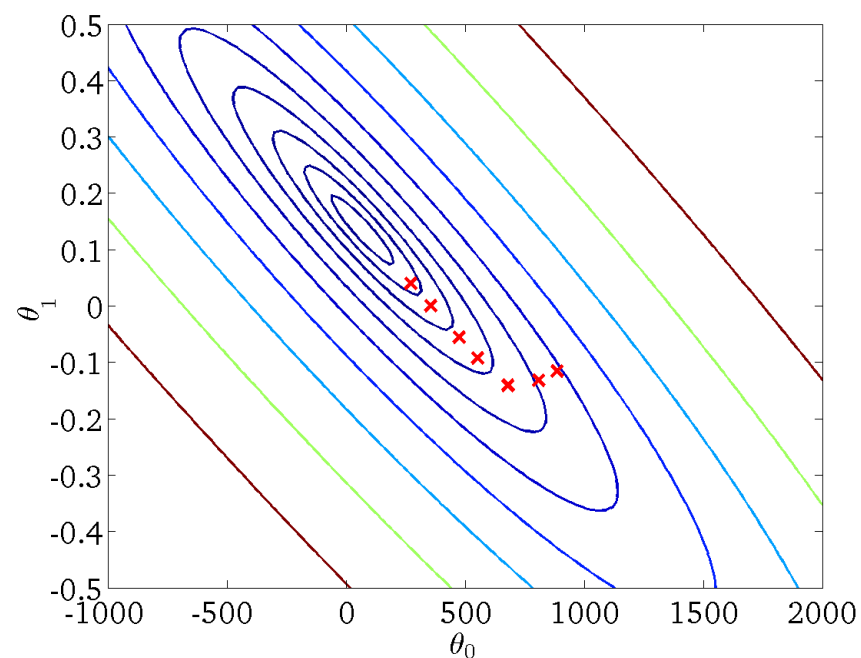
$$h_{\theta}(x)$$

(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)

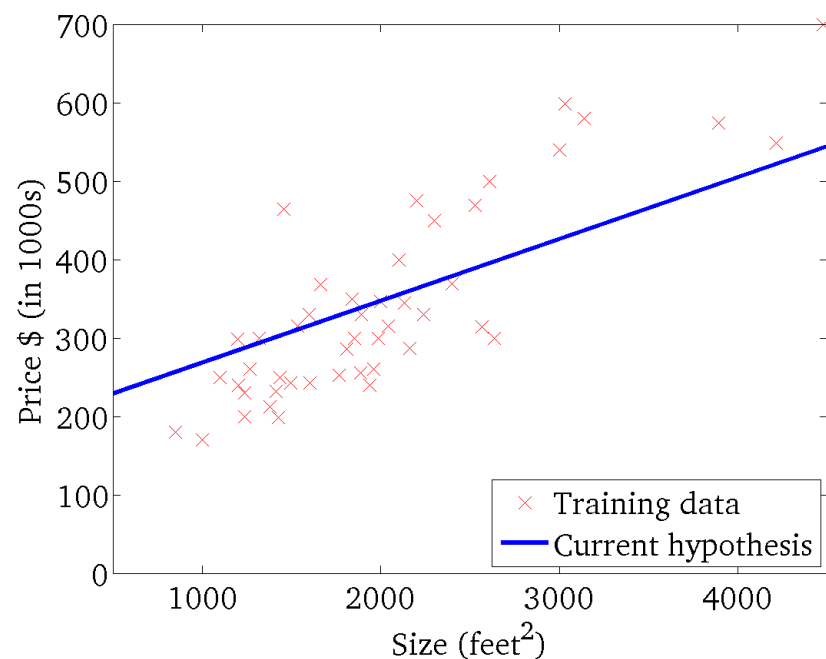


梯度下降法优化过程



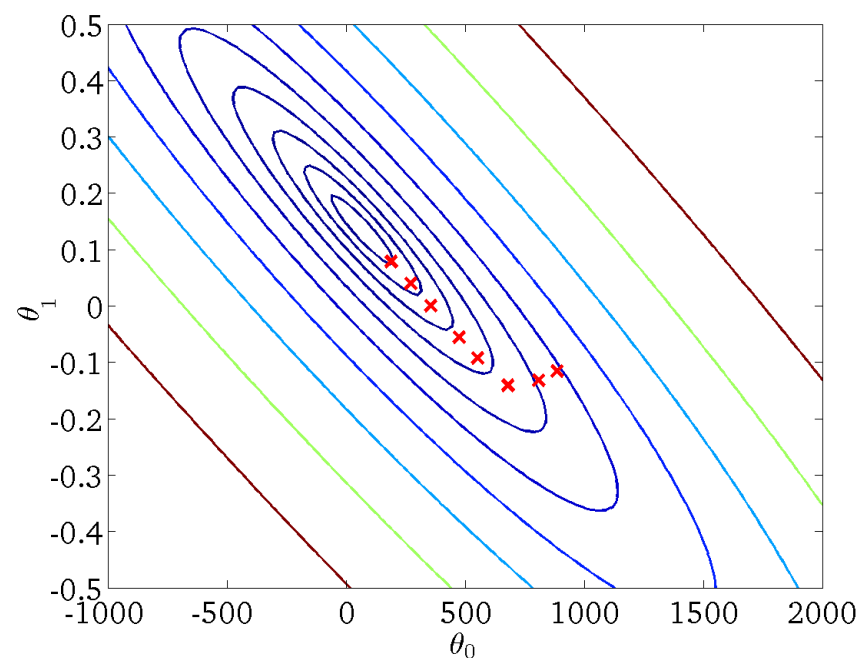
$$h_{\theta}(x)$$

(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)

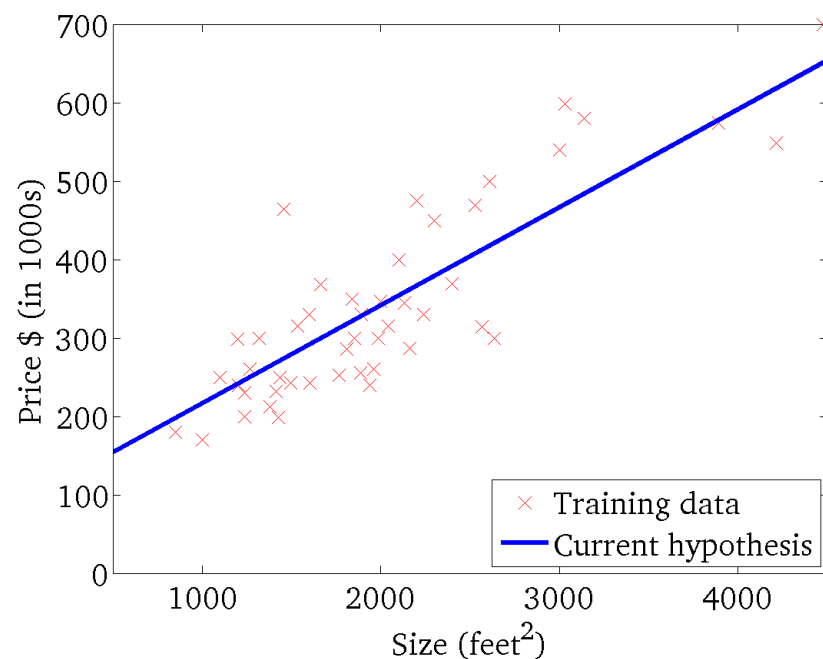


梯度下降法优化过程



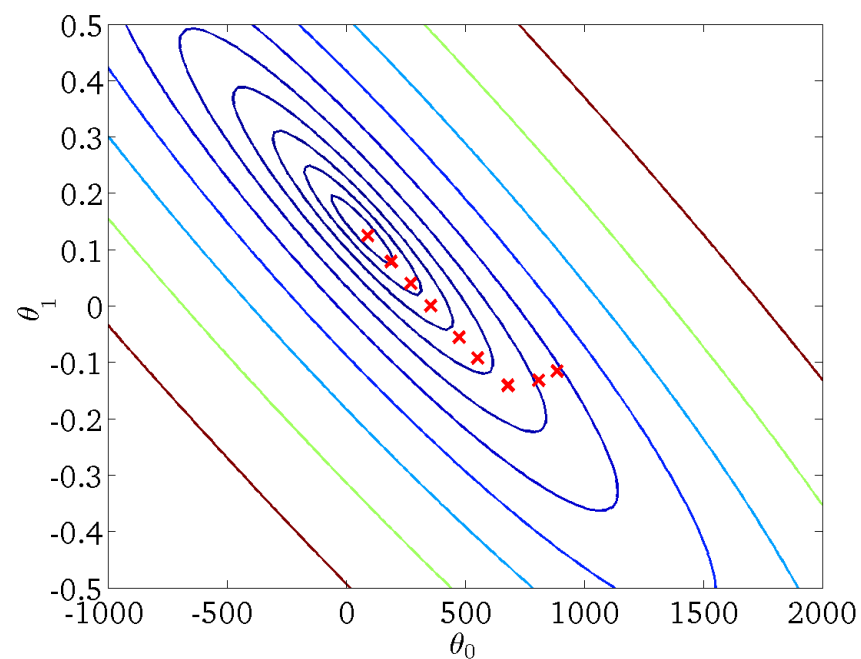
$$h_{\theta}(x)$$

(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)



矩阵运算



3行2列的矩阵：

$$A = \begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix}$$

$$\begin{aligned} A_{11} &= 6, A[1,1] = 6 \\ A_{20} &= 4, A[2,0] = 4 \\ A[0] &= [2, 3] \end{aligned}$$

2行3列的矩阵：

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$\begin{aligned} B_{02} &= 3, B[0,2] = 3 \\ B_{11} &= 5, B[1,1] = 5 \\ B[0] &= [1, 2, 3] \end{aligned}$$



正确的按位加减乘除，
两个矩阵的形状要一致：

$$\begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} = \begin{bmatrix} 3 & 5 \\ 2 & 10 \\ 9 & 6 \end{bmatrix}$$

形状不一致的两个矩阵
不能按位进行加减乘

除：

$$\begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$



$$3 * \begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} = \begin{bmatrix} 6 & 9 \\ -3 & 18 \\ 12 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} + 3 = \begin{bmatrix} 5 & 6 \\ 2 & 9 \\ 7 & 3 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} / 3 = \begin{bmatrix} 2/3 & 1 \\ -1/3 & 2 \\ 4/3 & 0 \end{bmatrix}$$



n行m列的矩阵乘以m行n列的矩阵得到n行n列的矩阵：

$$\begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} = \begin{bmatrix} 2 \times 1 + 3 \times 4 & 2 \times 2 + 3 \times 5 & 2 \times 3 + 3 \times 6 \\ -1 \times 1 + 6 \times 4 & -1 \times 2 + 6 \times 5 & -1 \times 3 + 6 \times 6 \\ 4 \times 1 + 0 \times 4 & 4 \times 2 + 0 \times 5 & 4 \times 3 + 0 \times 6 \end{bmatrix}$$
$$= \begin{bmatrix} 14 & 19 & 24 \\ 23 & 28 & 33 \\ 4 & 8 & 12 \end{bmatrix}$$



单位矩阵：

$$I_{3 \times 3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A \cdot I = I \cdot A = A$$



$$A = \begin{bmatrix} 2 & 3 \\ -1 & 6 \\ 4 & 0 \end{bmatrix}$$

$$A^T = \begin{bmatrix} 2 & -1 & 4 \\ 3 & 6 & 0 \end{bmatrix}$$

$$A_{ij} = A_{ji}^T$$



逆矩阵特点：

$$AA^{-1} = A^{-1}A = I$$

举例：

$$\begin{bmatrix} 3 & 4 \\ 2 & 16 \end{bmatrix} \begin{bmatrix} 0.4 & -0.1 \\ -0.05 & 0.075 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

多元线性回归



Size (feet ²)	Price (\$1000)
2104	460
1416	232
1534	315
852	178
...	...

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



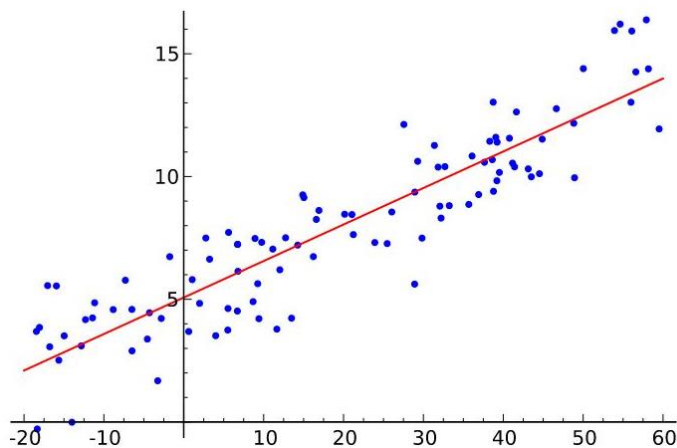
Size (feet ²)	Number of bedrooms	Number of floors	Age of home (years)	Price (\$1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
...

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

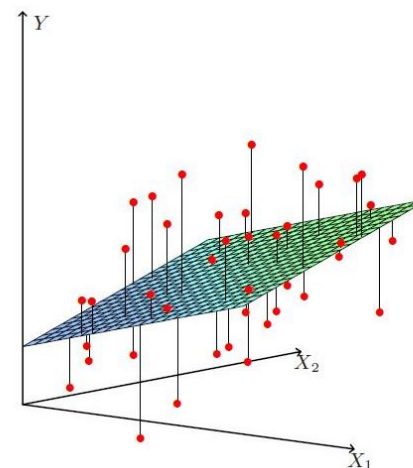


当Y值的影响因素不是唯一时，采用多元线性回归模型

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$



一元线性回归



二元线性回归



Hypothesis: $h_{\theta}(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

Parameters: $\theta_0, \theta_1, \dots, \theta_n$

Cost function:

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

Gradient descent:

Repeat {

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

} (simultaneously update for every $j = 0, \dots, n$)



Gradient Descent

Previously (n=1):

Repeat {

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

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

(simultaneously update θ_0, θ_1)

}

New algorithm ($n \geq 1$)

Repeat {

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

(simultaneously update θ_j for
 $j = 0, \dots, n$)

}

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

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_1^{(i)}$$

$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_2^{(i)}$$

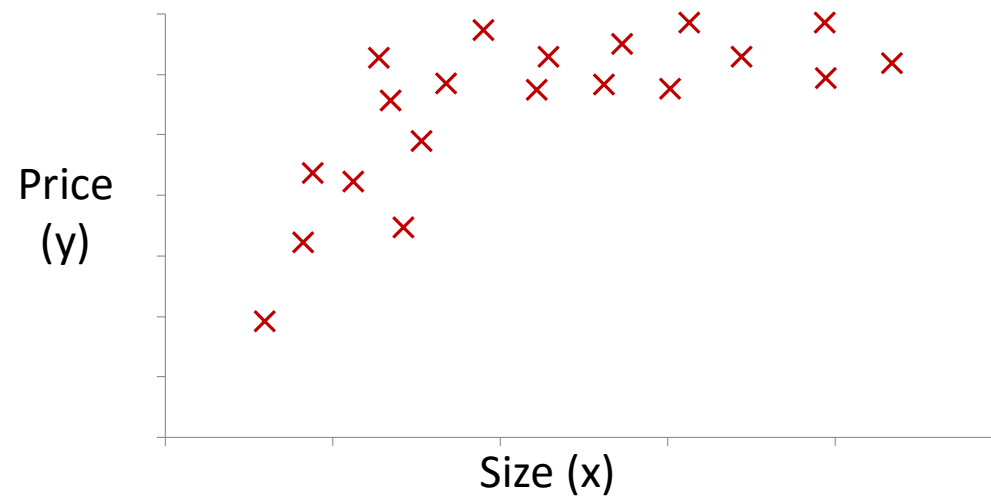


一家快递公司送货：X1：运输里程 X2：
运输次数 Y：总运输时间

Driving Assignment	X1=Miles Traveled	X2=Number of Deliveries	Y= Travel Time (Hours)
1	100	4	9.3
2	50	3	4.8
3	100	4	8.9
4	100	2	6.5
5	50	2	4.2
6	80	2	6.2
7	75	3	7.4
8	65	4	6.0
9	90	3	7.6
10	90	2	6.1

多项式回归

多项式回归

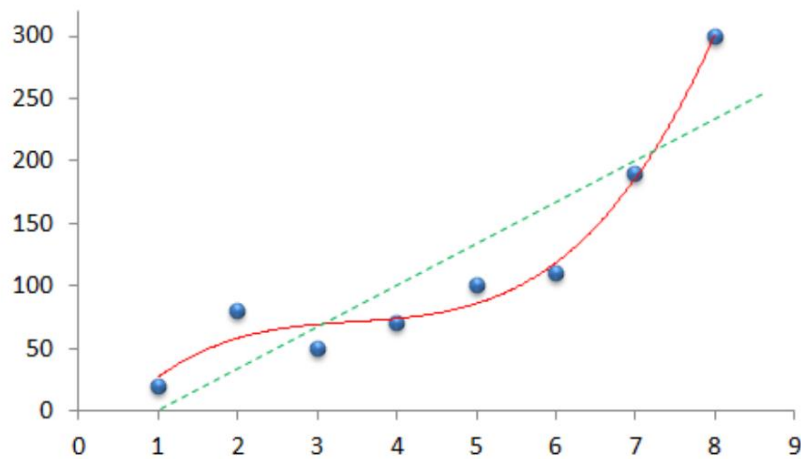


$$\theta_0 + \theta_1 x + \theta_2 x^2$$

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



假如我们不是要找直（或者超平面），而是一个需要找到一个用多项式所表示的曲线（或者超曲面），例如二次曲线：
 $y = at^2 + bt + c$



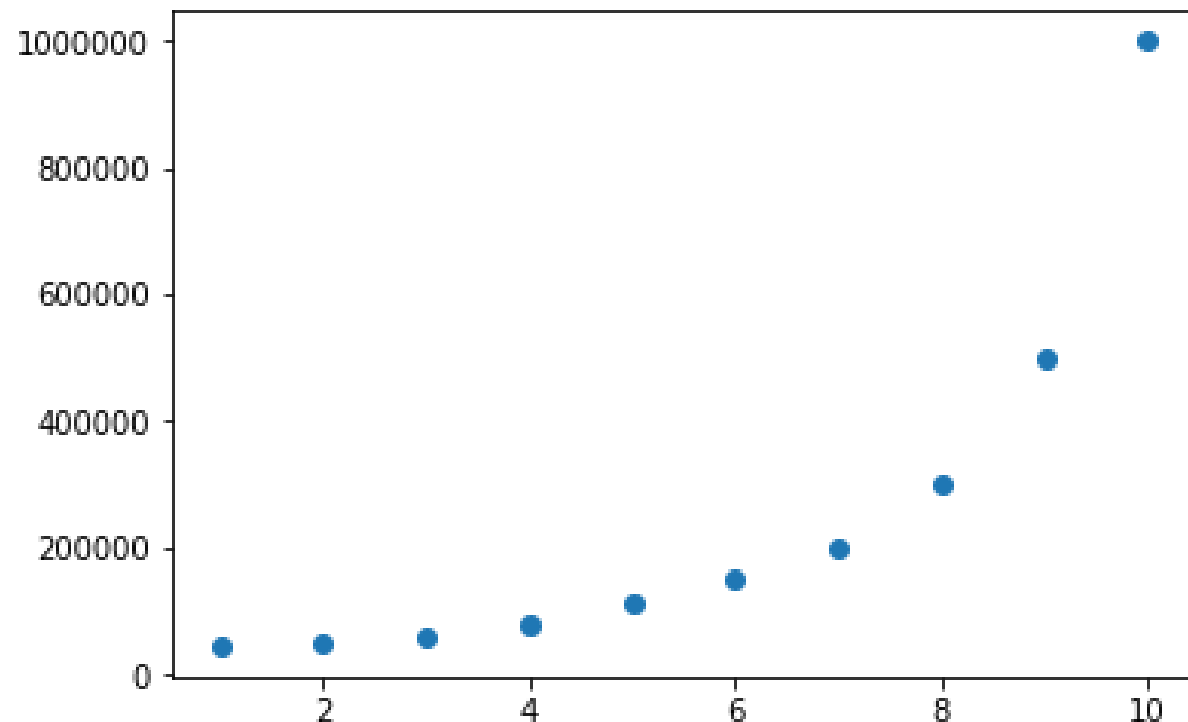
多项式回归可以写成下面这种形式：

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \cdots + \beta_k X_i^k$$

多项式回归



Position	Level	Salary
Business Analyst	1	45000
Junior Consultant	2	50000
Senior Consultant	3	60000
Manager	4	80000
Country Manager	5	110000
Region Manager	6	150000
Partner	7	200000
Senior Partner	8	300000
C-level	9	500000
CEO	10	1000000

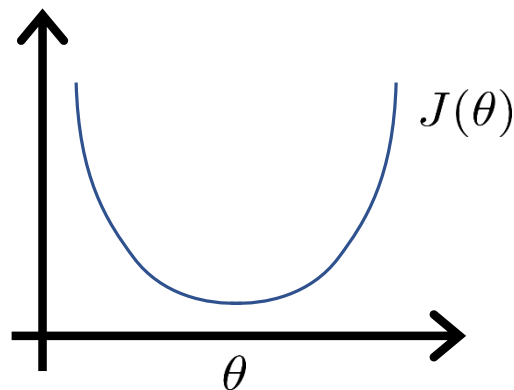


标准方程法

Normal Equation



$$J(\theta) = a\theta^2 + b\theta + c$$



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

$$\frac{\partial}{\partial \theta_j} J(\theta) = \dots = 0$$

求解： $\theta_0, \theta_1, \dots, \theta_n$

标准方程法



	Size (feet ²)	Number of bedrooms	Number of floors	Age of home (years)	Price (\$1000)
x_0	x_1	x_2	x_3	x_4	y
1	2104	5	1	45	460
1	1416	3	2	40	232
1	1534	3	2	30	315
1	852	2	1	36	178

$$X = \begin{matrix} & x_0 & x_1 & x_2 & x_3 & x_4 \\ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} & \begin{bmatrix} 2104 \\ 1416 \\ 1534 \\ 852 \end{bmatrix} & \begin{bmatrix} 5 \\ 3 \\ 3 \\ 2 \end{bmatrix} & \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix} & \begin{bmatrix} 45 \\ 40 \\ 30 \\ 36 \end{bmatrix} \end{matrix} \quad w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$



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

$$X = \begin{matrix} & x_0 & x_1 & x_2 & x_3 & x_4 \\ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} & \begin{bmatrix} 2104 \\ 1416 \\ 1534 \\ 852 \end{bmatrix} & \begin{bmatrix} 5 \\ 3 \\ 3 \\ 2 \end{bmatrix} & \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix} & \begin{bmatrix} 45 \\ 40 \\ 30 \\ 36 \end{bmatrix} \end{matrix} \quad w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$

$$\sum_{i=1}^m (h_w(x^i) - y^i)^2 = (y - Xw)^T (y - Xw)$$



https://en.wikipedia.org/wiki/Matrix_calculus#Scalar-by-vector_identities

分子布局 (Numerator-layout) : 分子为列向量或者分母为行向量)

分母布局 (Denominator-layout) : 分子为行向量或者分母为列向量)

$$\frac{\partial (y - Xw)^T (y - Xw)}{\partial w}$$

$$\frac{\partial (y^T y - y^T Xw - w^T X^T y + w^T X^T Xw)}{\partial w}$$

$$\frac{\partial y^T y}{\partial w} - \frac{\partial y^T Xw}{\partial w} - \frac{\partial w^T X^T y}{\partial w} + \frac{\partial w^T X^T Xw}{\partial w}$$



$$\frac{\partial y^T y}{\partial w} - \frac{\partial y^T X w}{\partial w} - \frac{\partial w^T X^T y}{\partial w} + \frac{\partial w^T X^T X w}{\partial w}$$

$$\textcircled{1} \quad \frac{\partial y^T y}{\partial w} = 0$$

$$\textcircled{2} \quad \frac{\partial y^T X w}{\partial w} = X^T y$$

$$\textcircled{3} \quad \frac{\partial w^T X^T y}{\partial w} = \frac{\partial (w^T X^T y)^T}{\partial w} = \frac{\partial y^T X w}{\partial w} = X^T y$$

$$\textcircled{4} \quad \frac{\partial w^T X^T X w}{\partial w} = 2X^T X w$$



$$\frac{\partial y^T y}{\partial w} - \frac{\partial y^T Xw}{\partial w} - \frac{\partial w^T X^T y}{\partial w} + \frac{\partial w^T X^T Xw}{\partial w} = 0 - X^T y - X^T y + 2X^T Xw$$

$$-2X^T y + 2X^T Xw = 0$$

$$X^T Xw = X^T y$$

$$(X^T X)^{-1} X^T Xw = (X^T X)^{-1} X^T y$$

$$w = (X^T X)^{-1} X^T y$$

$(X^T X)^{-1}$ 是 $X^T X$ 的逆矩阵

矩阵不可逆的情况



1. 线性相关的特征(多重共线性)。
例如：x1为房子的面积，单位是平方英尺
x2为房子的面积，单位是平方米
预测房价
1平方英尺 \approx 0.0929平方米
2. 特征数据太多（样本数 $m \leq$ 特征数量 n ）

梯度下降法VS标准方程法



梯度下降法

缺点：

需要选择合适的学习率
需要迭代很多个周期
只能得到最优解的近似值

优点：

当特征值非常多的时候也
可以很好的工作

标准方程法

优点：

不需要学习率
不需要迭代
可以得到全局最优解

缺点：

需要计算 $(X^T X)^{-1}$
时间复杂度大约是 $O(n^3)$
 n 是特征数量

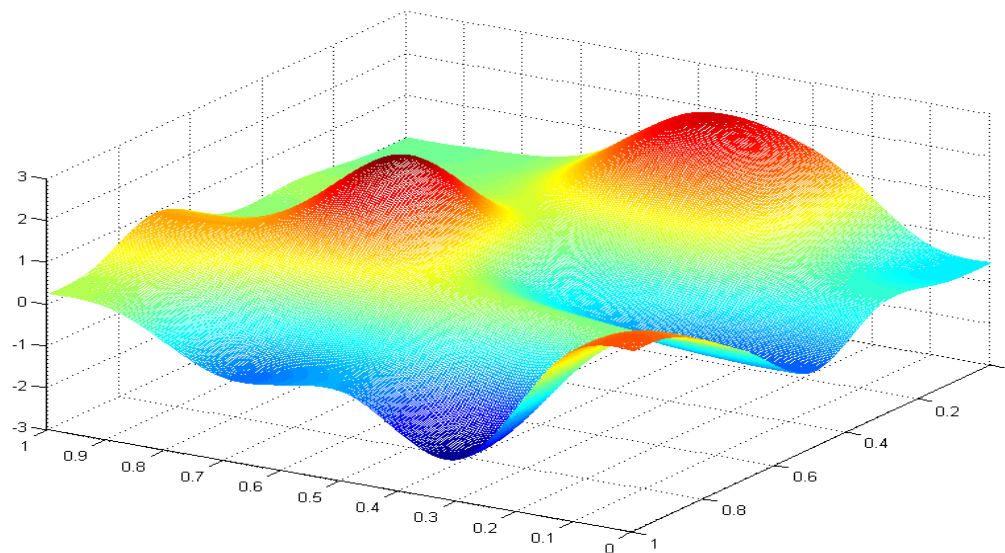
特征缩放 交叉验证法



假设有多个特征用于决定房屋的价格

x_1 = 房子面积(1000000cm^2 - 2000000cm^2)

x_2 = 房间数量(1-5)



数据归一化



数据归一化就是把数据的取值范围处理为0-1或者-1-1之间。

任意数据转化为0-1之间：

$$\text{newValue} = (\text{oldValue} - \text{min}) / (\text{max} - \text{min})$$

(1,3,5,7,9)

$$(1-1)/(9-1)=0$$

$$(3-1)/(9-1)=1/4$$

$$(5-1)/(9-1)=1/2$$

$$(7-1)/(9-1)=3/4$$

$$(9-1)/(9-1)=1$$

任意数据转化为-1-1之间：

$$\text{newValue} = ((\text{oldValue} - \text{min}) / (\text{max} - \text{min}) - 0.5) * 2$$

均值标准化



x为特征数据，u为数据的平均值，s为数据的方差

$$\text{newValue} = (\text{oldValue} - u) / s$$

(1,3,5,7,9)

$$u = (1 + 3 + 5 + 7 + 9) / 5 = 5$$

$$s = ((1-5)^2 + (3-5)^2 + (5-5)^2 + (7-5)^2 + (9-5)^2) / 5 = 8$$

$$(1-5)/\sqrt{8} = -1/\sqrt{2}$$

$$(3-5)/\sqrt{8} = -1/\sqrt{4}$$

$$(5-5)/\sqrt{8} = 0$$

$$(7-5)/\sqrt{8} = 1/\sqrt{4}$$

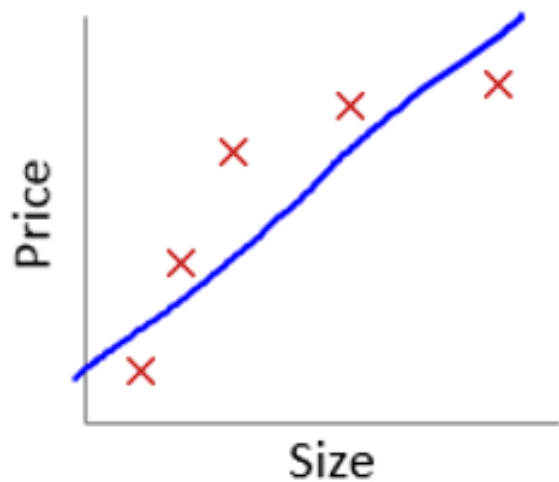
$$(9-5)/\sqrt{8} = 1/\sqrt{2}$$

交叉验证法



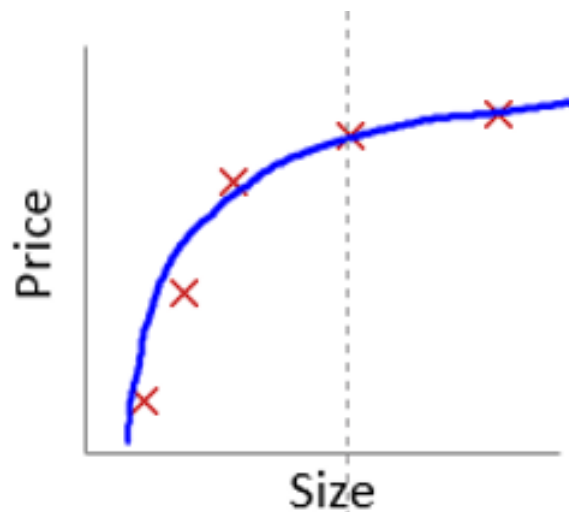
过拟合(Overfitting)
正则化(Regularized)

拟合



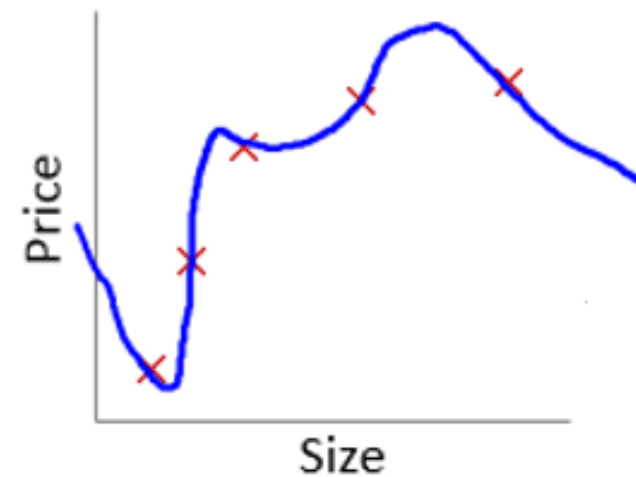
$$\theta_0 + \theta_1 x$$

欠拟合(Underfitting)



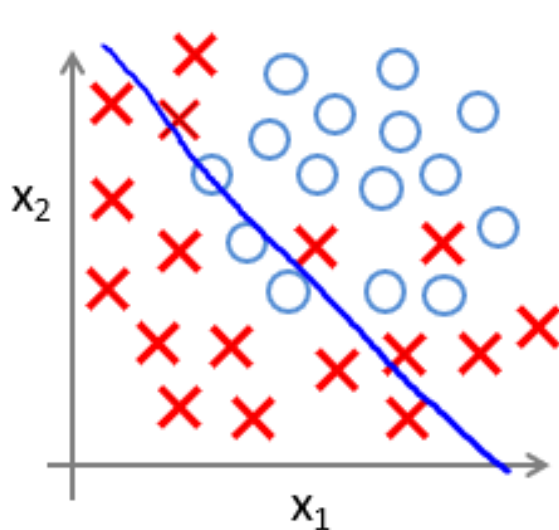
$$\theta_0 + \theta_1 x + \theta_2 x^2$$

正确拟合(Just right)



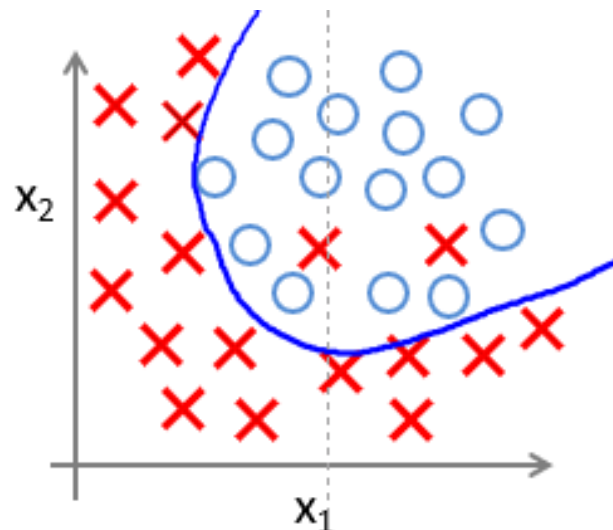
$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

过拟合
(Overfitting)



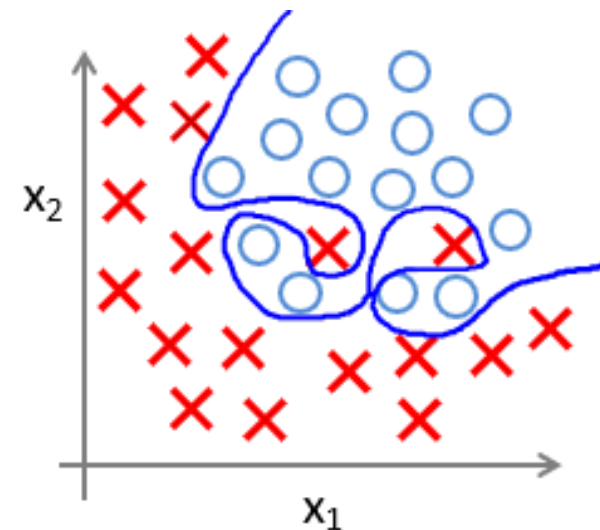
$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

欠拟合(Underfitting)



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

正确拟合(Just right)



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

过拟合
(Overfitting)



- 1.减少特征
- 2.增加数据量
- 3.正则化
(Regularized)



正则化代价函数：

$$\text{L2正则化： } J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2 \right]$$

$$\text{L1正则化： } J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n |\theta_j| \right]$$

岭回归

Ridge Regression

岭回归(Ridge Regression)



$$w = (X^T X)^{-1} X^T y$$

如果数据的特征比样本点还多，数据特征 n ，样本个数 m ，如果 $n > m$ ，则计算 $(X^T X)^{-1}$ 时会出错。因为 $(X^T X)$ 不是满秩矩阵，所以不可逆。

为了解决这个问题，统计学家引入了岭回归的概念。

$$w = (X^T X + \lambda I)^{-1} X^T y$$

λ 为岭系数， I 为单位矩阵(对角线上全为1，其他元素全为0)

岭回归(Ridge Regression)



$$J(\theta) = \frac{1}{2} \sum_{i=1}^n (h_{\theta}(x_i) - y_i)^2 + \lambda \sum_i^n \theta_i^2$$

$$\begin{aligned} J(\theta) &= \frac{1}{2} (X\theta - Y)^T (X\theta - Y) + \lambda \theta^T \theta \\ &= \frac{1}{2} [\theta^T X^T X \theta - \theta^T X^T Y - Y^T X \theta + Y^T Y] + \lambda \theta^T \theta \end{aligned}$$

$$\frac{\partial J(\theta)}{\partial \theta} = X^T X \theta - X^T Y + \lambda \theta$$

$$\theta = (X^T X + \lambda I)^{-1} X^T Y$$

岭回归(Ridge Regression)



岭回归最早是用来处理特征数多于样本的情况，现在也用于在估计中加入偏差，从而得到更好的估计。同时也可以解决多重共线性的问题。岭回归是一种有偏估计。

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

线性回归标准方程法：
$$w = (X^T X)^{-1} X^T y$$

岭回归求解：
$$w = (X^T X + \lambda I)^{-1} X^T y$$

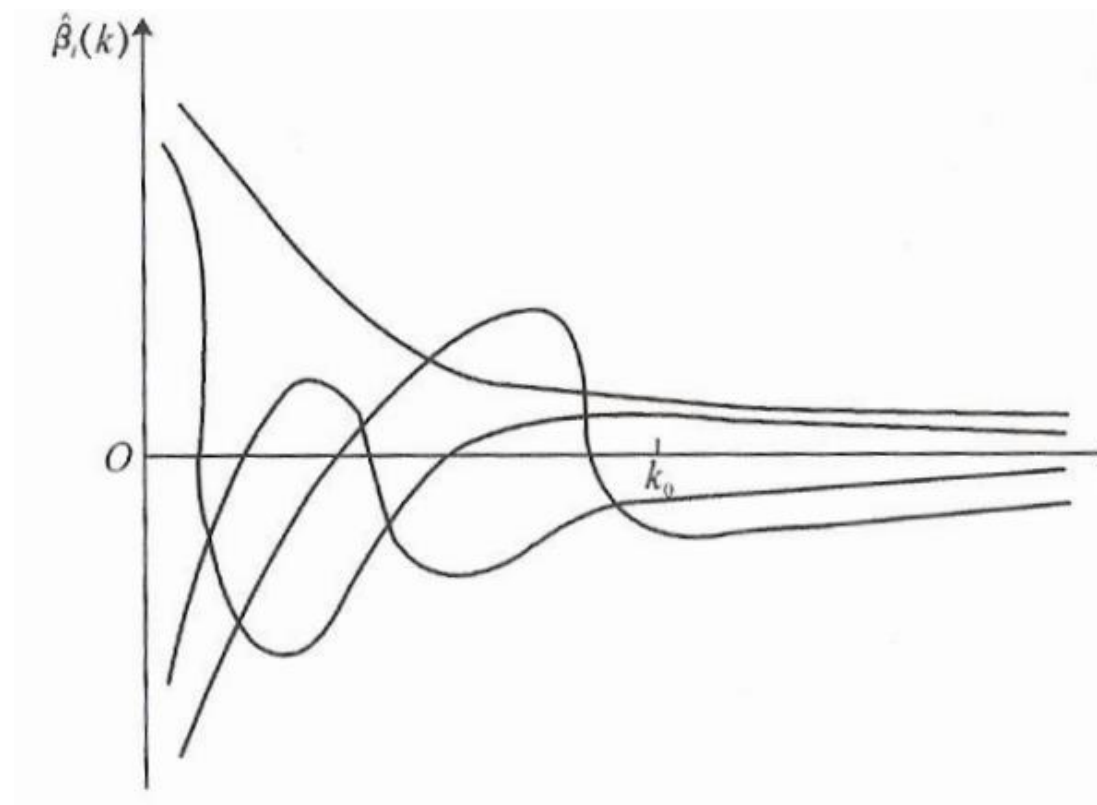
 λ 为岭系数。

岭回归(Ridge Regression)



选择 λ 值，使到：

- 1.各回归系数的岭估计基本稳定。
- 2.残差平方和增大不太多。



LASSO



- Tibshirani(1996)提出了Lasso(The Least Absolute Shrinkage and Selectionator operator)算法。
- 通过构造一个一阶惩罚函数获得一个精炼的模型；通过最终确定一些指标（变量）的系数为零（岭回归估计系数等于0的机会微乎其微，造成筛选变量困难），解释力很强。
- 擅长处理具有多重共线性的数据，与岭回归一样是有偏估计。



岭回归代价函数：

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

λ 的值可以用于限制 $\sum_{j=1}^n \theta_j^2 \leq t$

LASSO代价函数：

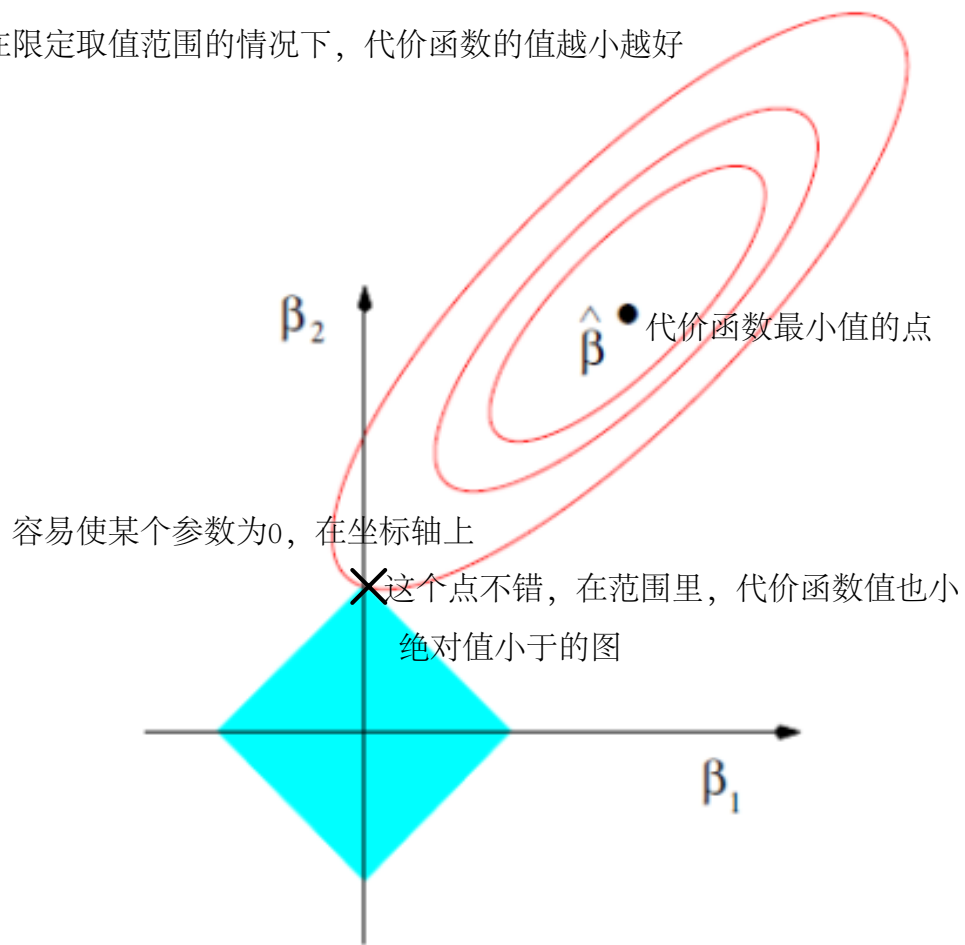
$$J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n |\theta_j| \right]$$

λ 的值可以用于限制 $\sum_{j=1}^n |\theta_j| \leq t$

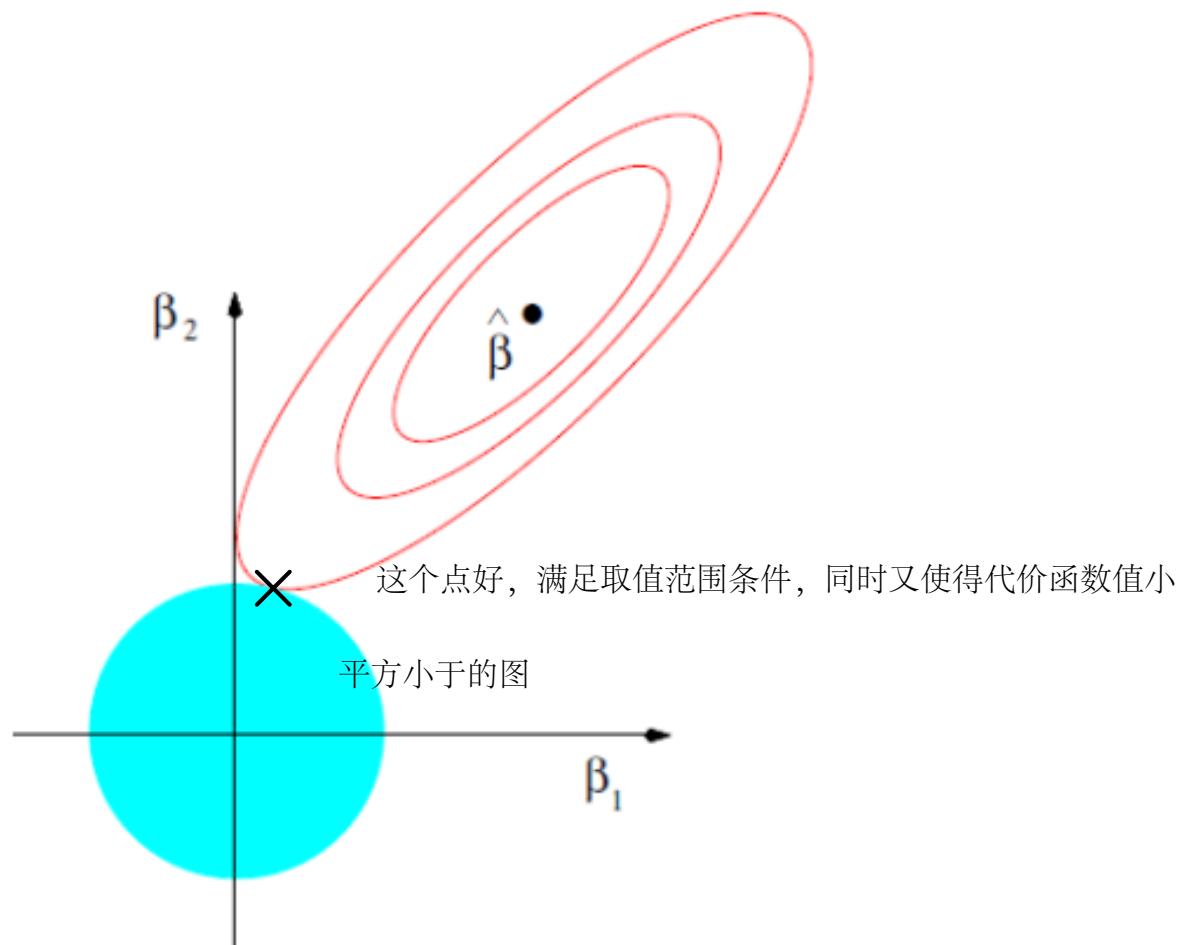
LASSO与岭回归



在限定取值范围的情况下，代价函数的值越小越好

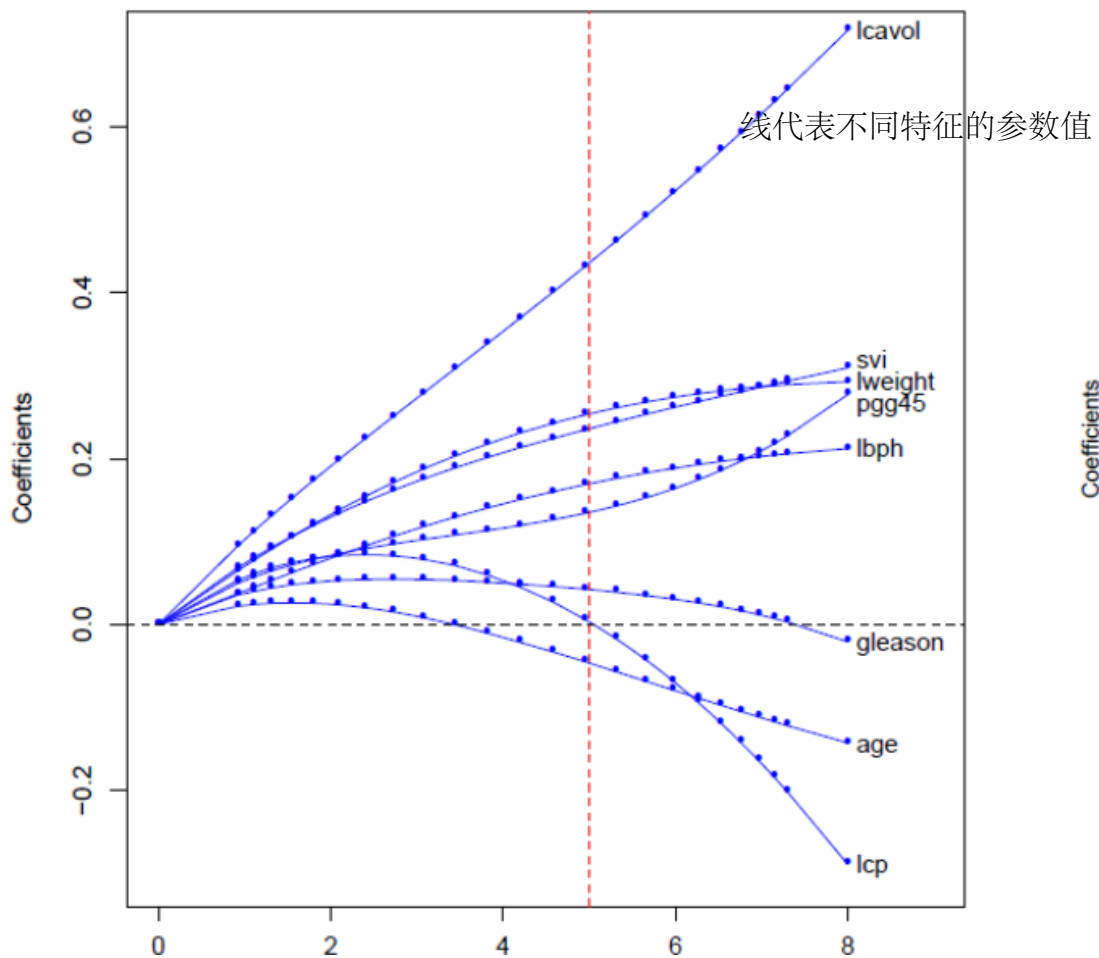


LASSO

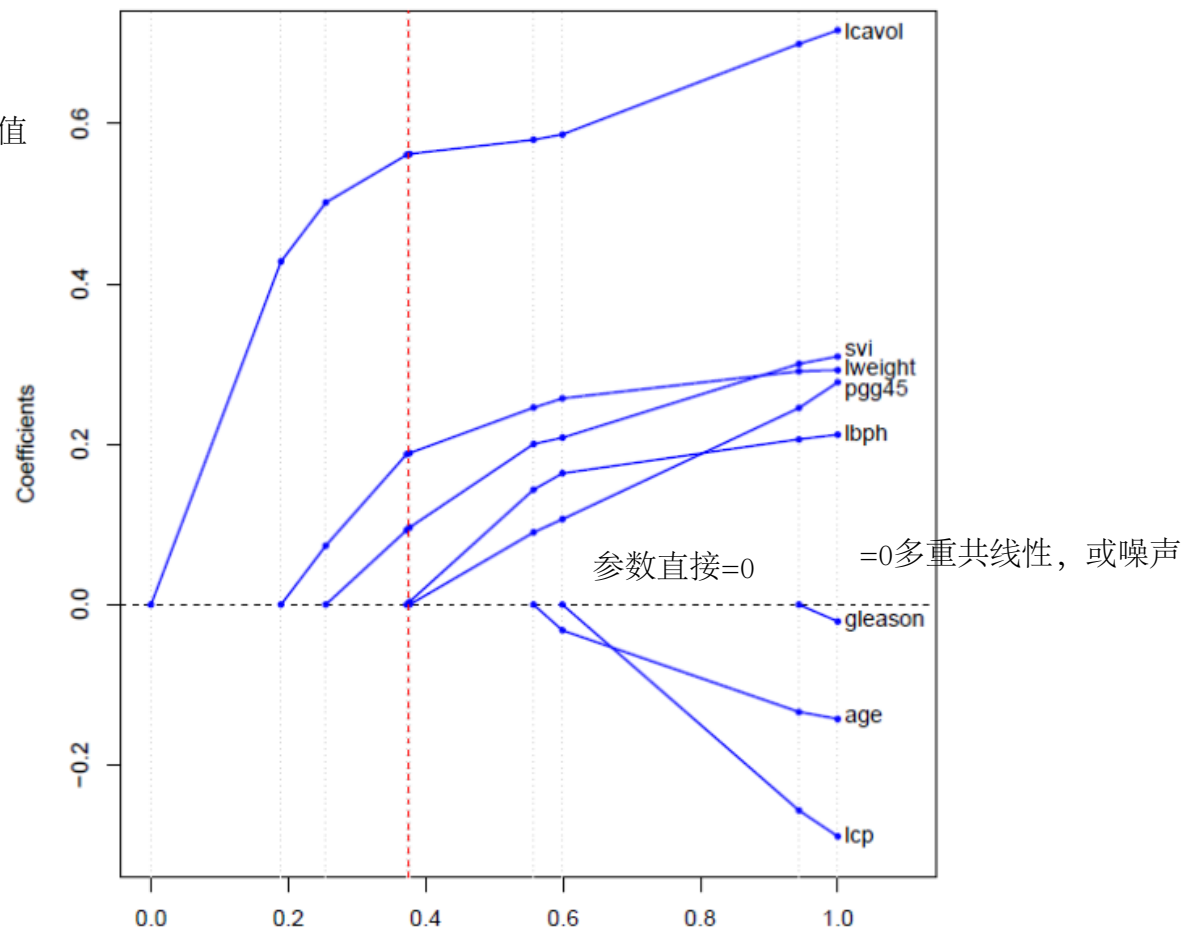


岭回归

LASSO与岭回归



岭回归



LASSO

弹性网

Elastic Net

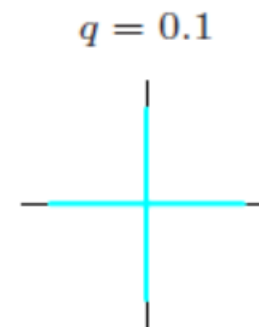
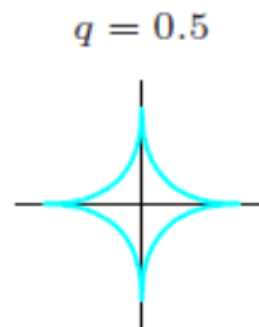
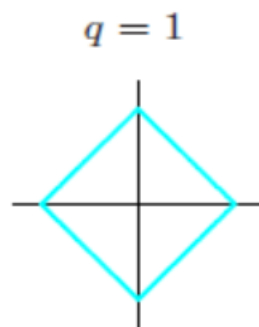
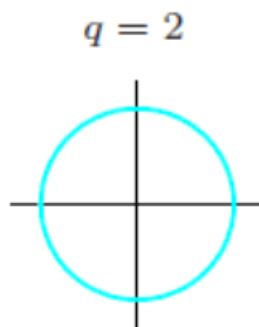
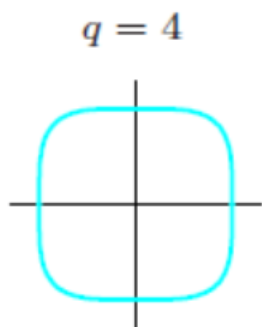
弹性网(Elastic Net)



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

q=2 岭回归

q=1 Lasso





Zou and Hastie (2005)提出elasticnet

只是正则项部分，还有其他

$$\lambda \sum_{j=1}^n (\alpha \theta_j^2 + (1 - \alpha) |\theta_j|)$$

结合了岭回归和Lasso

