

# 分类方法-丁兆云

## ——Regression

# 案例引入

## ◆ 买房子问题

房屋  
价格  
销售  
表

假设有一个房屋销售的数据如下：

面积( $m^2$ )	销售价钱 (万元)
123	250
150	320
87	160
102	220
...	...

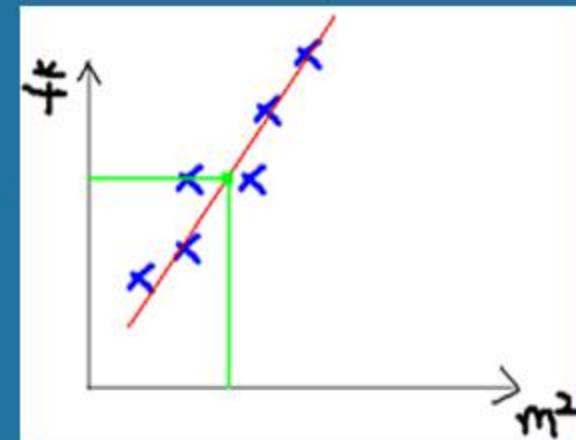


# 案例引入



一般想法：

我们可以用一条直线去尽量准的拟合这些数据，从而找到面积与房价之间的因果关系：直线的斜率就是每平的均价

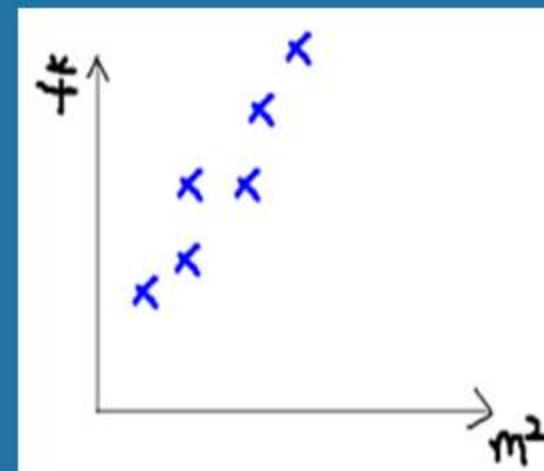


# 案例引入



一般想法：

画个图看看到底两者是什么关系，或者像  
是什么关系，总之就是期望能够找到两者  
之间的**因果关系**



# 案例引入



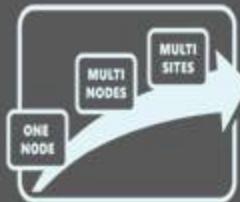
## 房屋销售记录表

训练集(training set)或者训练数据(training data), 是我们流程中的**输入数据**, 一般称为x



## 训练数据的条目数

一条训练数据是由一对输入数据和输出数据组成的输入数据的维度n  
(特征的个数, #features)



## 房屋销售价钱

**输出数据**, 一般称为y



## 拟合的函数 (假设或者模型)

一般写做  $y = h(x)$

# 主要内容

- ◆ 1. 线性回归
- ◆ 2. 优化求解
- ◆ 3. 逻辑回归
- ◆ 4. 决策树回归

## 1.1 回归问题

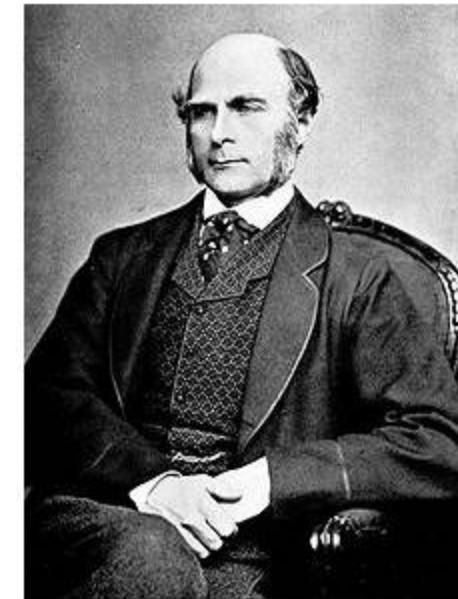
- ◆ 回归分析
  - 如果把其中的一些因素（房屋面积）作为自变量，而另一些随自变量的变化而变化的变量作为因变量（房价），研究他们之间的非确定因果关系，这种分析就称为**回归分析**（regression）。
  - 回归分析是研究一个或多个自变量与一个因变量之间是否存在某种线性关系或非线性关系的一种统计学方法。

## 1.1 回归问题的起源

### ◆ 回归问题的来源

英国著名的统计学家F.Galton研究了1078对夫妇及其一个成年儿子的身高关系。他们以儿子身高作为纵坐标、夫妇平均身高为横坐标作散点图，结果发现二者的关系近似于一条直线。经计算得到了如下方程：

$$y=0.8567+0.516x$$



Francis Galton  
英国19世纪统计学家

Galton引进“回归”(regression)一词来表达这种方程关系。

## 1.2. 线性回归

- ◆ 线性回归假设特征和结果满足线性关系
- ◆ 一元线性回归问题函数关系可表示

$$y = a + bx$$

- 根据上式，在确定a、b的情况下，给定一个x值，我们就能够得到一个确定的y值，然而根据上式得到的y值与实际的y值存在一个误差
- a、b为参数 (parameters)，或称回归系数 (regression coefficients)
- ◆ 用什么样的线性关系刻画更好呢？

## 1.3模型刻画

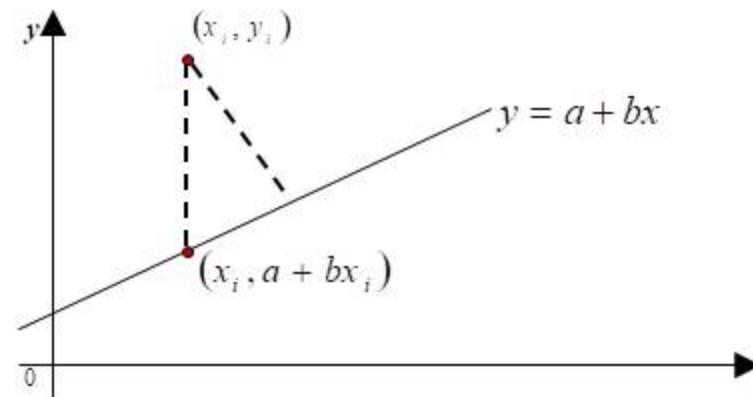
用什么样的方法刻画点与直线的距离会方便有效？

设直线方程为 $y=a+bx$ , 样本点**A** ( $x_i, y_i$ )

方法一、点到直线的距离公式

$$d = \frac{|bx_i - y_i + a|}{\sqrt{b^2 + 1}}$$

方法二、 $[y_i - (a + bx_i)]^2$



显然方法二能有效地表示点A与直线 $y=a+bx$ 的距离，而且比方法一更方便计算，所以我们用它来表示二者之间的接近程度

## 1.4 最小二乘法

- ◆ 基本思想

- 保证直线与所有点都近

- ◆ 详细做法

- 若有n个样本点:  $(x_1, y_1), \dots, (x_n, y_n)$ , 可以用下面的表达式来刻画这些点与直线 $y=a+bx$ 的接近程度:

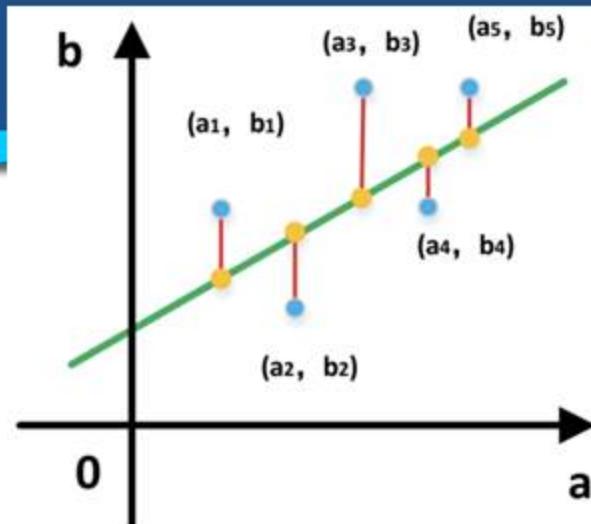
$$[y_1 - (a + bx_1)]^2 + \dots + [y_n - (a + bx_n)]^2$$

- 使上式达到最小值的直线 $y=a+bx$ 就是所求的直线, 这种方法称为最小二乘法。

- ◆ 求a和b的偏导数, 可得

如果用 $\bar{x}$ 表示 $\frac{x_1 + x_2 + \dots + x_n}{n}$ , 用 $\bar{y}$ 表示 $\frac{y_1 + y_2 + \dots + y_n}{n}$ 则可得到

$$b = \frac{x_1 y_1 + \dots + x_n y_n - n \bar{x} \bar{y}}{x_1^2 + \dots + x_n^2 - n \bar{x}^2}, \quad a = \bar{y} - b \bar{x}$$



## 1.4 最小二乘法

**例题1** 从某大学中随机选出8名女大学生，其身高和体重数据如下表：

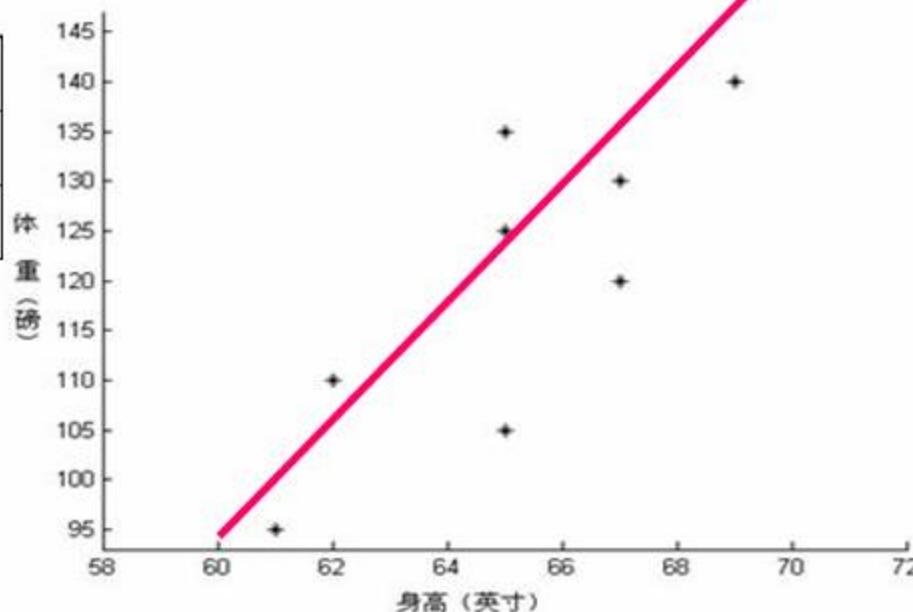
编号	1	2	3	4	5	6	7	8
身高	165	165	157	170	175	165	155	170
体重	48	57	50	54	64	61	43	59

求根据一名女大学生的身高预报她的体重的回归方程，并预报一名身高为172 cm的女大学生的体重。

## 1.4 最小二乘法

$$b = \frac{x_1 y_1 + \dots + x_n y_n - n \bar{x} \bar{y}}{x_1^2 + \dots + x_n^2 - n \bar{x}^2}, a = \bar{y} - b \bar{x}$$

编号	1	2	3	4	5	6	7	8
身高	165	165	157	170	175	165	155	170
体重	48	57	50	54	64	61	43	59



分析：

身高y为自变量

体重x为因变量.

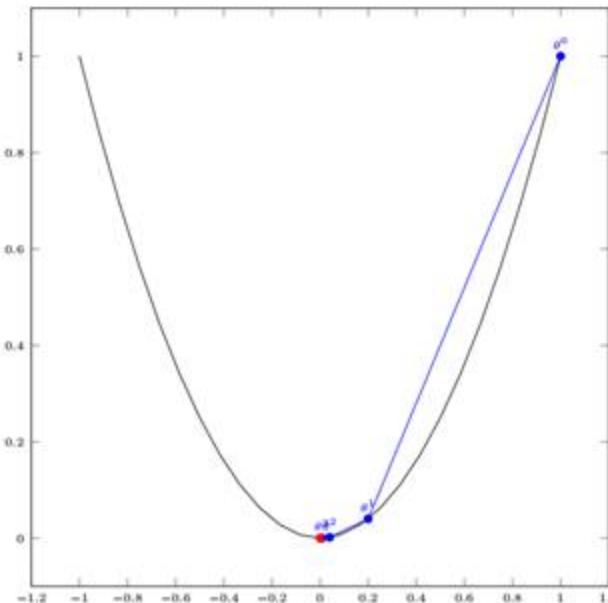
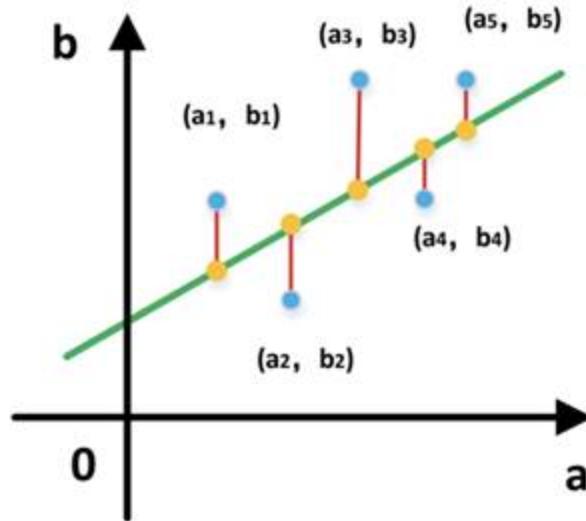
$$y = 0.849x - 85.172$$

身高172cm女大学生体重

$$\hat{y} = 0.849 \times 172 - 85.172 = 60.316 \text{ (kg)}$$

## 2.1 优化求解-例子

- ◆ 原函数  $J(\theta) = \theta^2$
- ◆ 误差最小化：如何求得theta使得 $\min(J)$
- ◆ 函数的微分  $J'(\theta) = 2\theta$ .
- ◆ 求解  $J' = 0$ : 简单方法即求解方程



## 2.2 优化求解-梯度下降法

- ◆ 梯度

- 梯度的本意是一个向量（矢量），表示某一函数在该点处的方向导数沿着该方向取得最大值，即函数在该点处沿着该方向（此梯度的方向）变化最快，变化率最大（为该梯度的模）。



## 2.2 优化求解-梯度下降

- ◆ 基本思想

- 向着梯度的反方向调整
- 步长不能太大，也不能太小

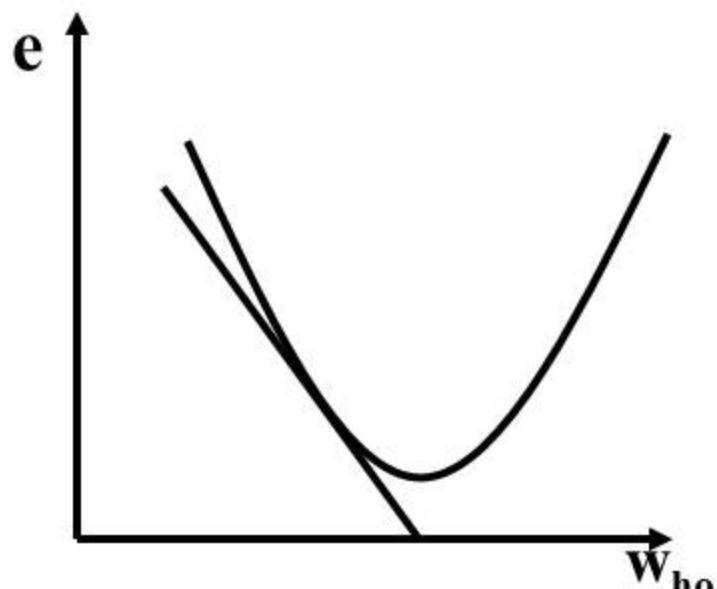
$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta) \quad \text{evaluated at } \Theta^0$$

Diagram illustrating the gradient descent update rule:

- current position (blue speech bubble)
- opposite direction (black speech bubble)
- next position (red speech bubble)
- small step (green speech bubble)
- direction of fastest increase (purple speech bubble)

## 2.2 优化求解-梯度下降

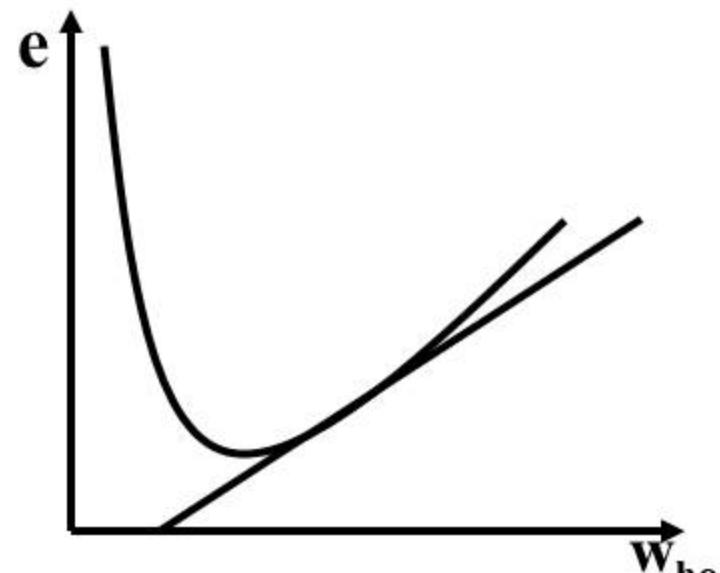
- ◆ 情况一直观表达
  - 当误差对权值的偏导数小于零时，权值调整量为正，实际输出少于期望输出，权值向增大方向调整，使得实际输出与期望输出的差减少。



$$\frac{\partial e}{\partial w_{ho}} < 0, \text{ 此时 } \Delta w_{ho} > 0$$

## 2.2 优化求解-梯度下降

- ◆ 情况二直观表达
  - 当误差对权值的偏导数大于零时，权值调整量为负，实际输出大于期望输出，权值向减少方向调整，使得实际输出与期望输出的差减少。



$$\frac{\partial e}{\partial w_{ho}} > 0, \text{ 此时 } \Delta w_{ho} < 0$$

## 2.3 优化求解-梯度下降法求解例子1

◆ 原函数

$$J(\theta) = \theta^2$$

◆ 函数的微分

$$J'(\theta) = 2\theta.$$

◆ 初始条件

$$\theta^0 = 1 \quad \alpha = 0.4$$

◆ 调整过程

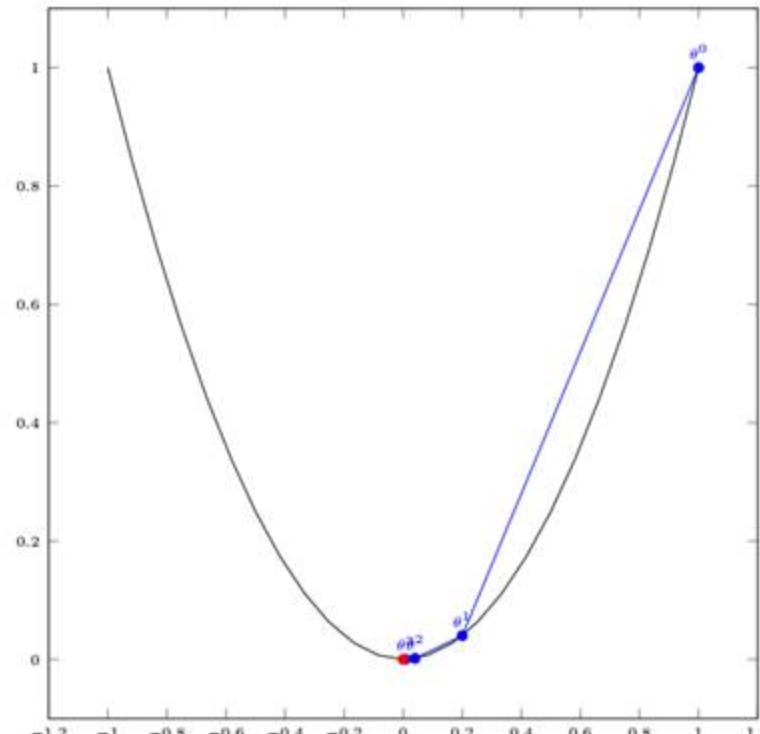
$$\theta^0 = 1$$

$$\begin{aligned}\theta^1 &= \theta^0 - \alpha * J'(\theta^0) \\ &= 1 - 0.4 * 2 \\ &= 0.2\end{aligned}$$

$$\begin{aligned}\theta^2 &= \theta^1 - \alpha * J'(\theta^1) \\ &= 0.04\end{aligned}$$

$$\theta^3 = 0.008$$

$$\theta^4 = 0.0016$$



## 2.3 优化求解-梯度下降法求解例子2

- ◆ 原函数

$$J(\Theta) = \theta_1^2 + \theta_2^2.$$

- ◆ 函数的微分

$$\nabla J(\Theta) = \langle 2\theta_1, 2\theta_2 \rangle$$

- ◆ 初始条件

$$\Theta^0 = (1, 3) \quad \alpha = 0.1.$$

- ◆ 调整过程

$$\Theta^0 = (1, 3)$$

$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta)$$

$$= (1, 3) - 0.1(2, 6)$$

$$= (0.8, 2.4)$$

$$\Theta^2 = (0.8, 2.4) - 0.1(1.6, 4.8)$$

$$= (0.64, 1.92)$$

$$\Theta^3 = (0.512, 1.536)$$

$$\Theta^4 = (0.4096, 1.2288000000000001)$$

⋮

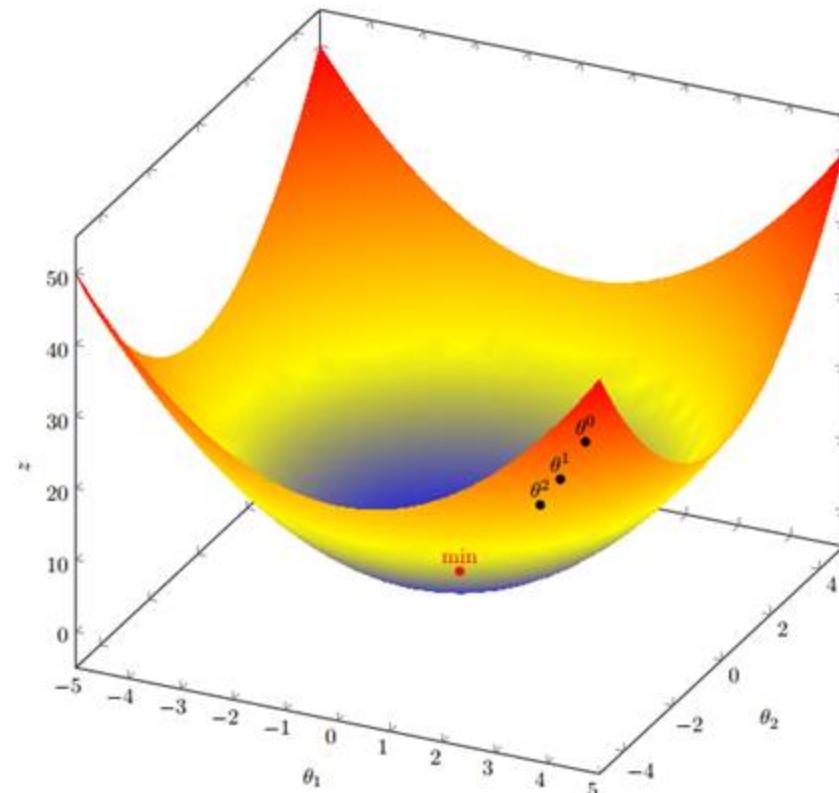
$$\Theta^{10} = (0.1073741824000003, 0.3221225472000005)$$

⋮

$$\Theta^{50} = (1.1417981541647683e^{-05}, 3.425394462494306e^{-05})$$

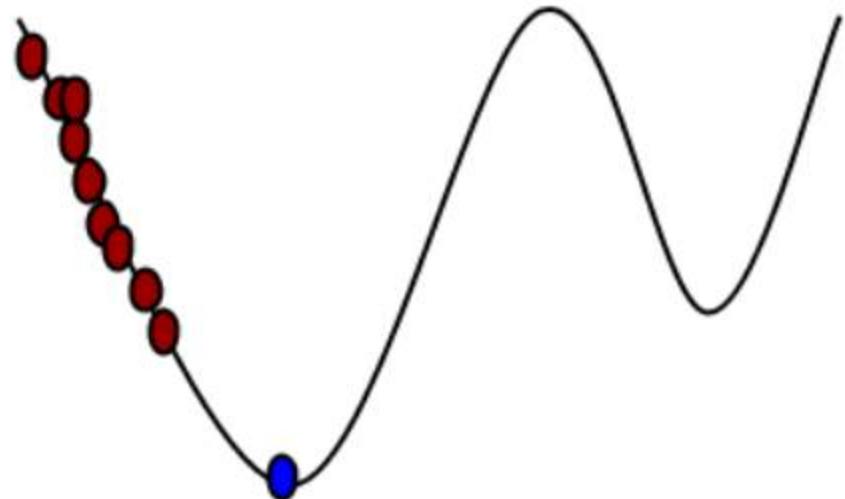
⋮

$$\Theta^{100} = (1.6296287810675902e^{-10}, 4.888886343202771e^{-10})$$



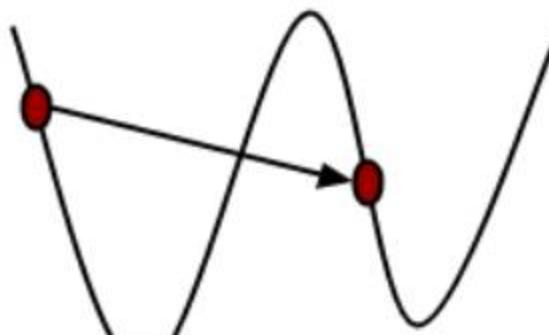
## 2.4 优化求解-学习速率的影响

学习率过小，  
收敛速度太慢。



very small learning  
rate needs lots of  
steps

学习率过大，  
不会收敛。



too big learning rate:  
missed the minimum

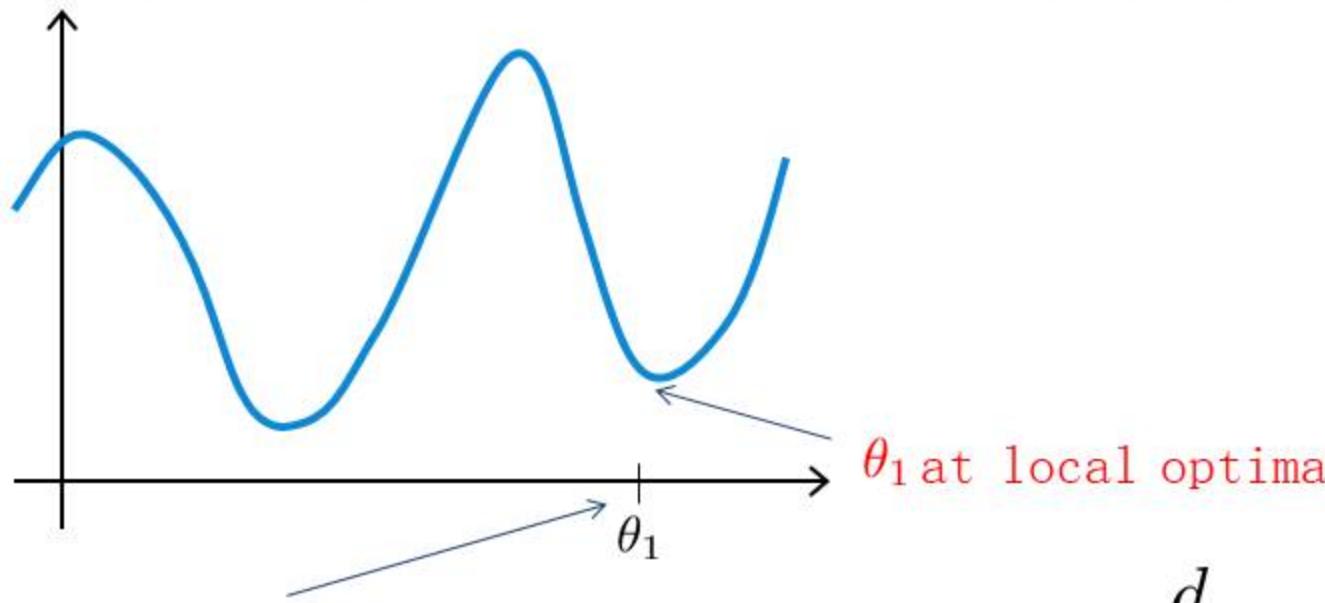
## 2.5 优化求解-梯度下降法收敛

$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta) \quad \text{evaluated at } \Theta^0$$

Annotations:

- current position:  $\Theta^0$
- next position:  $\Theta^1$
- opposite direction:  $\nabla J(\Theta)$
- small step:  $\alpha$
- direction of fastest increase:  $\nabla J(\Theta)$

梯度下降法不能保证一定收敛到全局最优值。



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

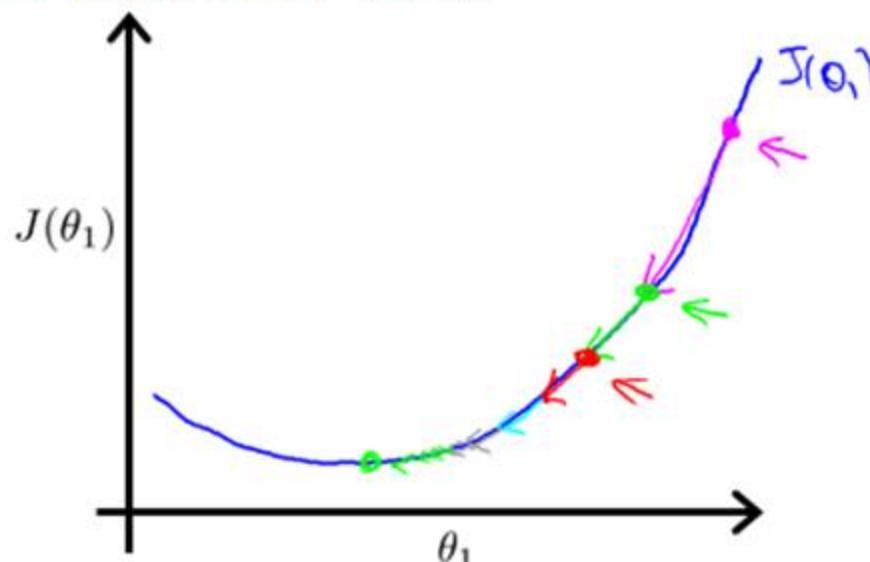
## 2.6 优化求解-变种梯度下降法1

$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta) \quad \text{evaluated at } \Theta^0$$

Annotations:

- current position
- opposite direction
- next position
- small step
- direction of fastest increase

自动调整学习率的梯度下降法



学习率随梯度大小调整

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

## 2.6 优化求解-变种梯度下降法2

### 批处理梯度下降法(一批数据后更新权值)

- 优点：由全数据集确定的方向能够更好地代表样本总体，从而更准确地朝向极值所在的方向、易并行
- 缺点：当样本数目 $m$ 很大时，每迭代一步都需要对所有样本计算，训练过程会很慢

```
Repeat until convergence {  
     $\theta_j := \theta_j + \alpha \sum_{i=1}^m (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$       (for every  $j$ ).  
}
```

### 随机梯度下降法(一个数据更新1次权值)

- 优点：每一轮参数更新快
- 缺点：准确度下降、可能会收敛到局部最优、不易于并行

```
Loop {  
    for i=1 to m, {  
         $\theta_j := \theta_j + \alpha (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$       (for every  $j$ ).  
    }  
}
```

当训练集规模较大时，更适合采用下列哪种优化算法估计权值

- A 批处理梯度下降
- B 随机梯度下降

 提交

### 3.1 逻辑回归-案例引入

#### ◆ 案例引入

表1 年龄(Age)和冠心病(CD)发病情况

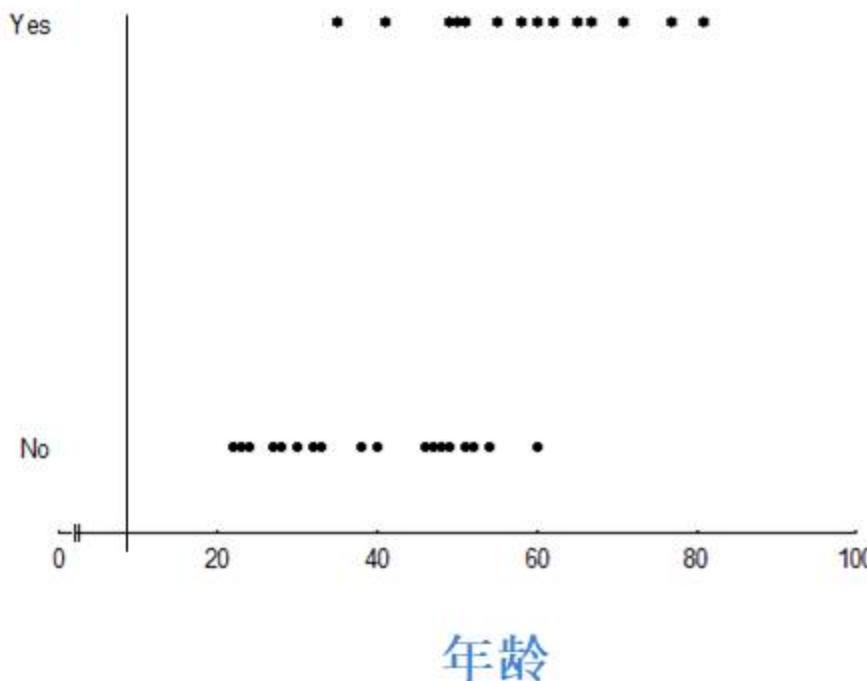
Age	CD	Age	CD	Age	CD
22	0	40	0	54	0
23	0	41	1	55	1
24	0	46	0	58	1
27	0	47	0	60	1
28	0	48	0	60	0
30	0	49	1	62	1
30	0	49	0	65	1
32	0	50	1	67	1
33	0	51	0	71	1
35	1	51	1	77	1
38	0	52	0	81	1

### 3.1 案例引入

- ◆ 比较病人和非病人的平均年龄

- 非病人: 38.6 岁
- 病人: 58.7 岁

- ◆ 能不能用线形回归?

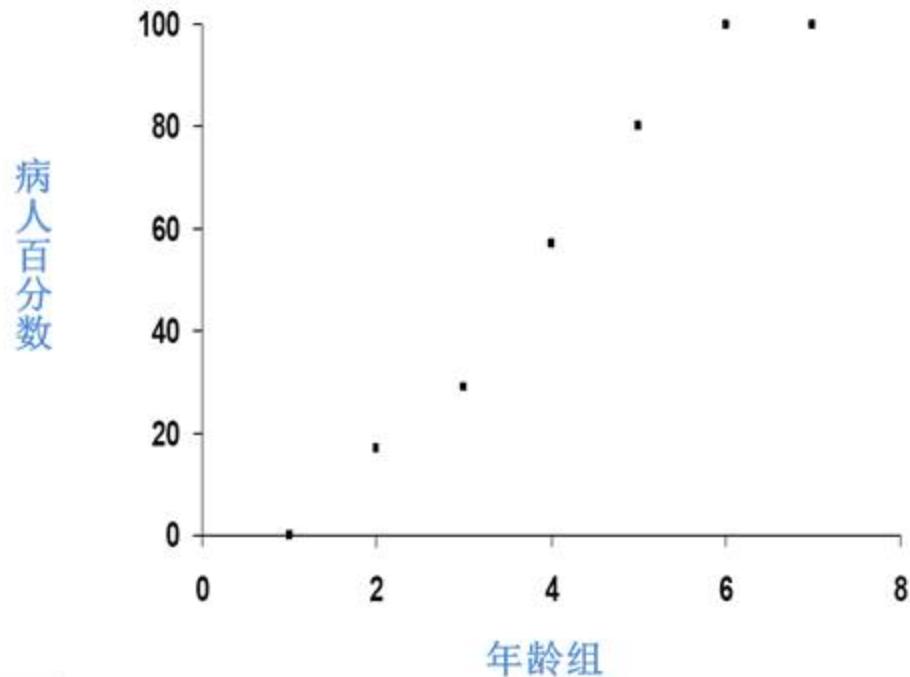


### 3.1 案例引入

表2 按年龄组划分的冠心病发病情况

年龄组	人数	冠心病人数	累积%
-----	----	-------	-----

20 - 29	5	0	0
30 - 39	6	1	17
40 - 49	7	2	29
50 - 59	7	4	57
60 - 69	5	4	80
70 - 79	2	2	100
80 - 89	1	1	100

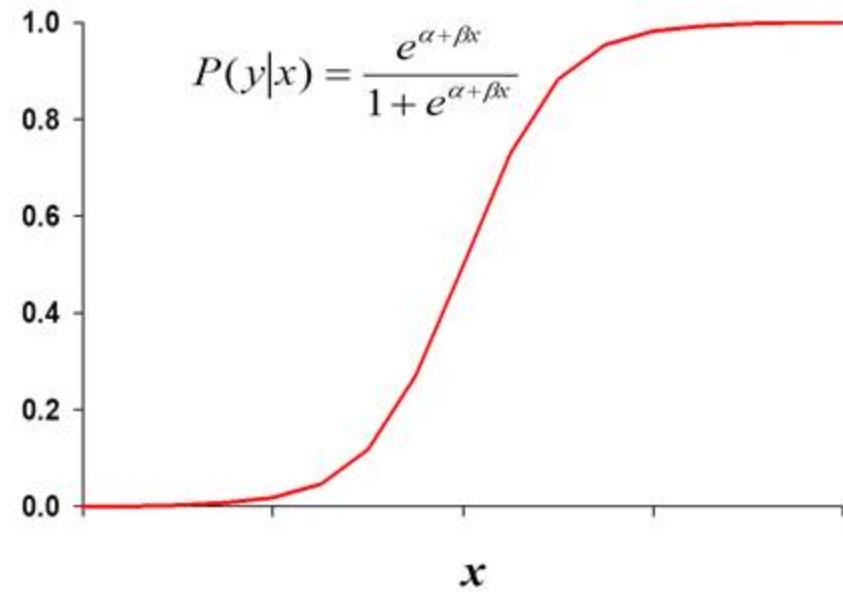


### 3.1 案例引入

表2 按年龄组划分的冠心病发病情况

年龄组	人数	冠心病人数	累积%
20 - 29	5	0	0
30 - 39	6	1	17
40 - 49	7	2	29
50 - 59	7	4	57
60 - 69	5	4	80
70 - 79	2	2	100
80 - 89	1	1	100

得病概率

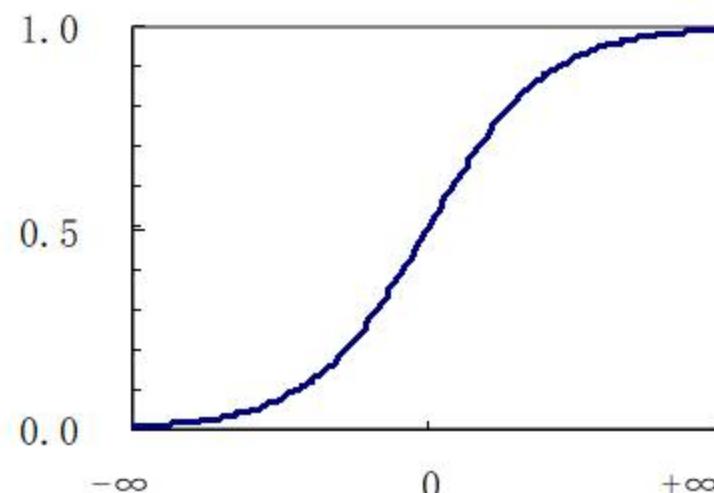


## 3.2 逻辑回归函数

1838年由比利时学者 Verhulst 首次提出。  
1920年美国学者 Bearl & Reed在研究果蝇的繁殖中发现和使用该函数，并在人口估计和预测中推广使用

logistic函数的值域为[0,1]

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



用  $p_i = P(y_i = 1 | x_{i1}, x_{i2}, \dots, x_{ip})$  作为因变量，得到logistic回归模型

$$p_i = \frac{\exp(\alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip})}{1 + \exp(\alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip})}$$

$$\ln \frac{p_i}{1 - p_i} = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$$

### 3.2 将逻辑回归转换成线性模型

Logit (logistic probability unit) 变换:

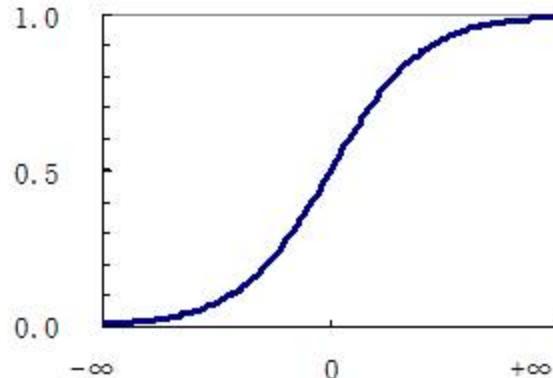
定义:

$$\text{Logit}(p_i) = \ln \frac{p_i}{1-p_i}$$

得到:

$$\text{Logit}(p_i) = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

Logit变换的特点:  $p_i \in [0,1]$



$$\frac{p_i}{1-p_i} \in [0, +\infty)$$

$$\text{Logit}(p_i) = \ln \frac{p_i}{1-p_i} \in (-\infty, +\infty)$$

较好地克服了概率因变量的取值受到限制的困难。

### 3.3 优势比OR

$$\ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_m X_m = \text{log it} P$$

## 优势比 **OR(odds ratio)**

流行病学衡量危险因素作用大小的**比数比例**指标。

计算公式为：

$$OR_j = \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)}$$

式中  $P_1$  和  $P_0$  分别表示在  $X_j$  取值为  $c_1$  及  $c_0$  时的发病概率，  $OR_j$  称作多变量调整后的优势比，表示扣除了其他自变量影响后危险因素的作用。

### 3.3 优势比OR

$$\ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_m X_m = \text{logit} P$$

与 logisticP 的关系：

对比某一危险因素两个不同暴露水平  $X_j = c_1$  与  $X_j = c_0$  的发病情况（假定其它因素的水平相同），其优势比的自然对数为：

$$\begin{aligned}\ln OR_j &= \ln \left[ \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)} \right] = \text{logit} P_1 - \text{logit} P_0 \\ &= (\beta_0 + \beta_j c_1 + \sum_{t \neq j}^m \beta_t X_t) - (\beta_0 + \beta_j c_0 + \sum_{t \neq j}^m \beta_t X_t) \\ &= \beta_j (c_1 - c_0)\end{aligned}$$

### 3.3 优势比OR

$$\begin{aligned}\ln OR_j &= \ln \left[ \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)} \right] = \text{logit} P_1 - \text{logit} P_0 \\ &= (\beta_0 + \beta_j c_1 + \sum_{t \neq j}^m \beta_t X_t) - (\beta_0 + \beta_j c_0 + \sum_{t \neq j}^m \beta_t X_t) \\ &= \beta_j (c_1 - c_0)\end{aligned}$$

即  $OR_j = \exp[\beta_j (c_1 - c_0)]$

若  $X_j = \begin{cases} 1 & \text{暴露} \\ 0 & \text{非暴露} \end{cases}, c_1 - c_0 = 1,$

则有  $OR_j = \exp \beta_j, \quad \beta_j \begin{cases} = 0, OR_j = 1 & \text{无作用} \\ > 0, OR_j > 1 & \text{危险因子} \\ < 0, OR_j < 1 & \text{保护因子} \end{cases}$

### 3.3 优势比OR

- ◆ **例子：** 在一个具有17个家庭的样本里，共有3家的收入为¥10000，5家的收入为¥11000，9家的收入为¥12000。在收入为¥10000的家庭里，1个主妇不工作，2个主妇工作；在收入为¥11000的家庭里，1个主妇不工作，4个主妇工作；在收入为¥12000的家庭里，1个主妇不工作，8个主妇工作。
- ◆ **主妇工作状态对家庭收入的影响**

收入	主妇工作状况		总计
	0 (不工作)	1 (工作)	
10	1	2	3
11	1	4	5
12	1	8	9
总计	3	14	17

收入	主妇工作状况		工作概率P
	0 (不工作)	1 (工作)	
10	1	2	2/3
11	1	4	4/5
12	1	8	8/9

$$OR_j = \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)}$$

X分别取10和11时，odd= [填空1]

X分别取12和11时，odd= [填空2]

正常使用填空题需3.0以上版本雨课堂

作答

### 3.3 优势比OR

收入	主妇工作状况		工作概率P
	0 (不工作)	1 (工作)	
10	1	2	2/3
11	1	4	4/5
12	1	8	8/9

$$OR_j = \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)}$$

X分别取10和11时，odd=4/2=2

X分别取12和11时，odd=8/4=2

### 3.3 优势比OR

收入	主妇工作状况		工作概率P
	0 (不工作)	1 (工作)	
10	1	2	2/3
11	1	4	4/5
12	1	8	8/9

$$OR_j = \frac{P_1 / (1 - P_1)}{P_0 / (1 - P_0)}$$

X分别取10和11时，odd=4/2=2

X分别取12和11时，odd=8/4=2

- 收入每增加1个单位，主妇工作的Odds增加到原来的2倍
- 说明收入对工作状态有正关系，收入越高，工作概率越高
- 在疾病检测中，说明一个因素越高，使得疾病概率越高

则有  $OR_j = \exp \beta_j$ ,  $\beta_j \begin{cases} = 0, OR_j = 1 & \text{无作用} \\ > 0, OR_j > 1 & \text{危险因子} \\ < 0, OR_j < 1 & \text{保护因子} \end{cases}$

### 3.3 优势比OR

$$\ln\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_m X_m = \text{logit} P$$

**常数项**  $\beta_0$  表示暴露剂量为0时个体发病与不发病概率之比的自然对数。

**回归系数**  $\beta_j (j = 1, 2, \dots, m)$   
表示自变量  $X_j$  改变一个单位时  
 $\text{logit} P$  的改变量。

### 3.4 logistic回归例子1

表 冠心病8个可能的危险因素与赋值

因素	变量名	赋值说明	序号	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$	$Y$
年龄(岁)	$X_1$	<45=1, 45~54=2, 55~64=3, 65~4	1	3	1	0	1	0	0	1	1	0
高血压史	$X_2$	无=0, 有=1	2	2	0	1	1	0	0	1	0	0
高血压家族史	$X_3$	无=0, 有=1	3	2	1	0	1	0	0	1	0	0
吸烟	$X_4$	不吸=0, 吸=1	4	2	0	0	1	0	0	1	0	0
高血脂史	$X_5$	无=0, 有=1	5	3	0	0	1	0	1	1	1	0
动物脂肪摄入	$X_6$	低=0, 高=1	6	3	0	1	1	0	0	2	1	0
体重指数(BMI)	$X_7$	<24=1, 24~<26=2, 26~3	7	2	0	1	0	0	0	1	0	0
A型性格	$X_8$	否=0, 是=1	8	3	0	1	1	1	0	1	0	0
冠心病	$Y$	对照=0, 病例=1	9	2	0	0	0	0	0	1	1	0
			10	1	0	0	1	0	0	1	0	0
			.	.	.	.	.	.	.	.	.	.
			.	.	.	.	.	.	.	.	.	.
			51	2	0	1	1	0	1	2	1	1
			52	2	1	1	1	0	0	2	1	1
			53	2	1	0	1	0	0	1	1	1
			54	3	1	1	0	1	0	3	1	1

建立冠心病的逻辑回归模型

### 3.5 logistic回归例子2

#### 例1、自变量是二值分类型变量

某医院为了研究导致手术切口感染的原因，收集了295例手术者情况，其中，手术时间小于或等于5小时的有242例，感染者13例；手术时间大于5小时的有53例，感染者7例。试建立手术切口感染(y)关于手术时间(x)的logistic回归模型。

y \ x	1 (>5小时)	0 ( $\leq$ 5小时)
1 (感染)	7	13
0 (未感染)	46	229
总和	53	242

### 3.6 逻辑回归参数估计

记得到一个实际观测值  $y_i (i = 1, 2, \dots, n)$  的概率为

$$P(y_i) = p_i^{y_i} (1 - p_i)^{1-y_i}$$

则似然函数为  $L = \prod_{i=1}^n P(y_i) = \prod_{i=1}^n p_i^{y_i} (1 - p_i)^{1-y_i}$

两边取对数： $\ln L = \sum_{i=1}^n [y_i \ln p_i + (1 - y_i) \ln(1 - p_i)] =$   
 $\sum_{i=1}^n [y_i \ln \frac{p_i}{1-p_i} + \ln(1 - p_i)]$

最后得到： $\ln L = \sum_{i=1}^n [y_i(\alpha + \beta x_i) - \ln(1 + \exp(\alpha + \beta x_i))]$

当使得  $\ln L$  取得最大值时， $-\frac{1}{m} \ln L$  最小时，参数估计值即为所求。

### 3.6 逻辑回归参数估计

- 使用梯度下降方法，迭代求解参数

参数估计： $\alpha = -2.869$  ( $p=0.0001$ ),

$\beta = 0.986$  ( $p=0.0468$ ).

回归模型：

如何解释系数的实际意义？

$$p(y=1|x) = \frac{e^{-2.869+0.986x}}{1+e^{-2.869+0.986x}}$$

### 3.6 逻辑回归参数解释

即  $OR_j = \exp[\beta_j(c_1 - c_0)]$

若  $X_j = \begin{cases} 1 & \text{暴露} \\ 0 & \text{非暴露}, c_1 - c_0 = 1, \end{cases}$

则有  $OR_j = \exp \beta_j, \beta_j \begin{cases} = 0, OR_j = 1 & \text{无作用} \\ > 0, OR_j > 1 & \text{危险因子} \\ < 0, OR_j < 1 & \text{保护因子} \end{cases}$

例如，手术感染问题

y	x	1 (>5小时)	0 (≤ 5小时)
1 (感染)		7	13
0 (未感染)		46	229
总和		53	242

Logistic 回归模型：

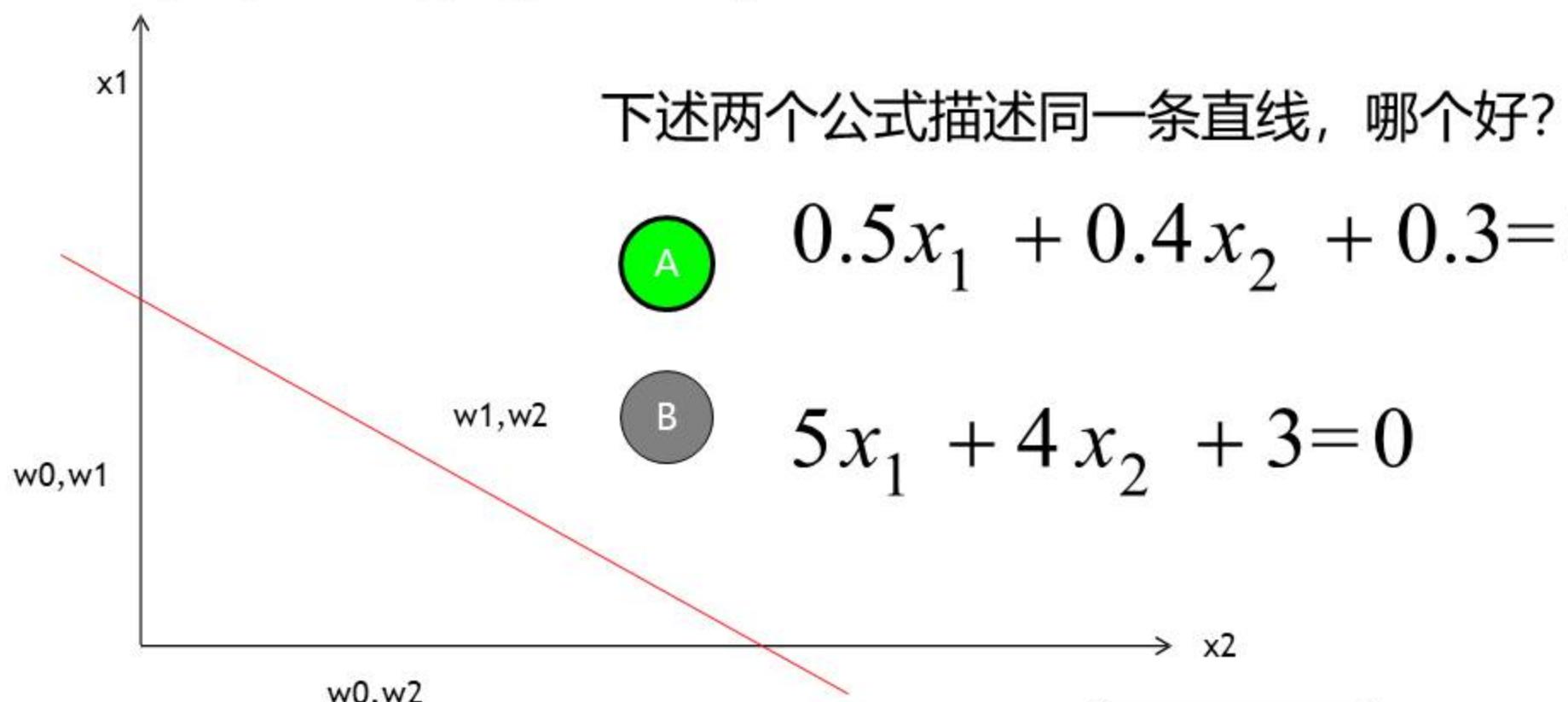
$$p(y=1|x) = \frac{e^{-2.869+0.986x}}{1+e^{-2.869+0.986x}}$$

从  $\beta = 0.986$ , 得到  $RR \approx OR = e^\beta = 2.681$ 。

所以, 手术时间大于5小时的感染率是手术时间小于或等于5小时的感染率的2.681倍, 即感染的可能性增加了186.1%。

$$\text{Logit}(p_i) = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

$w_1 x_1 + w_2 x_2 + w_0 = 0$  对应于平面的一根直线

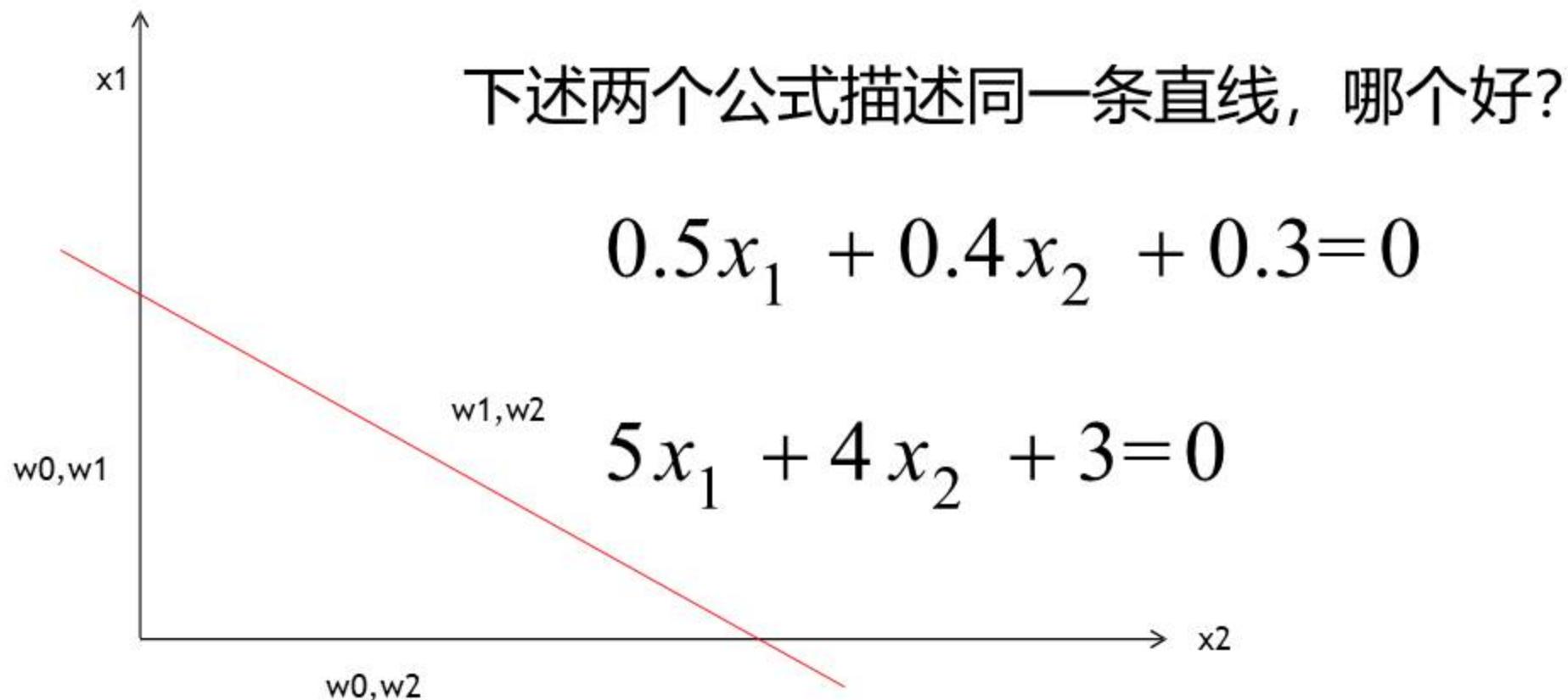


提交

### 3.7 逻辑回归正则化

$$\text{Logit}(p_i) = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

$w_1 x_1 + w_2 x_2 + w_0 = 0$  对应于平面的一根直线

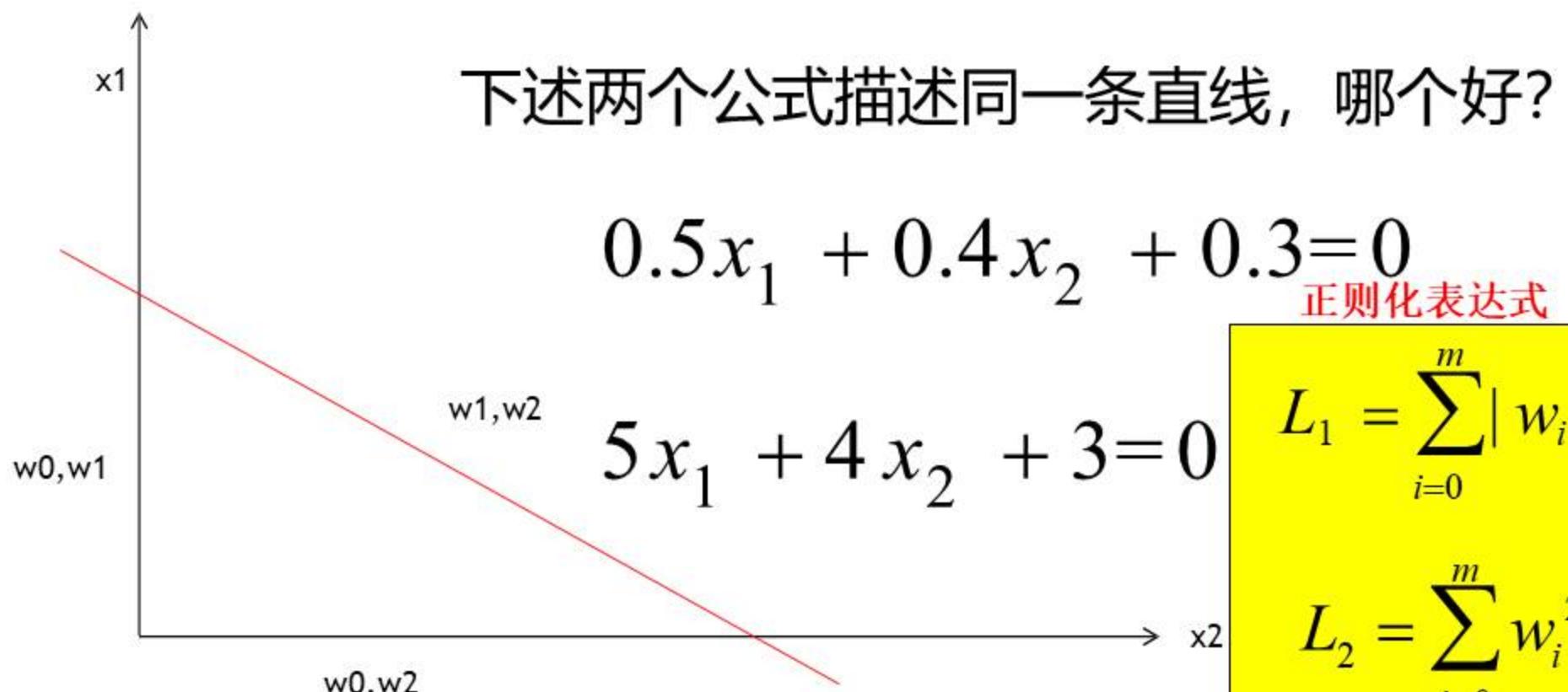


W在数值上越小越好，这样越能抵抗数据的扰动

### 3.7 逻辑回归正则化

$$\text{Logit}(p_i) = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

$w_1 x_1 + w_2 x_2 + w_0 = 0$  对应于平面的一根直线



W在数值上越小越好，这样越能抵抗数据的扰动

### 3.7 逻辑回归正则化

重写误差函数 lambda是W的权重

牺牲正确率来提高推广能力

$$E = \sum_{i=1}^n \left( y_i - \frac{1}{1 + e^{-(w_1x_{i1} + w_2x_{i2} + w_0)}} \right)^2 + \boxed{\lambda L_1}$$

$$E = \sum_{i=1}^n \left( y_i - \frac{1}{1 + e^{-(w_1x_{i1} + w_2x_{i2} + w_0)}} \right)^2 + \boxed{\lambda L_2}$$

正则化表达式

$$L_1 = \sum_{i=0}^m |w_i|$$

$$L_2 = \sum_{i=0}^m w_i^2$$

**惩罚项：**若学习到大权值使得误差小，但是再加上正则化式子以后使得上面E值变大。

因此，最小化E值使得求解的权值尽可能相对较小。

## 3.7 逻辑回归正则化

一个有趣的结论

$L_1$  倾向于使得  $w$  要么取1，要么取0

稀疏编码

$L_2$  倾向于使得  $w$  整体偏小

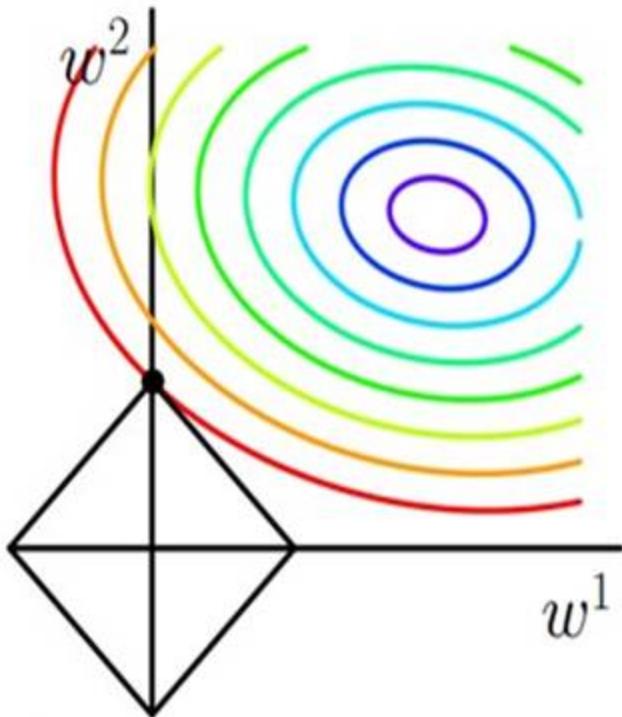
岭回归

正则化表达式

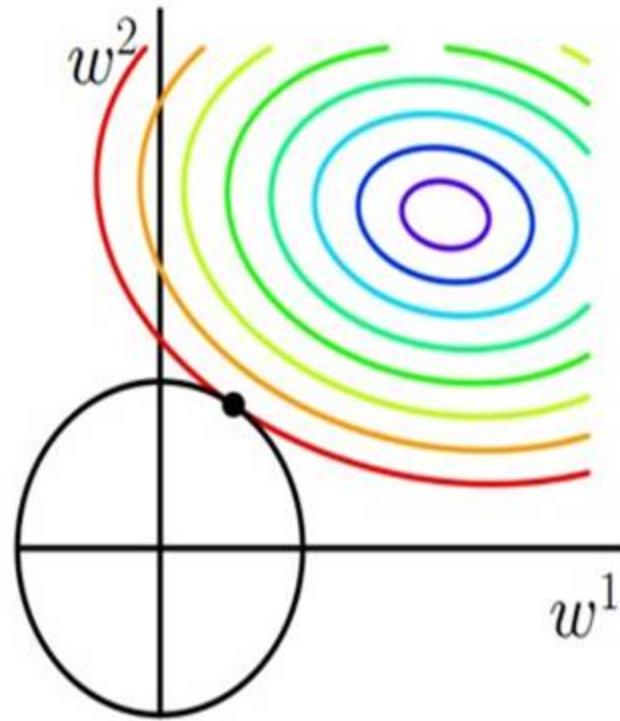
$$L_1 = \sum_{i=0}^m |w_i|$$

$$L_2 = \sum_{i=0}^m w_i^2$$

### 3.7 逻辑回归正则化



(a)  $\ell_1$ -ball meets quadratic function.  
 $\ell_1$ -ball has corners. It's very likely that  
the meet-point is at one of the corners.



(b)  $\ell_2$ -ball meets quadratic function.  
 $\ell_2$ -ball has no corner. It is very unlikely  
that the meet-point is on any of axes.

正则化表达式

$$L_1 = \sum_{i=0}^m |w_i|$$

$$L_2 = \sum_{i=0}^m w_i^2$$

### 3.7 逻辑回归正则化

适用场景：

$L_1$

适合降低维度

$L_2$

也称为岭回归，有很强的概率意义

正则化表达式

$$L_1 = \sum_{i=0}^m |w_i|$$

$$L_2 = \sum_{i=0}^m w_i^2$$

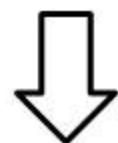
### 3.8逻辑回归数值优化1

某个地区的生态环境和动物数量的关系

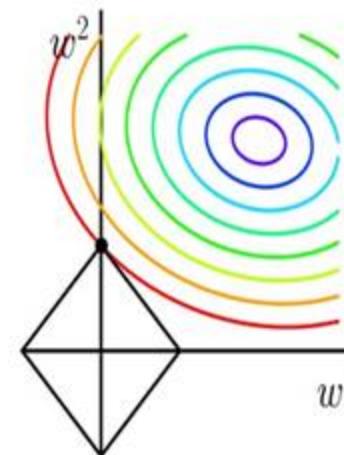
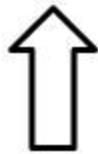
老虎数量	麻雀数量	是否污染
2	50640	1
3	55640	0
1	62020	0
0	54642	1

### 3.8 逻辑回归数值优化1

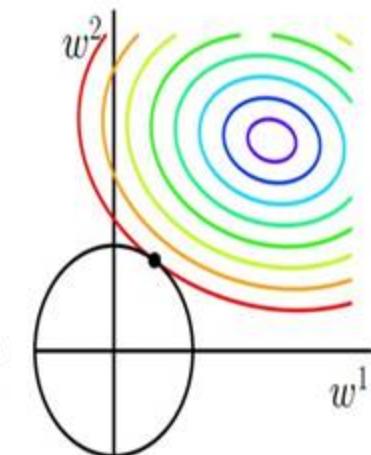
各个维度的输入如果在数值上差异很大，那么会引起正确的w在各个维度上数值差异很大



矛盾



(a)  $\ell_1$ -ball meets quadratic function.  
 $\ell_1$ -ball has corners. It's very likely that the meet-point is at one of the corners.



(b)  $\ell_2$ -ball meets quadratic function.  
 $\ell_2$ -ball has no corner. It is very unlikely that the meet-point is on any of axes."

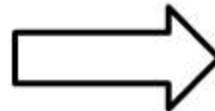
找寻w的时候，对各个维度的调整基本上是按照同一个数量级来进行调整的。

### 3.8逻辑回归数值优化1

某个地区的生态环境和动物数量的关系

老虎数量	麻雀数量	是否污染
2	50640	1
3	62020	0
1	55640	0
0	54642	1

Z得分规范



老虎数量	麻雀数量	是否污染
1.33	10.7	1
2	13.15	0
0.67	11.80	0
0	11.59	1

老虎数量	麻雀数量	是否污染
2	50640	1
3	55640	0
1	62020	0
0	54642	1

### 3.8 逻辑回归数值优化2

$$E = \sum_{i=1}^n \left( y_i - \frac{1}{1 + e^{-(w_1 x_{i1} + w_2 x_{i2} + w_0)}} \right)^2$$



$$\frac{\partial E}{\partial w} = -\sum_{i=1}^n 2 \left( y_i - \frac{1}{1 + e^{-w^T x_i}} \right) \frac{e^{-w^T x_i}}{(1 + e^{-w^T x_i})} x_i$$

$$w^{t+1} = w^t + \alpha \frac{\partial E}{\partial w}$$

$$= w^t + \alpha A x_i$$

### 3.8 逻辑回归数值优化2

$$E = \sum_{i=1}^n \left( y_i - \frac{1}{1 + e^{-(w_1 x_{i1} + w_2 x_{i2} + w_0)}} \right)^2$$



$$\frac{\partial E}{\partial w} = -\sum_{i=1}^n 2 \left( y_i - \frac{1}{1 + e^{-w^T x_i}} \right) \frac{e^{-w^T x_i}}{(1 + e^{-w^T x_i})} x_i$$

$$\begin{aligned} w^{t+1} &= w^t + \alpha \frac{\partial E}{\partial w} \\ &= w^t + \alpha A x_i \end{aligned}$$



$$\begin{pmatrix} w_0^{t+1} \\ w_1^{t+1} \\ w_2^{t+1} \end{pmatrix} = \begin{pmatrix} w_0^t \\ w_1^t \\ w_2^t \end{pmatrix} + \alpha A \begin{pmatrix} 1 \\ x_{i1} \\ x_{i2} \end{pmatrix}$$

### 3.8 逻辑回归数值优化2

$$E = \sum_{i=1}^n \left( y_i - \frac{1}{1 + e^{-(w_1 x_{i1} + w_2 x_{i2} + w_0)}} \right)^2$$



$$\frac{\partial E}{\partial w} = -\sum_{i=1}^n 2 \left( y_i - \frac{1}{1 + e^{-w^T x_i}} \right) \frac{e^{-w^T x_i}}{(1 + e^{-w^T x_i})} x_i$$

$$\begin{aligned} w^{t+1} &= w^t + \alpha \frac{\partial E}{\partial w} \\ &= w^t + \alpha A x_i \end{aligned}$$



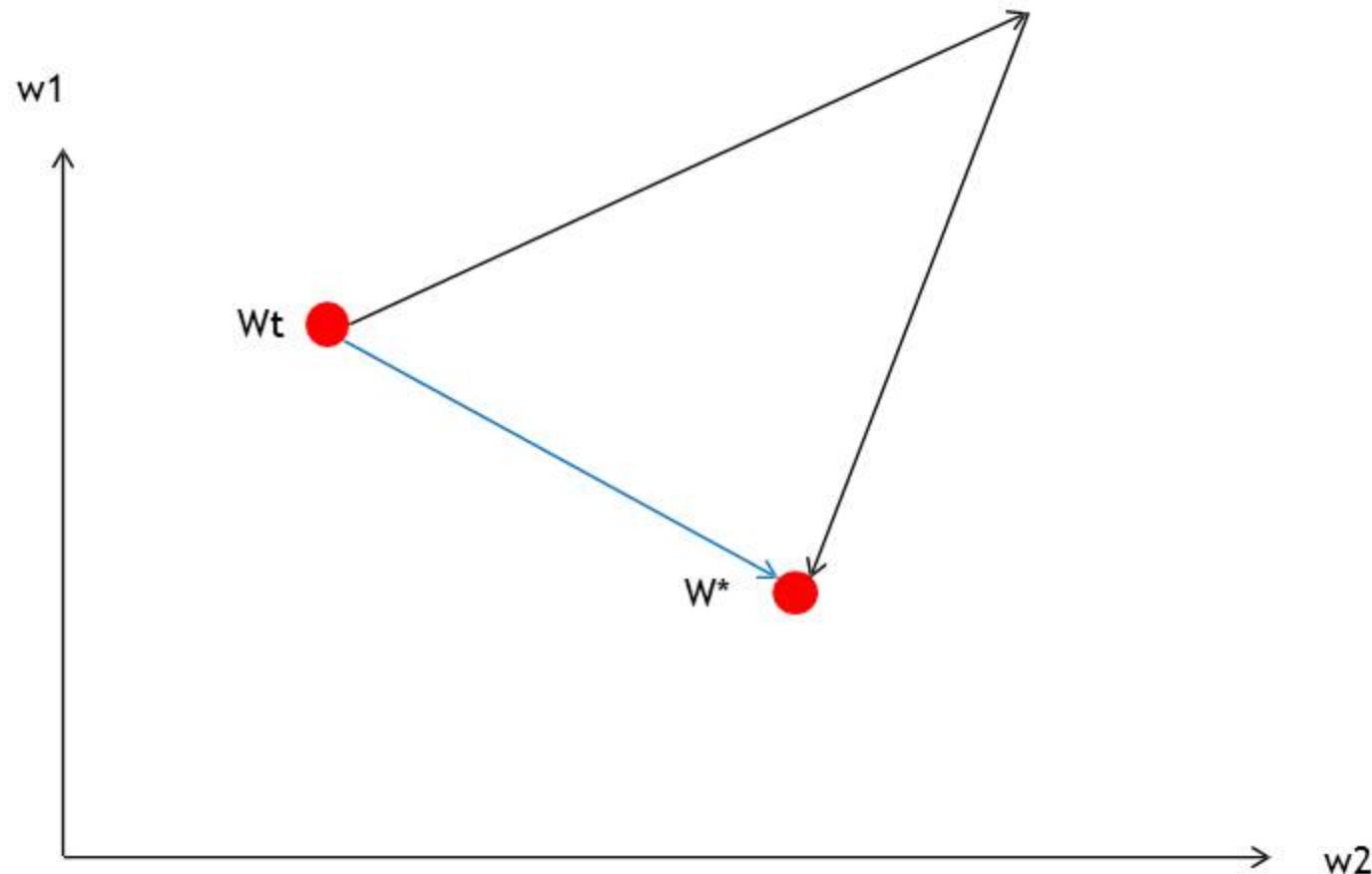
$$\begin{pmatrix} w_0^{t+1} \\ w_1^{t+1} \\ w_2^{t+1} \end{pmatrix} = \begin{pmatrix} w_0^t \\ w_1^t \\ w_2^t \end{pmatrix} + \alpha A \begin{pmatrix} 1 \\ x_{i1} \\ x_{i2} \end{pmatrix}$$

老虎数量	麻雀数量	是否污染
1.33	10.7	1
2	13.15	0
0.67	11.80	0
0	11.59	1

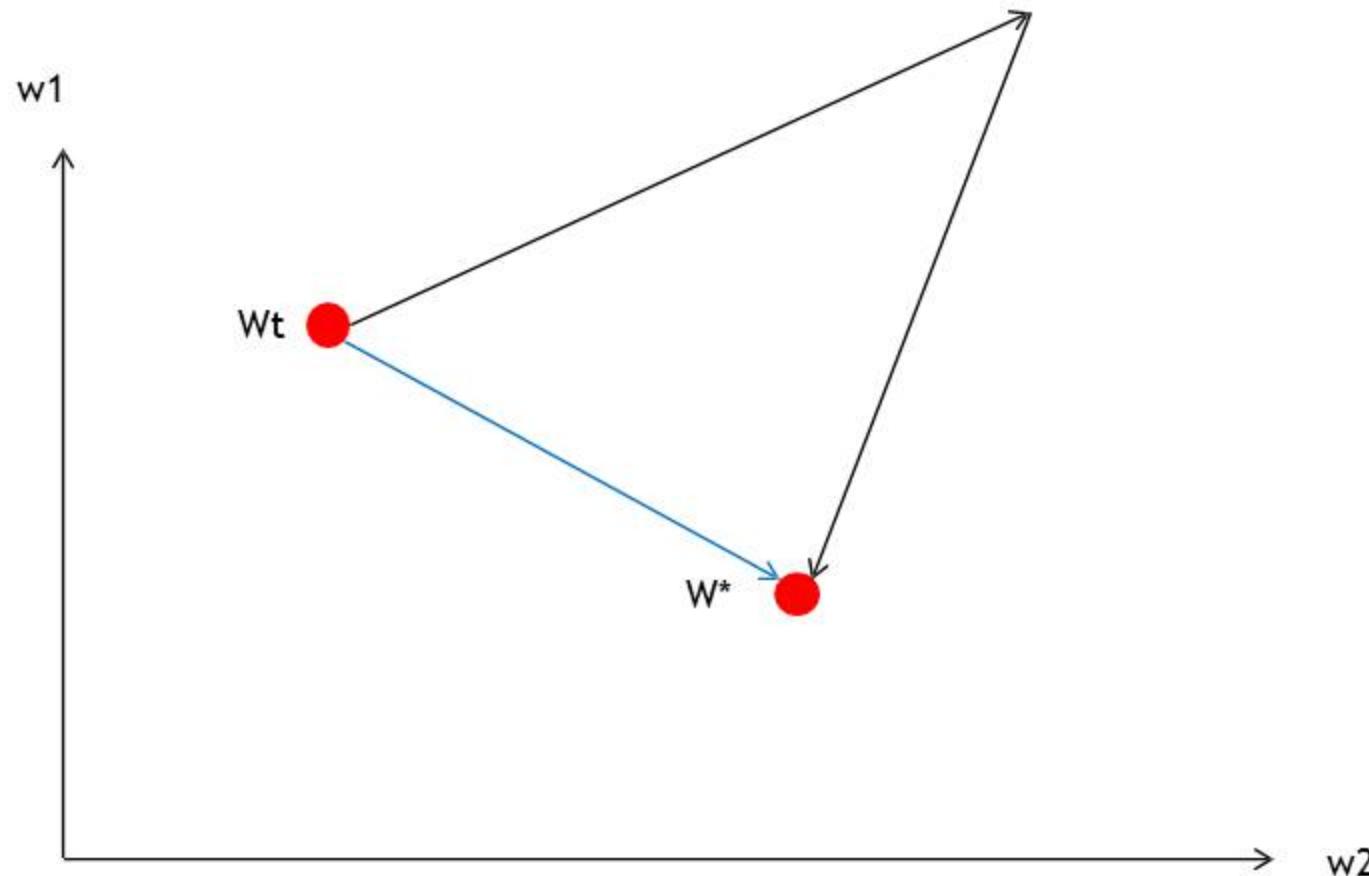
若训练集各属性值都为正

问题：w1和w2只能朝一个方向变化。  
要么同时变大，要么同时变小

### 3.8逻辑回归数值优化2



### 3.8 逻辑回归数值优化2

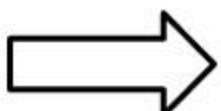


解决方法，尽可能让 $x$ 的各个维度取值上有正有负

### 3.8 逻辑回归数值优化2

老虎数量	麻雀数量	是否污染
1.33	10.7	1
2	11.80	0
0.67	13.15	0
0	11.59	1
1	11.81	均值

均值归一化

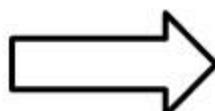


老虎数量	麻雀数量	是否污染
0.33	-1.17	1
1	-0.01	0
-0.33	1.28	0
-1	-0.22	1

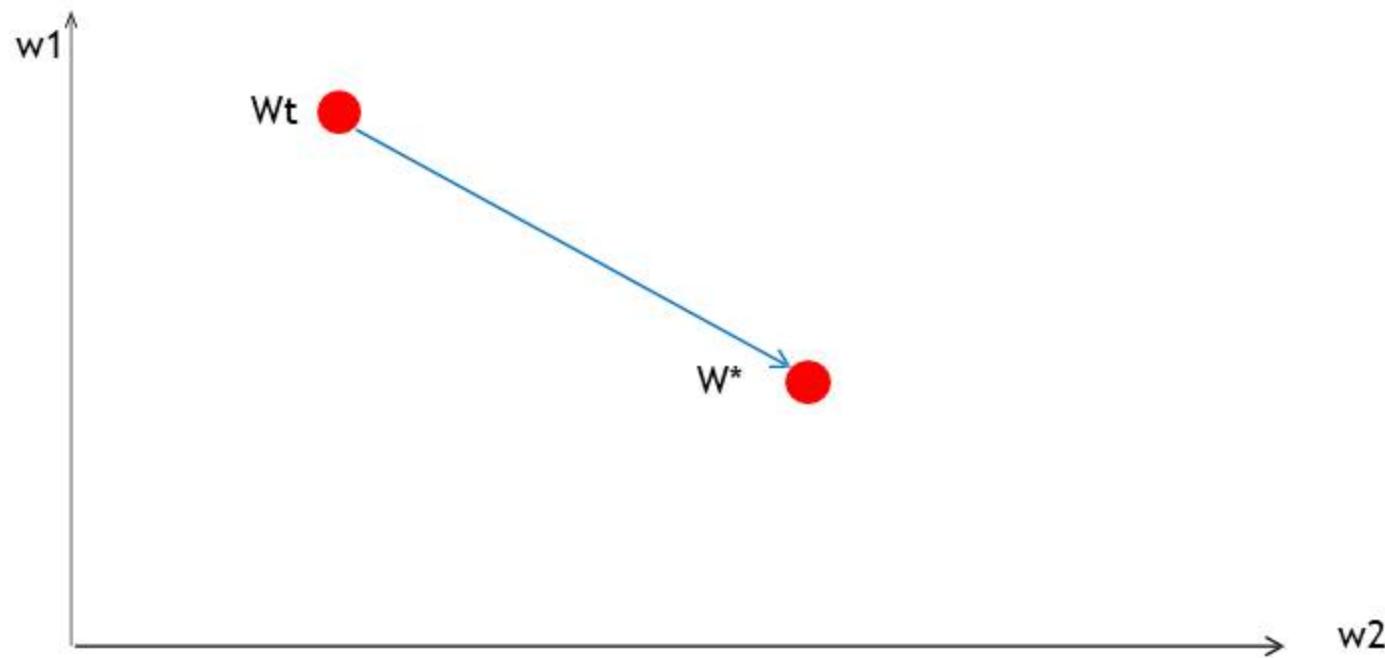
### 3.8 逻辑回归数值优化2

老虎数量	麻雀数量	是否污染
1.33	10.7	1
2	11.80	0
0.67	13.15	0
0	11.59	1
1	11.81	均值

均值归一化



老虎数量	麻雀数量	是否污染
0.33	-1.17	1
1	-0.01	0
-0.33	1.28	0
-1	-0.22	1



### 3.8 逻辑回归训练方法优化

## 梯度下降法的选择

两种梯度: 1. SGD    2. L-BFGS

L-BFGS为SGD的优化方法，它的训练速度比SGD快

	数值归一化	正则化	梯度下降法	分类个数	数据选择
LogisticRegressionWithLBFGS	需要均值归一化，算法融入方差归一化	支持L2	LBFGS (收敛快，考虑二阶导数)	支持多分类	加载所有数据，都参与训练
LogisticRegressionWithSGD	不归一化，需要专门在外面进行归一化	支持L1, L2	SGD	不支持多分类	随机从训练集选取 (支持 MiniBatch Fraction)

logistic回归是否对噪声敏感



是



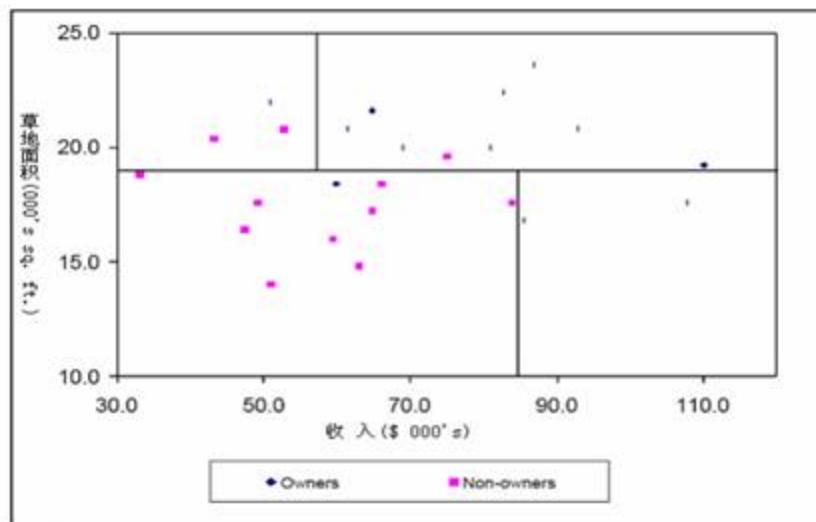
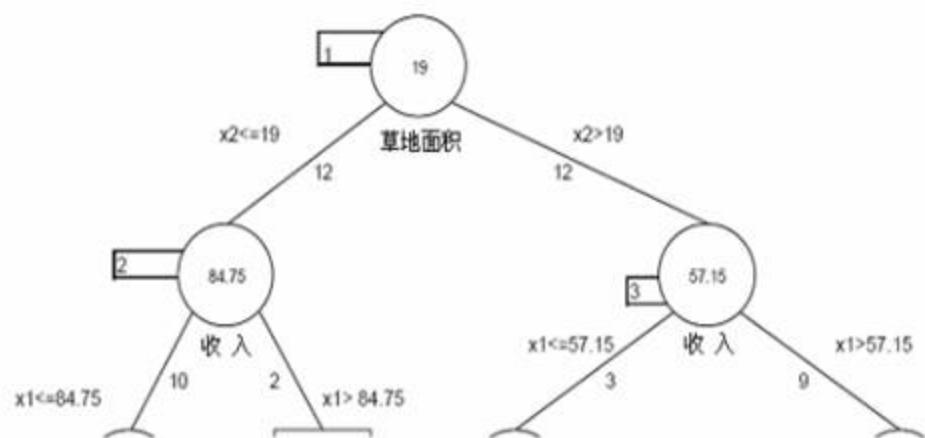
否

提交

## 4 决策树回归

- 决策树实际上是将空间用超平面进行划分的一种方法，每次分割的时候，都将当前的空间一分为二，这样使得每一个叶子节点都是在空间中的一个不相交的区域，在进行决策的时候，会根据输入样本每一维feature的值，一步一步往下，最后使得样本落入N个区域中的一个（假设有N个叶子节点），如下图所示。

图 7

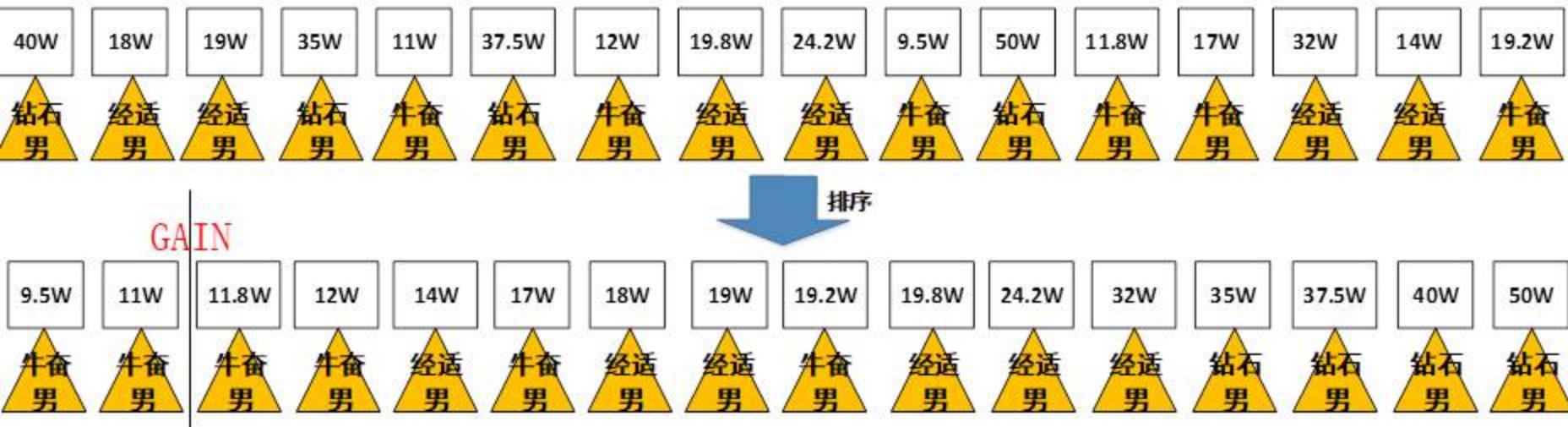


- 既然是决策树，那么必然存在以下两个核心问题：如何选择划分点？如何决定叶节点的输出值？**决策树分类选择划分点，使得信息增益最大，叶节点输出即类别**
- 一个回归树对应着输入空间（即特征空间）的一个划分以及在划分单元上的输出值。分类树中**采用信息增益等方法**，通过计算选择最佳划分点。**而在回归树中，采用的是启发式的方法。**

## 4.1 决策树分类最佳划分点选择

### ◆ 分割区间的策略

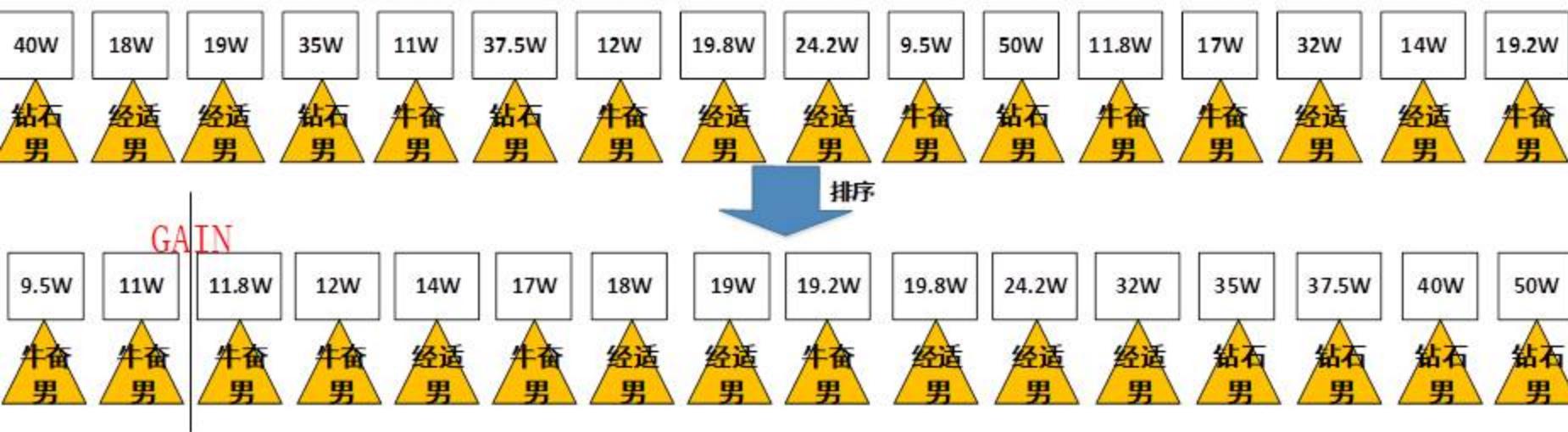
- 从最小值开始建立分割区间，开始计算各自的信息增益，  
选择**信息增益最大的一个分割区间**作为最佳划分点



## 4.2 决策树回归最佳划分点选择

### ◆ 分割区间的策略

- 从最小值开始建立分割区间，开始计算各自的信息增益，选择**信息增益最大的一个分割区间作为最佳划分点**



- 假如n个特征，每个特征有 $s_i$  ( $i \in (1, n)$ )个取值，则遍历所有特征，尝试该特征所有取值，对空间进行划分，直到**取到特征j的取值s，使得损失函数最小**，这样就得到了一个划分点。

$$\min_{js} \left[ \min_{c_1} Loss(y_i, c_1) + \min_{c_2} Loss(y_i, c_2) \right]$$

其中一个特征损失函数最小值

## 4.3 决策树回归-例子

- X的取值范围[0.5, 10.5],y的取值范围: [5.0, 10.10],用树桩做基函数;

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

- 求 $f_1(x)$ 回归树 $T_1(x)$ ,  
$$\min_s \left[ \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right]$$
- 求切分点s:  
 $R_1 = \{x \mid x \leq s\}, \quad R_2 = \{x \mid x > s\}$

## 4.3 决策树回归-例子

- X的取值范围[0.5, 10.5],y的取值范围: [5.0, 10.10],用树桩做基函数;

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

- 求 $f_1(x)$ 回归树 $T_1(x)$ ,  
$$\min_s \left[ \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right]$$
- 求切分点s:  $R_1 = \{x | x \leq s\}$ ,  $R_2 = \{x | x > s\}$

- 解题过程: 各切分点:

1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5

## 4.3 决策树回归-例子

- X的取值范围[0.5, 10.5],y的取值范围: [5.0, 10.10],用树桩做基函数;

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

- 求 $f_1(x)$ 回归树 $T_1(x)$ ,  
$$\min_s \left[ \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right]$$
- 求切分点s:  $R_1 = \{x | x \leq s\}$ ,  $R_2 = \{x | x > s\}$
- 求在 $R_1, R_2$ 内部使平方损失误差达到最小值的 $c_1, c_2$ :

$$c_1 = \frac{1}{N_1} \sum_{x_i \in R_1} y_i, \quad c_2 = \frac{1}{N_2} \sum_{x_i \in R_2} y_i$$

- 解题过程: 各切分点:

1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5

# 填空题

3分

设置

此题未设置答案，请点击右侧设置按钮

- X的取值范围[0.5, 10.5],y的取值范围: [5.0, 10.10],用树桩做基函数;

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

- 求 $f_1(x)$ 回归树 $T_1(x)$ ,  
$$\min_s \left[ \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right]$$
- 求切分点s:  $R_1 = \{x | x \leq s\}$ ,  $R_2 = \{x | x > s\}$
- 求在 $R_1, R_2$ 内部使平方损失误差达到最小值的 $c_1, c_2$ :

$$c_1 = \frac{1}{N_1} \sum_{x_i \in R_1} y_i, \quad c_2 = \frac{1}{N_2} \sum_{x_i \in R_2} y_i$$

- 解题过程: 各切分点:

1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5

当切分点为1.5时, 求 $c_1 =$  [填空1],  $c_2 =$  [填空2]  
均方误差= [填空3]

正常使用填空题需3.0以上版本雨课堂

作答

## 4.3 决策树回归-例子

- X的取值范围[0.5, 10.5],y的取值范围: [5.0, 10.10],用树桩做基函数;

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

- 求 $f_1(x)$ 回归树 $T_1(x)$ ,  
$$\min_s \left[ \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right]$$
- 求切分点s:  $R_1 = \{x | x \leq s\}$ ,  $R_2 = \{x | x > s\}$
- 求在R1,R2内部使平方损失误差达到最小值的 $c_1, c_2$ :

$$c_1 = \frac{1}{N_1} \sum_{x_i \in R_1} y_i, \quad c_2 = \frac{1}{N_2} \sum_{x_i \in R_2} y_i$$

- 解题过程: 各切分点:

1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5

$$m(s) = \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2$$

当  $s = 1.5$  时,  $R_1 = \{1\}$ ,  $R_2 = \{2, 3, \dots, 10\}$ ,  $c_1 = 5.56$ ,  $c_2 = 7.50$ ,

$$m(s) = \min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 = 0 + 15.72 = 15.72$$

## 4.3 决策树回归-例子

• 全部：

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
m(s)	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

回归树 T<sub>1</sub>

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases} \quad c_1 = \frac{1}{N_1} \sum_{x_i \in R_1} y_i, \quad c_2 = \frac{1}{N_2} \sum_{x_i \in R_2} y_i$$

$$f_1(x) = T_1(x)$$



• 全部:

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
m(s)	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

回归树  $T_1$ 

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases}$$

$$f_1(x) = T_1(x)$$

计算  
数据  
残差

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05
$r_{2i} = y_i - f_1(x_i)$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	
$x_i$	1	2	3	4	5	6	7	8	9	10
$r_{2i}$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	

用  $f_1$  拟合数据的平方误差:

$$L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$$

第二步: 求  $T_2$ ,

$$T_2(x) = \begin{cases} -0.52, & x < 3.5 \\ 0.22, & x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, & x < 3.5 \\ 6.46, & 3.5 \leq x < 6.5 \\ 9.13, & x \geq 6.5 \end{cases}$$

$$L(y, f_2(x)) = \sum_{i=1}^{10} (y_i - f_2(x_i))^2 = 0.79$$

第10个点的数据残差 = [填空1]

## 4.3 决策树回归-例子

• 全部:

<i>s</i>	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
$m(s)$	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

回归树  $T_1$

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases}$$

$$f_1(x) = T_1(x)$$

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05
$r_{2i} = y_i - f_1(x_i)$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

用  $f_1$  拟合数据的平方误差:

$$L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$$

第二步: 求  $T_2$ ,

$$T_2(x) = \begin{cases} -0.52, & x < 3.5 \\ 0.22, & x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, & x < 3.5 \\ 6.46, & 3.5 \leq x < 6.5 \\ 9.13, & x \geq 6.5 \end{cases}$$

$$L(y, f_2(x)) = \sum_{i=1}^{10} (y_i - f_2(x_i))^2 = 0.79$$

计算  
数据  
残差

## 4.3 决策树回归-例子

• 全部：

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
$m(s)$	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

回归树  $T_1$

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases}$$

$$f_1(x) = T_1(x)$$

$$r_{2i} = y_i - f_1(x_i)$$

$x_i$	1	2	3	4	5	6	7	8	9	10
$r_{2i}$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

用  $f_1$  拟合数据的平方误差：

$$L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$$

第二步：求  $T_2$ ，

$$T_2(x) = \begin{cases} -0.52, & x < 3.5 \\ 0.22, & x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, & x < 3.5 \\ 6.46, & 3.5 \leq x < 6.5 \\ 9.13, & x \geq 6.5 \end{cases}$$

$$L(y, f_2(x)) = \sum_{i=1}^{10} (y_i - f_2(x_i))^2 = 0.79$$

## 4.3 决策树回归-例子

• 全部：

s	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
m(s)	15.72	12.07	8.36	5.78	3.91	1.93	8.01	11.73	15.74

回归树  $T_1$

$$T_1(x) = \begin{cases} 6.24, & x < 6.5 \\ 8.91, & x \geq 6.5 \end{cases}$$

$$f_1(x) = T_1(x)$$

每一次进行回归树生成时采用的训练数据  $r$  都是上次预测结果  $f_m(x)$  与训练数据值  $y_i$  之间的残差。这个残差会逐渐的减小。

$$r_{2i} = y_i - f_1(x_i)$$

$x_i$	1	2	3	4	5	6	7	8	9	10
$r_{2i}$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

用  $f_1$  拟合数据的平方误差：

$$L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$$

第二步：求  $T_2$ ，

$$T_2(x) = \begin{cases} -0.52, & x < 3.5 \\ 0.22, & x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, & x < 3.5 \\ 6.46, & 3.5 \leq x < 6.5 \\ 9.13, & x \geq 6.5 \end{cases}$$

$$L(y, f_2(x)) = \sum_{i=1}^{10} (y_i - f_2(x_i))^2 = 0.79$$

## 4.3 决策树回归-例子

- 则接下来

$$T_3(x) = \begin{cases} 0.15, & x < 6.5 \\ -0.22, & x \geq 6.5 \end{cases} \quad L(y, f_3(x)) = 0.47$$

$$T_4(x) = \begin{cases} -0.16, & x < 4.5 \\ 0.11, & x \geq 4.5 \end{cases} \quad L(y, f_4(x)) = 0.30$$

$$T_5(x) = \begin{cases} 0.07, & x < 6.5 \\ -0.11, & x \geq 6.5 \end{cases} \quad L(y, f_5(x)) = 0.23$$

$$T_6(x) = \begin{cases} -0.15, & x < 2.5 \\ 0.04, & x \geq 2.5 \end{cases}$$

$$f_6(x) = f_5(x) + T_6(x) = T_1(x) + \dots + T_5(x) + T_6(x)$$

$$= \begin{cases} 5.63, & x < 2.5 \\ 5.82, & 2.5 \leq x < 3.5 \\ 6.56, & 3.5 \leq x < 4.5 \\ 6.83, & 4.5 \leq x < 6.5 \\ 8.95, & x \geq 6.5 \end{cases}$$

$$L(y, f_6(x)) = \sum_{i=1}^{10} (y_i - f_6(x_i))^2 = 0.17$$

- 此时已满足误差要求，则那么  $f(x)=f_6(x)$  即为所求提升树。

## 4.3 决策树回归-例子

- 则接下来

$$T_3(x) = \begin{cases} 0.15, & x < 6.5 \\ -0.22, & x \geq 6.5 \end{cases} \quad L(y, f_3(x)) = 0.47$$

$$T_4(x) = \begin{cases} -0.16, & x < 4.5 \\ 0.11, & x \geq 4.5 \end{cases} \quad L(y, f_4(x)) = 0.30$$

$$T_5(x) = \begin{cases} 0.07, & x < 6.5 \\ -0.11, & x \geq 6.5 \end{cases} \quad L(y, f_5(x)) = 0.23$$

$$T_6(x) = \begin{cases} -0.15, & x < 2.5 \\ 0.04, & x \geq 2.5 \end{cases}$$

$$f_6(x) = f_5(x) + T_6(x) = T_1(x) + \dots + T_5(x) + T_6(x)$$

$$= \begin{cases} 5.63, & x < 2.5 \\ 5.82, & 2.5 \leq x < 3.5 \\ 6.56, & 3.5 \leq x < 4.5 \\ 6.83, & 4.5 \leq x < 6.5 \\ 8.95, & x \geq 6.5 \end{cases}$$

$$L(y, f_6(x)) = \sum_{i=1}^{10} (y_i - f_6(x_i))^2 = 0.17$$

- 此时已满足误差要求，则那么  $f(x)=f_6(x)$  即为所求提升树。

$x_i$	1	2	3	4	5	6	7	8	9	10
$y_i$	5.56	5.70	5.91	6.40	6.80	7.05	8.90	8.70	9.00	9.05

对于训练集以为的新数据，可以实现回归预测

## 4.3 决策树回归算法

- 每一次进行回归树生成时采用的训练数据r都是上次预测结果 $f_{m-1}(x)$ 与训练数据值 $y_i$ 之间的残差。这个残差会逐渐的减小。
- 算法流程：
  - 初始化 $f_0(x) = 0$ ；
  - 对于 $m=1, 2, \dots, M$ 
    - 按照  $r = y_i - f_{m-1}(x)$  计算残差作为新的训练数据的  $y$
    - 拟合残差  $r$  学习一颗回归树，得到这一轮的回归树  $T(x_i; \Theta_m)$
    - 更新
  - 得到回归提升树：  $f_m(x) = f_{m-1}(x) + T(x; \Theta_m)$

$x_i$	1	2	3	4	5	6	7	8	9	10
$r_{2i}$	-0.68	-0.54	-0.33	0.16	0.56	0.81	-0.01	-0.21	0.09	0.14

用  $f_1$  拟合数据的平方误差：

$$L(y, f_1(x)) = \sum_{i=1}^{10} (y_i - f_1(x_i))^2 = 1.93$$

第二步：求  $T_2$ ，

$$T_2(x) = \begin{cases} -0.52, & x < 3.5 \\ 0.22, & x \geq 3.5 \end{cases}$$

$$f_2(x) = f_1(x) + T_2(x) = \begin{cases} 5.72, & x < 3.5 \\ 6.46, & 3.5 \leq x < 6.5 \\ 9.13, & x \geq 6.5 \end{cases}$$

$$L(y, f_2(x)) = \sum_{i=1}^{10} (y_i - f_2(x_i))^2 = 0.79$$

$$T_6(x) = \begin{cases} -0.15, & x < 2.5 \\ 0.04, & x \geq 2.5 \end{cases}$$

均方误差变小

$$f_6(x) = f_5(x) + T_6(x) = T_1(x) + \dots + T_5(x) + T_6(x)$$

$$= \begin{cases} 5.63, & x < 2.5 \\ 5.82, & 2.5 \leq x < 3.5 \\ 6.56, & 3.5 \leq x < 4.5 \\ 6.83, & 4.5 \leq x < 6.5 \\ 8.95, & x \geq 6.5 \end{cases}$$

$$L(y, f_6(x)) = \sum_{i=1}^{10} (y_i - f_6(x_i))^2 = 0.17$$

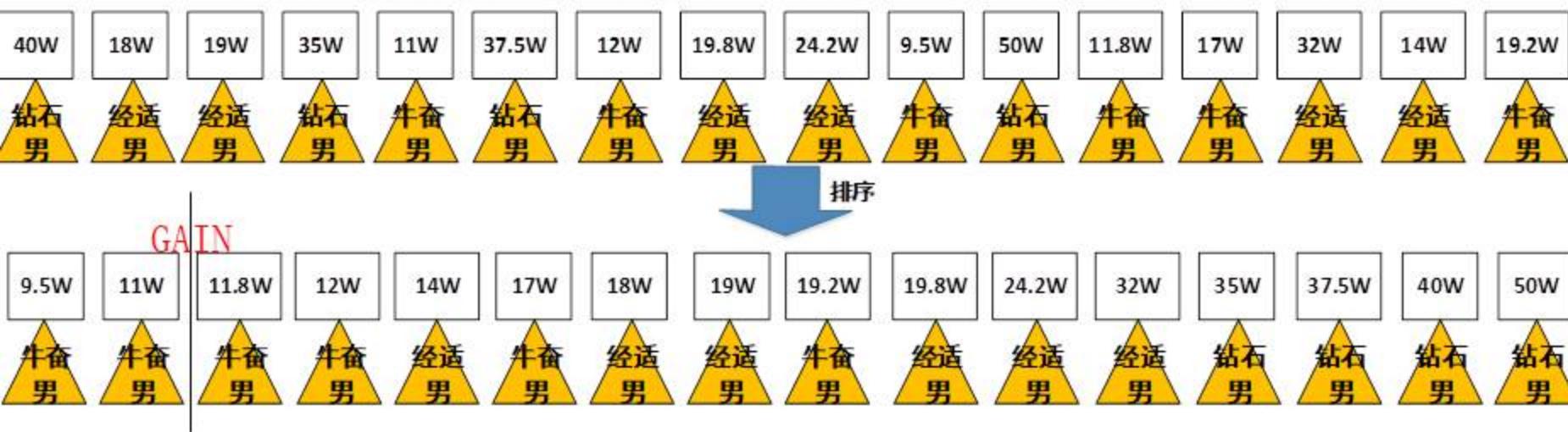
均方误差变小

$$f_m(x) = f_{m-1}(x) + T(x; \Theta_m)$$

# 4 决策树回归总结-最佳划分点选择

## ◆ 分割区间的策略

- 从最小值开始建立分割区间，开始计算各自的信息增益，选择**信息增益最大的一个分割区间作为最佳划分点**



- 假如n个特征，每个特征有 $s_i$  ( $i \in (1, n)$ )个取值，则遍历所有特征，尝试该特征所有取值，对空间进行划分，直到**取到特征j的取值s，使得损失函数最小**，这样就得到了一个划分点。

$$\min_{js} \left[ \min_{c_1} Loss(y_i, c_1) + \min_{c_2} Loss(y_i, c_2) \right]$$

其中一个特征损失函数最小值

- ◆ 线性回归
  - 连续型因变量态-数值预测
- ◆ 逻辑回归
  - 分类型因变量-类别预测
- ◆ 决策树回归-数值预测
  - 连续型因变量

# 回归编程实践

- ◆ [https://scikit-learn.org/stable/modules/linear\\_model.html#logistic-regression](https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)
- ◆ [https://scikit-learn.org/stable/modules/linear\\_model.html#ridge-regression-and-classification](https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification)
- ◆ [https://scikit-learn.org/stable/modules/linear\\_model.html#stochastic-gradient-descent-sgd](https://scikit-learn.org/stable/modules/linear_model.html#stochastic-gradient-descent-sgd)
- ◆ <https://scikit-learn.org/stable/modules/tree.html#regression>
- 同学们可以尝试利用python读入本地房价数据集和iris，来完成回归，分析其分类效果

# 第11次课后作业

- ◆ 第十一次课后作业-在educoder平台上完成作业
- ◆ <https://www.educoder.net/shixuns/w638nygr/challenges>
- ◆ <https://www.educoder.net/shixuns/4fhemfr9/challenges>
- ◆ <https://www.educoder.net/shixuns/cbsfh3r5/challenges>
- ◆ <https://www.educoder.net/shixuns/4awq25iv/challenges>
- ◆ <https://www.educoder.net/shixuns/tw9up75v/challenges>
- ◆ <https://www.educoder.net/shixuns/ya8h7utx/challenges>
- ◆ <https://www.educoder.net/shixuns/4ryluh2f/challenges>
- ◆ <https://www.educoder.net/shixuns/fglhnx5a/challenges>
- ◆ <https://www.educoder.net/shixuns/bei9fzxc/challenges>
- ◆ <https://www.educoder.net/shixuns/wexm87cf/challenges>

提交作业截至时间：2020年3月26日