

# 逻辑回归

## Logistic Regression



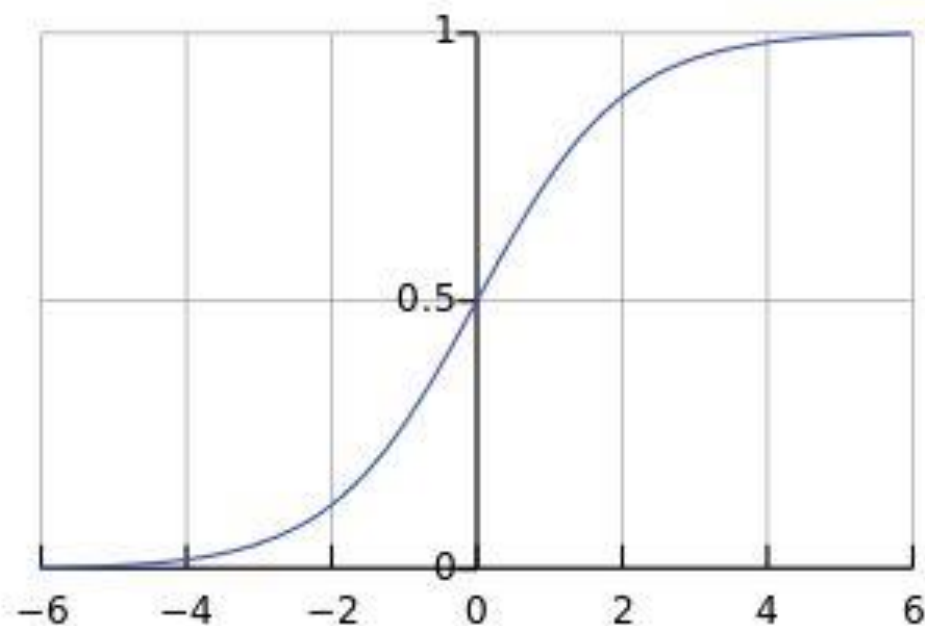
- 垃圾邮件分类
- 预测肿瘤是良性还是恶性
- 预测某人的信用是否良好

# Sigmoid/Logistic Function



我们定义逻辑回归的预测函数为 $h_{\theta}(x) = g(\theta^T x)$  ,  
其中 $g(x)$ 函数是sigmoid函数。

$$g(x) = \frac{1}{1 + e^{-x}} \qquad h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}}$$

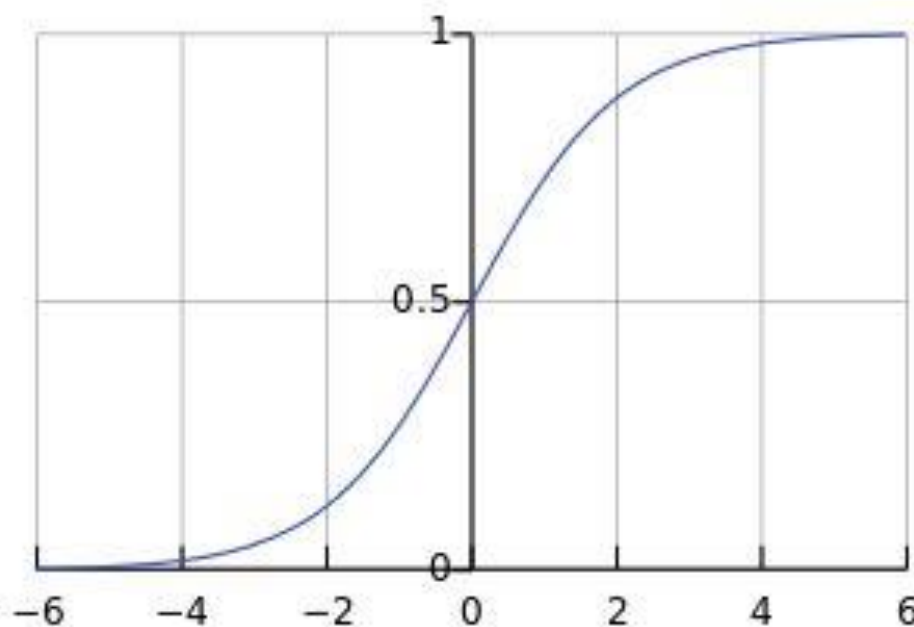


# Sigmoid/Logistic Function



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

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T X}}$$



0.5可以作为分类的边界

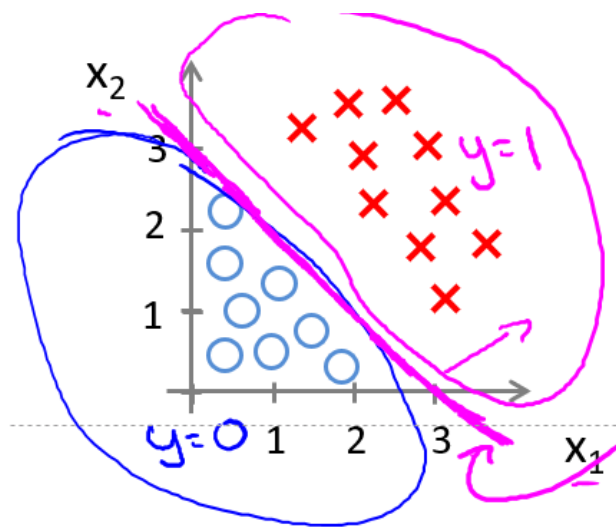
当 $z \geq 0$ 的时候 $g(z) \geq 0.5$

当 $\theta^T X \geq 0$ 的时候 $g(\theta^T X) \geq 0.5$

当 $z \leq 0$ 的时候 $g(z) \leq 0.5$

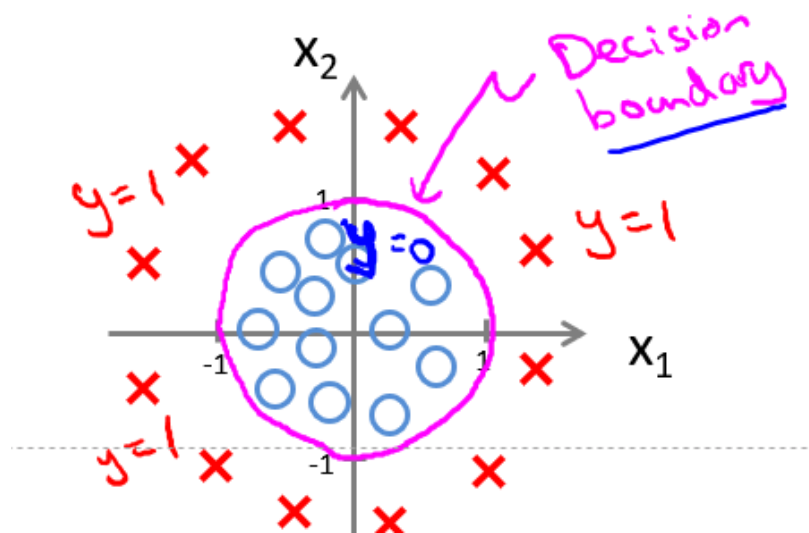
当 $\theta^T X \leq 0$ 的时候 $g(\theta^T X) \leq 0.5$

# 决策边界



$$h_{\theta}(x) = g(\underbrace{\theta_0}_{-3} + \underbrace{\theta_1}_{1}x_1 + \underbrace{\theta_2}_{1}x_2)$$

当  $-3 + x_1 + x_2 \geq 0$  则  $y = 1$

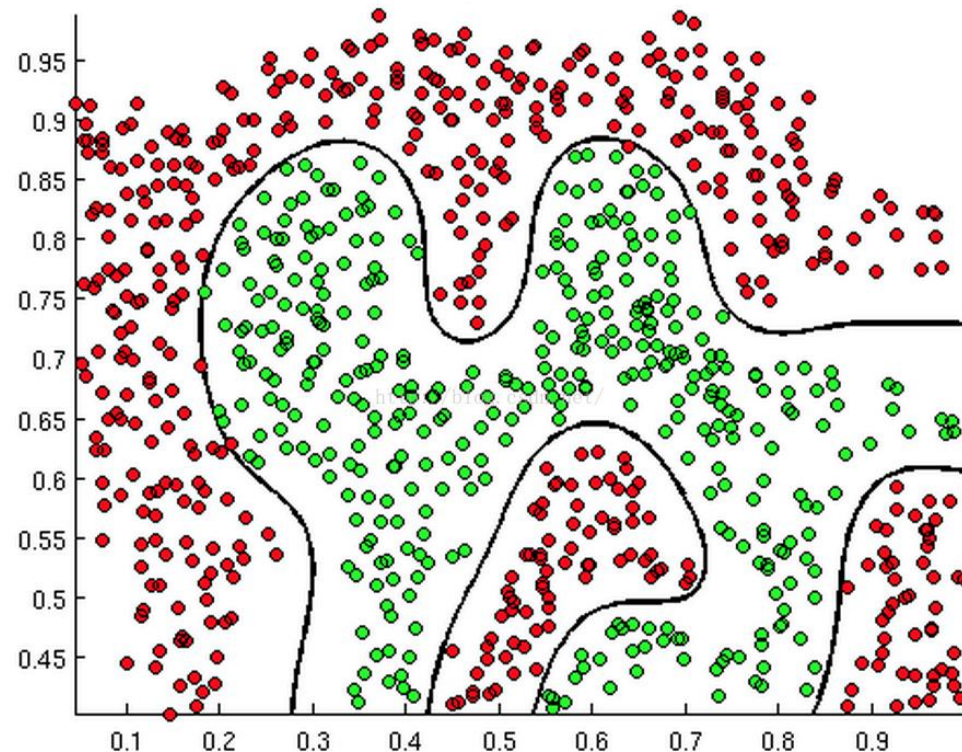


$$h_{\theta}(x) = g(\underset{-1}{\theta_0} + \underset{0}{\theta_1}x_1 + \underset{0}{\theta_2}x_2 + \underset{1}{\theta_3}x_1^2 + \underset{1}{\theta_4}x_2^2)$$

当  $-1 + x_1^2 + x_2^2 \geq 0$  则  $y = 1$



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

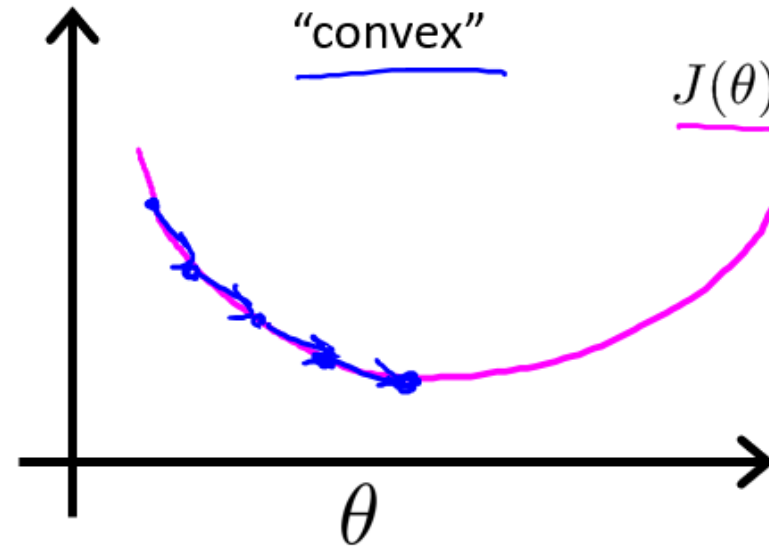
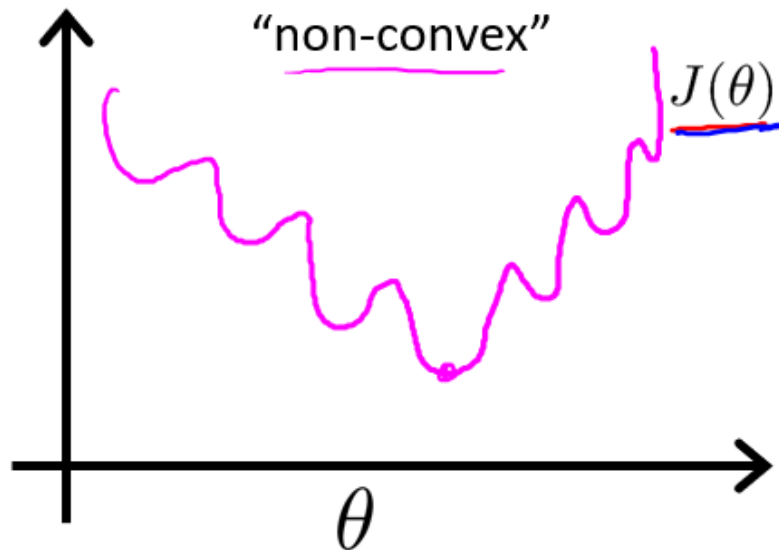


# 逻辑回归的代价函数



Linear regression:

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

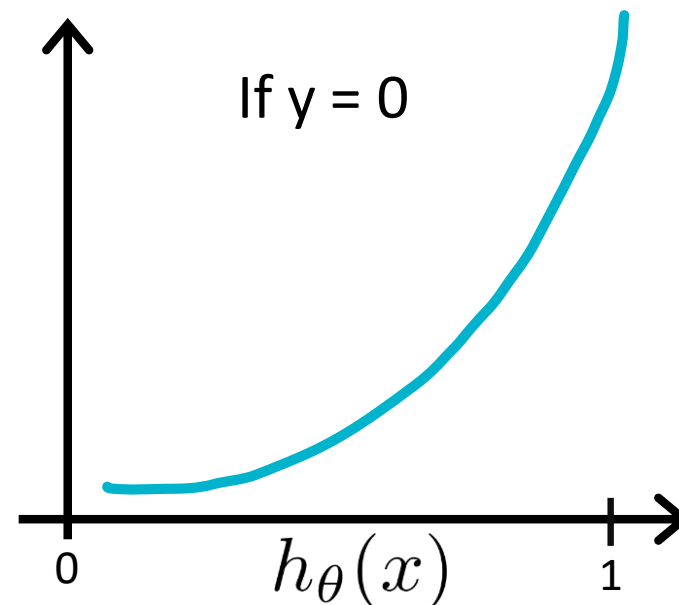
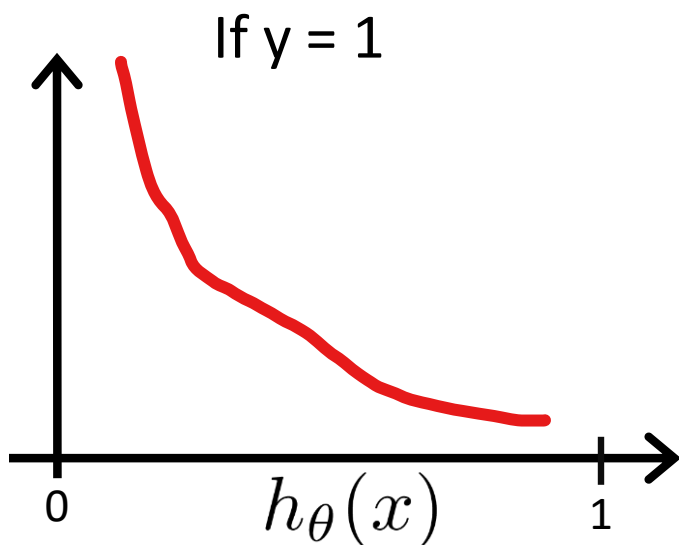




# 逻辑回归的代价函数



$$\text{Cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & \text{if } y = 1 \\ -\log(1 - h_{\theta}(x)) & \text{if } y = 0 \end{cases}$$



当  $y = 1$  ,  $h_{\theta}(x) = 1$  时 ,  $\text{cost} = 0$     当  $y = 0$  ,  $h_{\theta}(x) = 1$  时 ,  $\text{cost} = \infty$   
当  $y = 1$  ,  $h_{\theta}(x) = 0$  时 ,  $\text{cost} = \infty$     当  $y = 0$  ,  $h_{\theta}(x) = 0$  时 ,  $\text{cost} = 0$

# 逻辑回归的代价函数



$$\text{Cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & \text{if } y = 1 \\ -\log(1 - h_{\theta}(x)) & \text{if } y = 0 \end{cases}$$



$$\text{Cost}(h_{\theta}(x), y) = -y \log(h_{\theta}(x)) - (1 - y) \log(1 - h_{\theta}(x))$$

当 $y=1$ 时,  $\text{Cost}(h_{\theta}(x), y) = -\log(h_{\theta}(x))$

当 $y=0$ 时,  $\text{Cost}(h_{\theta}(x), y) = -\log(1 - h_{\theta}(x))$



## Gradient Descent

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

Want  $\min_{\theta} J(\theta)$ :

Repeat {

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

} (simultaneously update all  $\theta_j$ )

# 逻辑回归的代价函数



$$Cost(h_{\theta}(x), y) = -\frac{1}{m} \sum_{i=1}^m y \log(h_{\theta}(x)) + (1 - y) \log(1 - h_{\theta}(x))$$

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

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

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

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

$\lambda \cdot h_{\theta}(x)(1 - h_{\theta}(x))$

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

$$= \frac{1}{m} \sum_{i=1}^m \frac{h_{\theta}'(x) \cdot}{h_{\theta}(x)(1 - h_{\theta}(x))} (h_{\theta}(x) - y)$$

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



## Gradient Descent

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

Want  $\min_{\theta} J(\theta)$ :

Repeat {

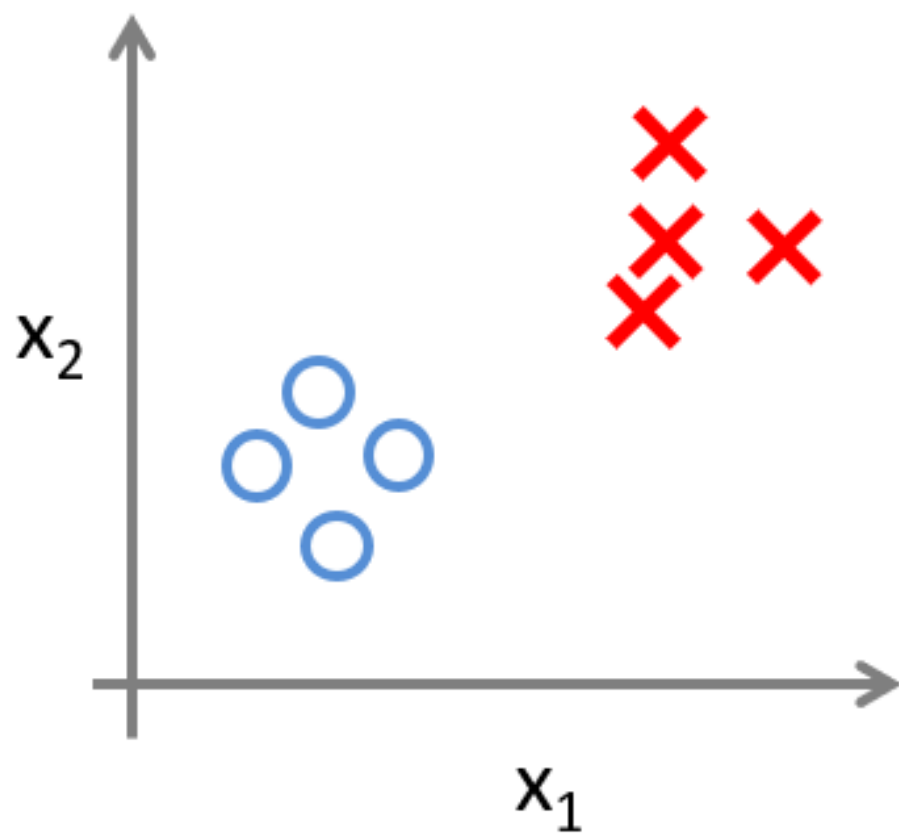
$$\theta_j := \theta_j - \alpha \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

} (simultaneously update all  $\theta_j$ )

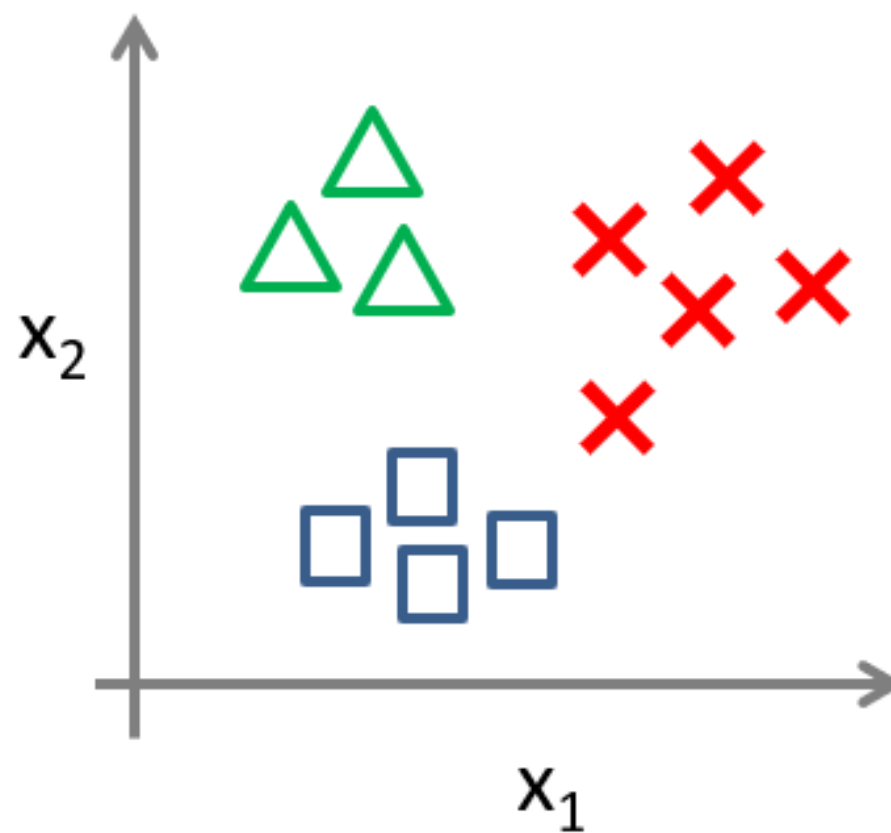
# 多分类



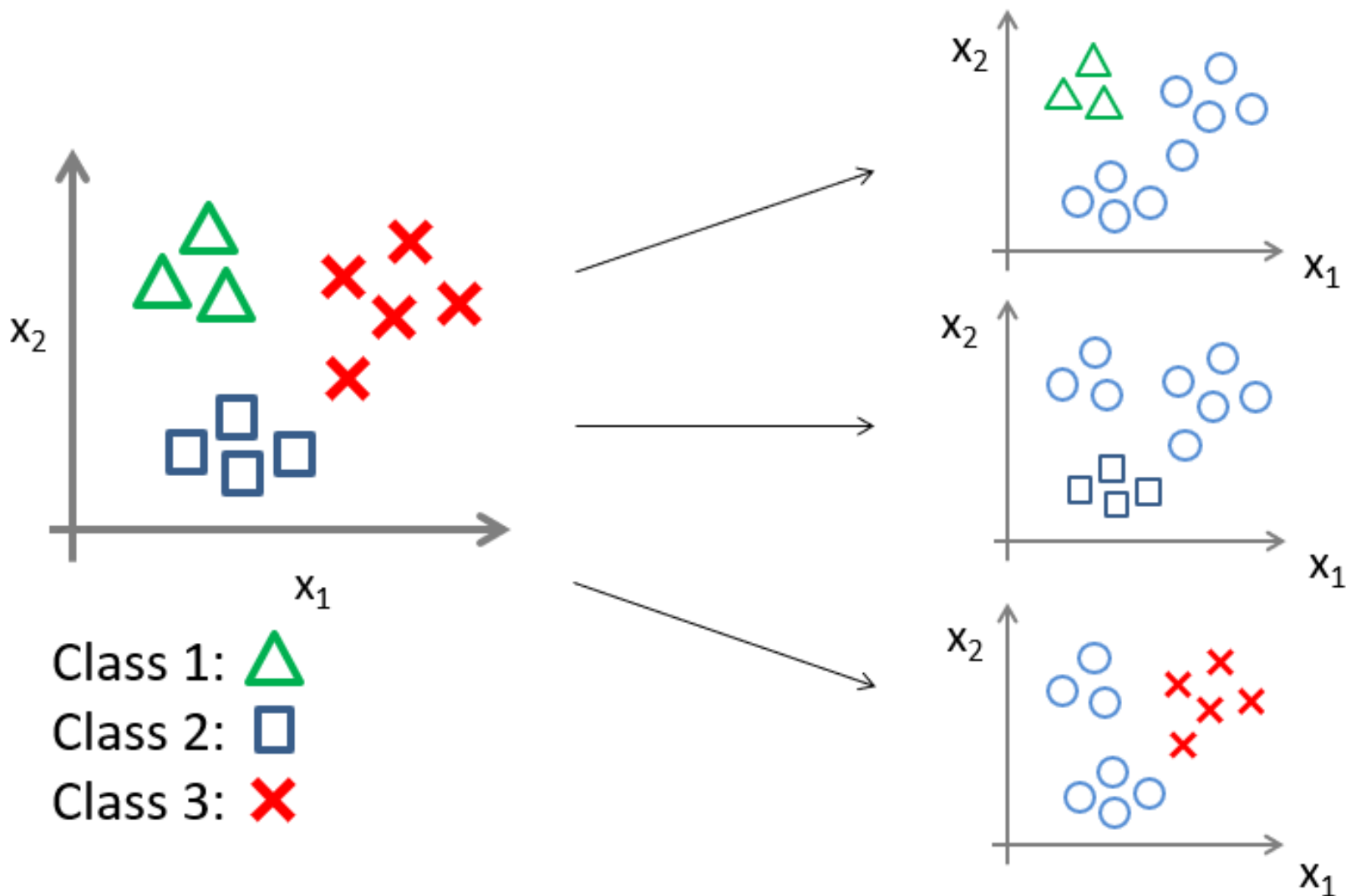
Binary classification:



Multi-class classification:



# 多分类





普通逻辑回归代价函数：

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

正则化逻辑回归代价函数：

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

求导：

$$\begin{aligned} & \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_0^{(i)} \\ & \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_1^{(i)} - \frac{\lambda}{m} \theta_1 \\ & \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_2^{(i)} - \frac{\lambda}{m} \theta_2 \\ & \dots\dots \end{aligned}$$



**正确率/召回率/F1指标**

# 正确率与召回率



正确率与召回率 ( Precision & Recall ) 是广泛应用于信息检索和统计学分类领域的两个度量值，用来评价结果的质量。

一般来说，正确率就是检索出来的条目有多少是正确的，召回率就是所有正确的条目有多少被检索出来了。

$$F1 \text{ 值} = 2 * \frac{\text{正确率} * \text{召回率}}{\text{正确率} + \text{召回率}}$$
。是综合上面二个指标的评估指标，用于综合反映整体的指标。

这几个指标的取值都在0-1之间，数值越接近于1，效果越好。



某池塘有1400条鲤鱼，300只虾，300只鳖。现在以捕鲤鱼为目的。撒一大网，逮着了700条鲤鱼，200只虾，100只鳖。那么，这些指标分别如下：

$$\text{正确率} = 700 / (700 + 200 + 100) = 70\%$$

$$\text{召回率} = 700 / 1400 = 50\%$$

$$\text{F值} = 70\% * 50\% * 2 / (70\% + 50\%) = 58.3\%$$



某池塘有1400条鲤鱼，300只虾，300只鳖。现在以捕鲤鱼为目的。  
撒一大网，逮着了所有的鱼虾鳖：

$$\text{正确率} = 1400 / (1400 + 300 + 300) = 70\%$$

$$\text{召回率} = 1400 / 1400 = 100\%$$

$$\text{F值} = 70\% * 100\% * 2 / (70\% + 100\%) = 82.35\%$$



我们希望检索结果Precision越高越好，同时Recall也越高越好，但事实上这两者在某些情况下有矛盾的。比如极端情况下，我们只搜索出了一个结果，且是准确的，那么Precision就是100%，但是Recall就很低；而如果我们把所有结果都返回，那么比如Recall是100%，但是Precision就会很低。

因此在不同的场合中需要自己判断希望Precision比较高或是Recall比较高。



正确率与召回率指标有时候会出现的矛盾的情况，这样就需要综合考虑他们，最常见的方法就是F-Measure（又称为F-Score）：

$$F_{\beta} = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

当 $\beta = 1$ 时，就是常见的F1指标：

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

# 梯度下降法-逻辑回归



# sklearn-逻辑回归





# 非线性逻辑回归



# sklearn-非线性逻辑回归





## 「关注 AI MOOC 官方公众号」

这里持续分享  
Python、机器学习和深度学习的  
各种资源干货及  
有趣又实用的硬知识



覃秉丰

AI MOOC创始人

网易计算机视觉微专业核心讲师

机器学习、深度学习多年开发经验

受邀为中国移动、国家电网、华夏银行、  
太平洋保险等世界五百强企业总部做AI内训

