4）证明概率 $p(\mathbf{x_1},...,\mathbf{x_n} \mid \Sigma) = \dfrac{1}{(2\pi)^{nd/2} \mid \hat{\Sigma} \mid^{n/2}} (\lambda_1 ... \lambda_d)^{n/2} exp\left[-\dfrac{n}{2}(\lambda_1 + ... + \lambda_d) \right]$

在 $\lambda_1 = 1 ... \lambda_d = 1$ 时达到最大值。

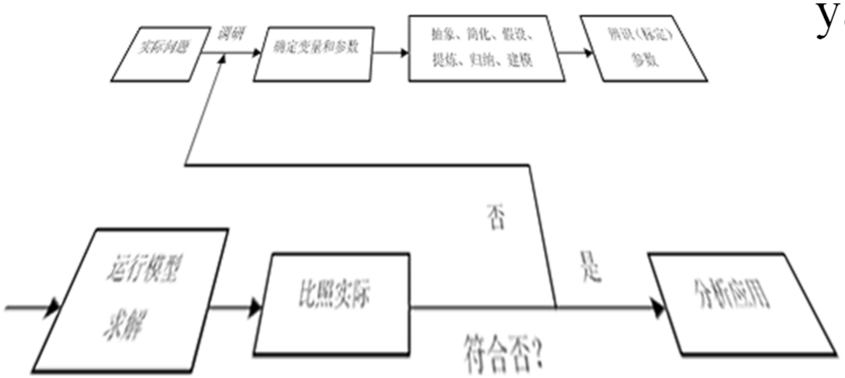
$$\prod_{i=1}^{d} \lambda_i^{n/2} exp\left(-\frac{n}{2}\lambda_i\right)$$

$$(\lambda_i e^{-\lambda_i})^{n/2}$$

统计学习基础

姚 远

yaoyuan@shu.edu.cn



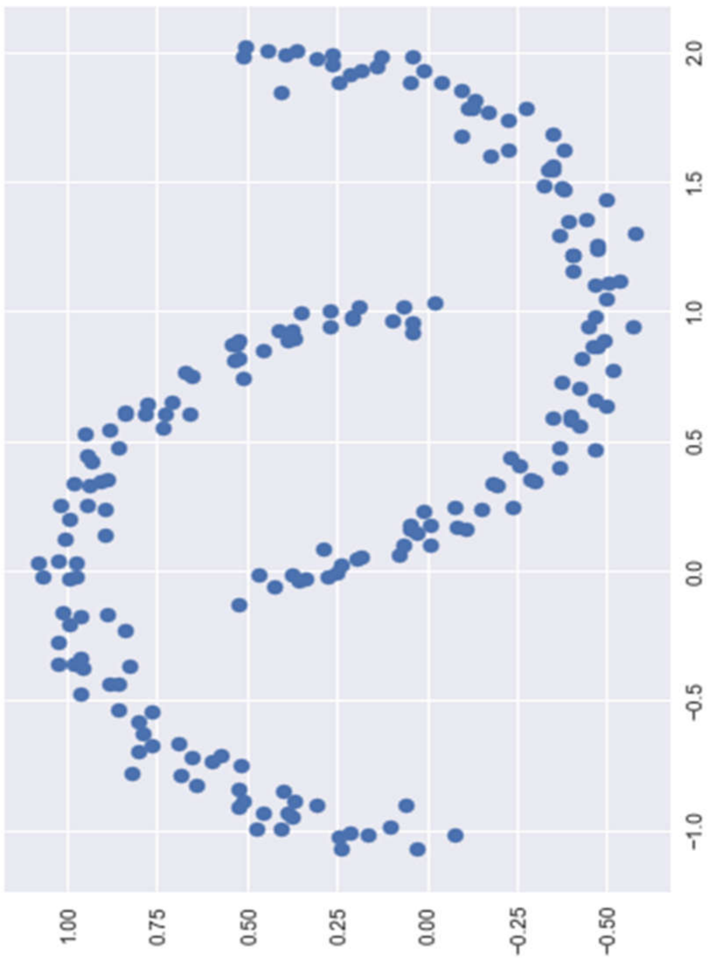
上海大学机自学院

2020/10/22

大纲

- 统计学习
- 线性回归方法
- 线性分类方法
- 扩展

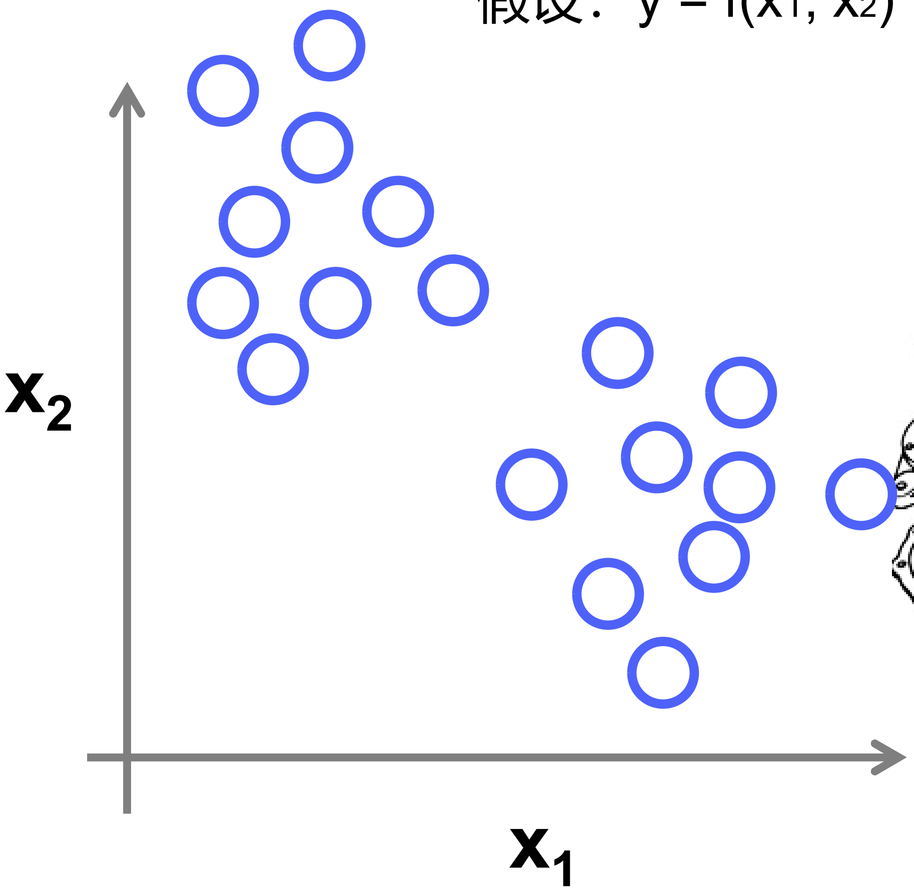
回顾: 作业1

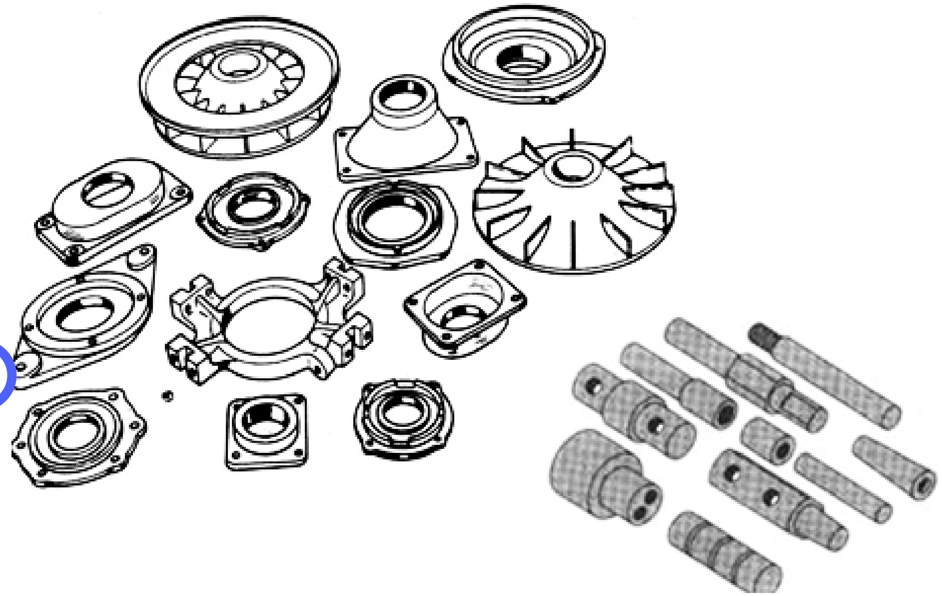


- 根据左面图例生成二维点云数据
- 采用K-means方法聚类，K=2
- 完成聚类算法实现，可采用matlab/octave/c++/python
- 分别采用曼哈顿距离/欧式距离/马氏距离作为距离评价方法
- 人是否可以很好的分类?

统计学习

假设：$y = f(x_1, x_2)$

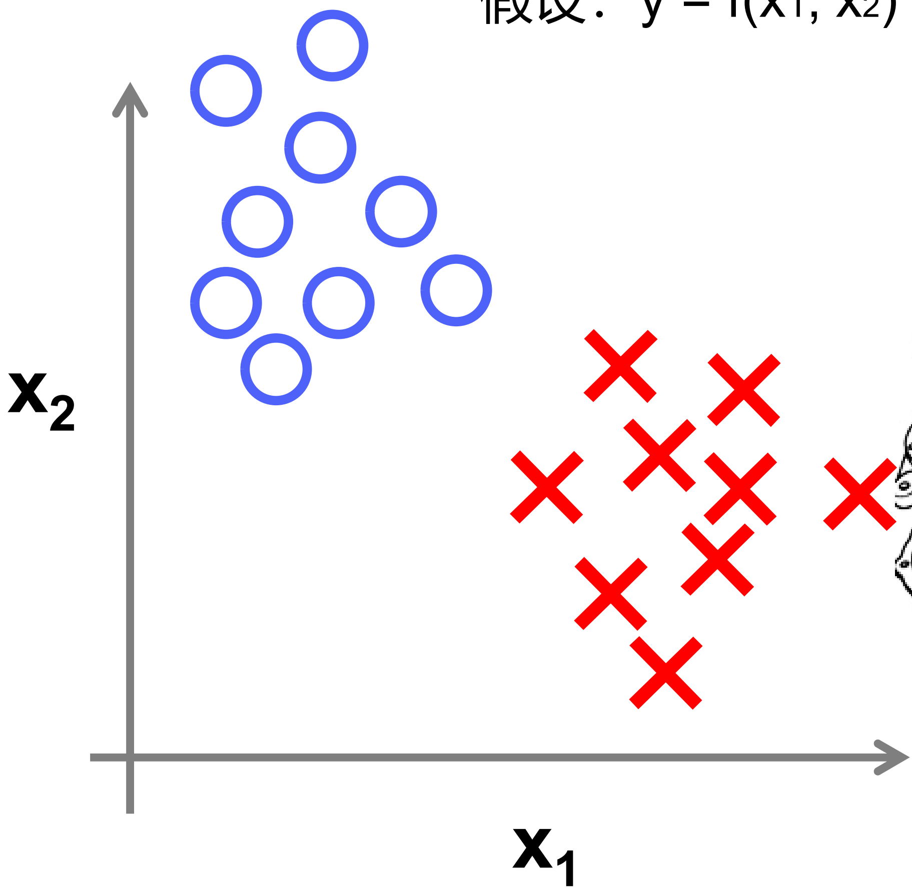


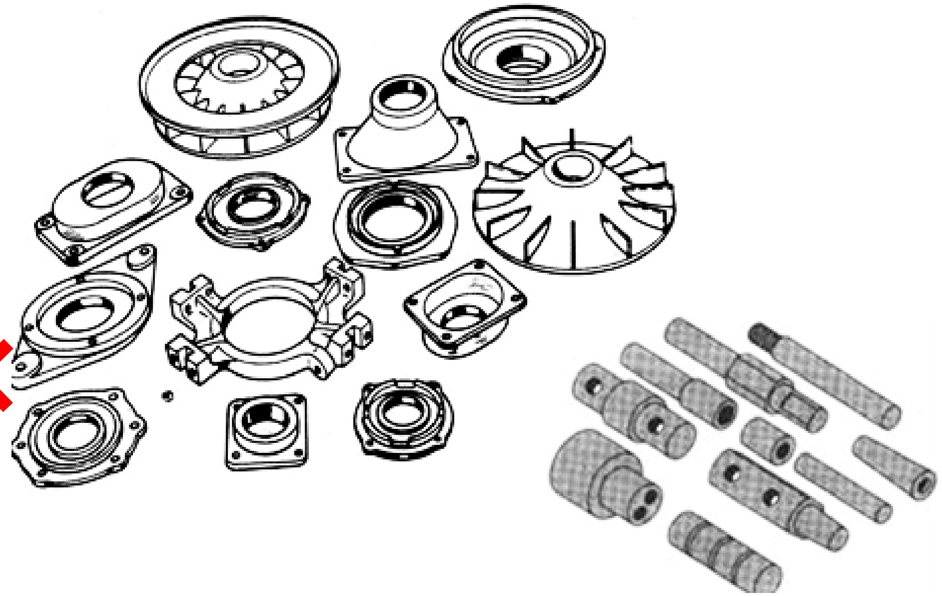


零件分组

统计学习

假设：$y = f(x_1, x_2)$



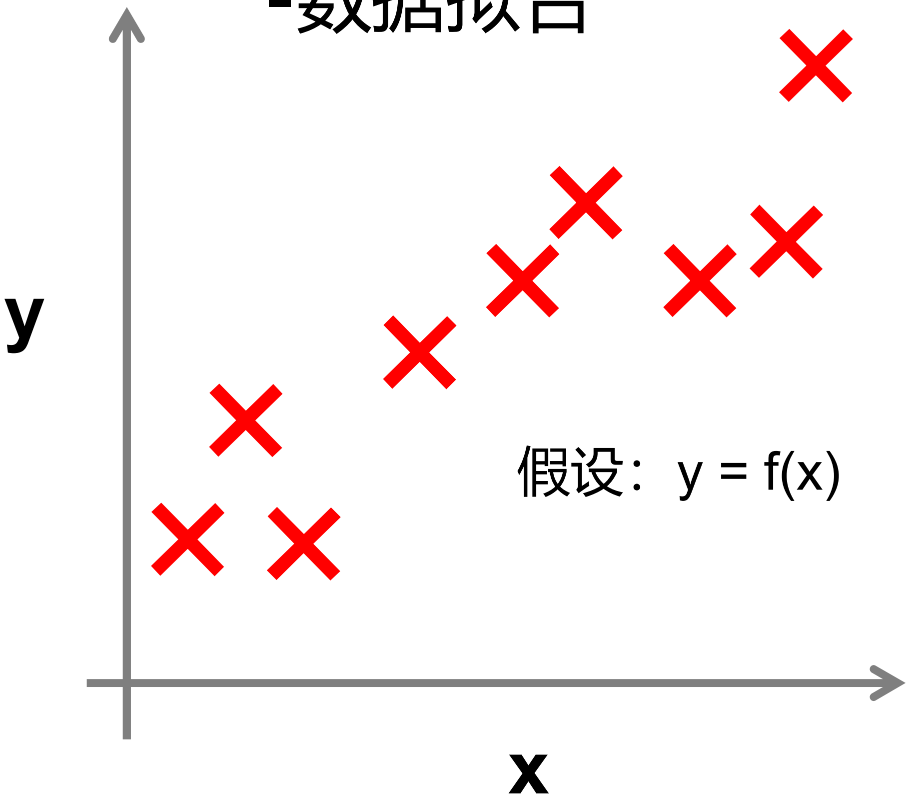


零件分组（聚类）

统计学习

如何找到合适的分割线?

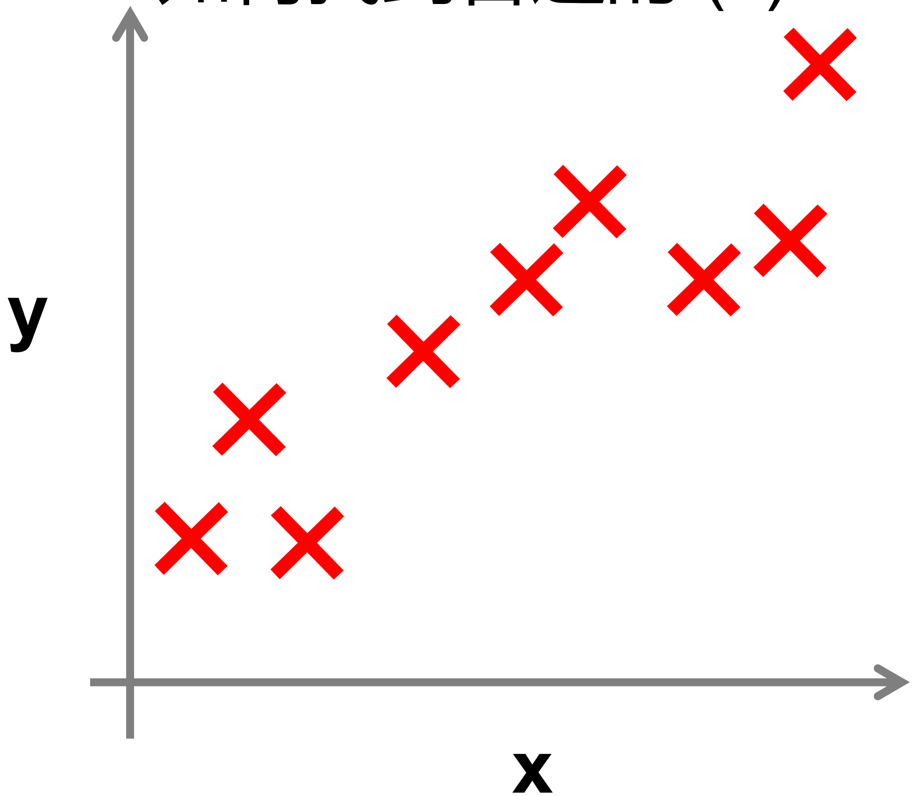
- 数据拟合



$$\min_f ||\delta||_2 = \Sigma_{i=1}^m ||\delta_i||_2^2$$

统计学习

如何找到合适的f(x)

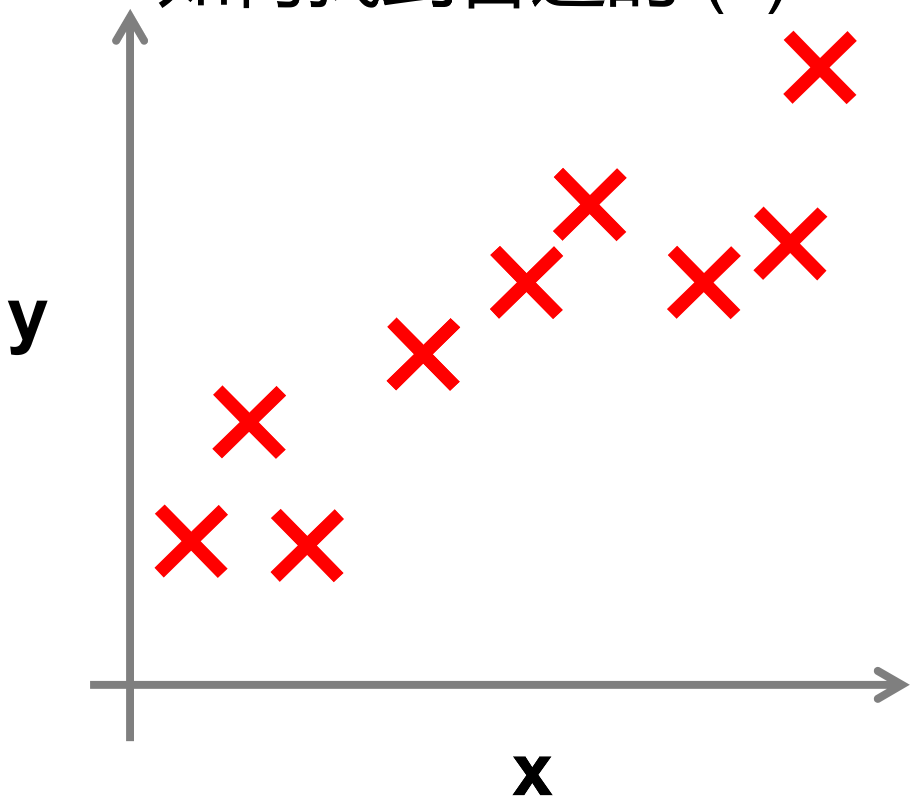


$$\min_{f} ||\delta||_2 = \Sigma_{i=1}^{m} ||\delta_i||_2^2$$

房价

统计学习

如何找到合适的f(x)



房价

$$\min_f ||\delta||_2 = \Sigma_{i=1}^m ||\delta_i||_2^2$$

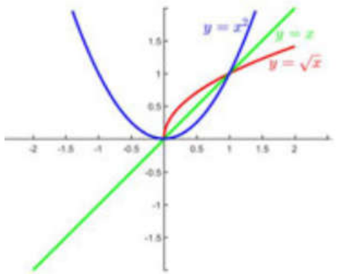
- 函数逼近问题：在选定的一类函数中寻找某个函数*g*，使它与已知函数*f*（或观测数据）在一定意义下为最佳近似表示，并求出用*g*近似*f*表示而产生的误差。

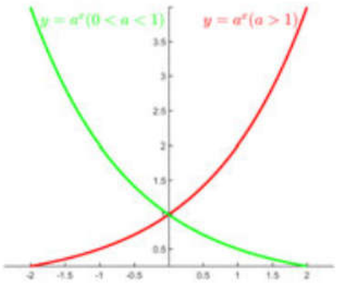
$$f(x) = wg(x)$$

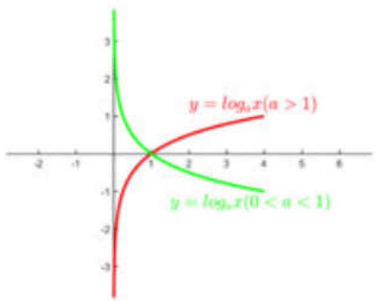
- **函数g的类型的选择和表示问题，这是函数拟合中的一个比较困惑的问题。**

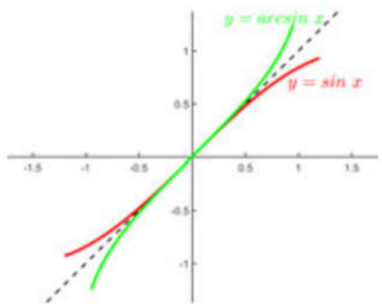
统计学习

如何找到合适的f(x)


幂函数


指数函数


对数函数


(反)三角函数

$$\min_f ||\delta||_2 = \Sigma_{i=1}^{m}||\delta_i||_2^2$$

- **函数的类型的选择和表示：**
  - **可选的（基函数）有很多**
  - **如何确定?**

- 函数逼近问题：
  - 可使用各类函数
  - 或函数的组合

统计学习

如何找到合适的f(x)：函数空间

- 函数空间是基函数的线性表达 $\{B_i(x), i = 0, 1, \ldots, n\}$

例如：n次多项式函数 $\{1, x, x^2, \ldots, x^n\}$

- 空间的思想：将复杂的量表达为若干（一般为有限）简单的量（基、字典）的（线性）组合

$$f(x) = \sum_{i=1}^{n} w_i B_i(x)$$

将求f(x)的求解转化为权系数w_i的求解

统计学习

困难

- 选择什么样的函数（函数空间）

  例如：选择多项式空间？三角函数空间

- 如果确立了空间，选择哪些基函数？

  例如：*x*^*n*中*n*应该是多少？是否要有xy项？

$$f(x) = \sum_{i=1}^{n} w_i B_i(x)$$

这就是“调参”中关注的问题

将求f(x)的求解转化为权系数w_i的求解

统计学习

- 选择一个函数空间
  - 基函数的线性表达
  - 建立假设

$$y = f(x) = \sum_{i=1}^{n} w_i B_i(x)$$

- 求解参数
  - 最小二乘法
  - 其它方法

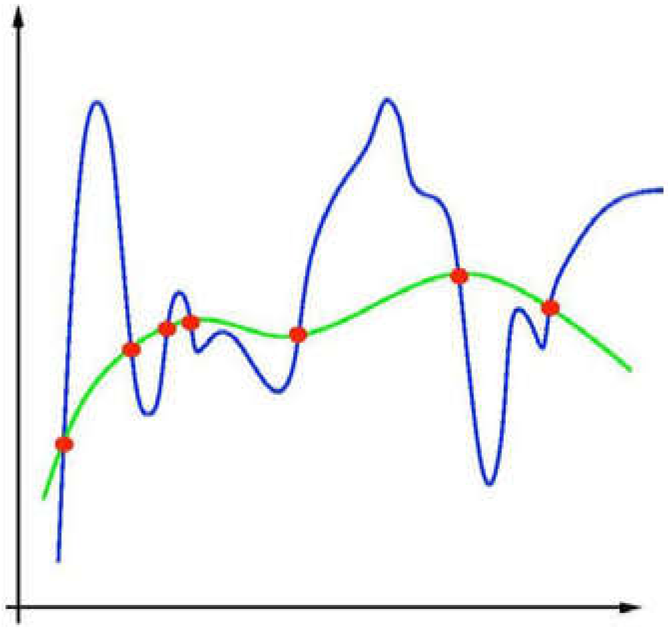
$$\min_{W} \|Y - XW\|^2$$

$$X^T X W = X^T Y$$

$$W = (X^T X)^{-1} X^T Y$$

统计学习-过拟合

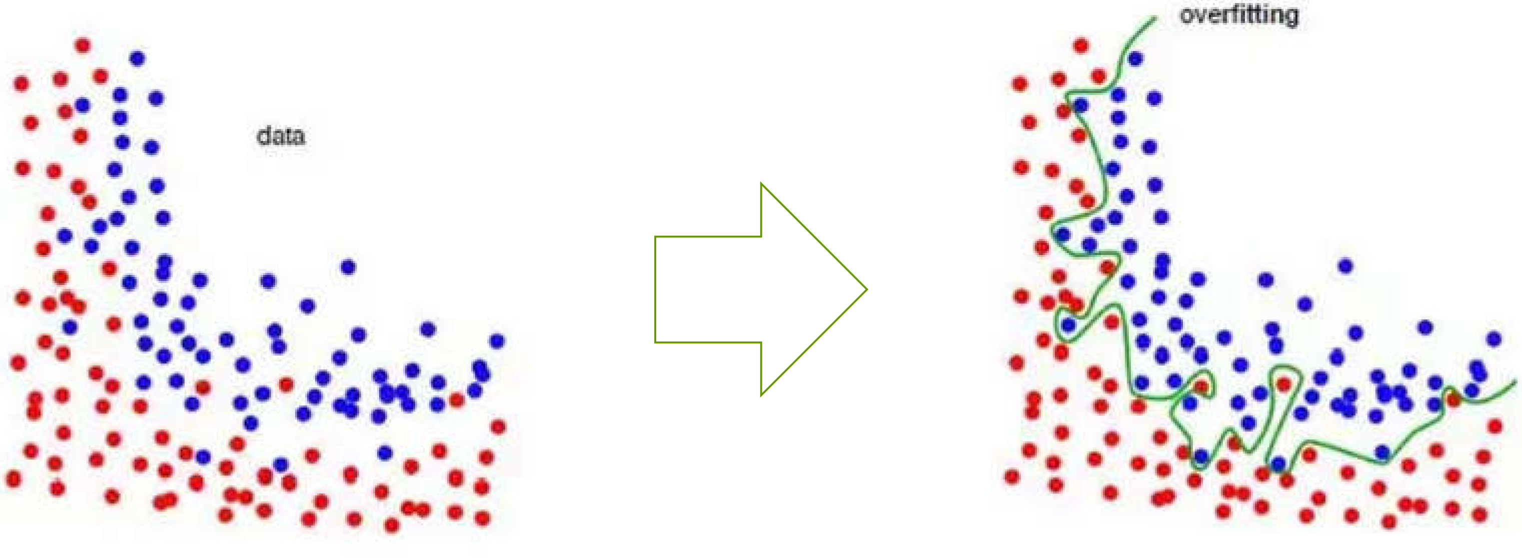
误差为0，但拟合程度并无使用价值



$$f(x) = \sum_{i=1}^{n} w_i B_i(x)$$

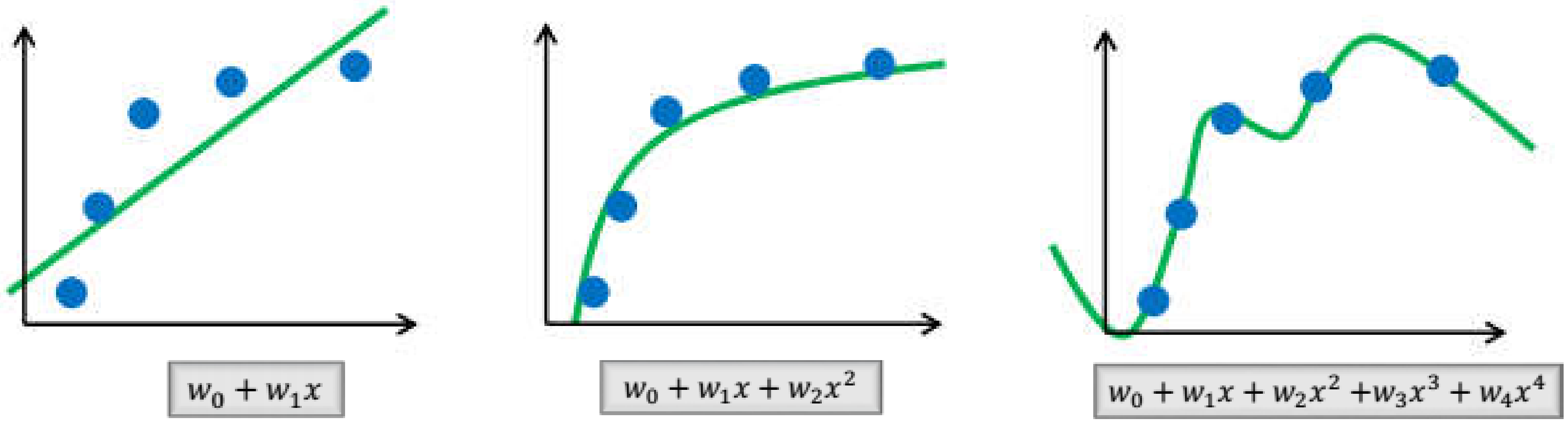
统计学习

- 分类问题的过拟合



统计学习-过拟合

如何选择合适的基函数



高偏置

低方差
低偏置

高方差

统计学习-避免过拟合

- 数据去噪

  - 提出训练样本中的噪声

$$f(x) = \sum_{i=1}^{n} w_i B_i(x)$$

- 数据扩增

  - 增加样本数，或者增加样本代表性和多样性

- 模型简化

  - 选用更简单的模型，避免拟合数据中的噪声，或者进行模型剪裁

- 正则约束

  - 选用更简单的模型，避免拟合数据中的噪声

统计学习

为什么是学习？

- 学习是通过观察和分析环境，构造响应模式的一种方法

数据：证据

假设：

如何起作用

（概率理论）

统计学习

统计学习是基于数据构建统计模型的一种方法

- 监督式学习
- 非监督式学习
- 半监督学习
- 强化学习

⇨

1. **模型**
2. **策略**
3. **算法**

数据模型

- 如何从数据中恢复信息?

- **<u>回归分析问题</u>**
- **预测真实的房价**

$x \rightarrow \boxed{M} \rightarrow$ **房屋价格**

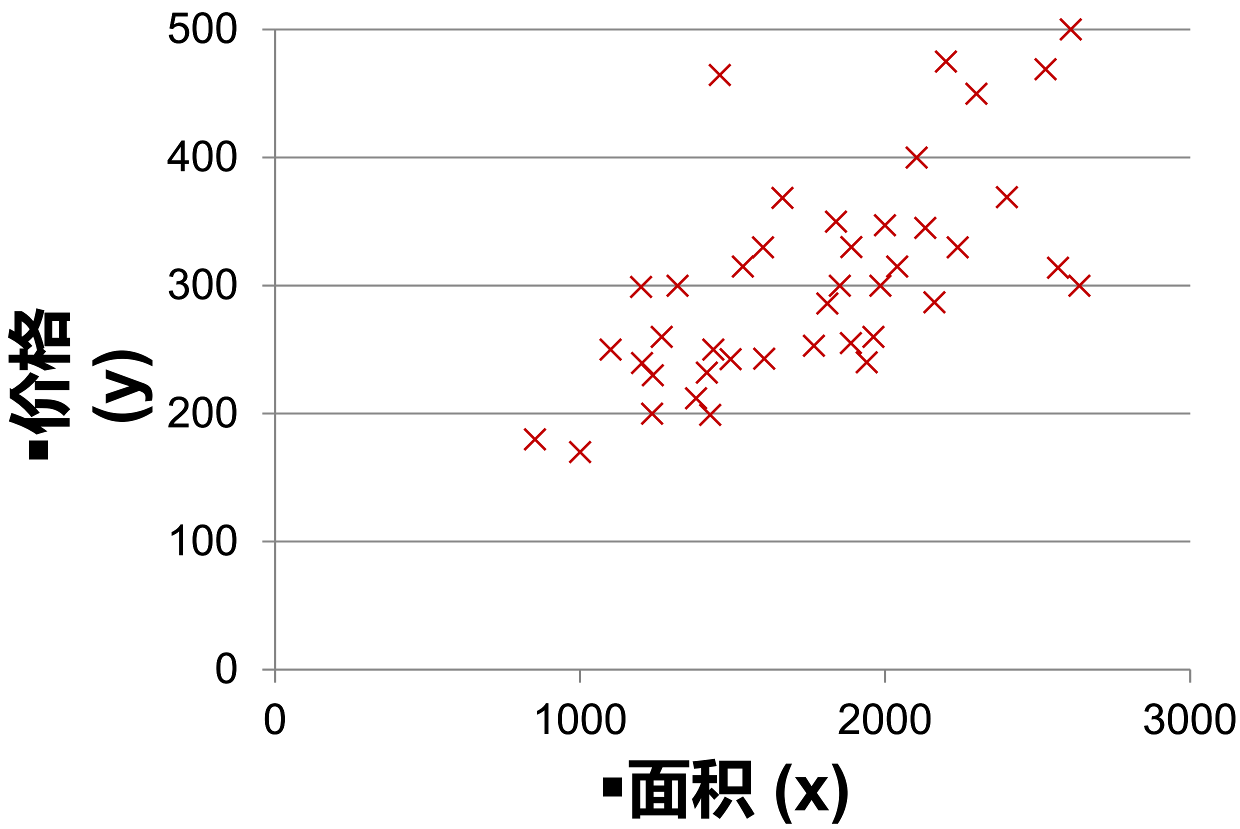
- **<u>分类问题</u>**
- **为对象给出所述类别**

$(x_1,y_1) \rightarrow \boxed{M} \rightarrow$ **零件类别**

大纲

- 数据模型
- 线性回归方法
- 线性分类方法
- 扩展

■房　价



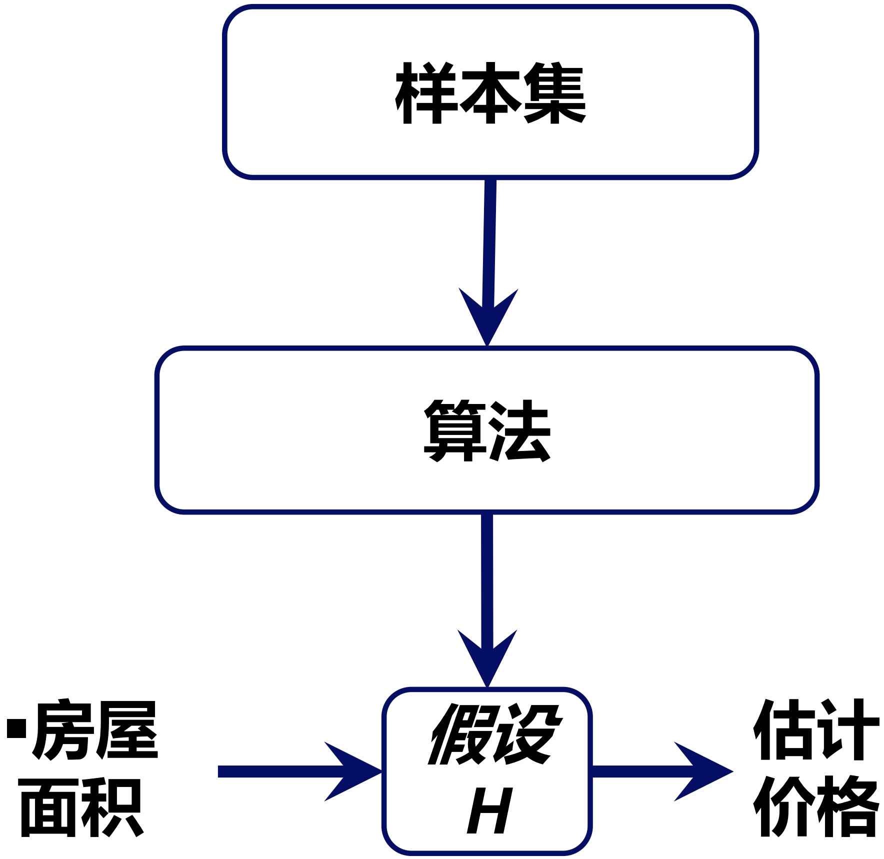
- 房价数据

样本集

	面积 (x)	价格 (y)
S^1	2104	460
S^2	1416	232
S^3	1534	315
	…	…
S^m	852	178

- 定义:

- m = 样本的数量
- x = 输入的“样本” / “特征”
- y = “输出变量” / “目标变量”



▪如何表达模型的假设H?

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

▪单变量线性回归.

▪多变量线性回归.

样本集合

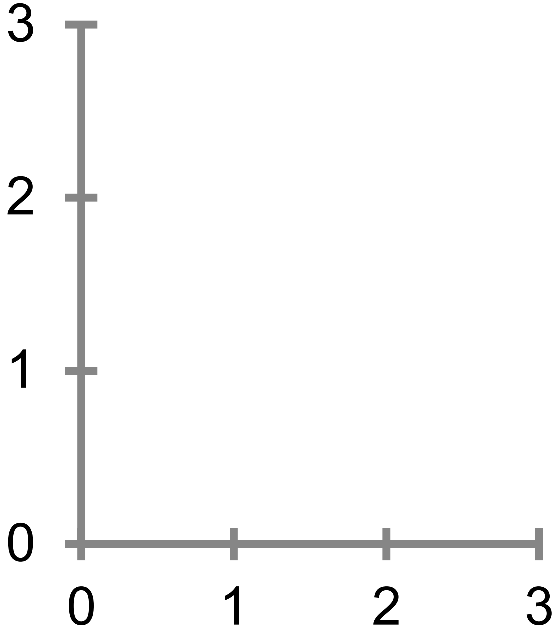
面积 (x)	价格 (y)
2104	460
1416	232
1534	315
852	178
…	…

- **假设H:** $h_\theta(x) = \theta_0 + \theta_1 x$

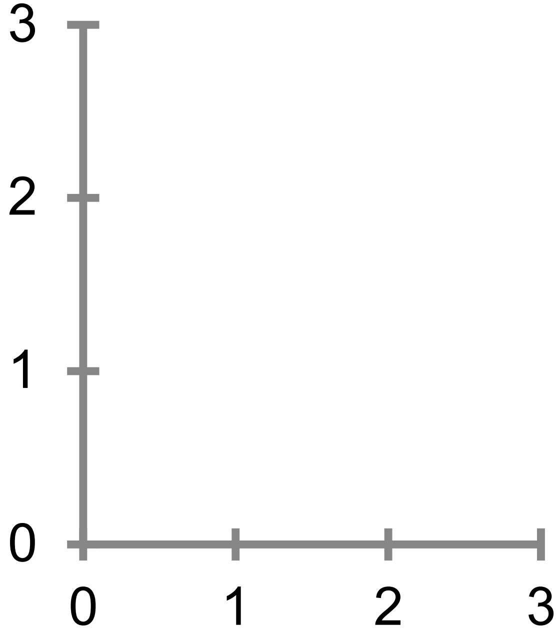
  θ_i : **模型参数**

- **如何选择** θ_i **？**

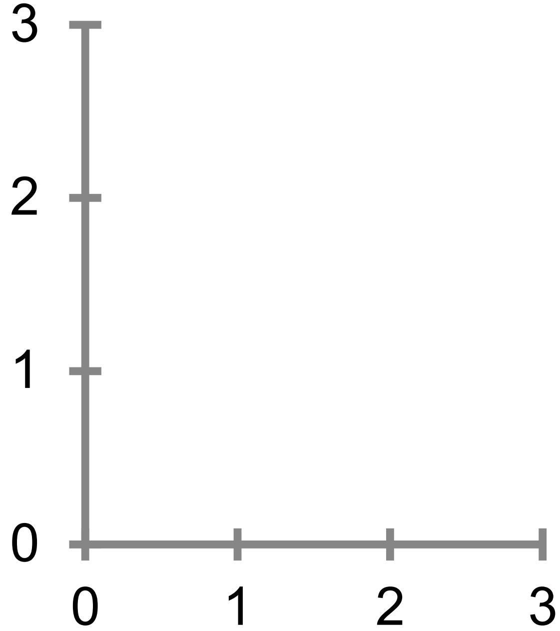
$$h_\theta(x) = \theta_0 + \theta_1 x$$



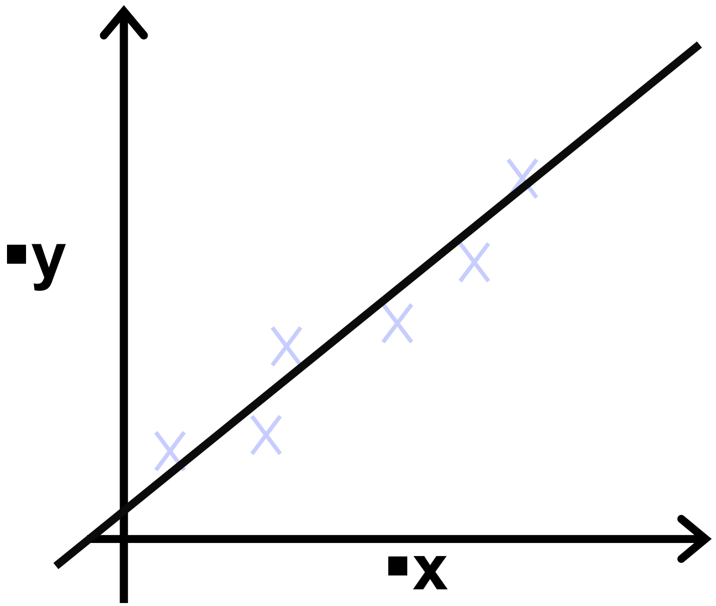
$\theta_0 = 1.5$
$\theta_1 = 0$



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



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



- **出发点：选择** θ_0, θ_1 **使** $h_\theta(x)$ **与样本集** y **最为近似**

▪**模型假设:**

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

▪**参数:**

$$\theta_0, \theta_1$$

▪**目标函数:** **所有目标点距离假设最近**

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

▪**目标:**

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

▪**<u>简化形式</u>**

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

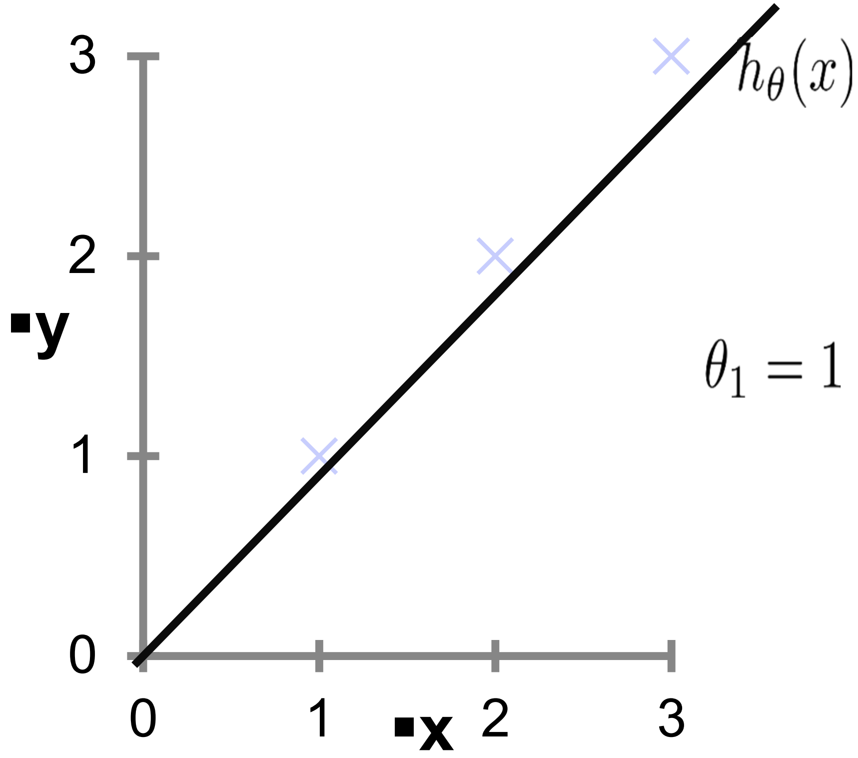
$$\theta_1$$

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

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

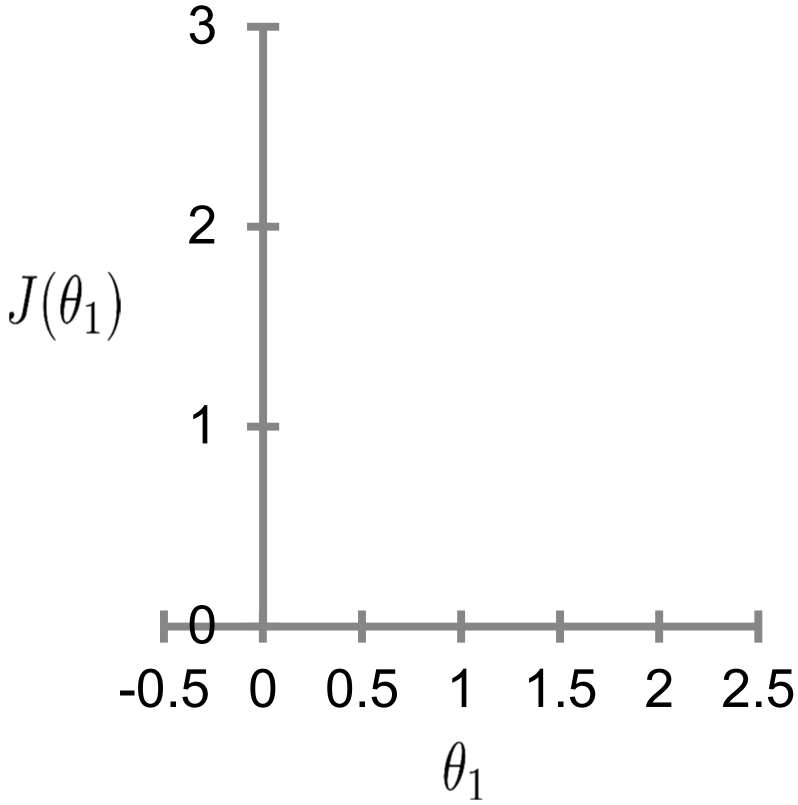
$h_\theta(x)$

固定 θ_1, 模型假设是x的函数



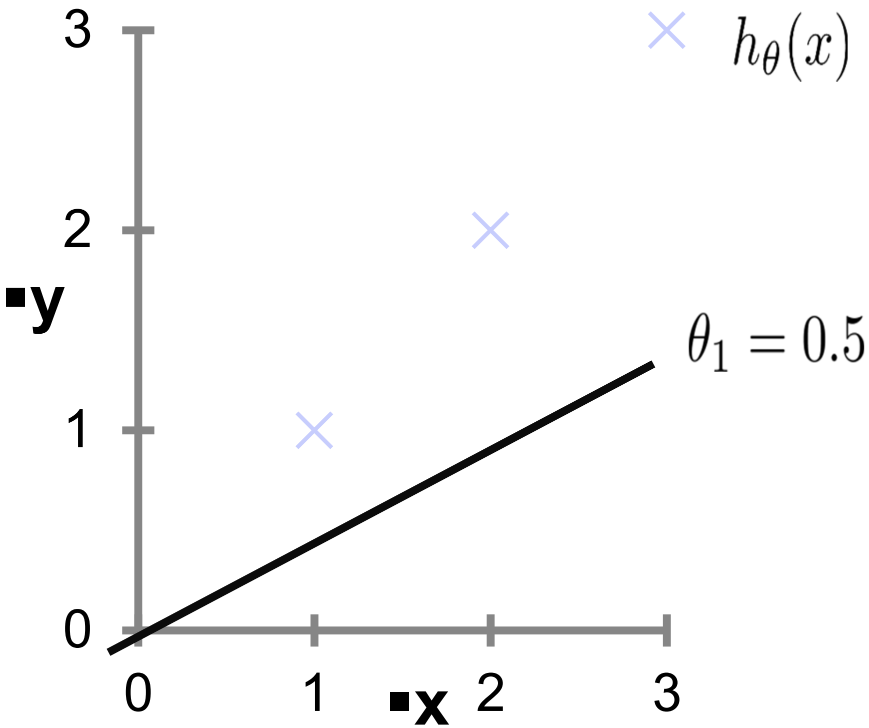
$J(\theta_1)$

目标函数是关于 θ_1 的函数



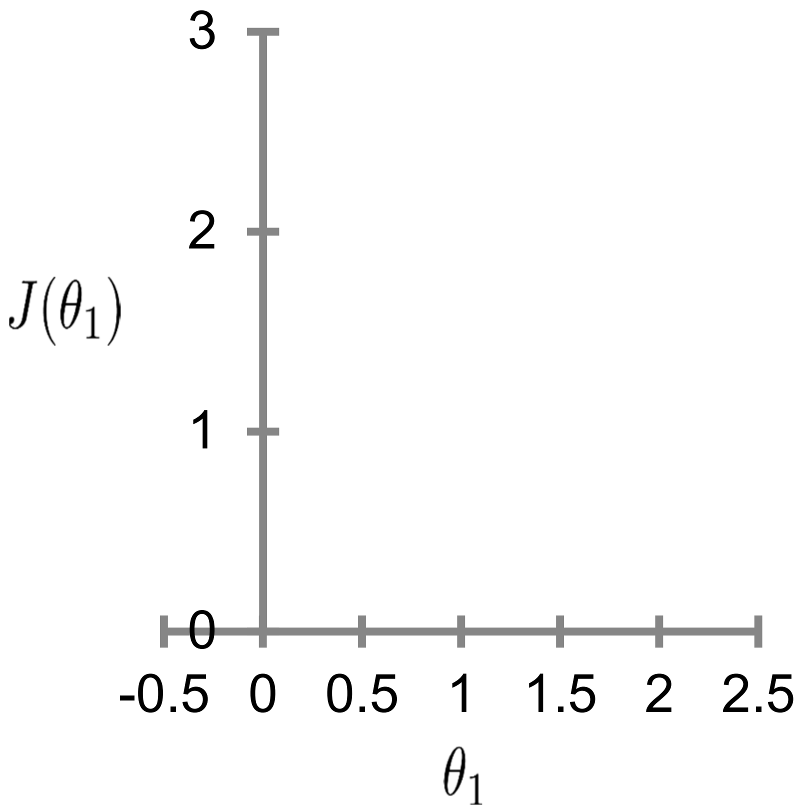
$h_\theta(x)$

固定 θ_1，模型假设是x的函数



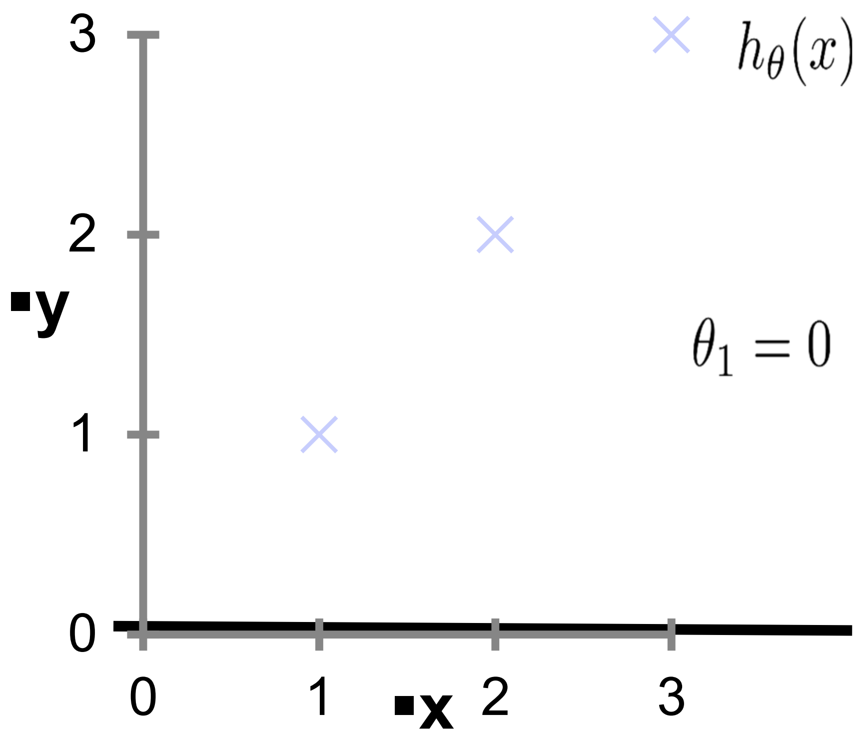
$J(\theta_1)$

目标函数是关于 θ_1的函数



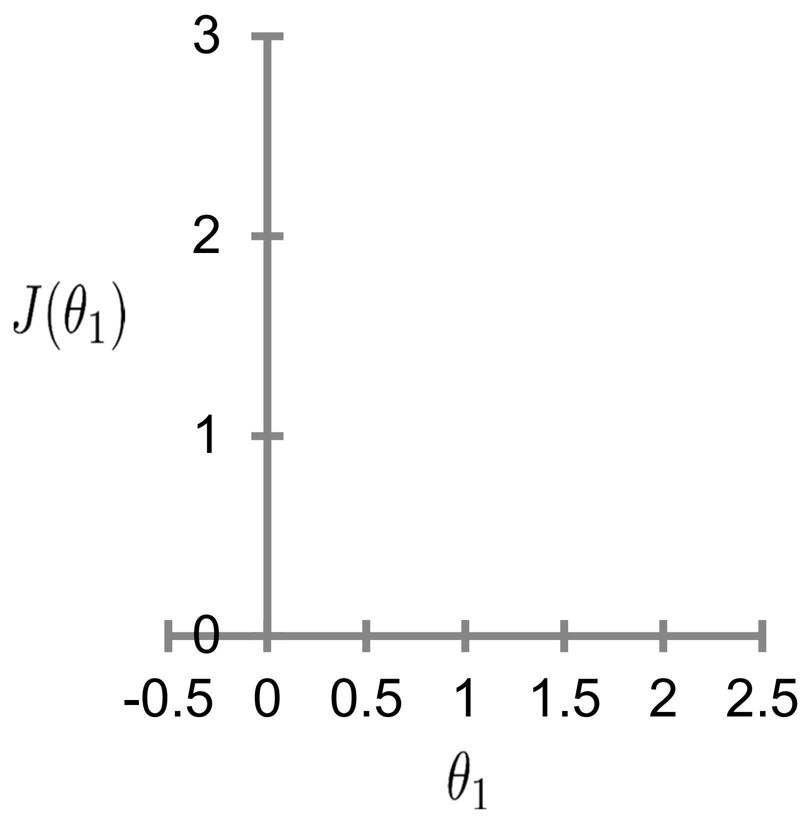
$$h_\theta(x)$$

固定 θ_1 ，模型假设是x的函数



$$J(\theta_1)$$

目标函数是关于 θ_1 的函数



多变量的情况

- **模型假设:** $h_\theta(x) = \theta_0 + \theta_1 x$

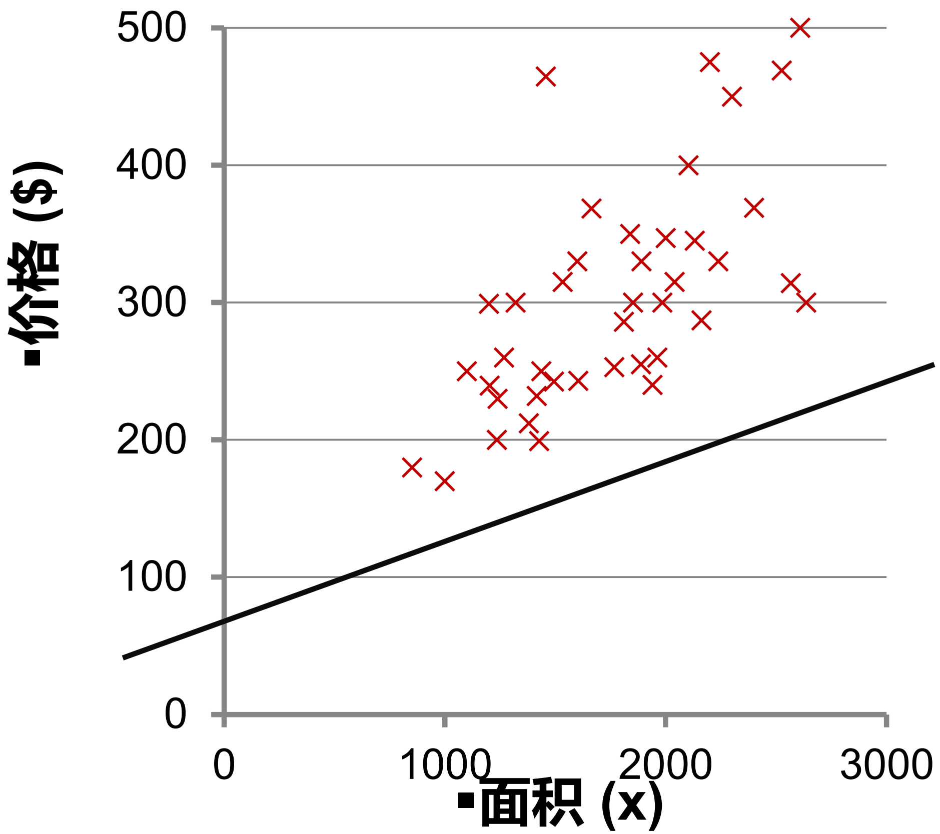
- **参数:** θ_0, θ_1

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

- **目标:** $\underset{\theta_0, \theta_1}{\text{minimize}}\, J(\theta_0, \theta_1)$

$h_\theta(x)$

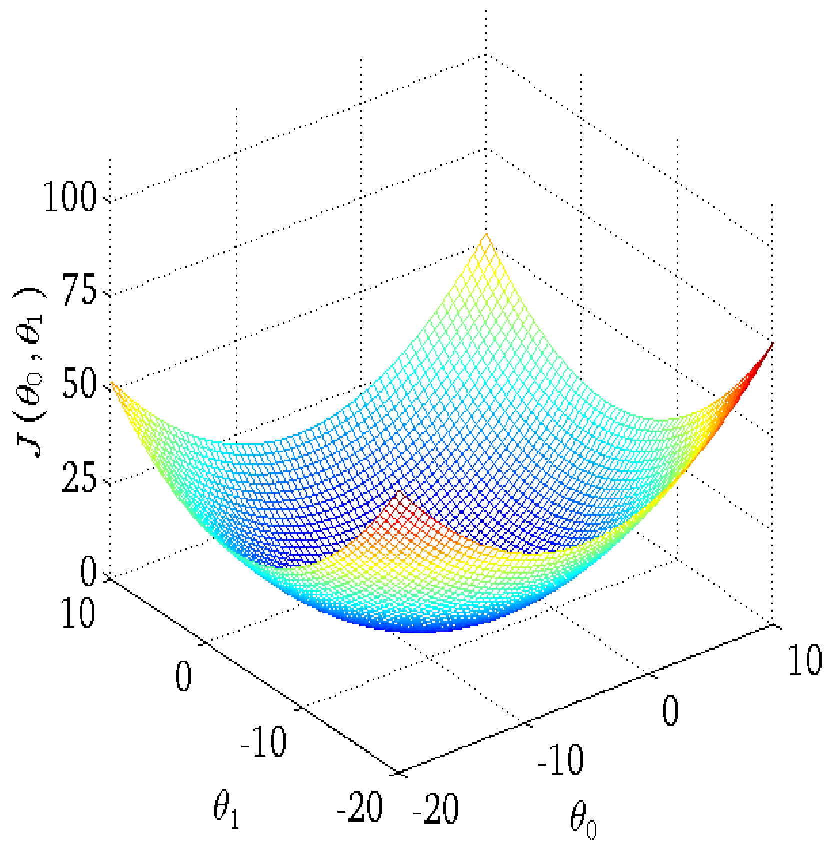
固定 θ_0, θ_1，模型假设是x的函数



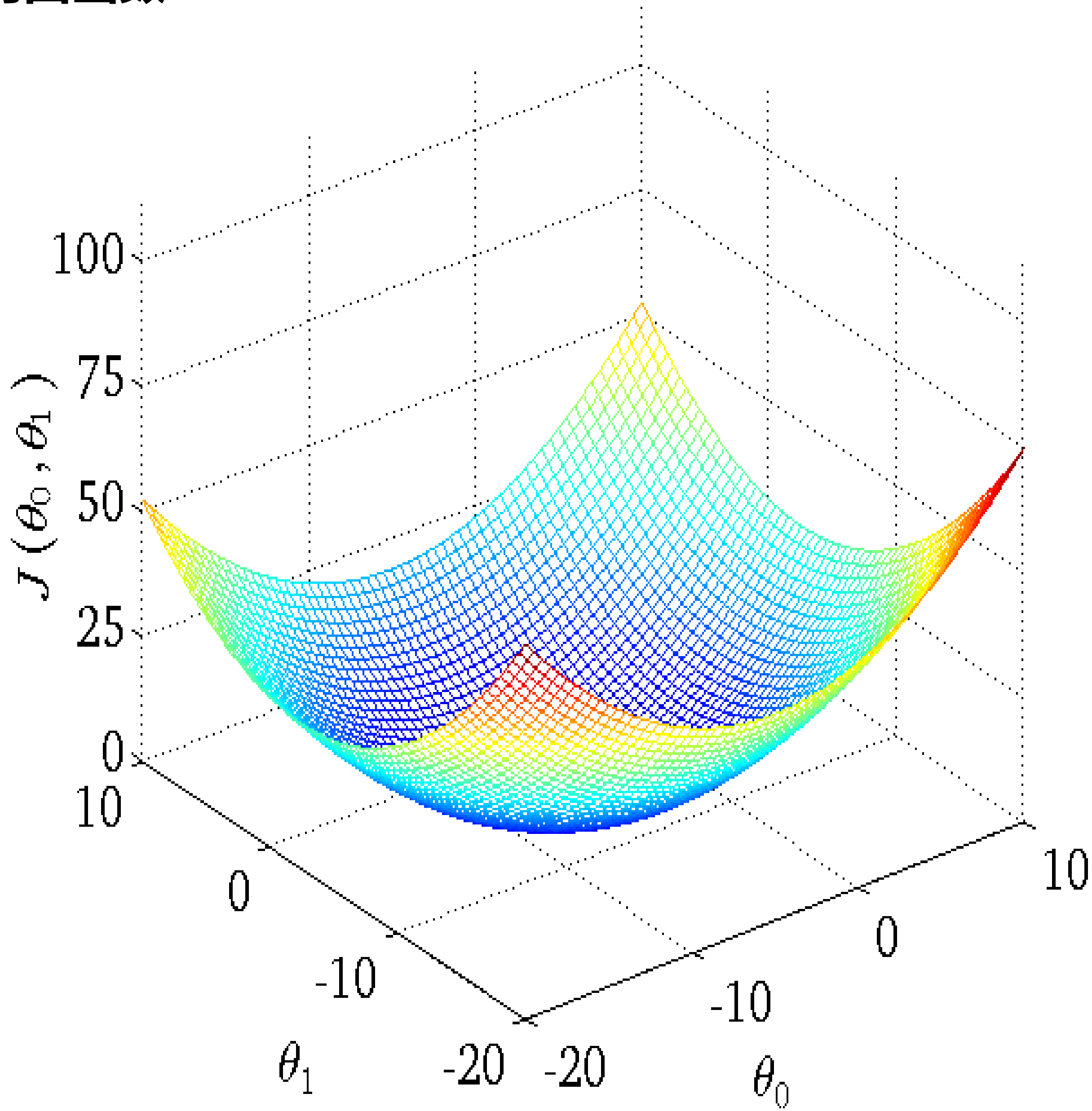
$$h_\theta(x) = 50 + 0.06x$$

$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 的函数

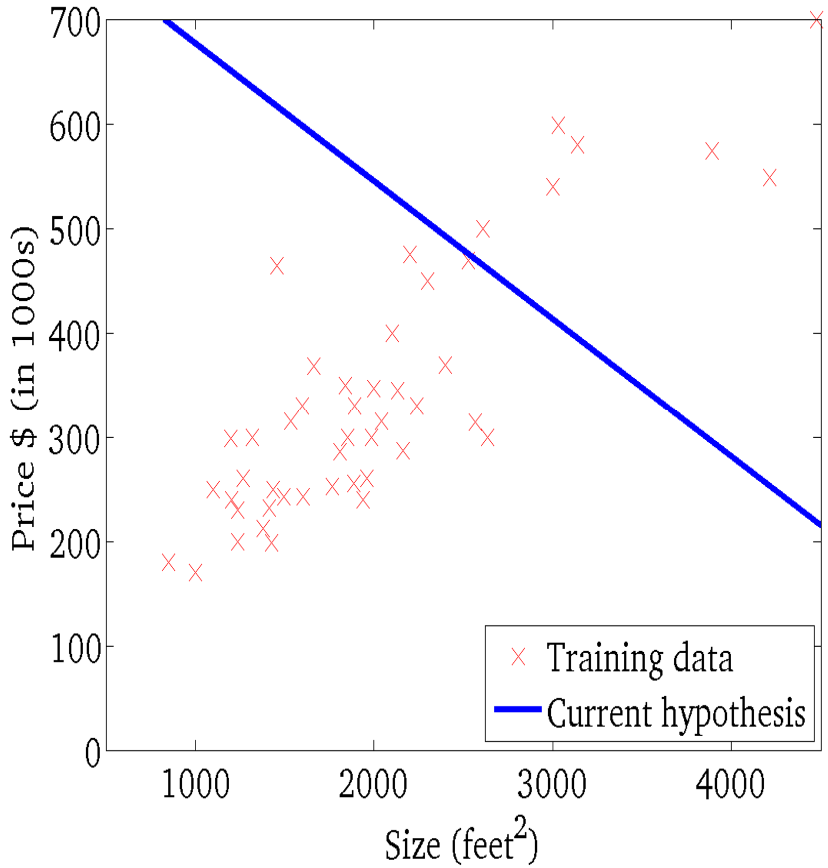


- 一个典型的凸函数



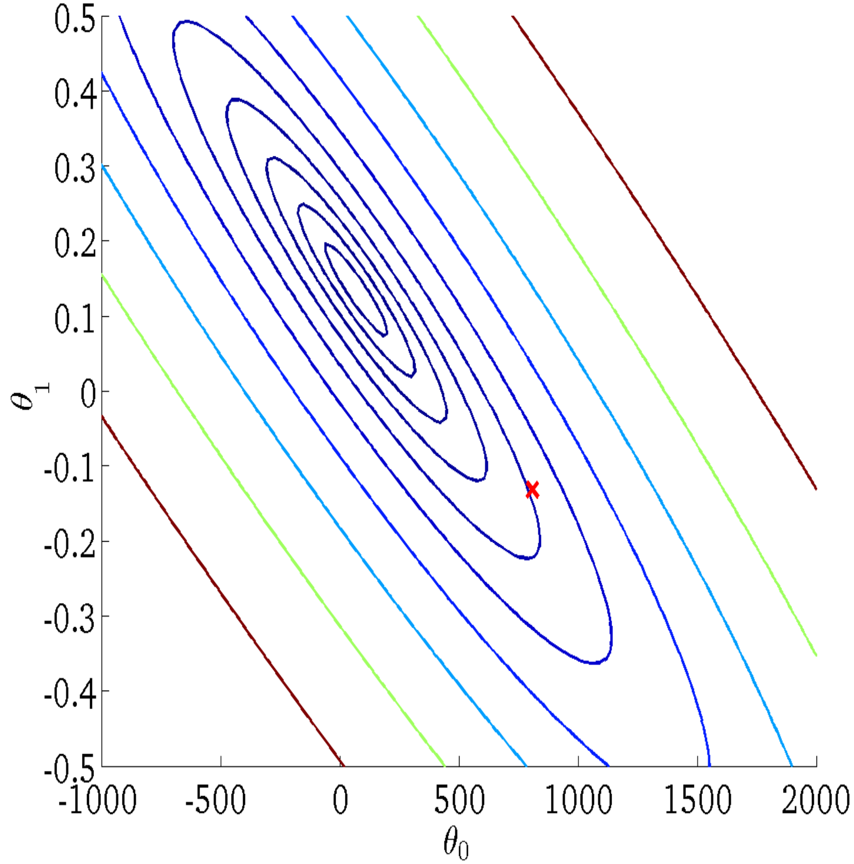
$h_\theta(x)$

固定 θ_0, θ_1，模型假设是x的函数



$J(\theta_0, \theta_1)$

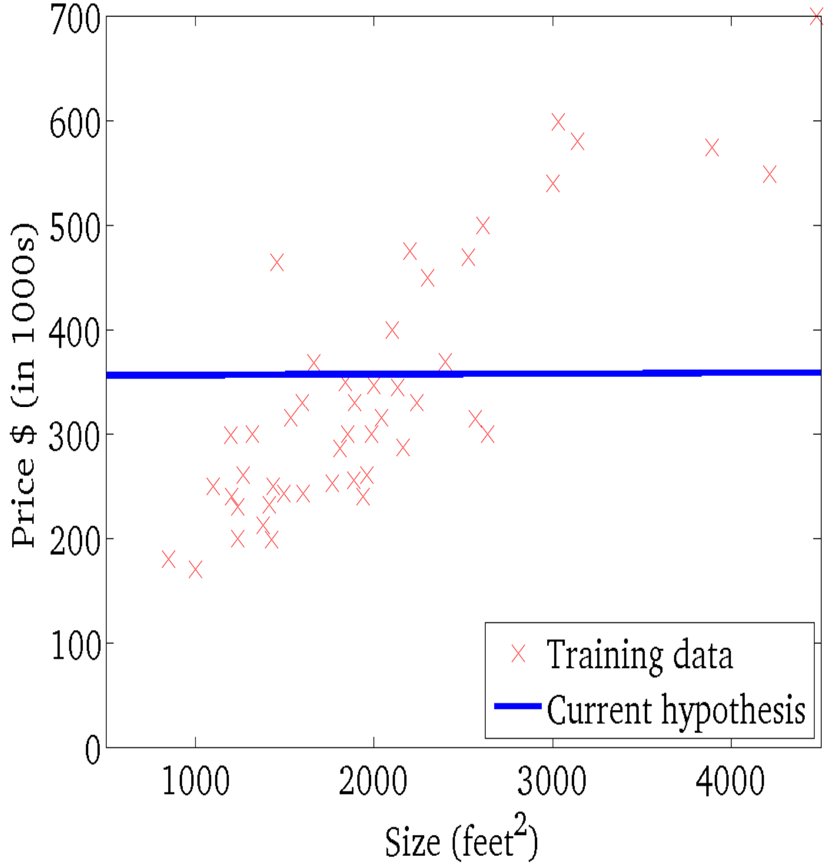
目标函数是关于 θ_0, θ_1 的函数



- **通常使用等高线表达三维函数**

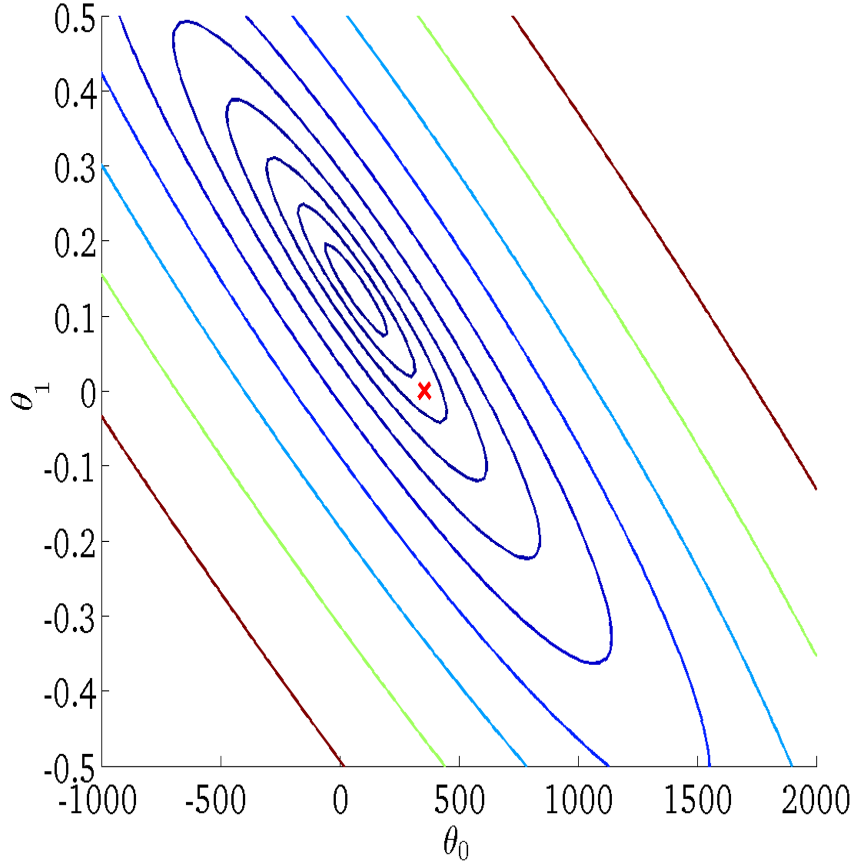
$h_\theta(x)$

固定 θ_0, θ_1**，模型假设是x的函数**



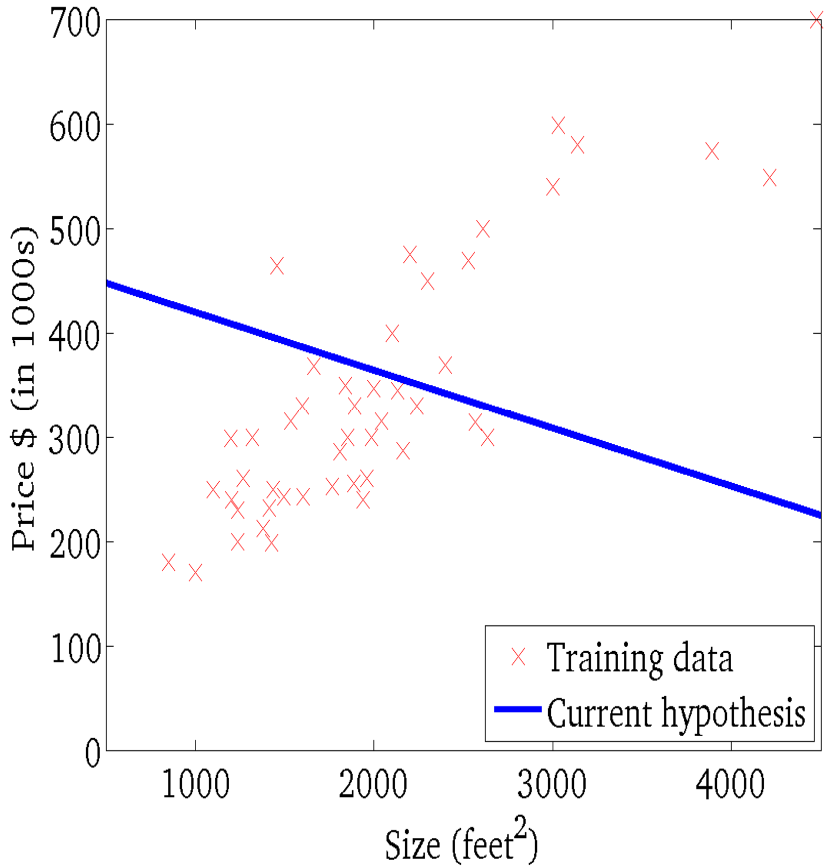
$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 **的函数**



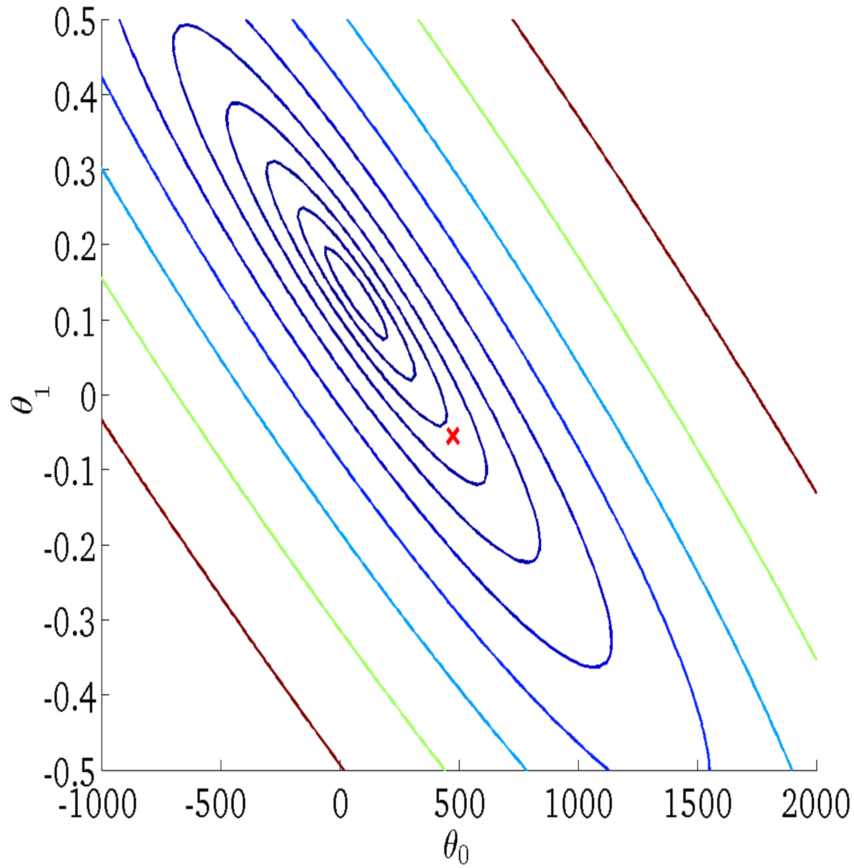
$h_\theta(x)$

固定 θ_0, θ_1**，模型假设是x的函数**



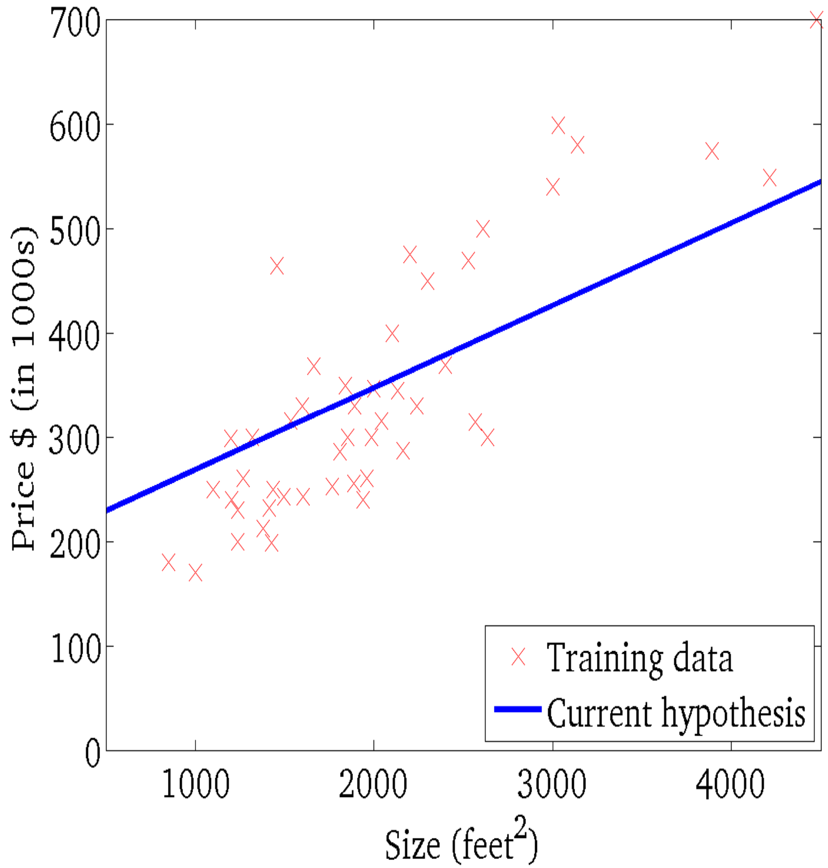
$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 **的函数**



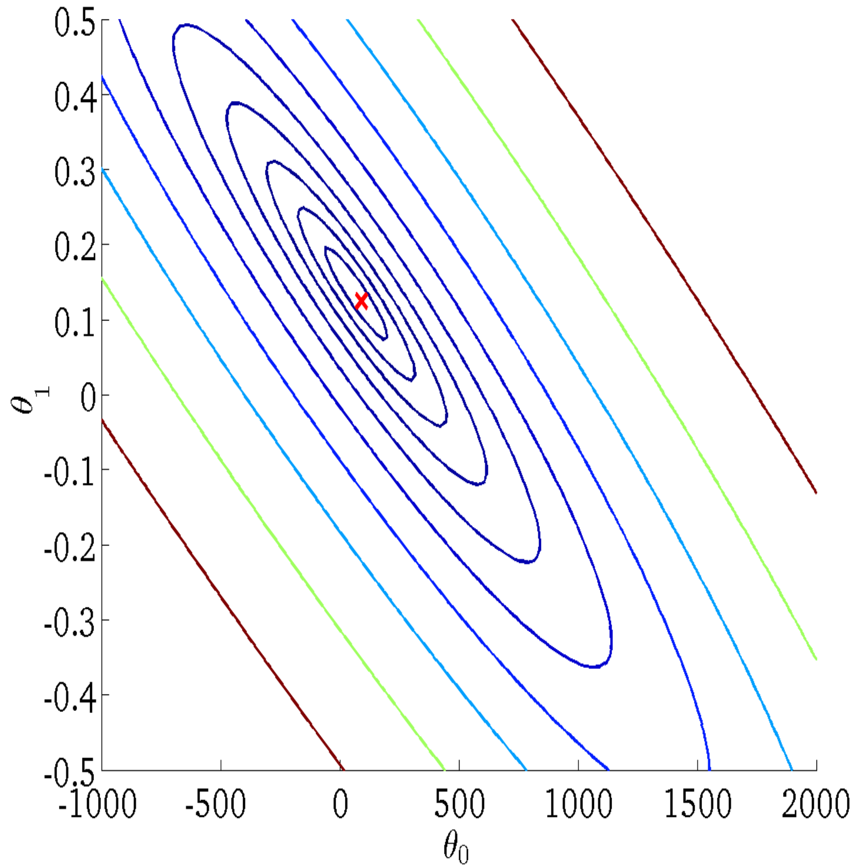
$$h_\theta(x)$$

固定 θ_0, θ_1，模型假设是x的函数



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

目标函数是关于 θ_0, θ_1 的函数



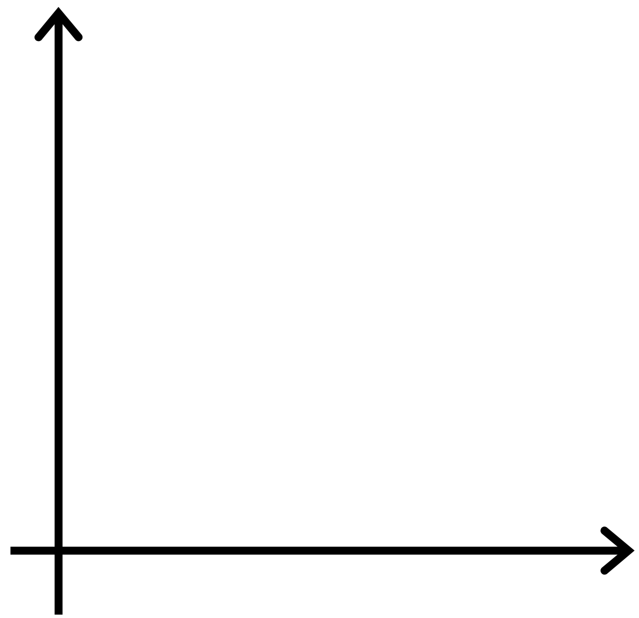
多变量学习的步骤

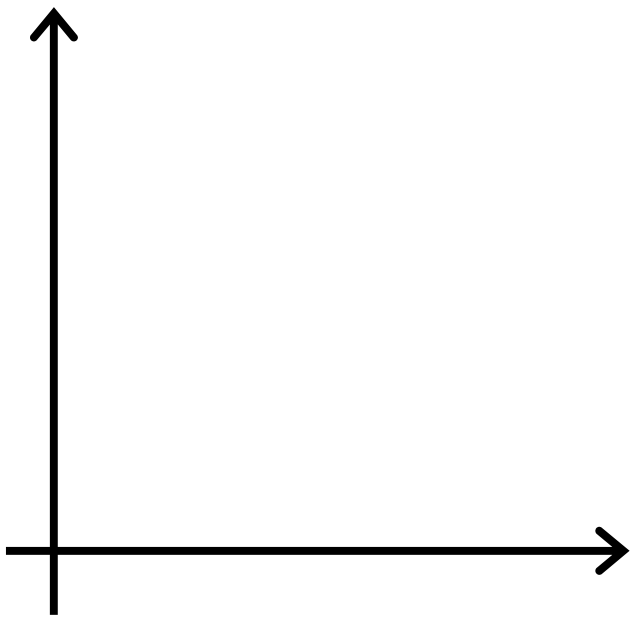
- 得到函数 $J(\theta_0, \theta_1)$
- 希望 $\min\limits_{\theta_0, \theta_1} J(\theta_0, \theta_1)$
- 步骤:
  - 从一组值 θ_0, θ_1 开始
  - 不断改变 θ_0, θ_1 来减小 $J(\theta_0, \theta_1)$, 直到达到一个我们希望得到的最小值。

▪最速下降算法

$$
\begin{aligned}
&\text{repeat until convergence } \{ \\
&\quad \theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \qquad \text{(simultaneously update } j = 0 \text{ and } j = 1\text{)} \\
&\}
\end{aligned}
$$

■最速下降算法-搜索方向

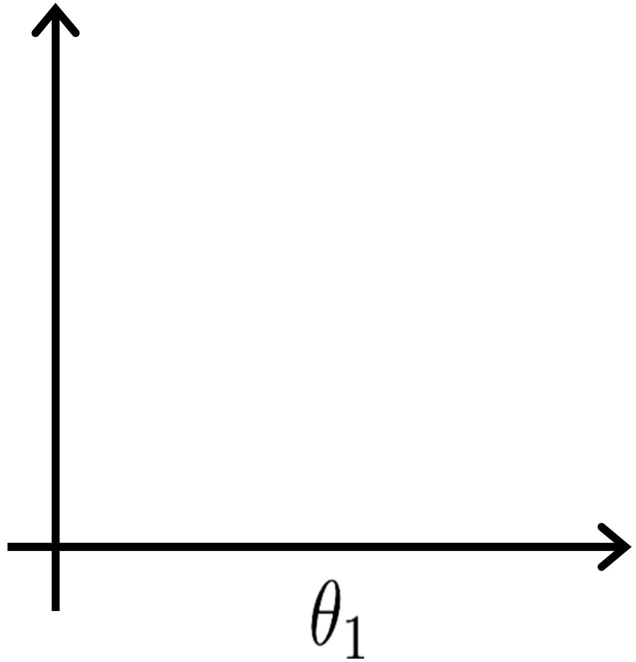




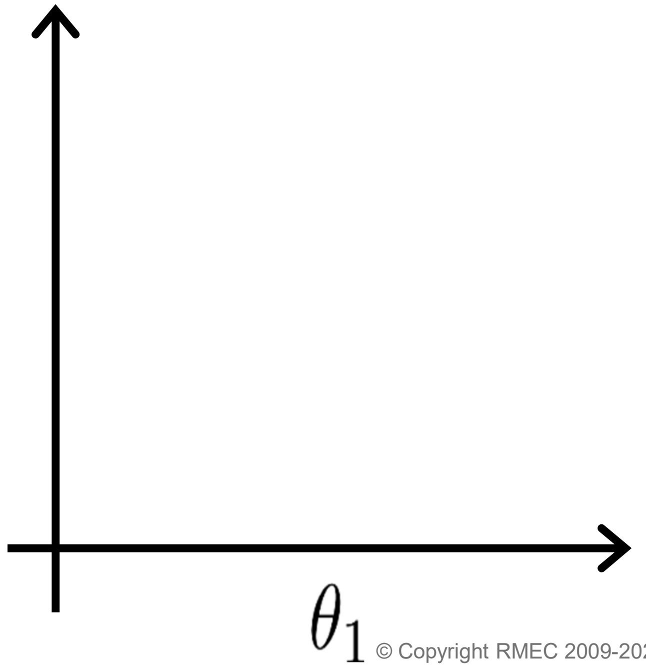
▪最速下降算法- α

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

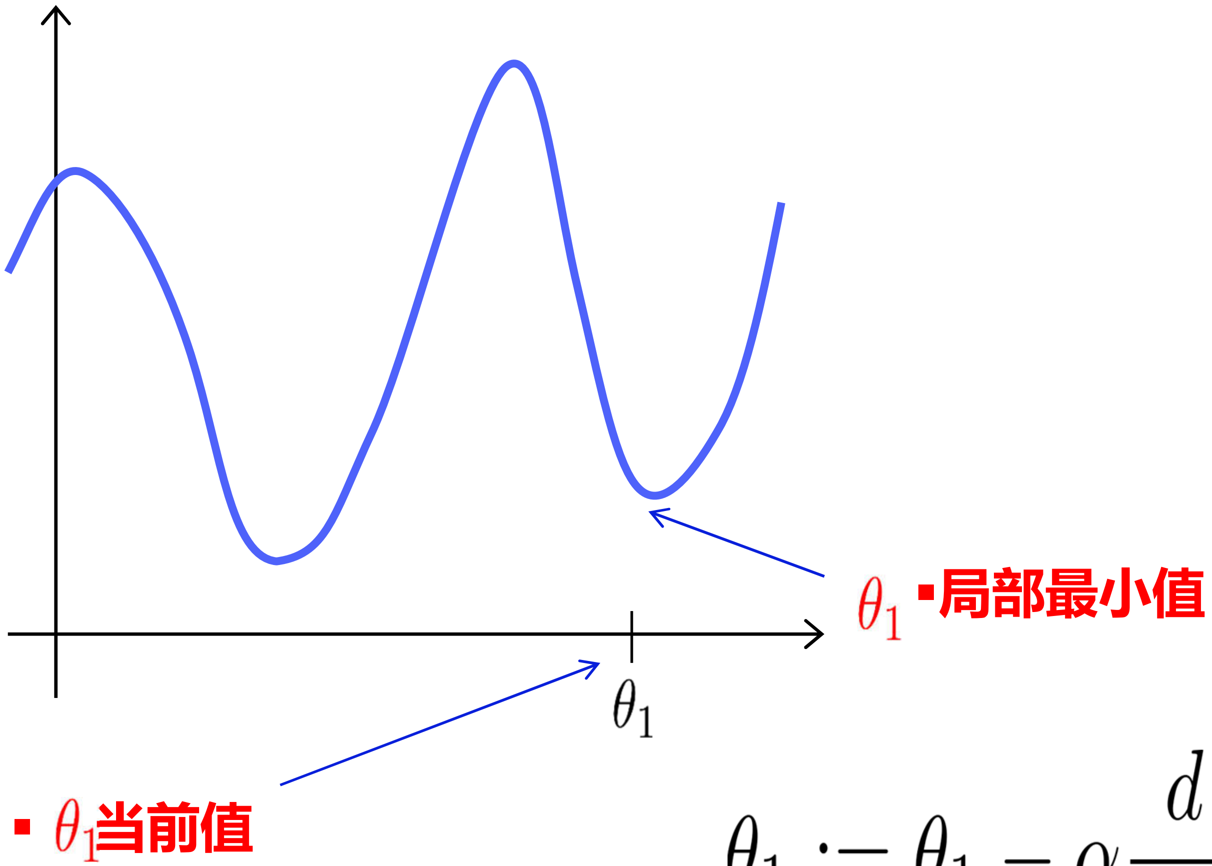
▪如果 α 太小，最速下降法收敛的速度会很慢。



▪如果 α 太大，在下降过程中会越过最小值点，可能会出现不收敛的情况。



▪最速下降算法- 局部最小值



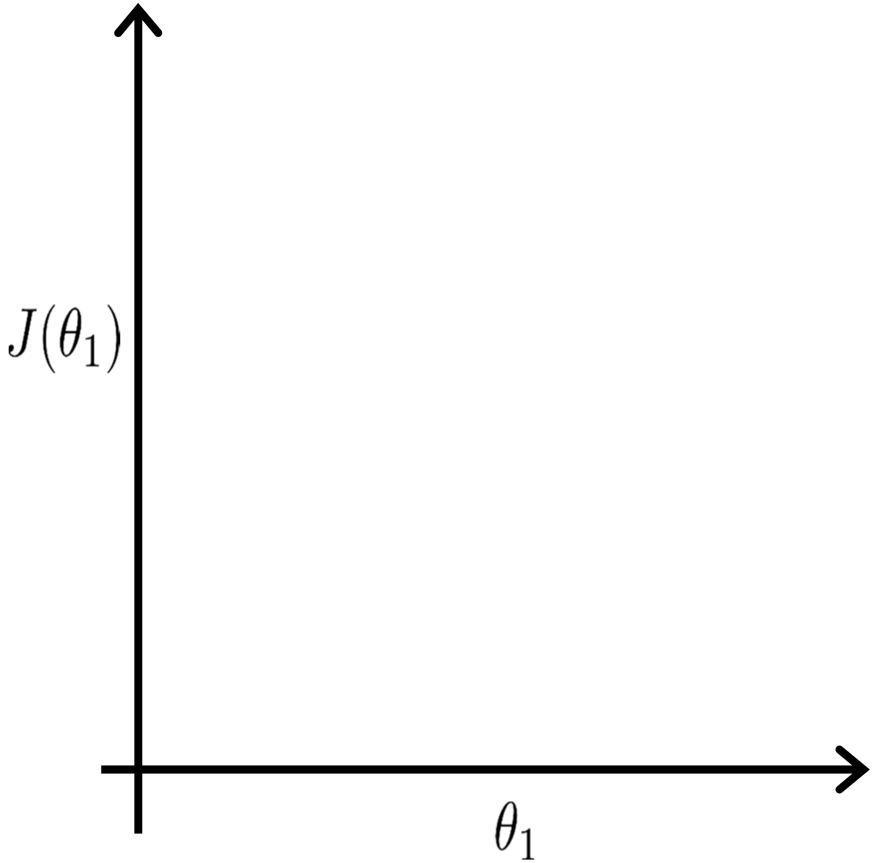
$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

▪最速下降法会收敛到局部最小值，即便修正学习速率α，仍不会改变这个结果。

▪最速下降算法-学习速率

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

▪当接近局部最小值时，因为斜率减小，最速下降法会自动减少每步的补偿，因此不需随时调整学习速率 α



▪最速下降算法

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

▪线性回归模型

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

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

▪最速下降法-梯度计算

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

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

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

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

▪最速下降法

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

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

同步更新 θ_0 和 θ_1

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

▪最速下降算法

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

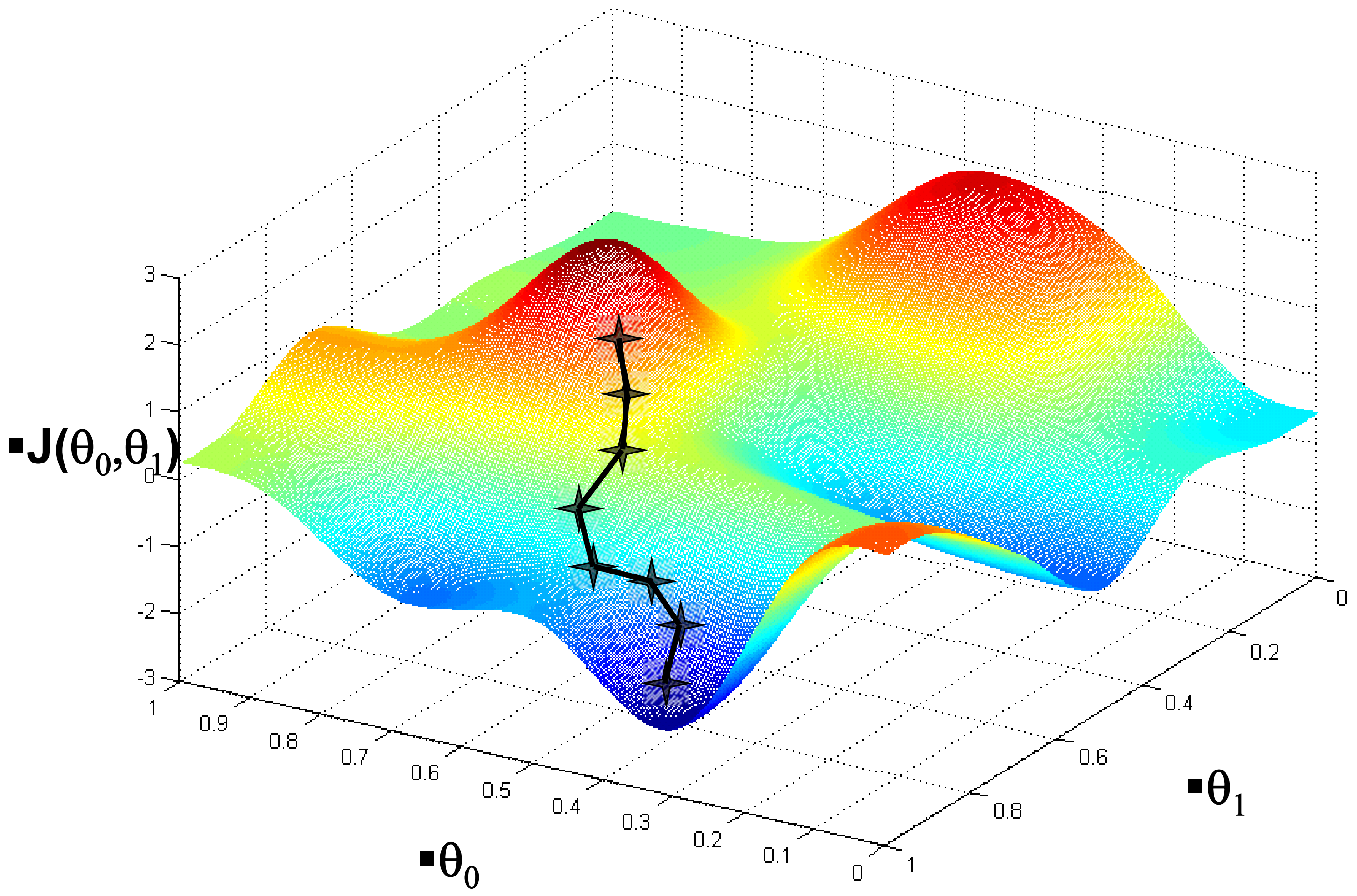
▪不正确做法:

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

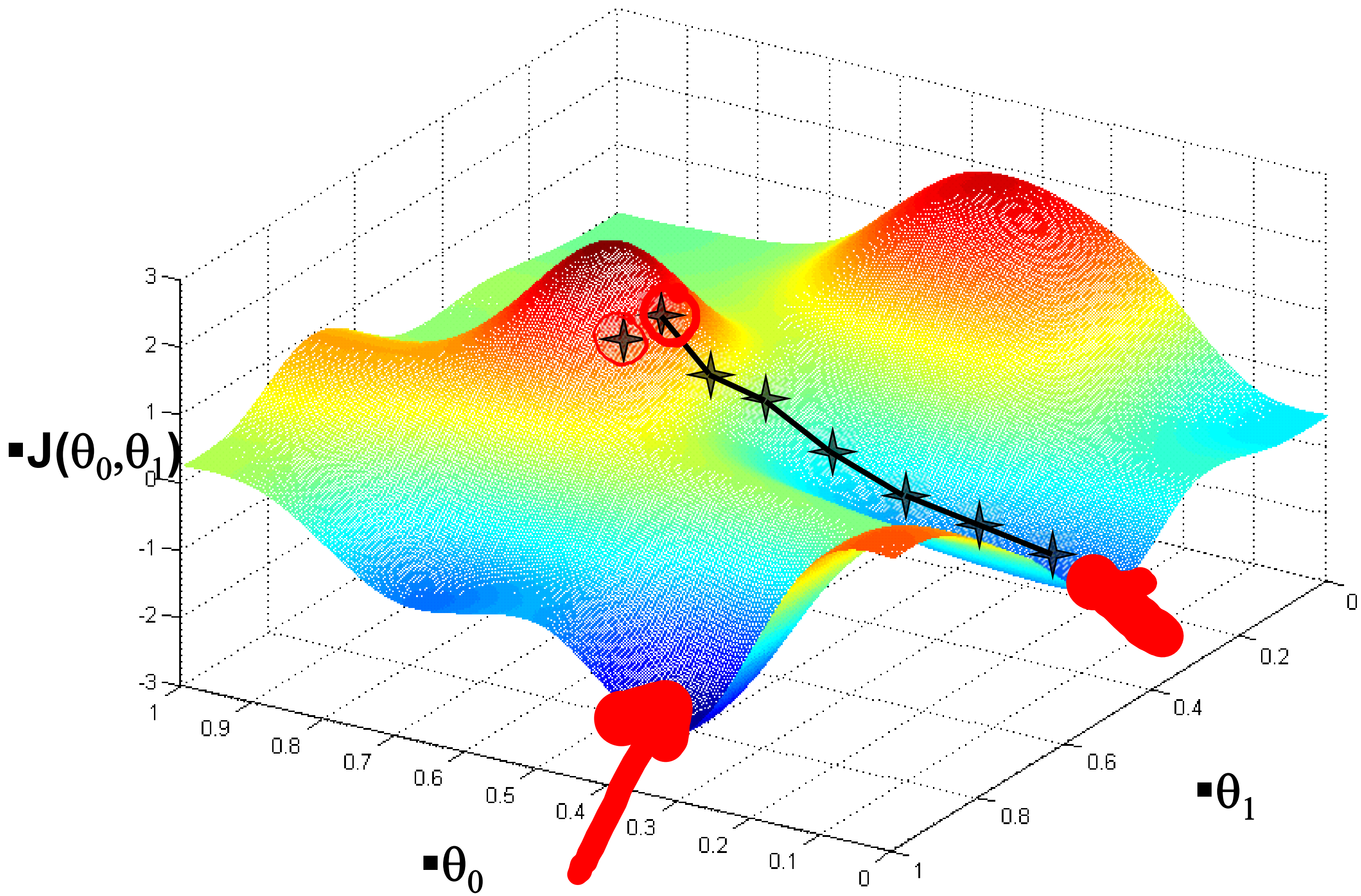
▪正确的做法:同时更新

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

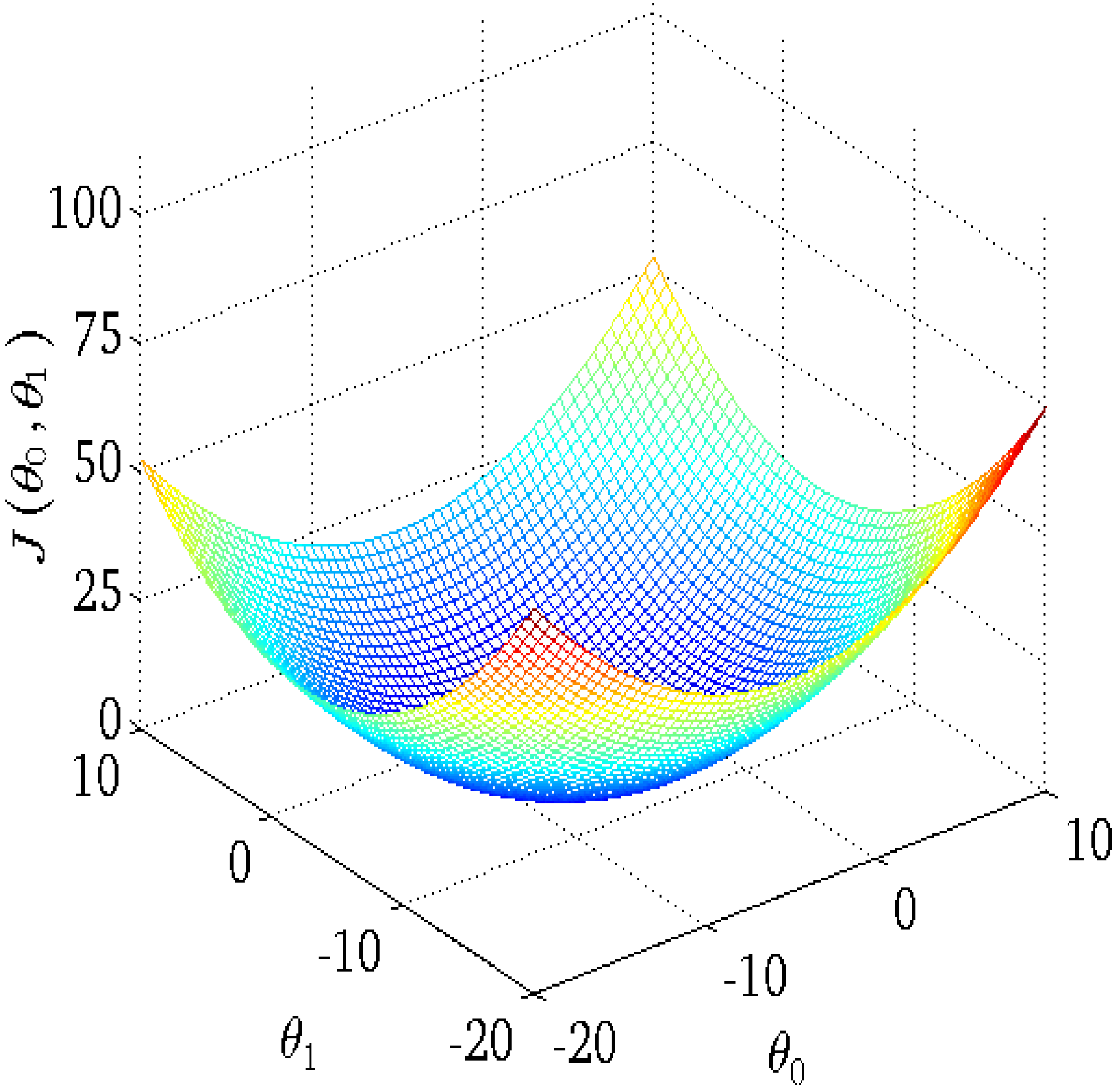
▪初始值



▪初始值

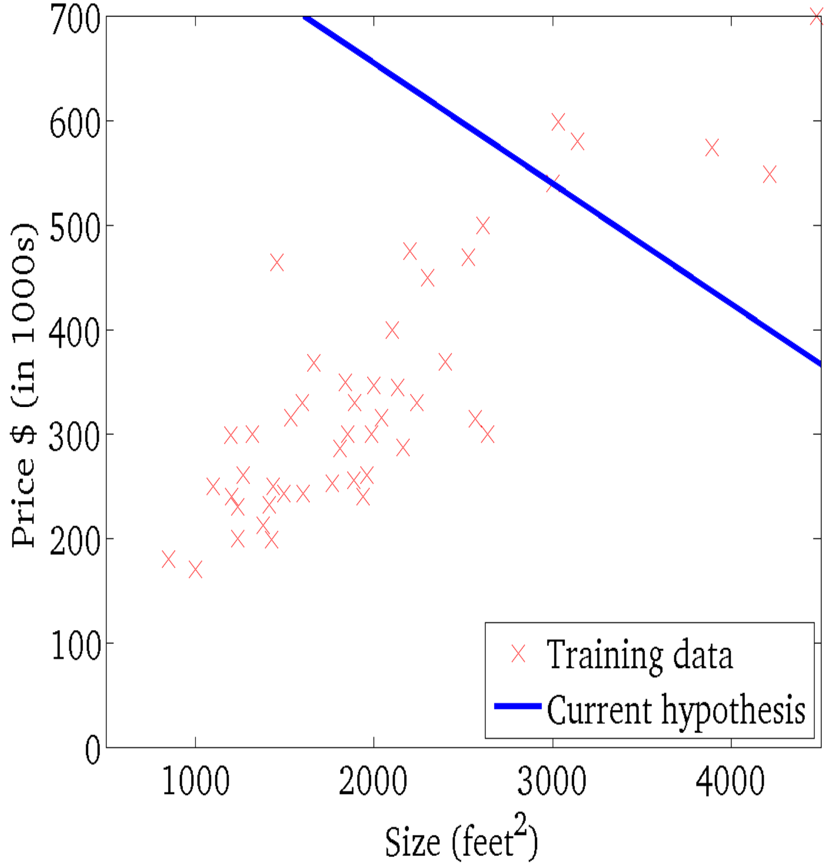


- 尽量构建一个凸函数形式的目标函数



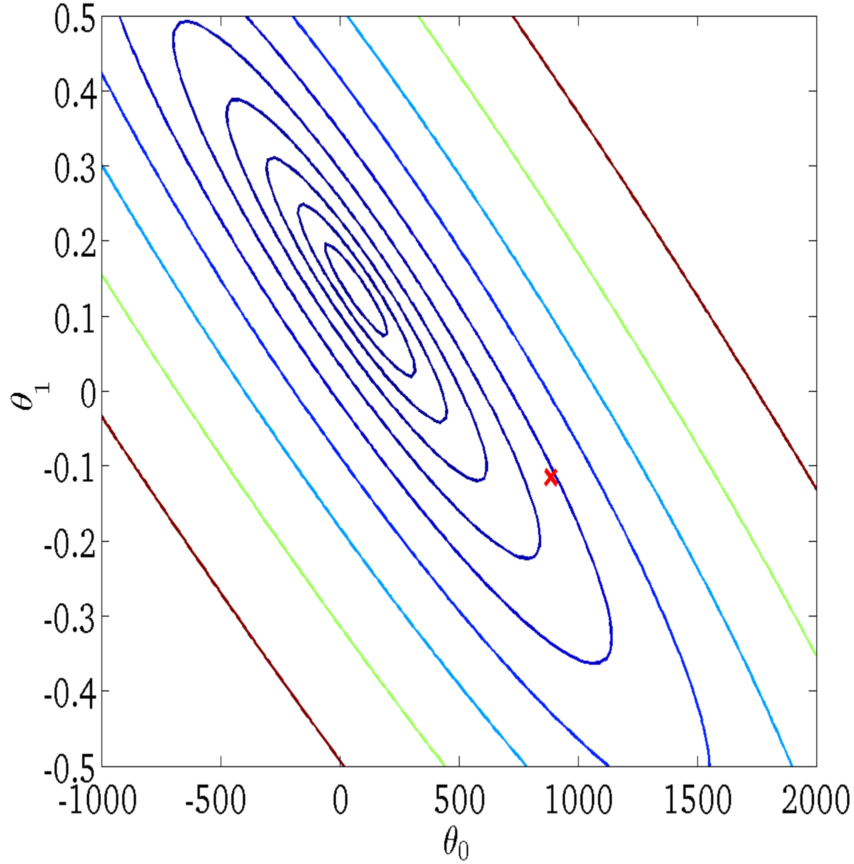
$$h_\theta(x)$$

固定θ_0, θ_1**，模型假设是x的函数**



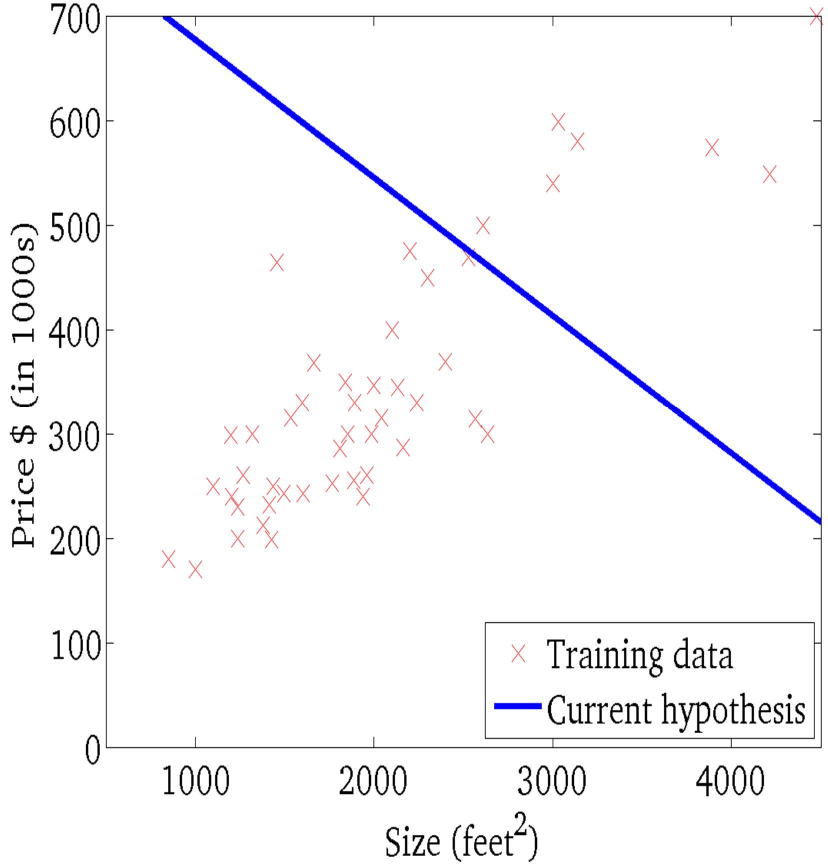
$$J(\theta_0, \theta_1)$$

目标函数是关于 θ_0, θ_1 **的函数**



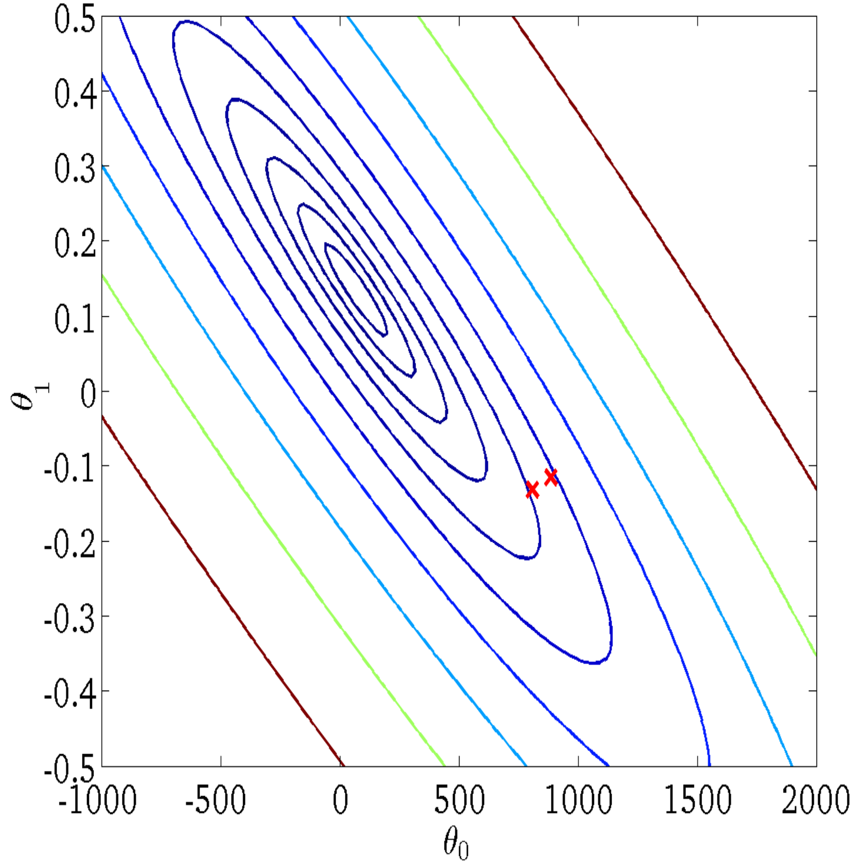
$h_\theta(x)$

固定 θ_0, θ_1，模型假设是x的函数



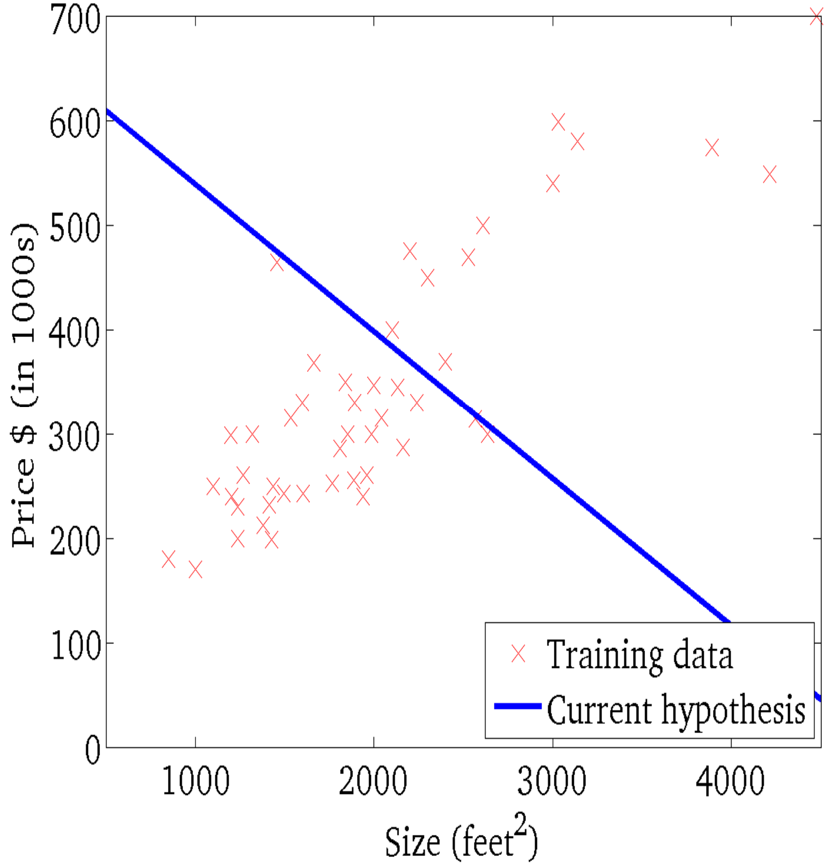
$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 的函数



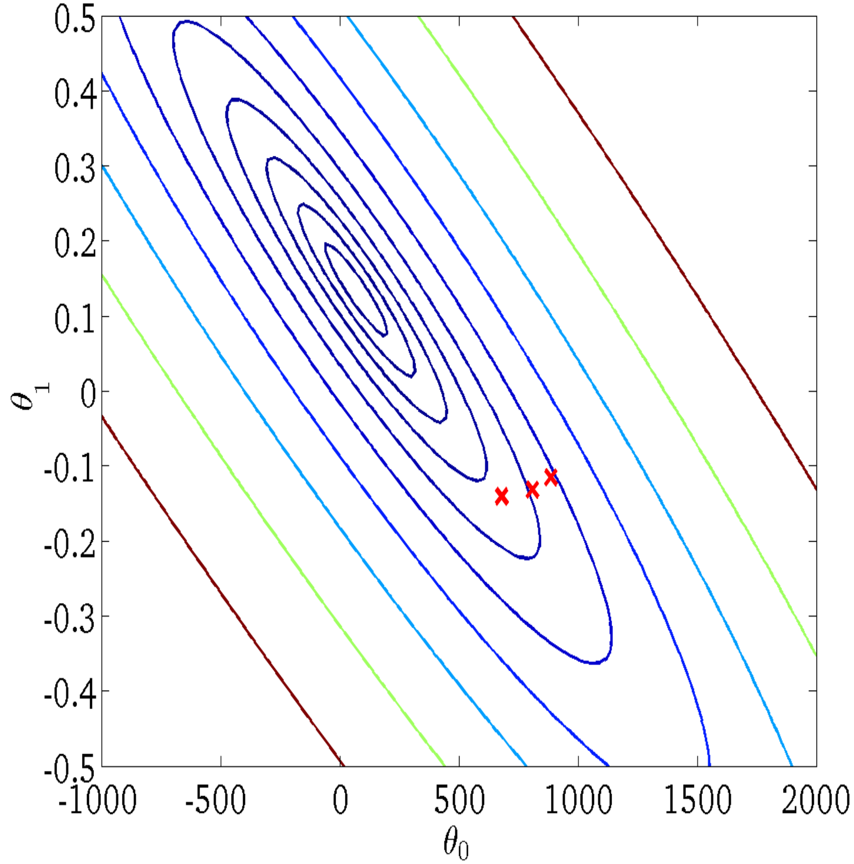
$$h_\theta(x)$$

固定 θ_0, θ_1，模型假设是x的函数



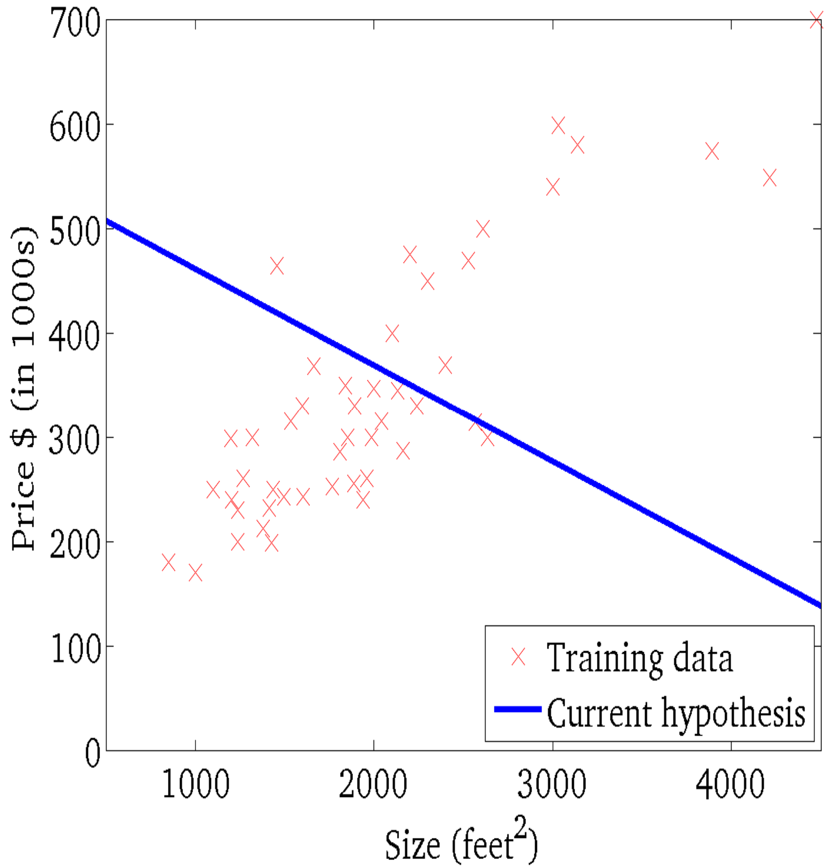
$$J(\theta_0, \theta_1)$$

目标函数是关于 θ_0, θ_1 的函数



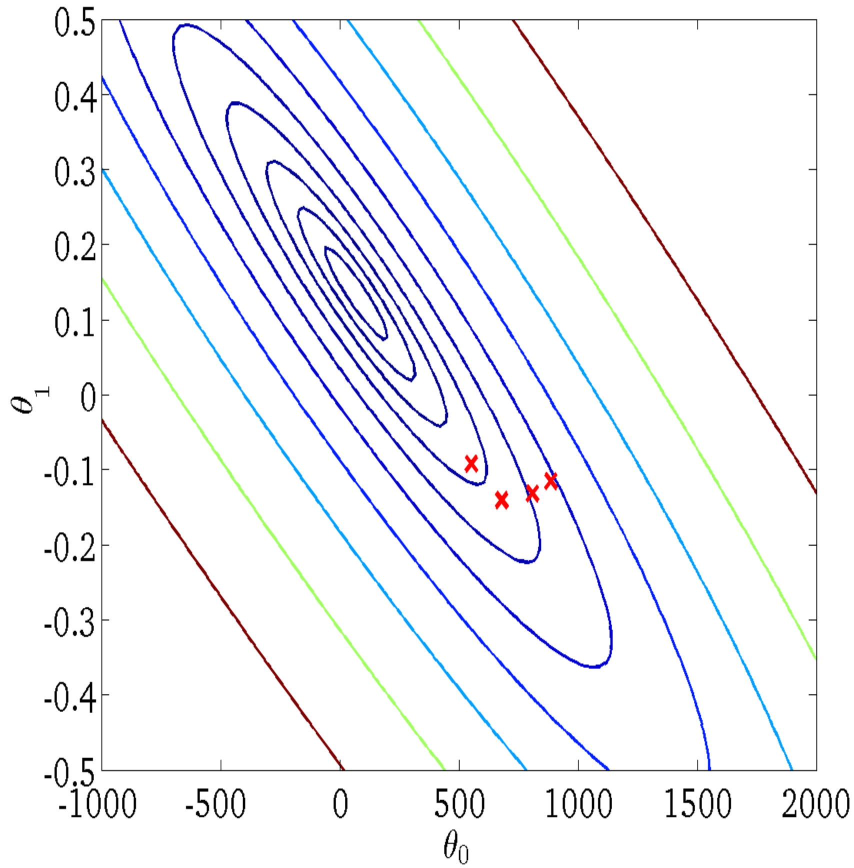
$h_\theta(x)$

固定 θ_0, θ_1**，模型假设是x的函数**



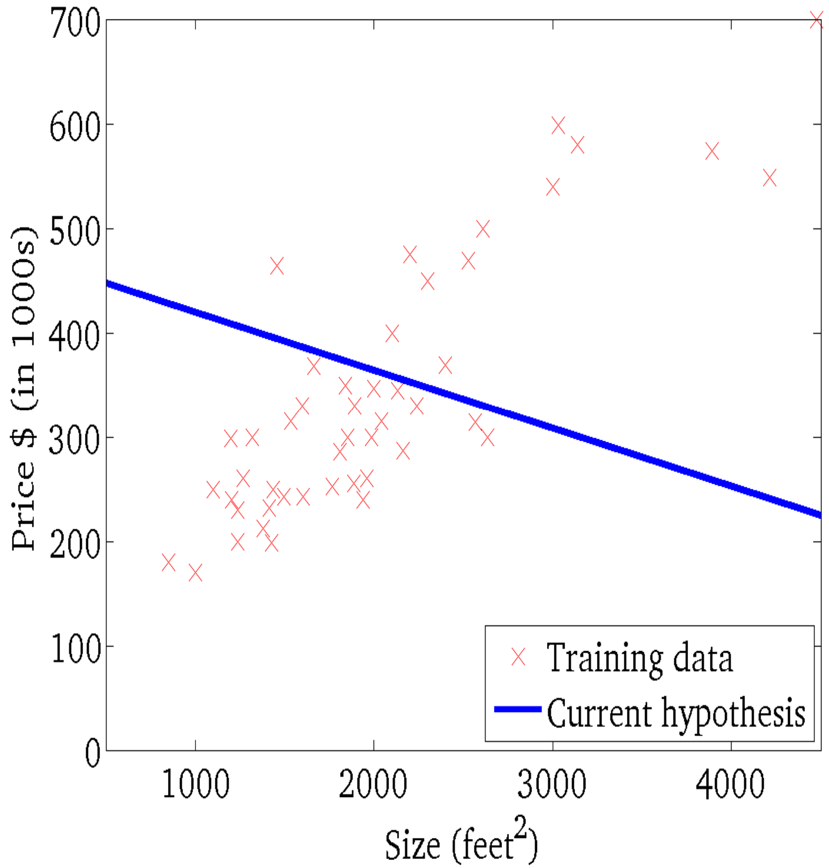
$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 **的函数**



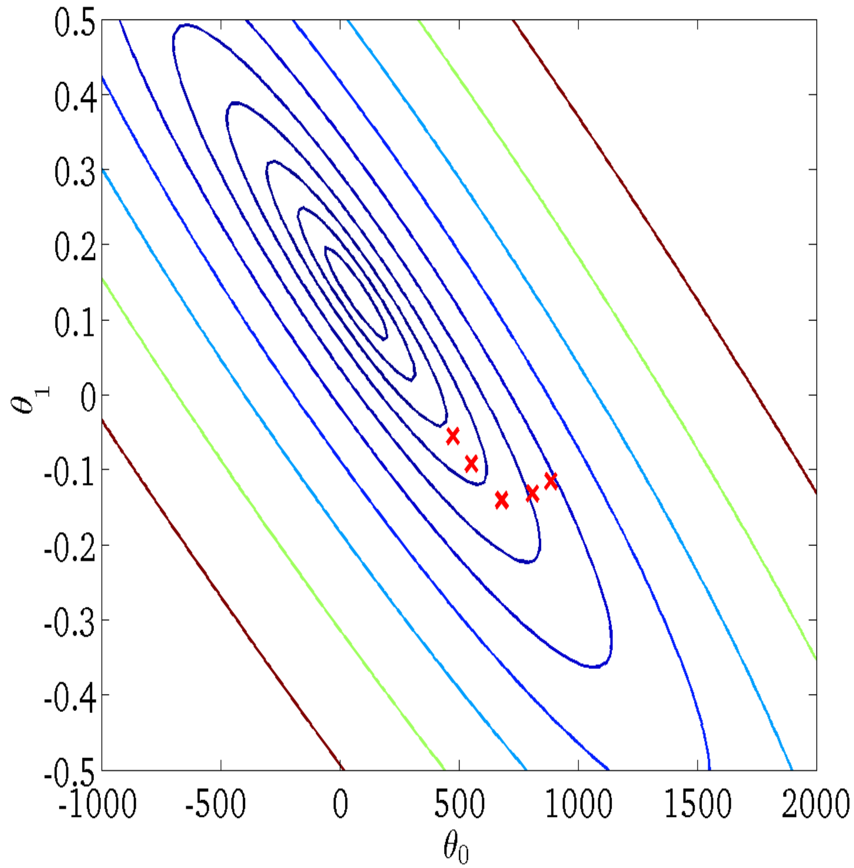
$$h_\theta(x)$$

固定 θ_0, θ_1，模型假设是x的函数



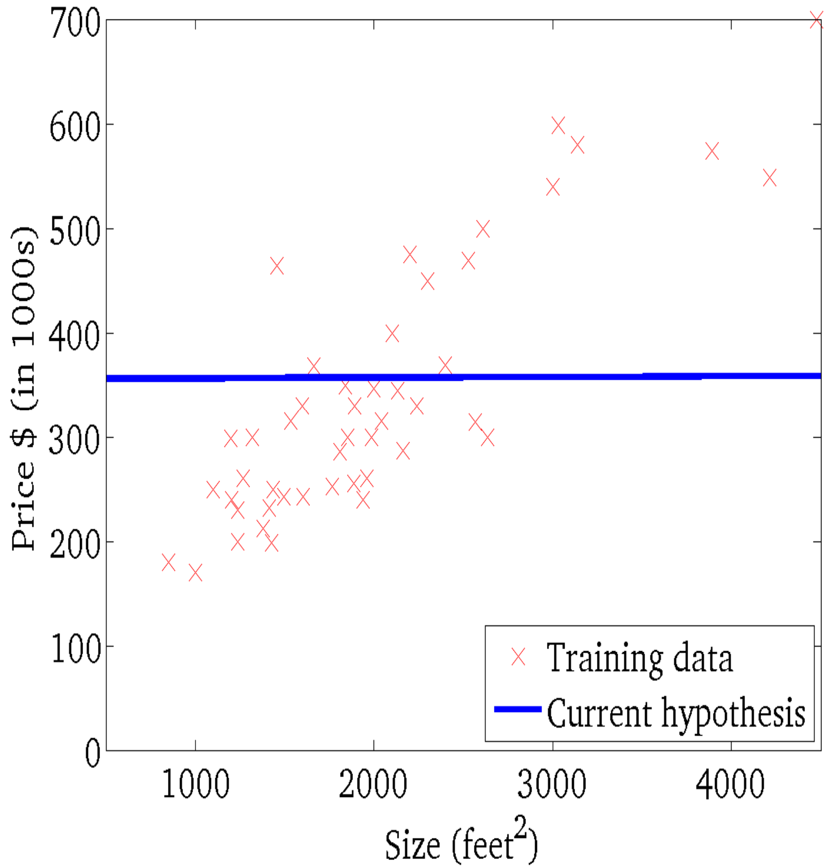
$$J(\theta_0, \theta_1)$$

目标函数是关于 θ_0, θ_1 的函数



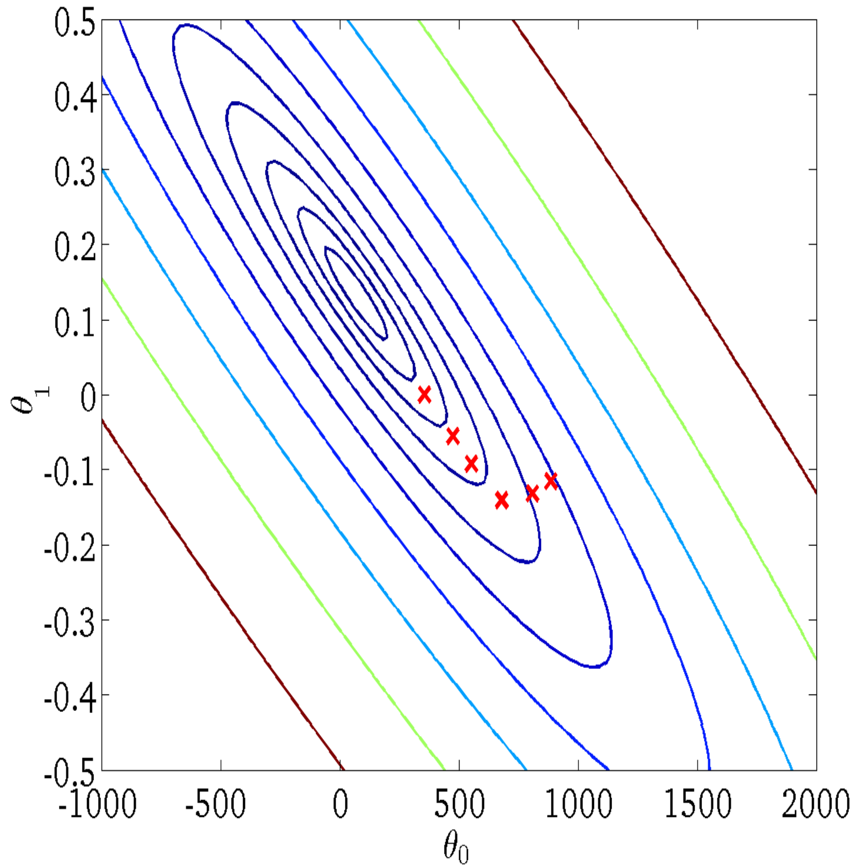
$$h_\theta(x)$$

固定 θ_0, θ_1**，模型假设是x的函数**



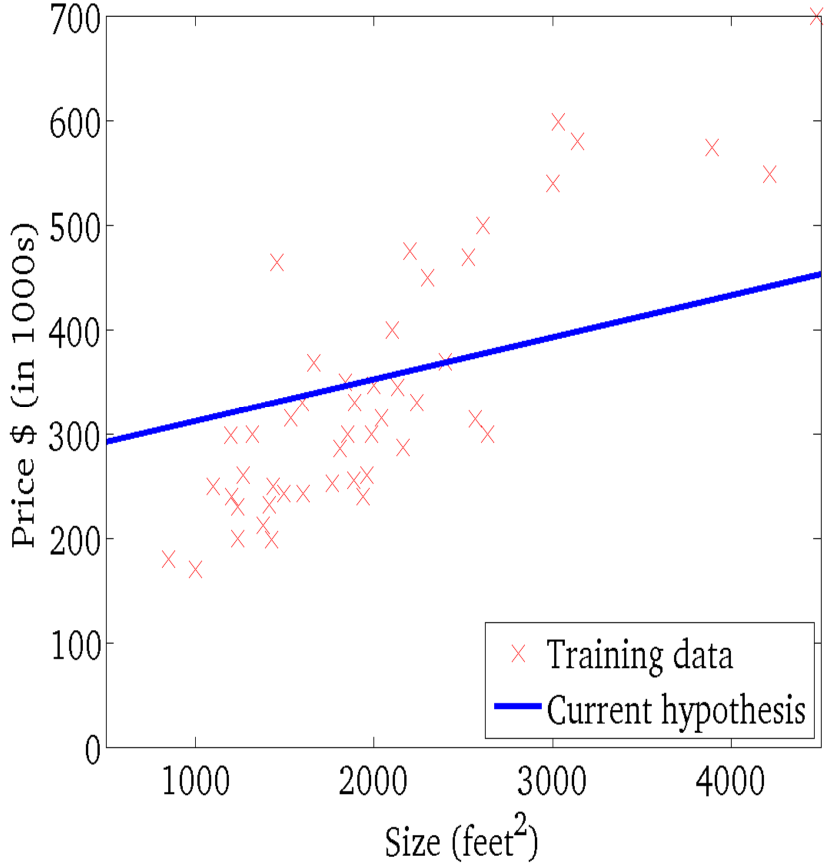
$$J(\theta_0, \theta_1)$$

目标函数是关于 θ_0, θ_1 **的函数**



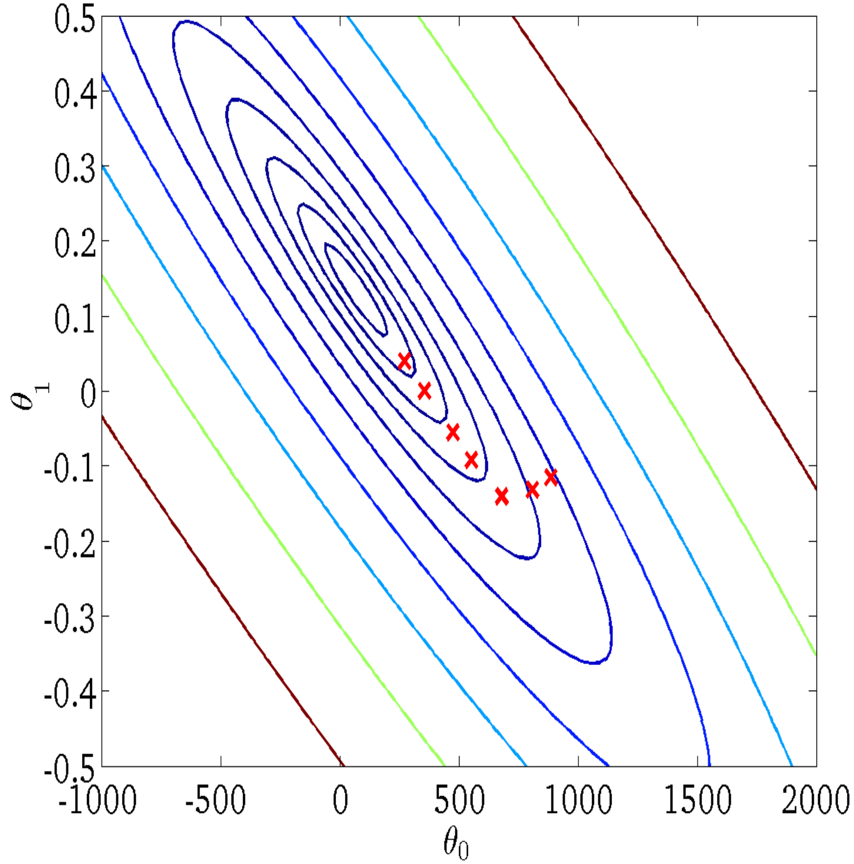
$h_\theta(x)$

固定 θ_0, θ_1**，模型假设是x的函数**



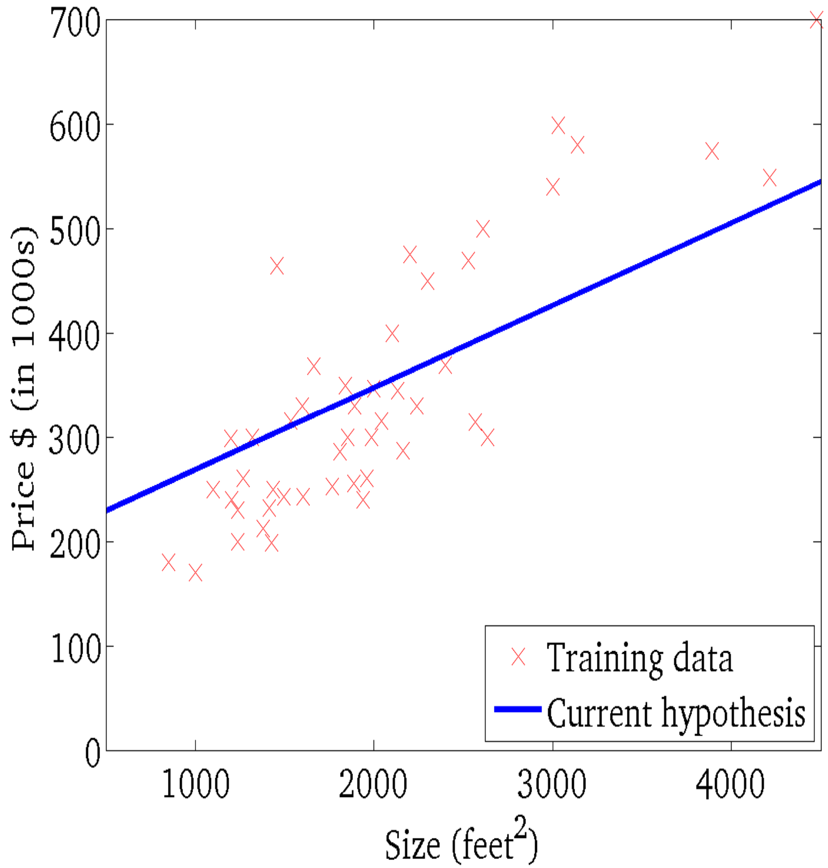
$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 **的函数**



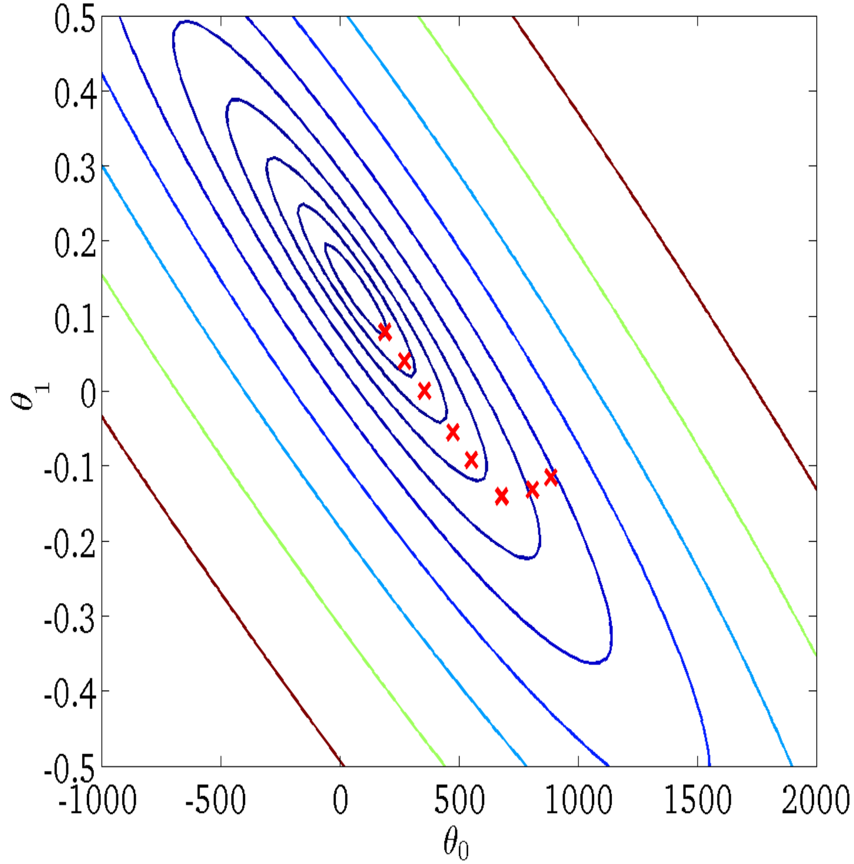
$$h_\theta(x)$$

固定 θ_0, θ_1**，模型假设是x的函数**



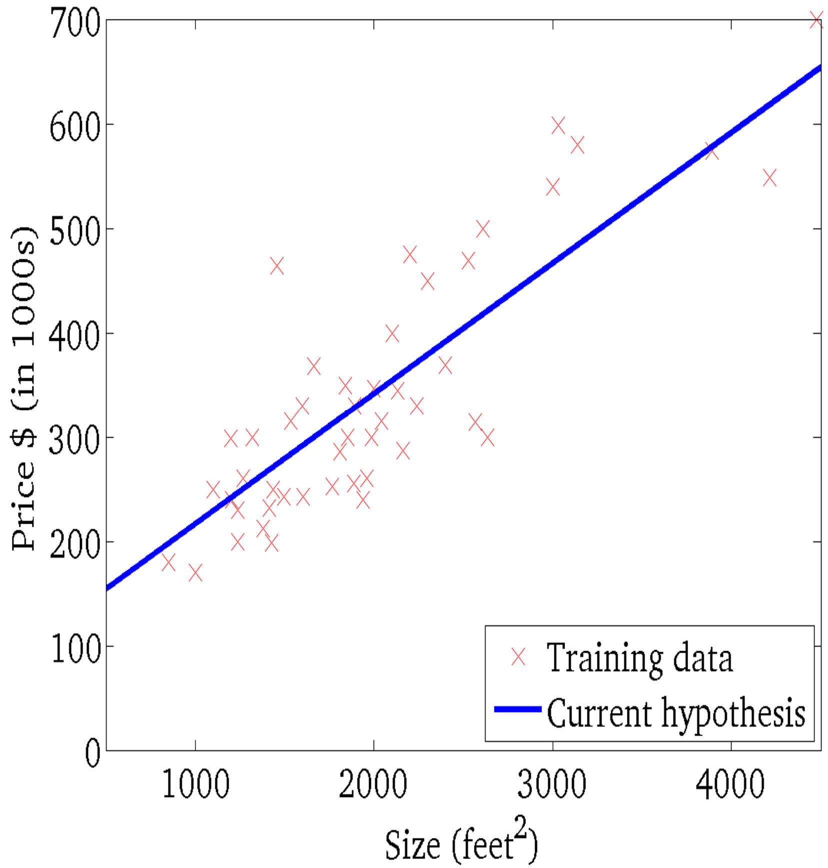
$$J(\theta_0, \theta_1)$$

目标函数是关于 θ_0, θ_1 **的函数**



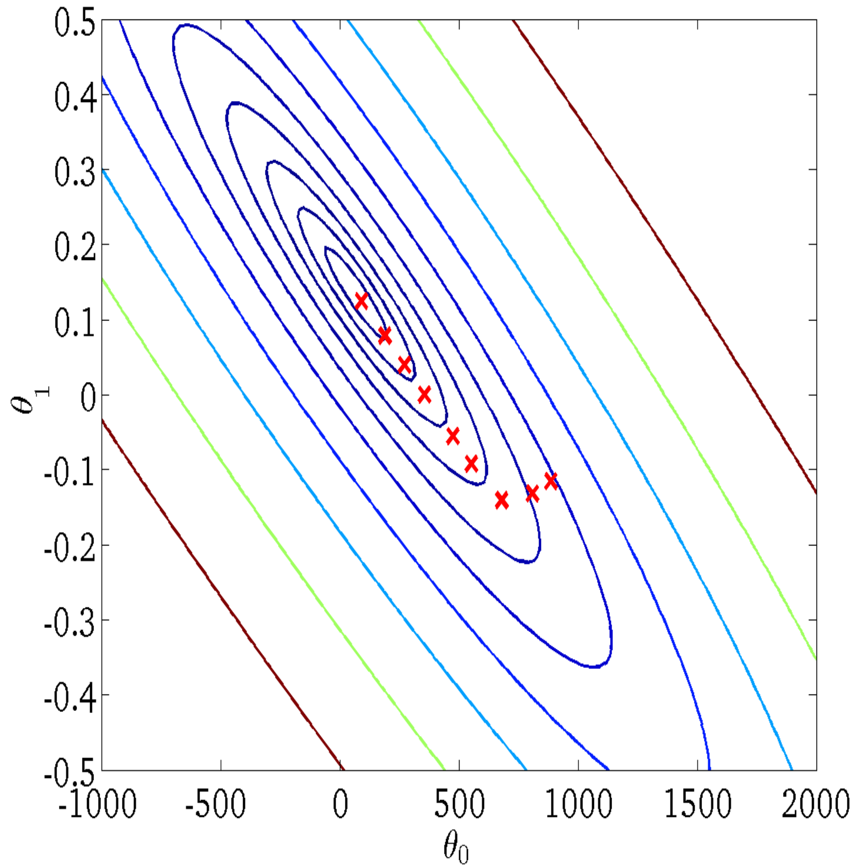
$h_\theta(x)$

固定 θ_0, θ_1**，模型假设是x的函数**



$J(\theta_0, \theta_1)$

目标函数是关于 θ_0, θ_1 **的函数**



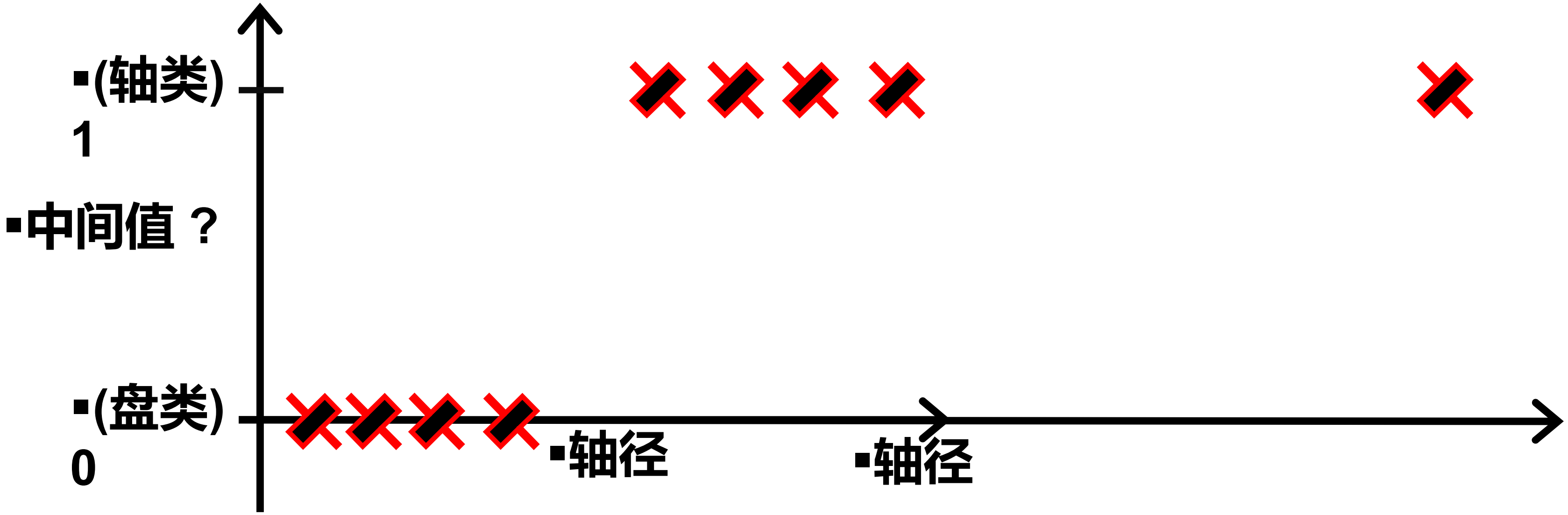
大纲

- 数据的表达
- 数据模型
- 线性回归方法
- 线性分类方法
- 扩展

- 分类
  - 零件: 盘类 / 轴类?
  - 邮件: 垃圾 / 正常?
  - 肿瘤: 恶性 / 良性 ?
  - 恐怖事件：分级

$$y \in \{0, 1\}$$

- 0: 类别0(例如：盘类 零件)
- 1: 类别1 (例如：轴类零件)



▪分类阈值 0.5: $h_\theta(x)$

▪如果 $h_\theta(x) \geq 0.5$ ，预测 “y = 1”

▪如果 $h_\theta(x) < 0.5$ ，预测 “y = 0”

- 分类: y = 0 or 1

  $h_\theta(x)$ ▪可以 > 1 或 < 0

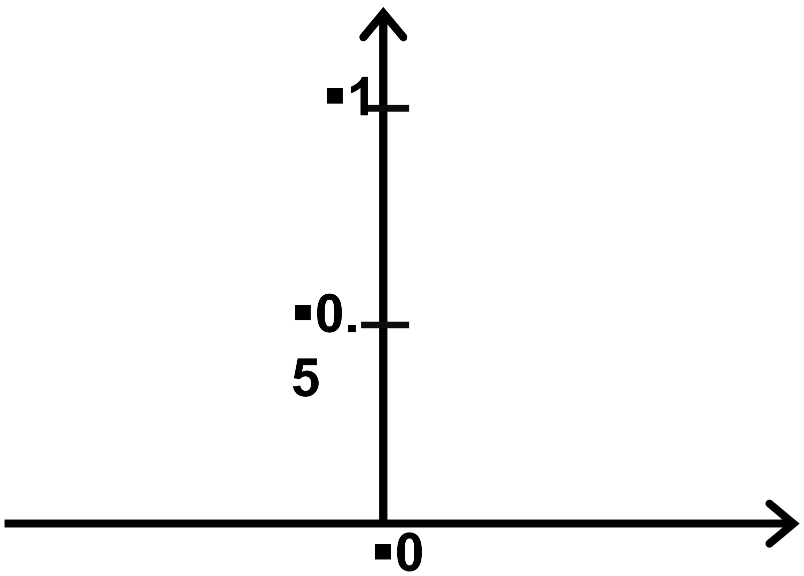
- Logistic 回归: $0 \leq h_\theta(x) \leq 1$

- Logistic 回归模型
- 希望 $0 \leq h_\theta(x) \leq 1$

$$h_\theta(x) = g(\ \theta^T x\ )$$

$$g(z) = \frac{1}{1+e^{-z}}$$

- S型函数
- Logistic 函数



▪模型假设输出的含义

▪输出

$h_\theta(x)$ ▪= 估算输入为x，输出 y = 1的概率

▪例子: 如果 ▪$x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{轴径} \end{bmatrix}$

$$h_\theta(x) = 0.7$$

▪告诉我们，此零件有70% 机会是轴类零件

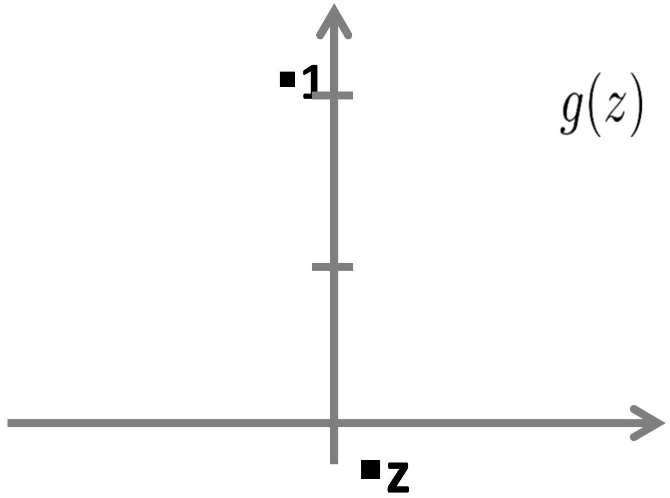
▪确定参数 θ “给定x， y = 1 的概率为70% ”

$$P(y=0|x;\theta) + P(y=1|x;\theta) = 1$$
$$P(y=0|x;\theta) = 1 - P(y=1|x;\theta)$$

▪Logistic 回归

$$h_\theta(x) = g(\theta^T x)$$

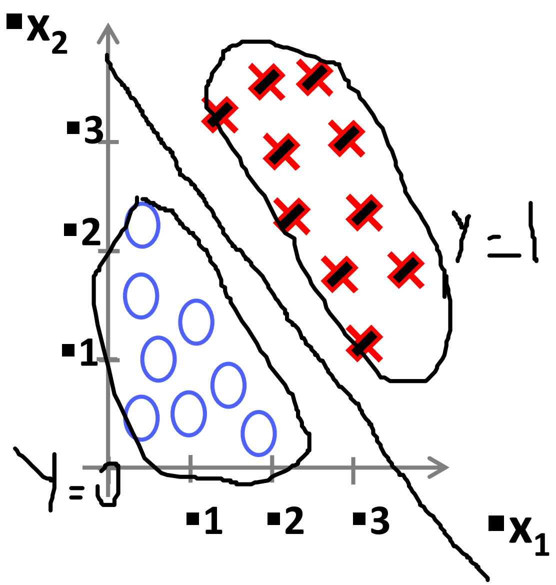
$$g(z) = \frac{1}{1+e^{-z}}$$



▪ 假设预测“$y=1$“ 如果 $h_\theta(x) \geq 0.5$

▪ 预测 “$y=0$“ 如果 $h_\theta(x) < 0.5$

▪决策边界（Decision Boundary）

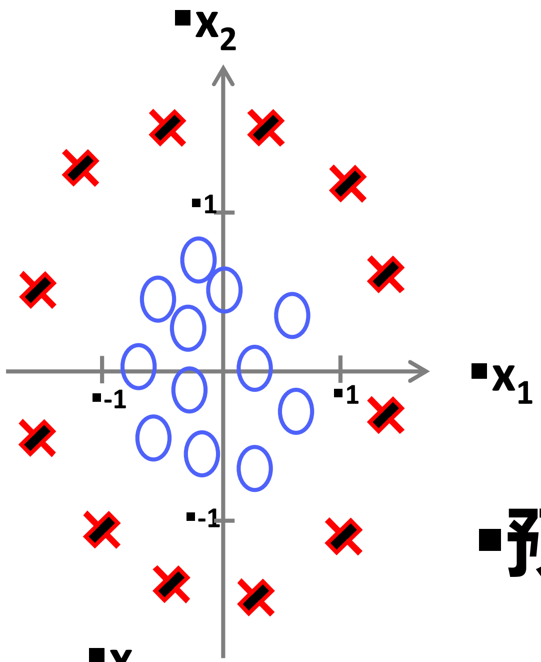


$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

▪预测 "$y = 1$" 如果 $-3 + x_1 + x_2 \geq 0$

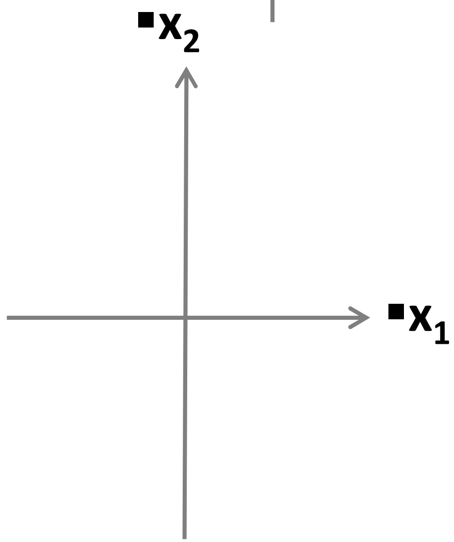
$x_1 + x_2 \geq 3$

- 非线性的决策边界



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

- 预测“ $y = 1$” 如果 $-1 + x_1^2 + x_2^2 \geq 0$

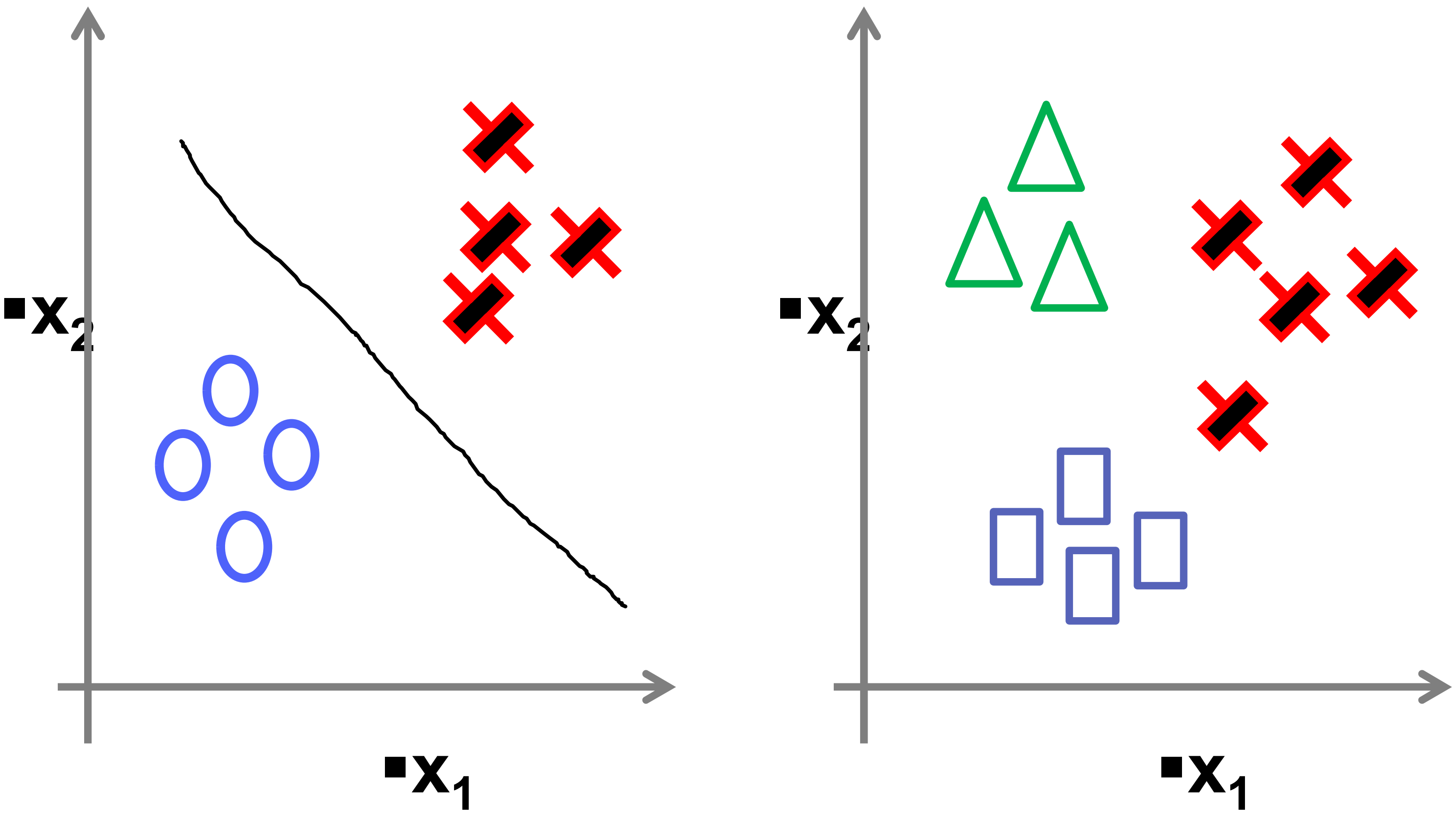


$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_1^2 x_2 + \theta_5 x_1^2 x_2^2 + \theta_6 x_1^3 x_2 + \dots)$$

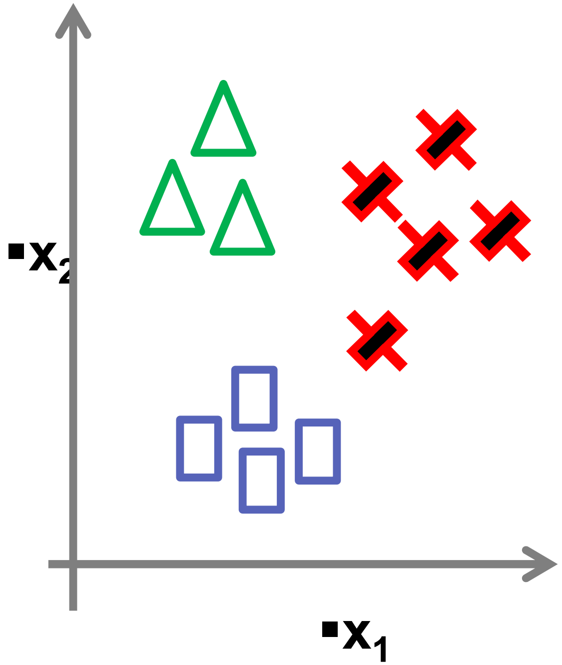
- 多类别问题
- Email目录/标签:工作, 朋友, 家庭, 临时
- 医疗诊断: 健康, 感冒, 传染病
- 天气: 晴朗, 多云, 阴雨, 下雪

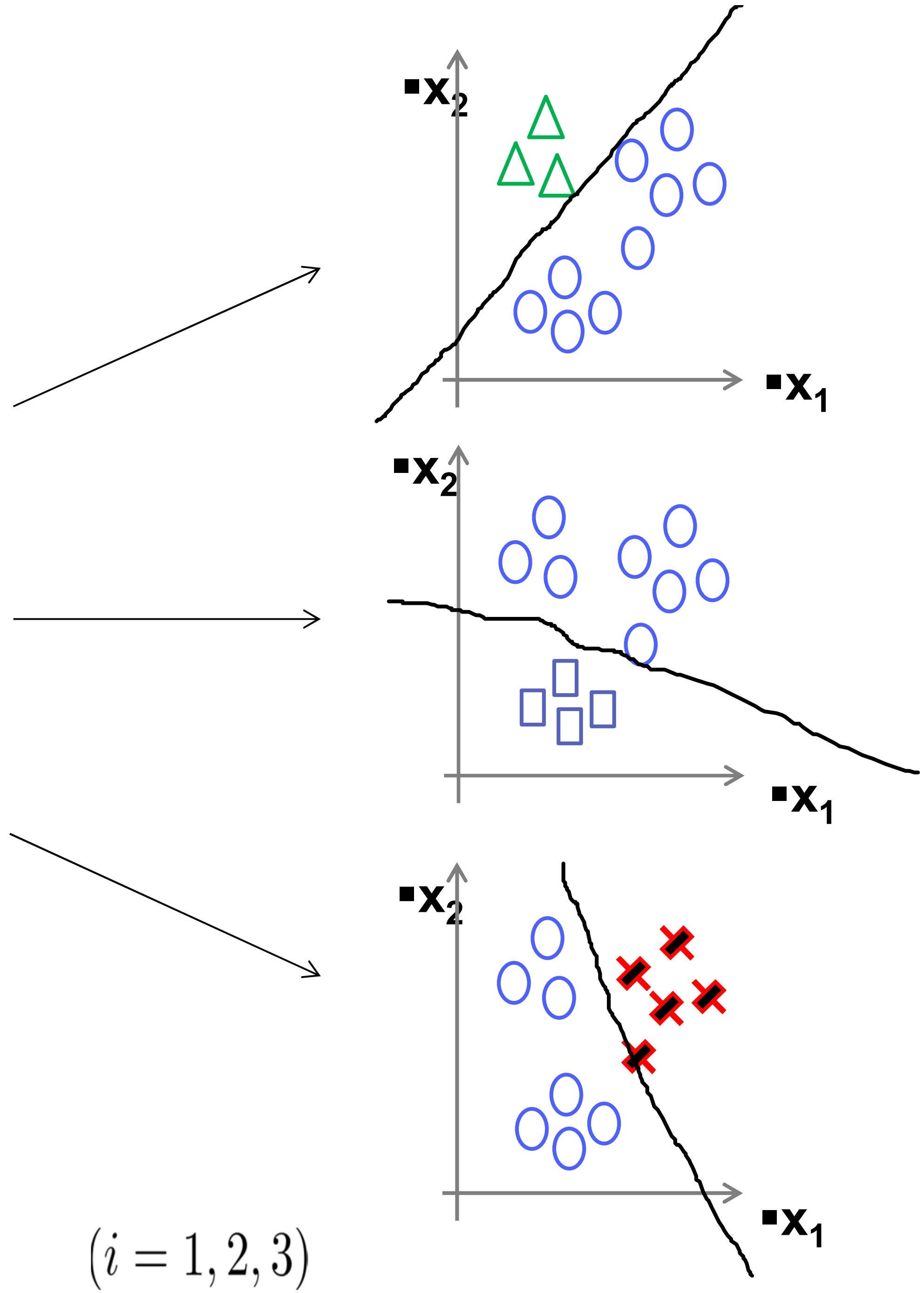
▪2类边界:

▪多类别:



- 一对多:





- Class 1:
- Class 2:
- Class 3:

$$h_\theta^{(i)}(x) = P(y = i | x; \theta) \qquad (i = 1, 2, 3)$$

▪一对多:

▪对所有类别分别训练Logistic分类器 $h_\theta^{(i)}(x)$ ，分别预测 $y = i$.

▪对于每一个输入 x ，分别预测是否属于某个分类，取假设输出概率最大的一个类别 i

$$\max_i h_\theta^{(i)}(x)$$

Bagging

大纲

- 数据的表达
- 数据模型
- 线性回归方法
- 线性分类方法
- 扩展

统计学习

统计学习方法有很多

- 神经网络/深度网络
- K近邻
- 朴素贝叶斯方法
- 决策树
- 支持向量机
- EM方法
- 隐马尔科夫模型
- 条件随机场
- Boosting
- Deep Learning
- 流形学习
- …

Boosting

- 学习复杂分类器的简单方法
  - Freund & Shapire, 1995
  - Friedman, Hastie, Tibshhirani, 1998

- 容易实现，不需要外部工具支持

Boosting

❿定义:

$$F(x) = w_1 f_1(x) + w_2 f_2(x) + w_3 f_3(x) + \ldots$$

强分类器

特征向量

▪权值

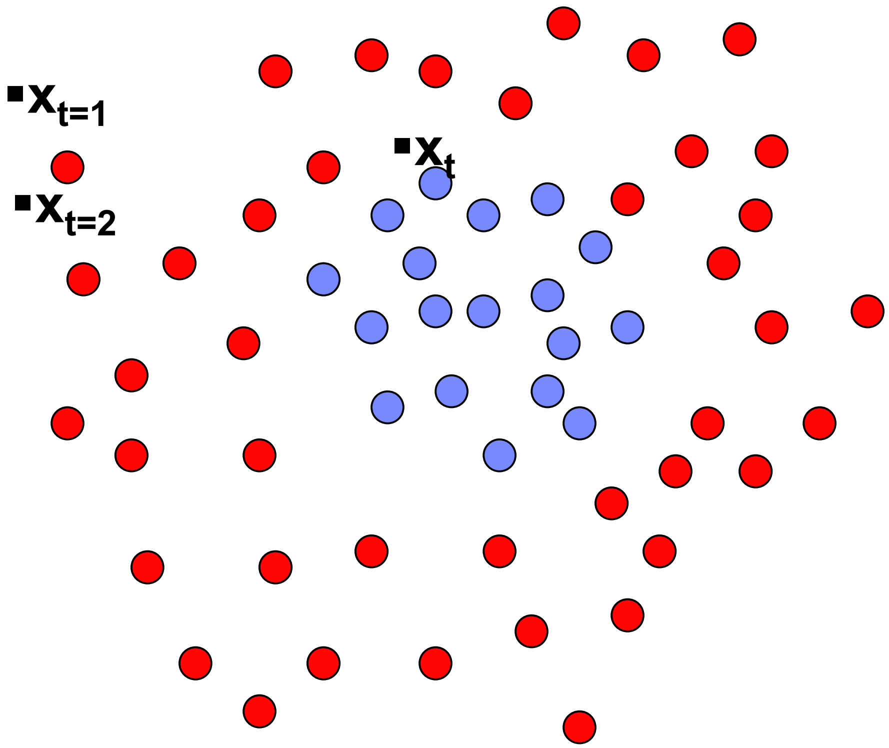
▪弱分类器

- 我们需要定义一组弱分类器

$f_k(x)$ **<50%的错误率**

Boosting

▪ 一个有序叠加的分类过程:



▪每个数据x_t都有一个标记y_{it} :

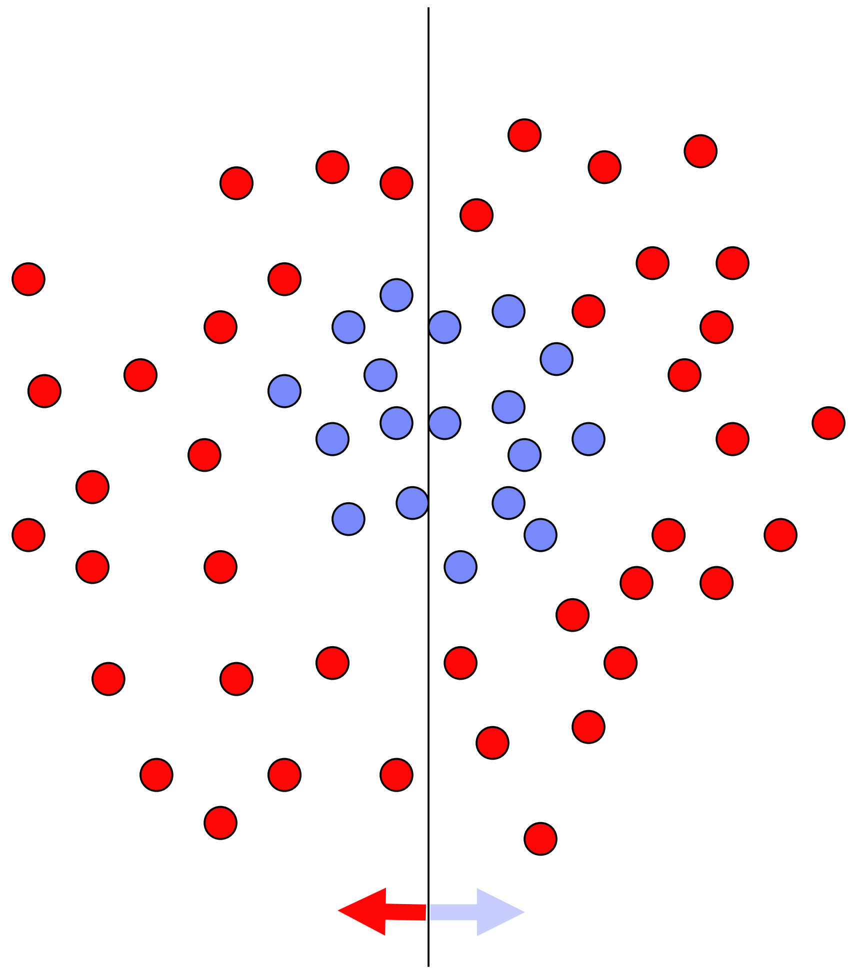
$$y_t = \begin{cases} +1 \ (\ ) \\ -1 \ (\ ) \end{cases}$$

▪以及权值:

▪$w_t = 1$

一个简单例子

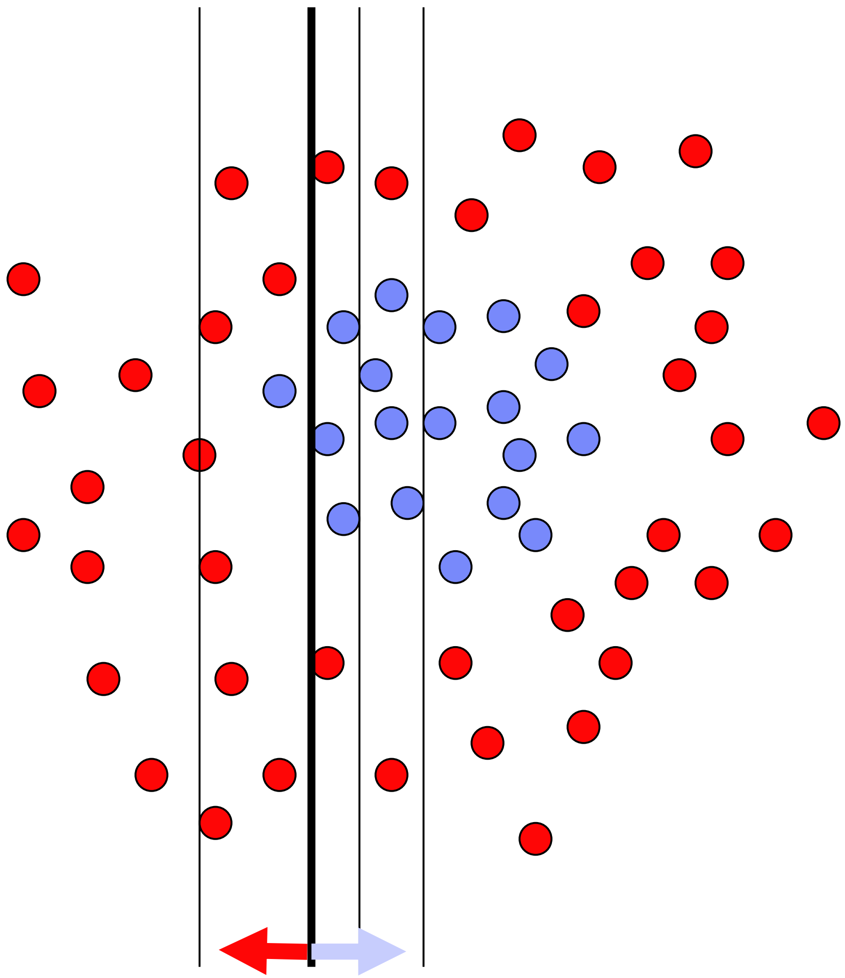
- 弱分类器



- 每个数据x_t都有一个标记y_t :
  - y_t =
    - +1 ( )
    - -1 ( )
- 以及样本权值:
  - w_t =1

- h => p(error) = 0.5 这是一个随机划分

一个简单例子



▪黑线代表一个弱分类器

它仅仅比随机划分的性能稍好一点

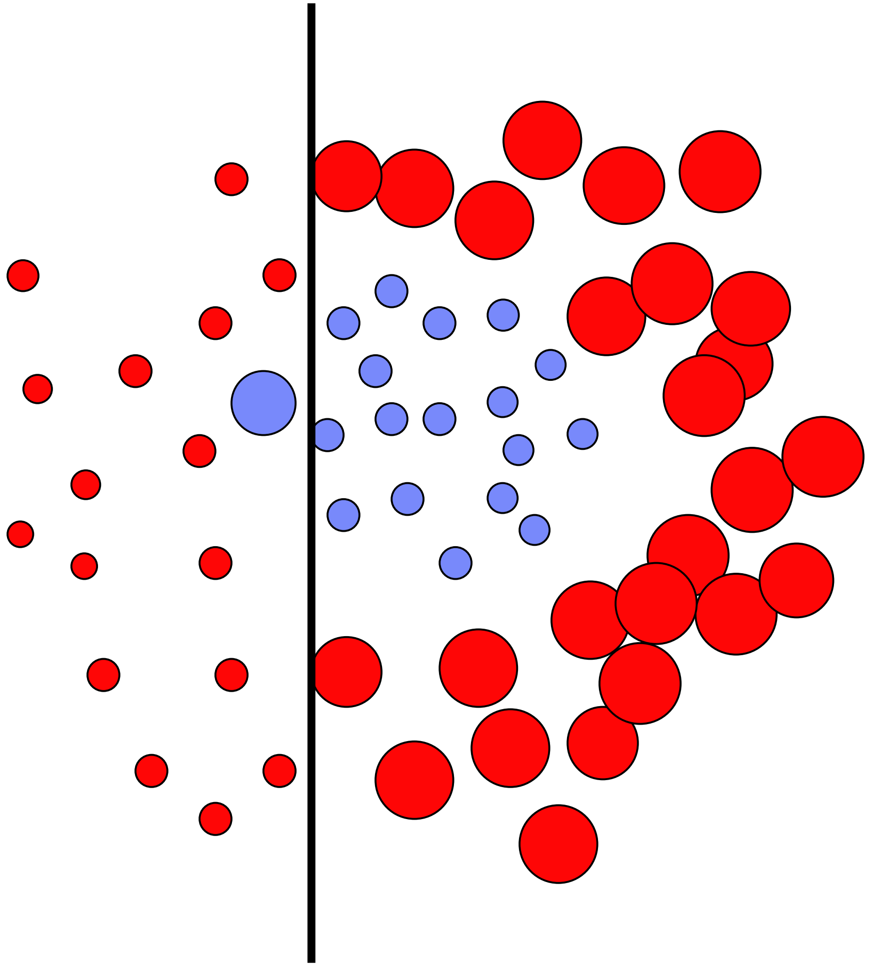
▪每个数据x_t都有一个标记y_t：

$$y_t = \begin{cases} +1 \ (\text{red}) \\ -1 \ (\text{blue}) \end{cases}$$

▪以及样本权值:

▪$w_t = 1$

一个简单例子



- 每个数据x_t都有一个标记y_t：

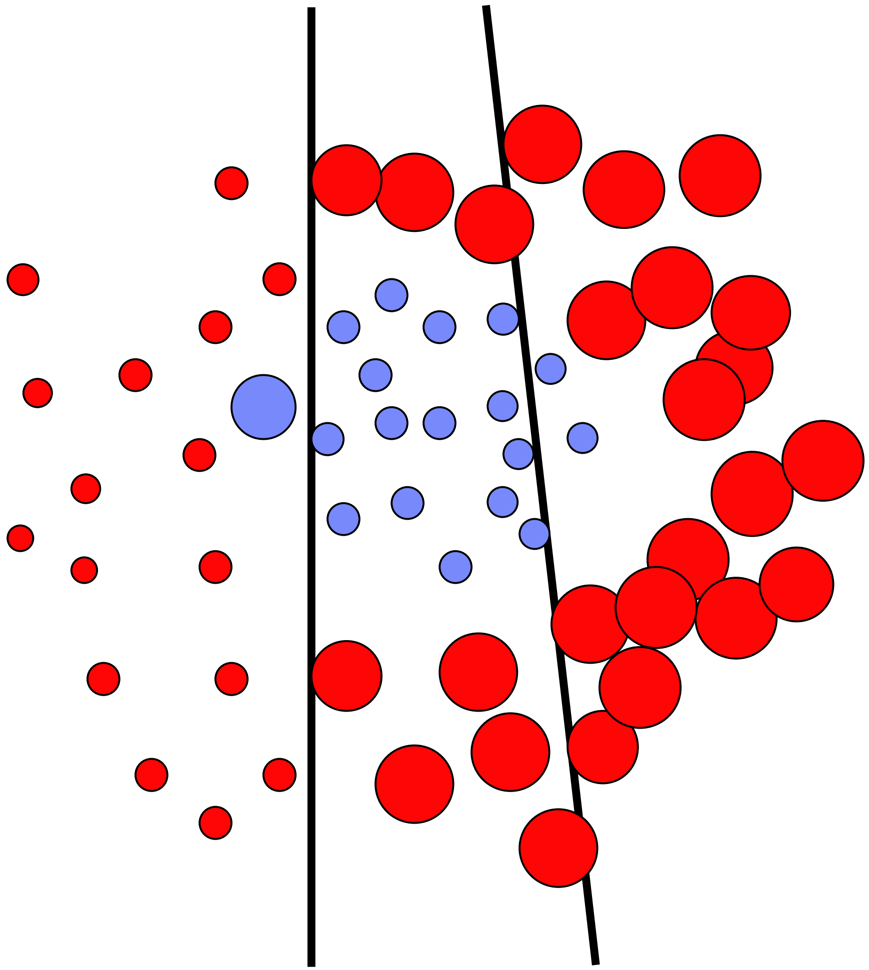
$$y_t = \begin{cases} +1\ (\ ) \\ -1\ (\ ) \end{cases}$$

- 更新权值:

$$w_t \leftarrow w_t \exp\{-y_t H_t\}$$

- 针对现有分类结果，我们添加一个新的分类器

一个简单例子



- 每个数据x_t都有一个标记y_t：

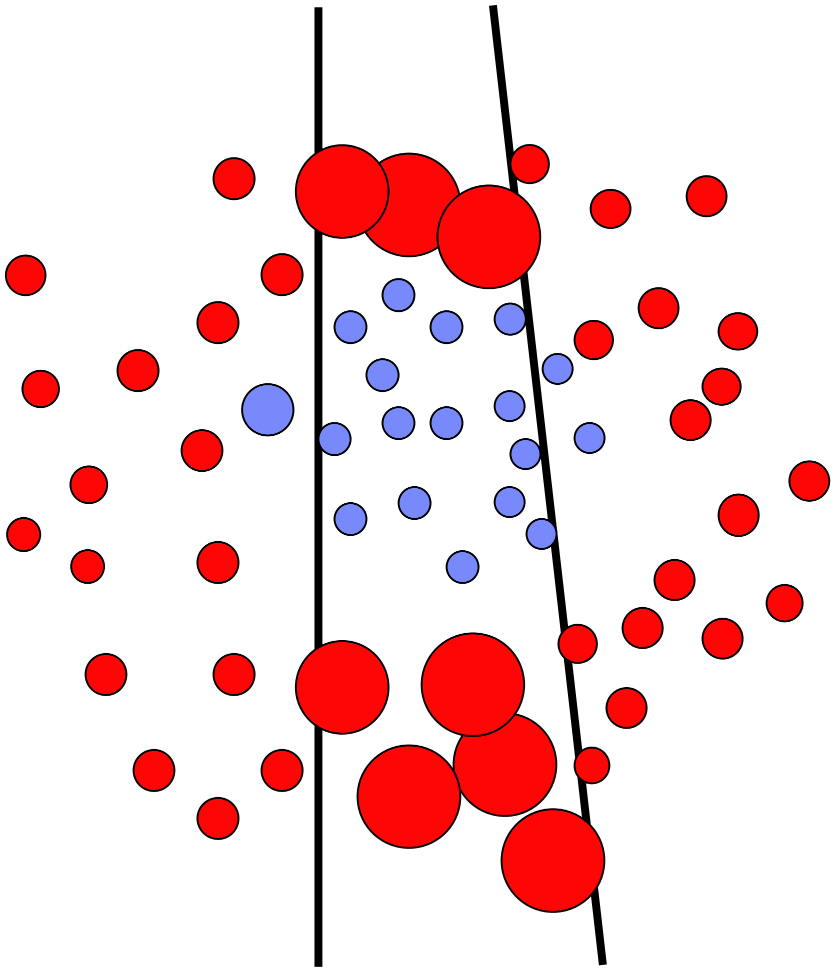
- y_t = { ▪+1 ( ) ; ▪-1 ( ) }

- 更新样本权值:

- $w_t \leftarrow w_t \exp\{-y_t H_t\}$

- 再次针对已有分类结果建立一个新的分类器

一个简单例子



- 每个数据x_t都有一个标记y_t:

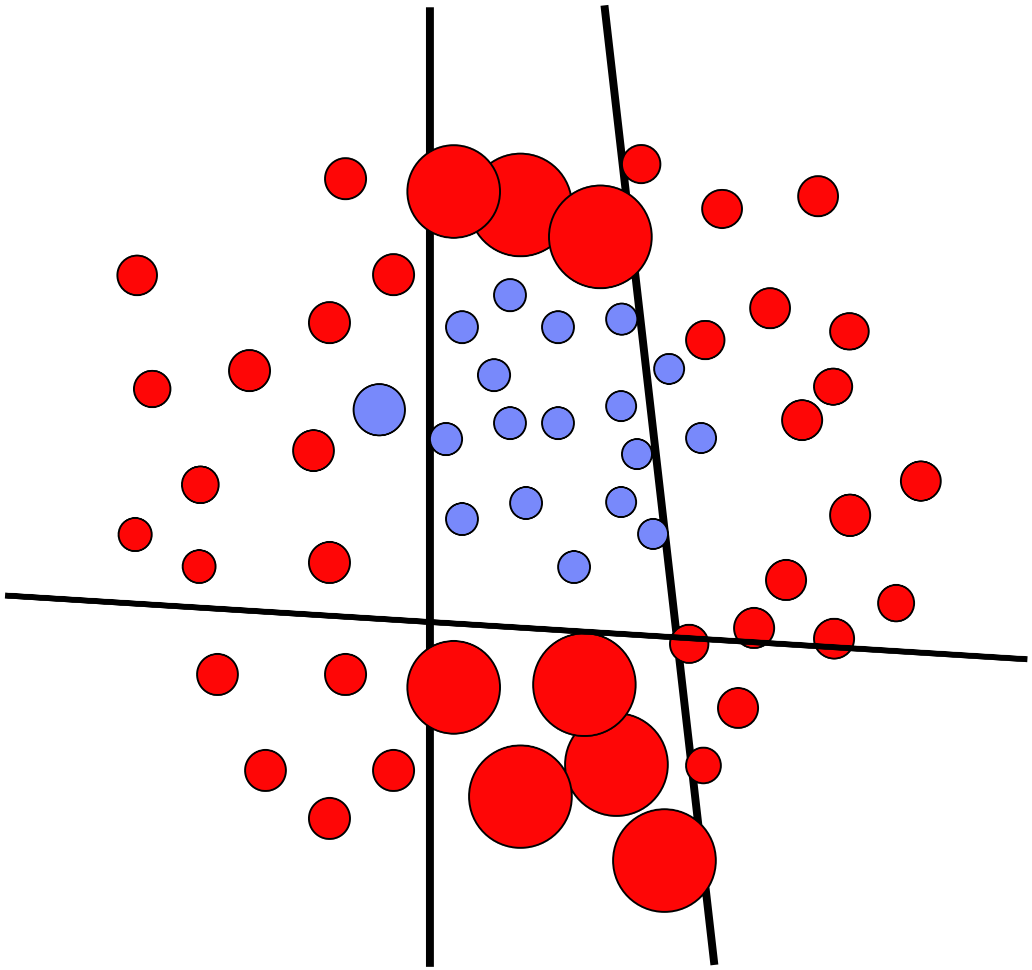
- $y_t = \begin{cases} +1 \ (\ ) \\ -1 \ (\ ) \end{cases}$

- 更新样本权值:

- $w_t \leftarrow w_t \exp\{-y_t H_t\}$

- 再次重新标记

一个简单例子



- 每个数据x_t都有一个标记y_t：

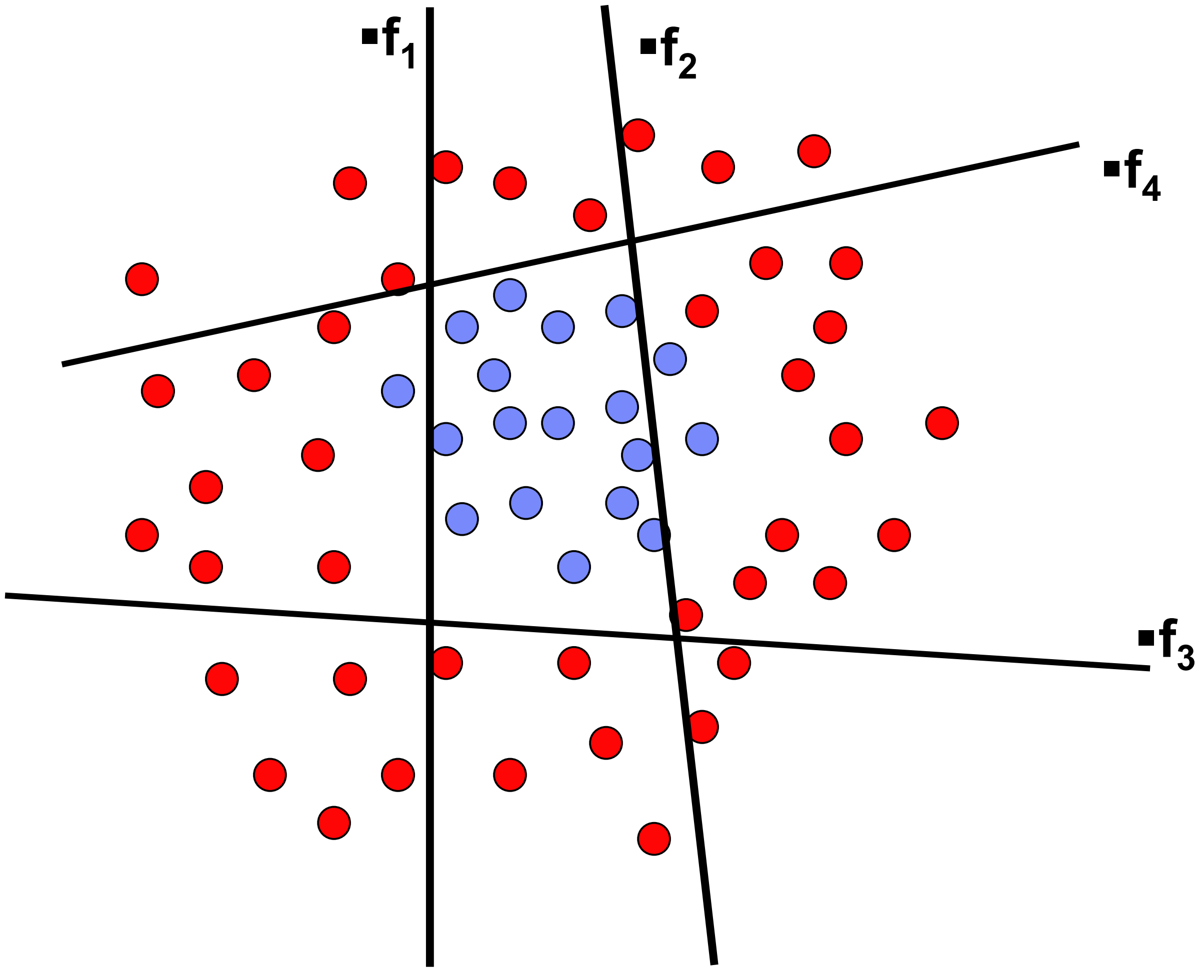
- $y_t = \begin{cases} +1 \ (\ ) \\ -1 \ (\ ) \end{cases}$

- 更新权值:

- $w_t \leftarrow w_t \exp\{-y_t H_t\}$

- 建立新的弱分类器

一个简单例子



▪通过弱分类器(线性)的叠加建立强分类器 (非线性)

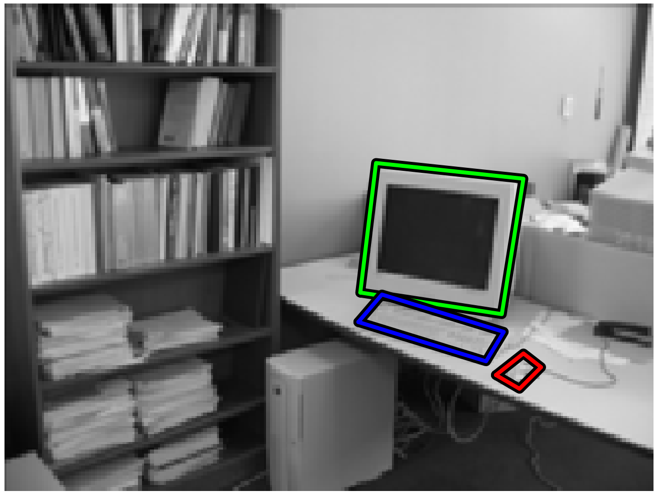
Boosting

- 不同的目标函数和最小化算法将得到不同性质的Boosting分类器

- 稳定性好，适用于工业领域

应用实例

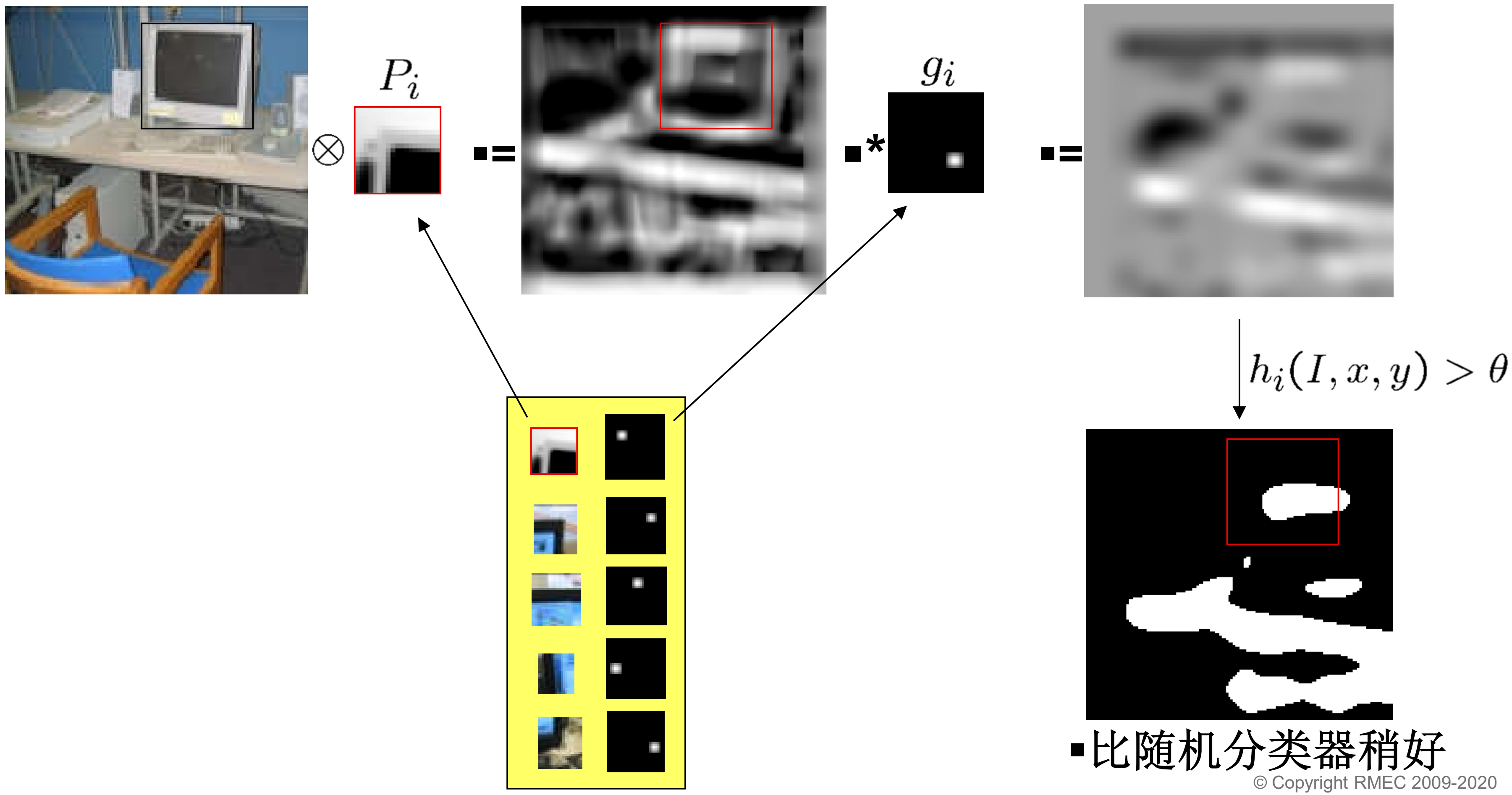
- 目标识别：识别图像中的目标物体
- **Vidal-Naquet, Ullman (2003)**



弱分类器

- 针对显示器，弱分类器可定义为:

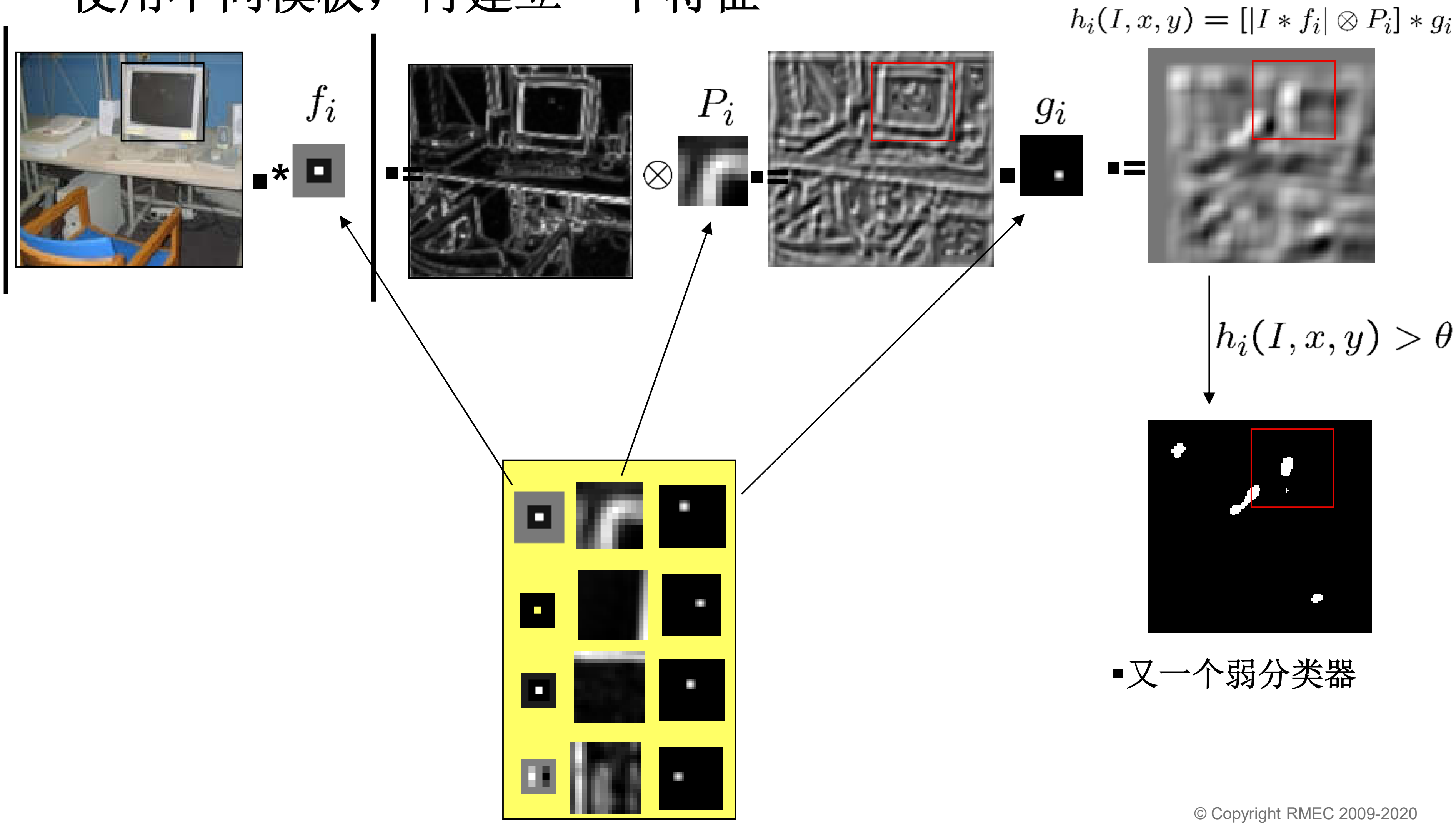
$$h_i(I,x,y) = [I \otimes P_i] * g_i$$



- 比随机分类器稍好

弱分类器

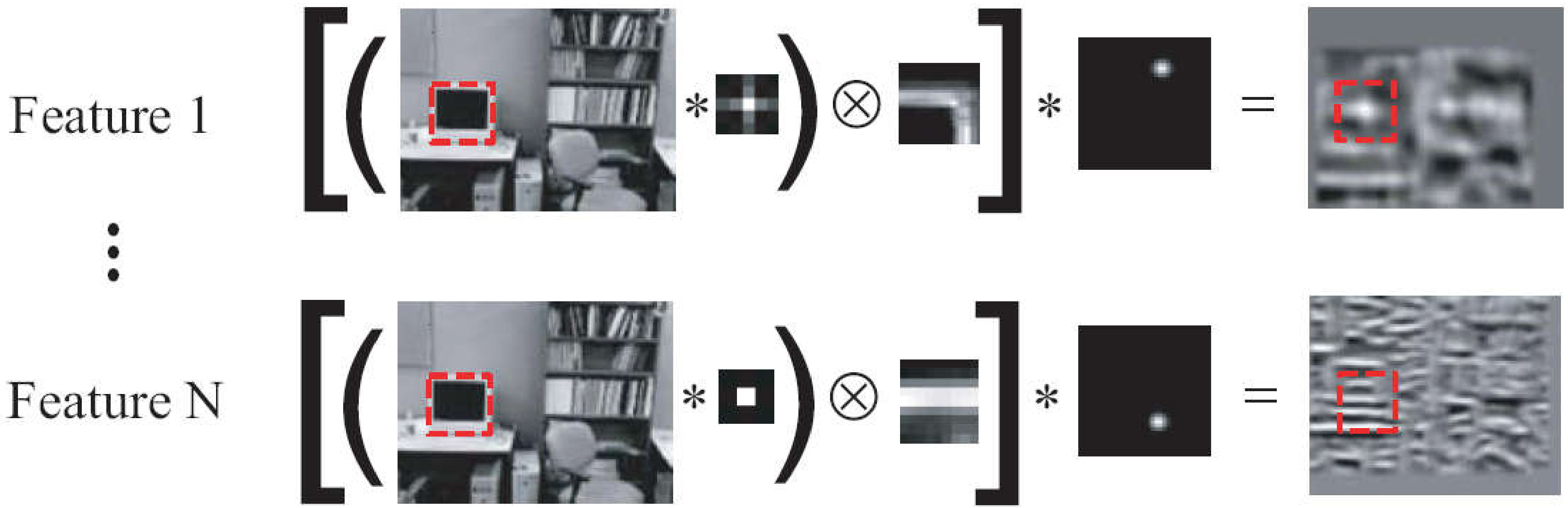
- 使用不同模板，再建立一个特征



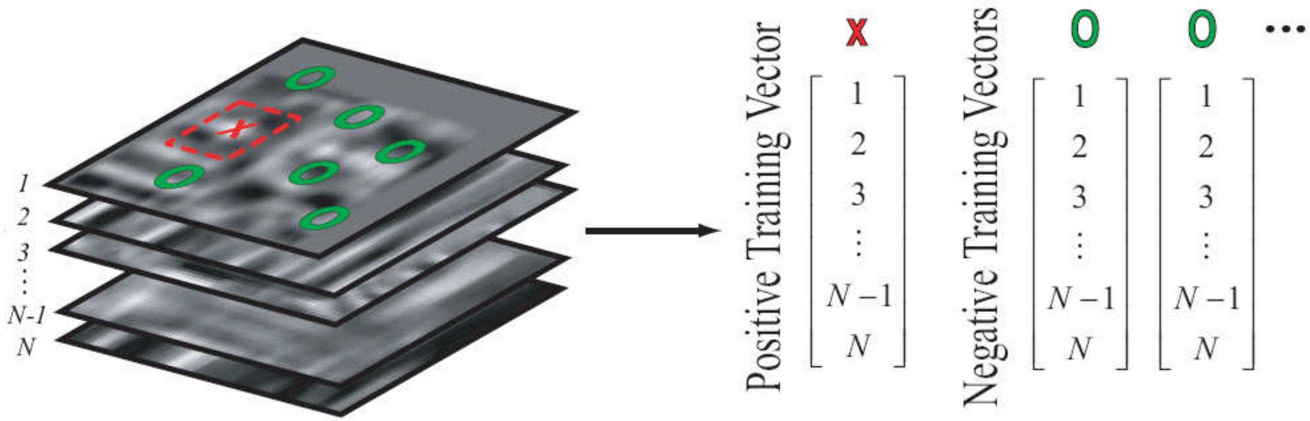
- 又一个弱分类器

训练

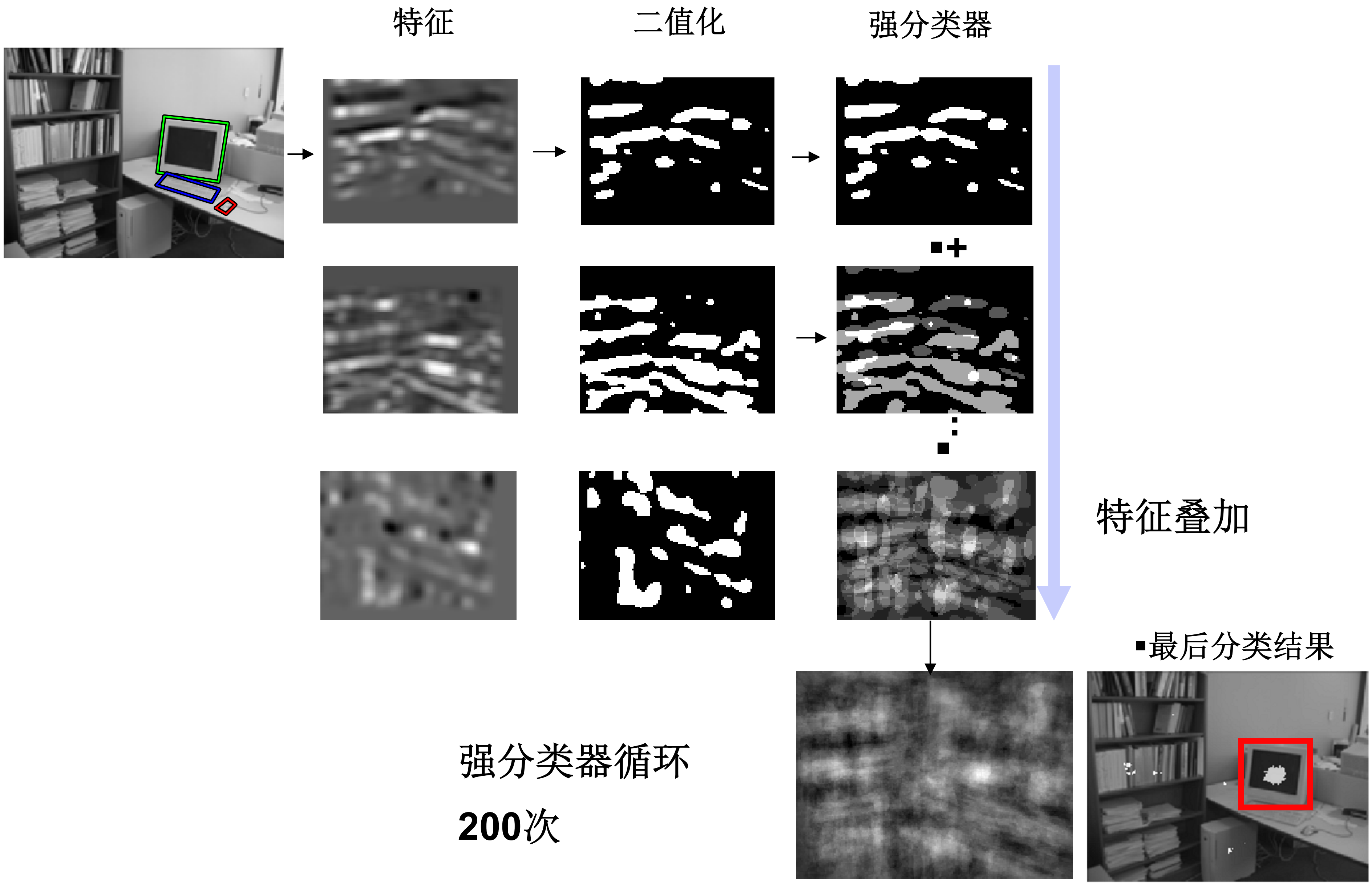
▪我们首先在图像集合上评估***N***个特征.



▪接着, 在目标物体和背景上分别采样，作为不同的分类



应用实例



总结

我们已经熟悉基本的统计学习方法

- 如何确定模型
- 根据数据选择策略
- 选择和使用算法

总结

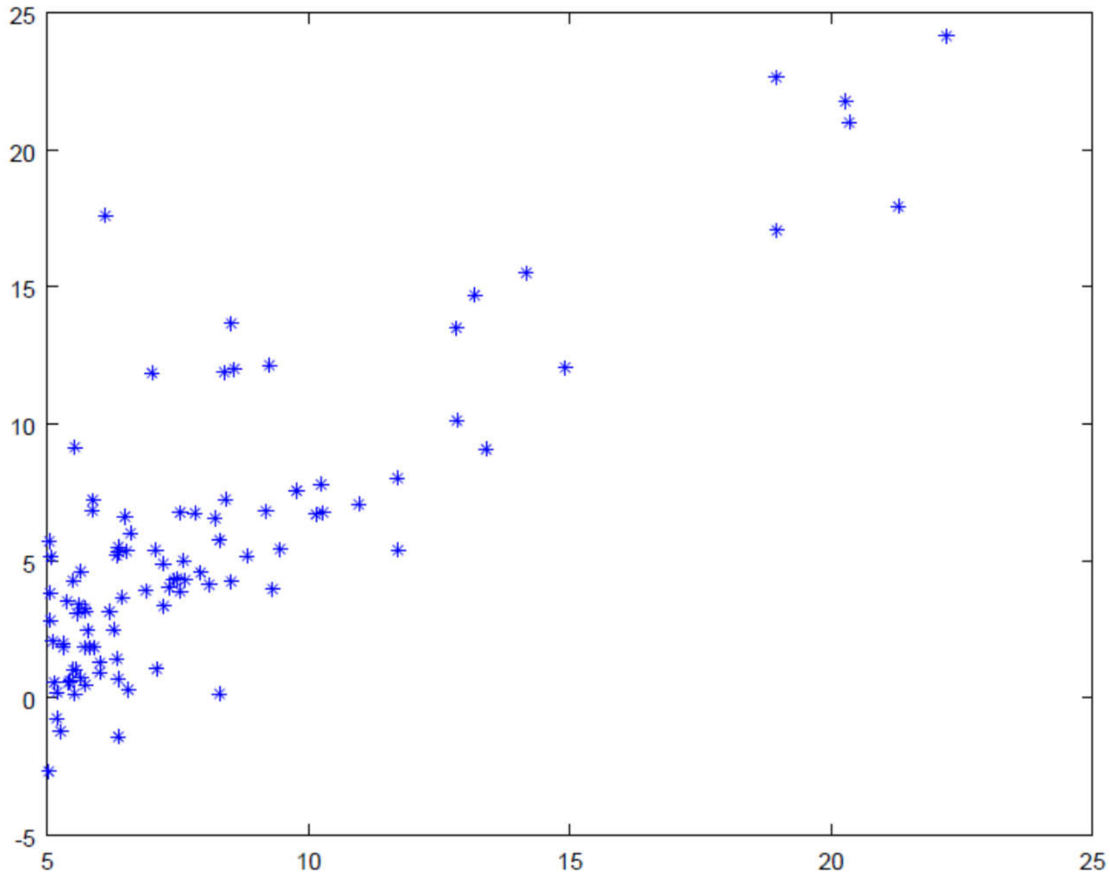
进一步学习

- 《统计学习方法》李航著
- 《机器学习》周志华著

作业

我们已经熟悉基本的统计学习方法

- 给定一组有意义的数据
- 建立回归模型
- 给出计算过程



所有数据和代码在课程网站下载