人工智能之机器学习

线性模型（Linear Model）

上海育创网络科技股份有限公司

主讲人：赵翌臣

课程内容

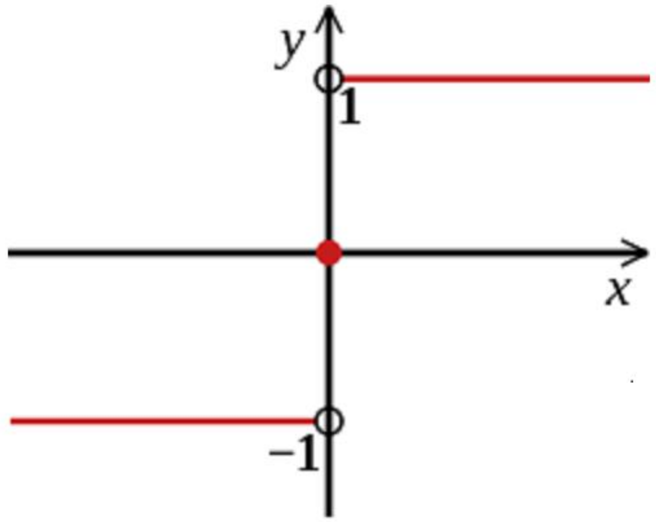
- Linear Regression 回归
- Logistic Regression 分类
- Softmax Regression*

符号介绍

- R：实数集
- R^n：n维向量空间
- X：输入空间
- Y：输出空间
- $x \in \mathrm{X}$：输入，实例
- $y \in \mathrm{Y}$：输出，标记
- $T = \{(x_1, y_1), \cdots, (x_N, y_N)\}$：训练数据集
- N：样本容量

- (x_i, y_i)：第 i 个训练数据点
- $x = (x^{(1)}, \cdots, x^{(n)})^T$：输入向量
- $x_i^{(j)}$：输入向量 i 的第 j 个分量
- θ：抽象模型参数
- $w = (w_1, \cdots, w_n)^T$：模型参数
- b：模型参数

符号函数和指示函数



$$I(x)=\begin{cases}1, x=\text{true}\\0, x=\text{false}\end{cases}$$

$$1(x)=\begin{cases}1, x=\text{true}\\0, x=\text{false}\end{cases}$$

符号函数sign(x)

指示函数 / 示性函数1(x)

其功能是取某个数的符号（正或负）：

当x>0，sign(x)=1

当x=0，sign(x)=0

当x<0， sign(x)=-1

当x=真，1(x)=1

当x=假，1(x)=0

什么是回归算法

- 回归算法是一种有监督算法
- 回归算法是一种比较常用的机器学习算法，用于构建一个模型来做特征向量到标签的映射。在算法的学习过程中，试图寻找一个模型，最大程度拟合训练数据。
- 回归算法在使用时，接收一个n维度特征向量，输出一个连续的数据值

一元线性回归

房屋面积(m^2)	租赁价格(1000￥)
10	0.8
15	1
20	1.8
30	2
50	3.2
60	3
60	3.1
70	3.5

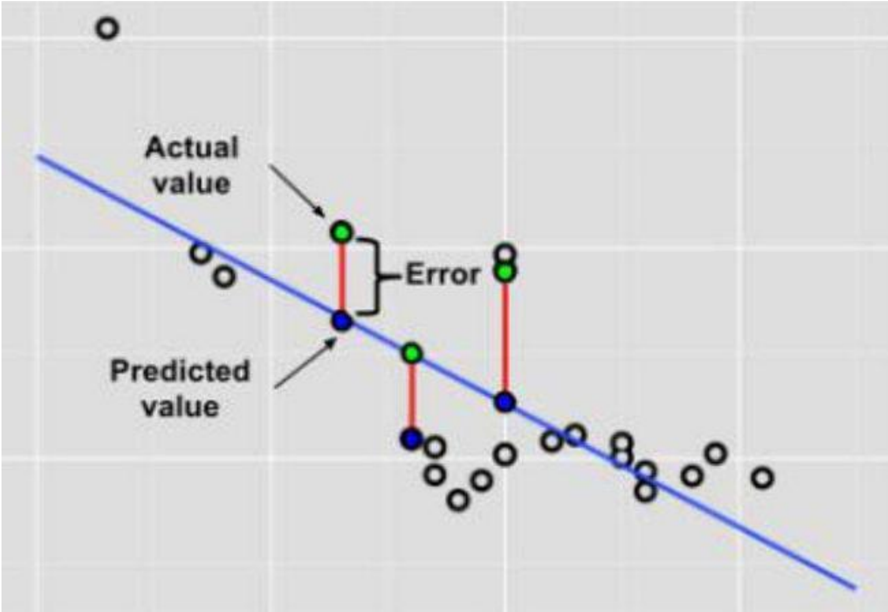
hypothesis 假设函数

$$h(x) = w_1 x^{(1)} + b$$

- 输入： x 特征向量
- 输出： $h_\theta(x)$即预测值

最小二乘法

- 最小二乘法（又称最小平方法）是一种数学优化技术。它由两部分组成：
  - 一、计算所有样本误差的平均（代价函数）
  - 二、使用最优化方法寻找数据的最佳函数匹配（抽象的）
- 最小二乘法是抽象的，具体的最优化方法有很多，比如正规方程法、梯度下降法、牛顿法、拟牛顿法等等

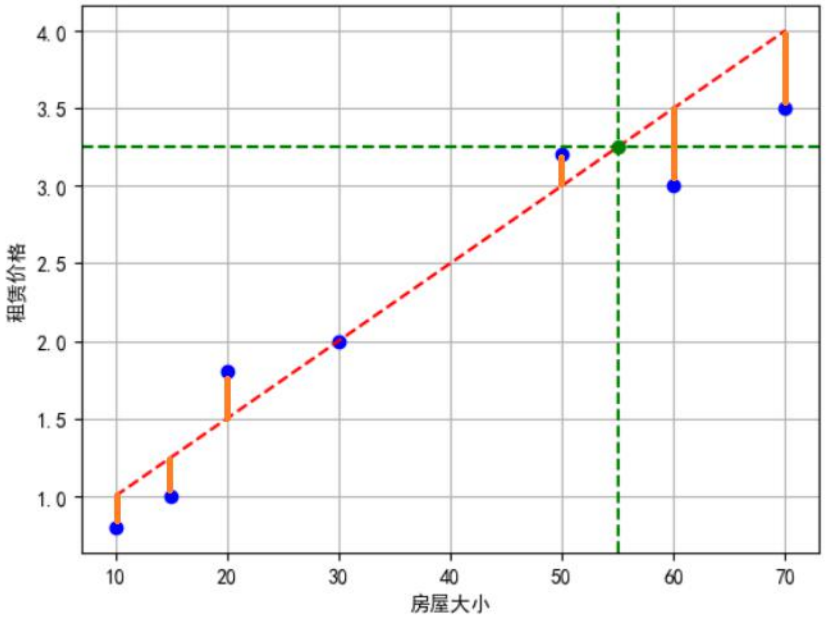


损失函数，代价函数，目标函数

- 损失函数（Loss Function）定义在单个样本上，算的是一个样本的误差。比如： $loss(\theta)=(\hat{y}_i-y_i)^2$,其中$\hat{y}_i=h_\theta(x_i)$
- 代价函数（Cost Function）定义在整个训练集上，是所有样本误差的平均，也就是损失函数的平均，比如：
$$Cost(\theta)=\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i-y_i)^2,\text{其中}\hat{y}_i=h_\theta(x_i)$$
- 目标函数（Object Function）是最终需要优化的函数。
  - 即：经验风险+正则化项（Cost Function + Regularization）。

$$Obj(\theta)=\frac{1}{N}\sum_{i=1}^{N}Loss(\hat{y}_i,y_i)+\lambda R_\theta$$

一元线性回归的代价函数

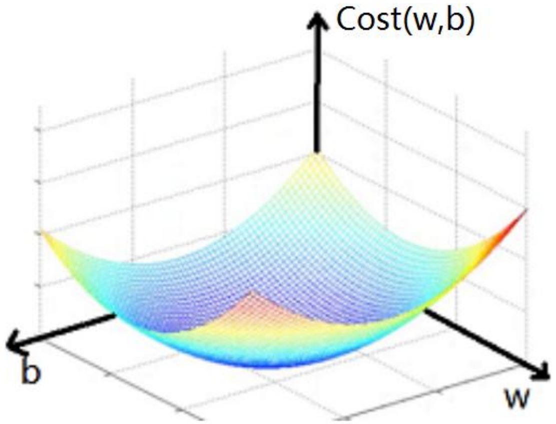


$$h(x) = w_1 x^{(1)} + b$$

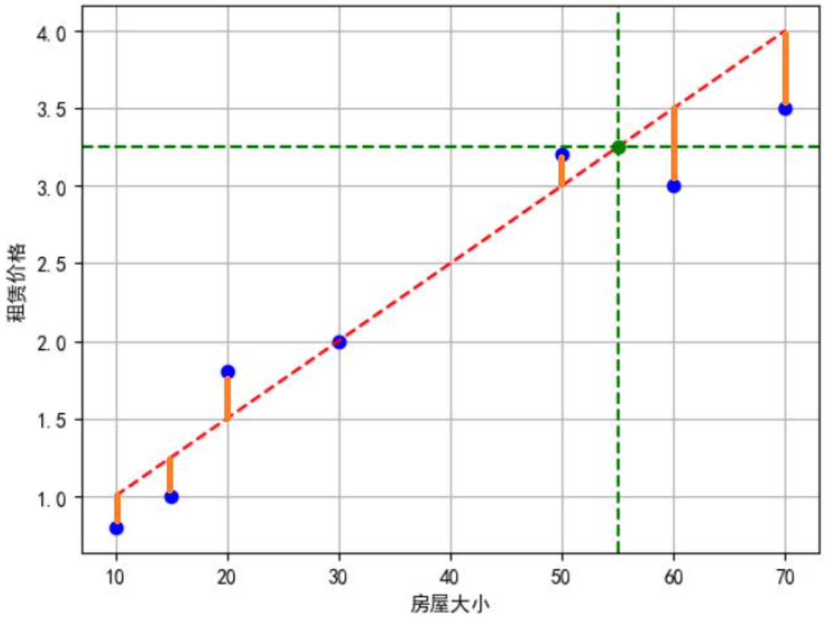
- 代价函数:

$$Cost(w,b) = \frac{1}{N}\sum_{i=1}^{N}(h(x_i) - y_i)^2$$

$$= \frac{1}{N}\sum_{i=1}^{N}(w_1 x_i^{(1)} + b - y_i)^2$$



一元线性回归的代价函数

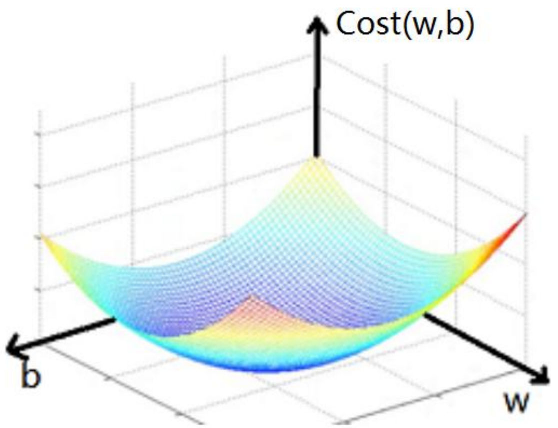


$$h(x) = w_1 x^{(1)} + b$$

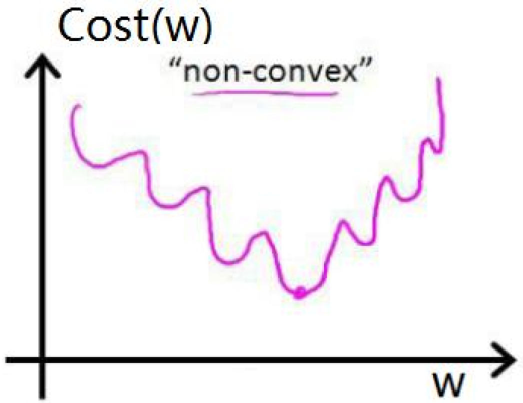
- 代价函数:

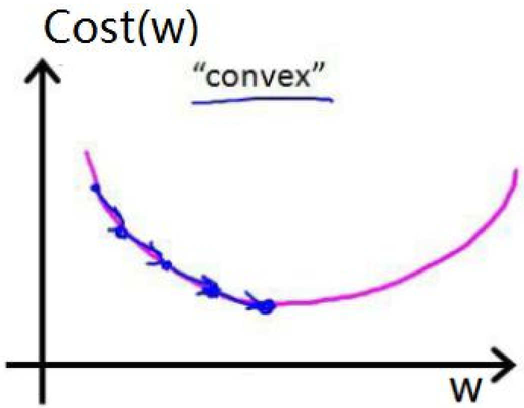
$$Cost(w,b) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(h(x_i) - y_i)^2$$

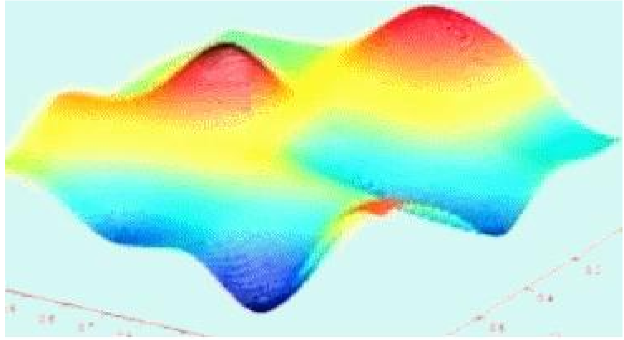
$$= \frac{1}{2N}\sum_{i=1}^{N}(w_1 x_i^{(1)} + b - y_i)^2$$

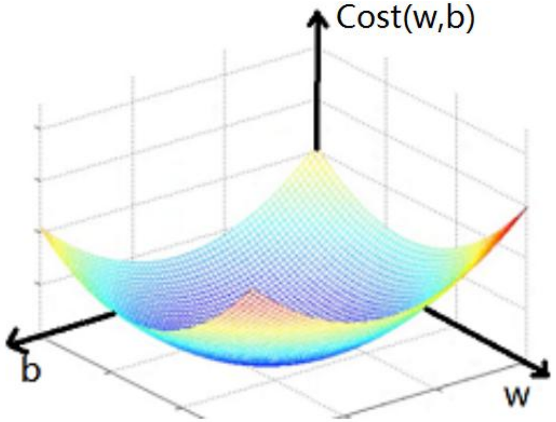


代价函数图像







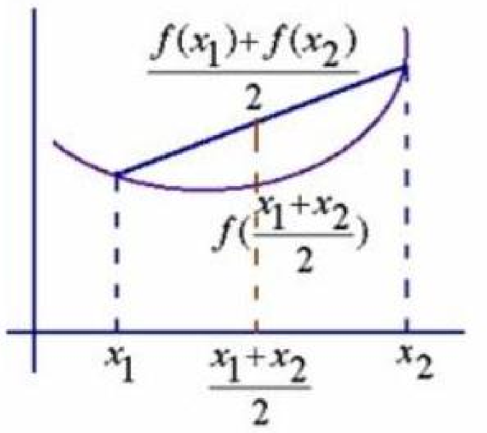


猜猜是哪个?

- 使用凸优化方法就能找到Cost(w,b)的极小值点，也就是反解出了 w^*, b^*
- 那么，线性模型为： $h(x) = w_1^* x^{(1)} + b^*$

附：线性回归模型的代价函数是凸函数的证明

- 定义
  - 对区间[a,b]上定义的函数 f ，若它对区间中任意两点x_1，x_2均有 $f(\frac{x_1+x_2}{2}) \le \frac{f(x_1)+f(x_2)}{2}$ 则称 f 为区间[a,b]上的凸函数
  - U型曲线的函数如 $f(x)=x^2$，通常是凸函数



- 判别
  - 对实数集上的函数，可以通过求二阶导数来判别：
  - 若二阶导数在区间上非负，则称为凸函数；若恒大于0，则称为严格凸函数

$$Cost(w) = \frac{1}{2N}\sum_{i=1}^{N}(w \cdot x_i - y_i)^2$$

$$Cost'(w) = \frac{1}{N}\sum_{i=1}^{N}(w \cdot x_i - y_i)x_i$$

$$Cost''(w) = \frac{1}{N}\sum_{i=1}^{N}(x_i)^2 > 0$$

模型使用

- 请问，如果现在有一个房屋面积为55平，请问最终的租赁价格是多少比较合适?
- （55，？）->　$h(x)=w_1^* x^{(1)}+b^*$ ->（55，3.3）

多元线性回归

房屋面积	房间数量	租赁价格
10	1	0.8
20	1	1.8
30	1	2.2
30	2	2.5
70	3	5.5
70	2	5.2
.....	......	....

$$h(x) = w_1 x^{(1)} + b$$

$$h(x) = w_1 x^{(1)} + w_2 x^{(2)} + b$$

写成向量的形式

$$h(x) = w \cdot x + b$$

简化符号，可以令:

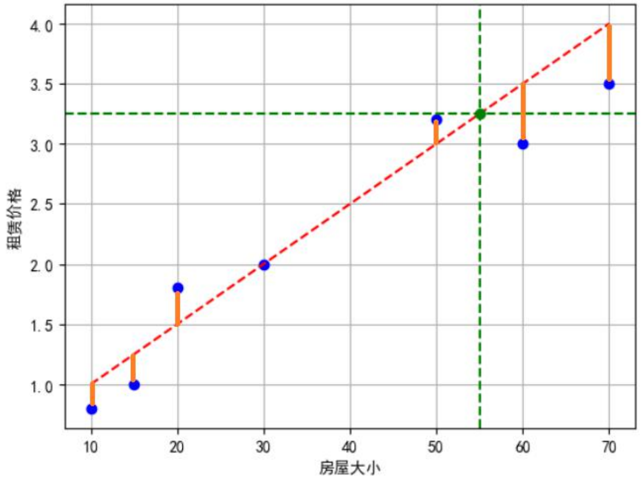
$$w = (w_1, \cdots, w_n, b)$$

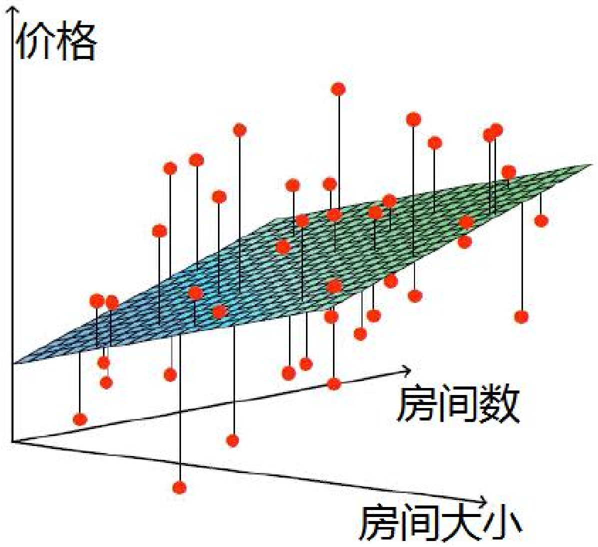
$$x = (x^{(1)}, \cdots, x^{(n)}, 1)$$

数学模型可以表示为

$$h(x) = w \cdot x$$

代价函数





$$h(x) = w_1 x^{(1)} + b$$

$$h(x) = w \cdot x$$

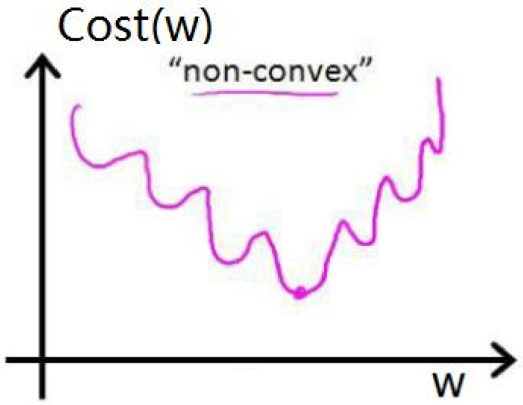
$$Cost(w,b) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(h(x_i) - y_i)^2$$

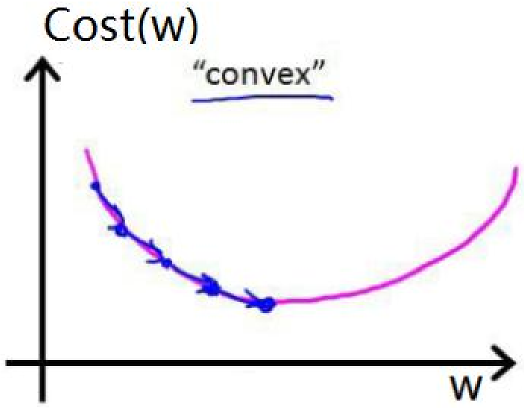
$$= \frac{1}{2N}\sum_{i=1}^{N}(w_1 x_i^{(1)} + b - y_i)^2$$

$$Cost(w) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(h(x_i) - y_i)^2$$

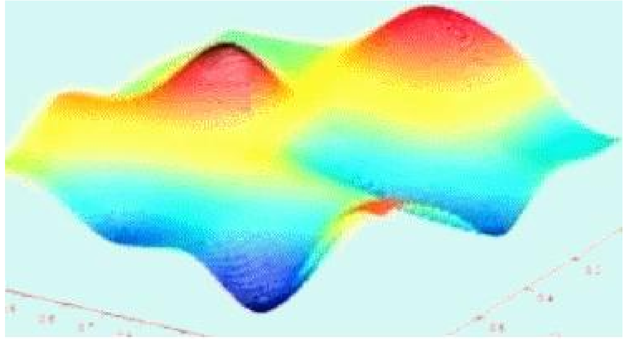
$$= \frac{1}{2N}\sum_{i=1}^{N}(w \cdot x - y_i)^2$$

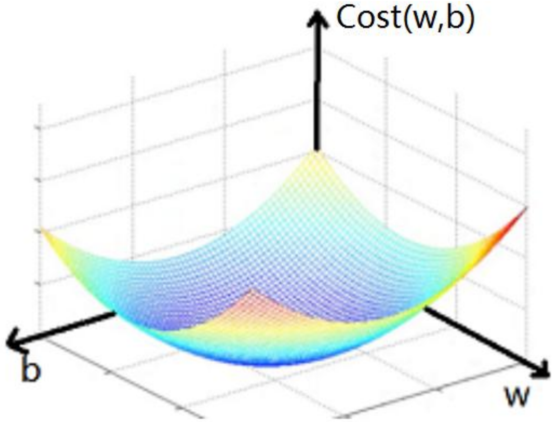
代价函数图像





猜猜是哪个?





- ，使用凸优化方法就能找到Cost(w)的极小值点，也就是反解出了w^*
- 那么，线性模型为： $h(x) = w^* \cdot x$

模型使用

- 请问，如果现在有一个房屋面积为55平，2个房间，请问最终的租赁价格是多少比较合适?
- （55，3，？）-> $h(x) = w^* \cdot x = w_1^* x^{(1)} + w_2^* x^{(2)} + b^*$

-> （55，3，3.3)

w的求解过程(正规方程法)

$$w^* = \underset{w}{\operatorname{argmin}}\ Cost(w)$$

$$Cost(w) = \frac{1}{2N}\sum_{i=1}^{N}\left(h_w(x_i) - y_i\right)^2 = \frac{1}{2N}(Xw - Y)^T(Xw - Y)$$

$$= \frac{1}{2N}\left(w^T X^T - Y^T\right)(Xw - Y)$$

$$= \frac{1}{2N}\left(w^T X^T Xw - w^T X^T Y - Y^T Xw + Y^T Y\right)$$

$$\xRightarrow{\text{对w求导}} \frac{\partial Cost(w)}{\partial w} = \frac{1}{2N}\left(2X^T Xw - X^T Y - \left(Y^T X\right)^T\right)$$

$$= \frac{1}{N}\left(X^T Xw - X^T Y\right) \xRightarrow{\text{令}=0} w^* = \left(X^T X\right)^{-1} X^T Y$$

附：w的求解过程(正规方程法)

$$\sum_{i=1}^{N}\left(h_w(x_i)-y_i\right)^2 \overset{?}{=} (Xw-Y)^T(Xw-Y)$$

$$Xw-Y=\begin{bmatrix} x_1^{(1)} & x_1^{(2)} & 1 \\ x_2^{(1)} & x_2^{(2)} & 1 \\ & \cdots & \\ x_N^{(1)} & x_N^{(2)} & 1 \end{bmatrix}_{N\times 3}\begin{bmatrix} w_1 \\ w_2 \\ b \end{bmatrix}_{3\times 1}-\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_N \end{bmatrix}=\begin{bmatrix} w_1x_1^{(1)}+w_2x_1^{(2)}+b \\ w_1x_2^{(1)}+w_2x_2^{(2)}+b \\ \cdots \\ w_1x_N^{(1)}+w_2x_N^{(2)}+b \end{bmatrix}_{N\times 1}-\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_N \end{bmatrix}=\begin{bmatrix} \hat{y}_1-y_1 \\ \hat{y}_2-y_2 \\ \cdots \\ \hat{y}_N-y_N \end{bmatrix}_{N\times 1}$$

$$(Xw-Y)^T(Xw-Y)=[\hat{y}_1-y_1 \quad \hat{y}_2-y_2 \quad \cdots \quad \hat{y}_N-y_N]_{1\times N}\begin{bmatrix} \hat{y}_1-y_1 \\ \hat{y}_2-y_2 \\ \cdots \\ \hat{y}_N-y_N \end{bmatrix}_{N\times 1}$$

$$=(\hat{y}_1-y_1)^2+(\hat{y}_2-y_2)^2+\cdots+(\hat{y}_N-y_N)^2$$

附：w的求解过程(正规方程法)

$$w^T X^T Y \overset{\text{对w求导?}}{\Rightarrow} X^T Y$$

$$[w_1 \quad w_2]\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} A \\ B \end{bmatrix}$$

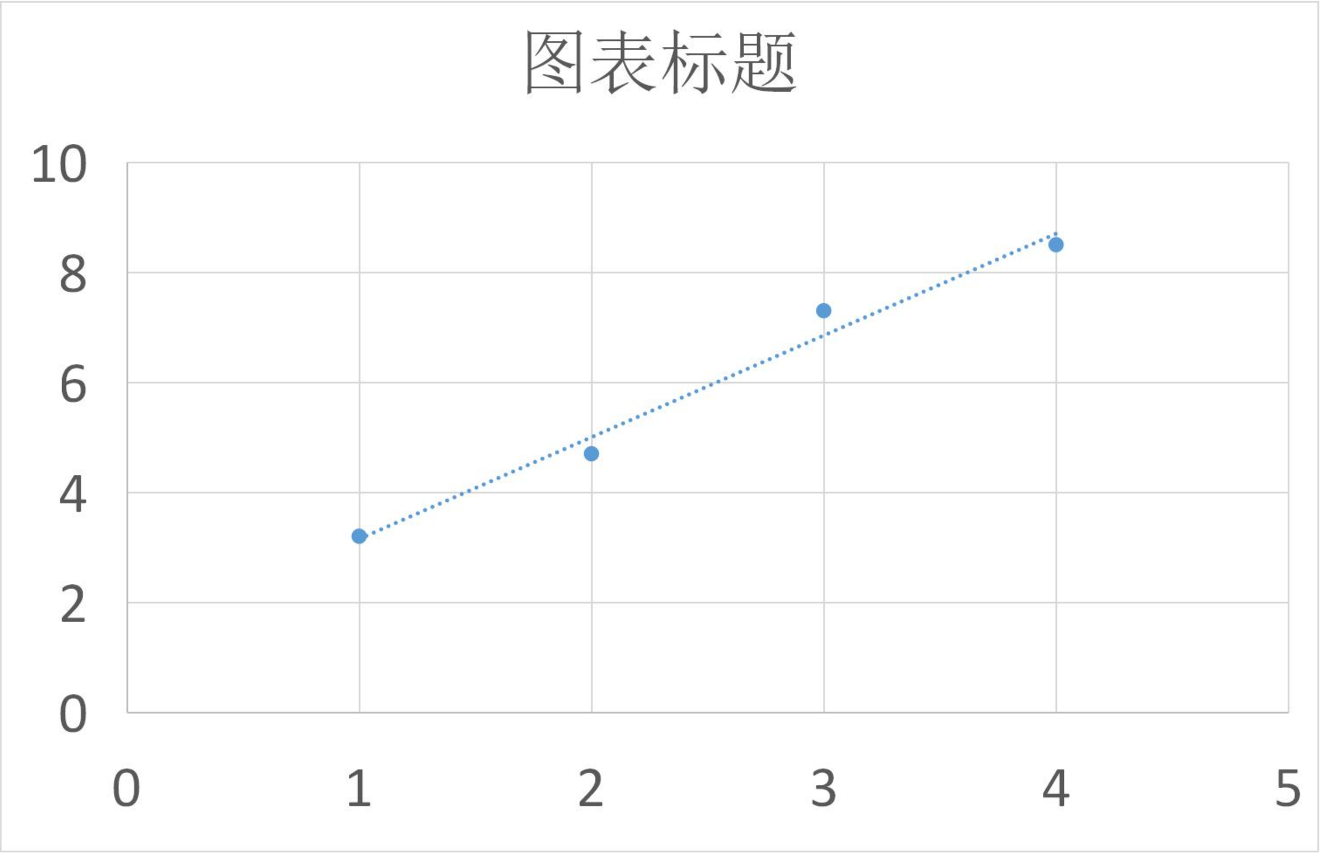
$$= [w_1 a + w_2 c \quad w_1 b + w_2 d]\begin{bmatrix} A \\ B \end{bmatrix}$$

$$= (w_1 a + w_2 c)A + (w_1 b + w_2 d)B(\text{标量})$$

$$\overset{\text{对w求导}}{\Rightarrow} \begin{bmatrix} aA + bB \\ cA + dB \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} A \\ B \end{bmatrix}$$

编程——正规化方程求解线性回归模型

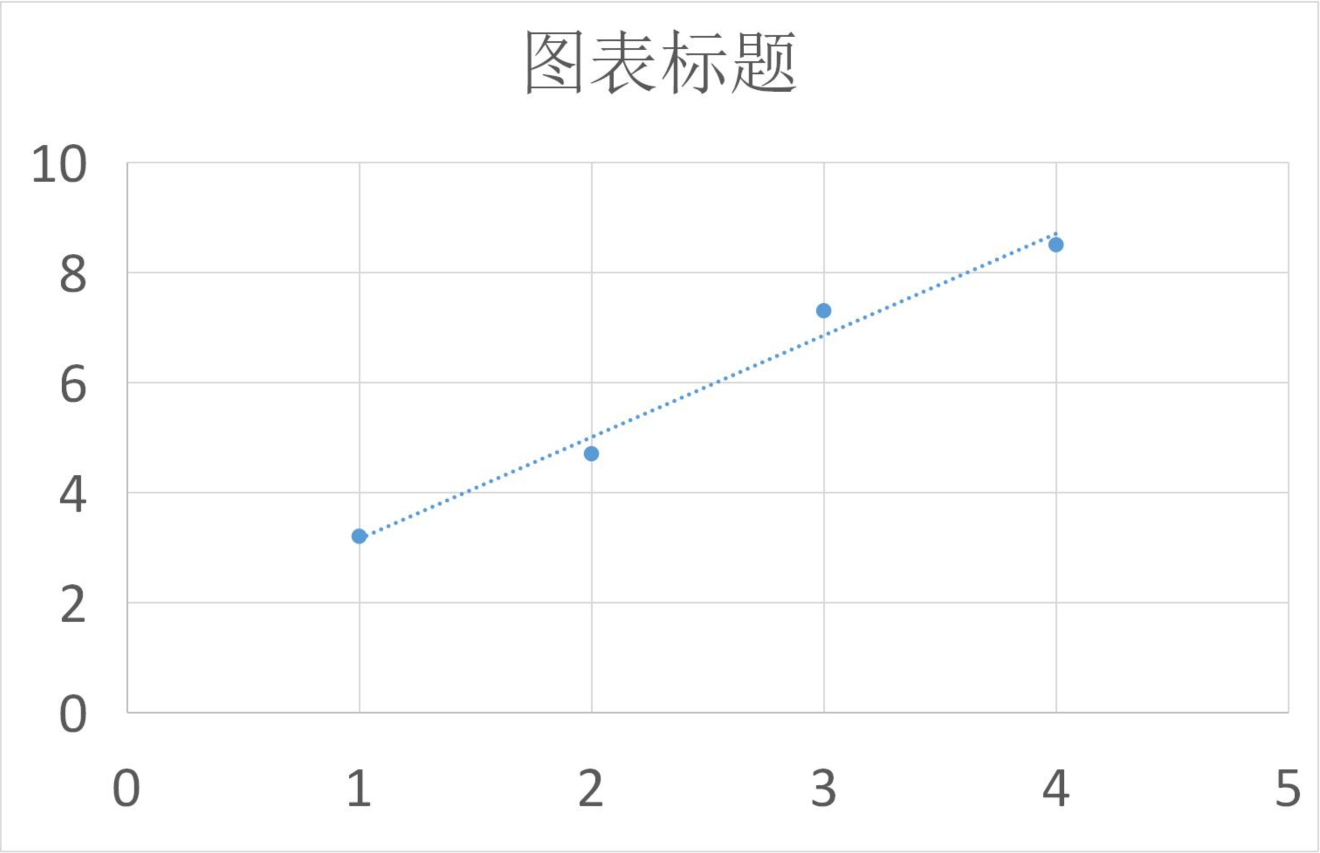
	y=2x+1
x	y
1	3. 2
2	4. 7
3	7. 3
4	8. 5



matrix([[1.85],
[1.3]])

调库——正规化方程求解线性回归模型

	y=2x+1
x	y
1	3. 2
2	4. 7
3	7. 3
4	8. 5



matrix([[1.85],
[1.3]])

编程——正规化方程求解家庭用电案例

北风网
IBEIFENG.COM

- 现有一批描述家庭用电情况的数据，请使用Global_active_power和Global_reactive_power对Global_intensity进行预测

- 数据来源: https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption

Individual household electric power consumption Data Set

Download: Data Folder, Data Set Description

Abstract: Measurements of electric power consumption in one household with a one-minute sampling rate over a period of almost 4 years. Different electrical quantities and some sub-metering values are available.

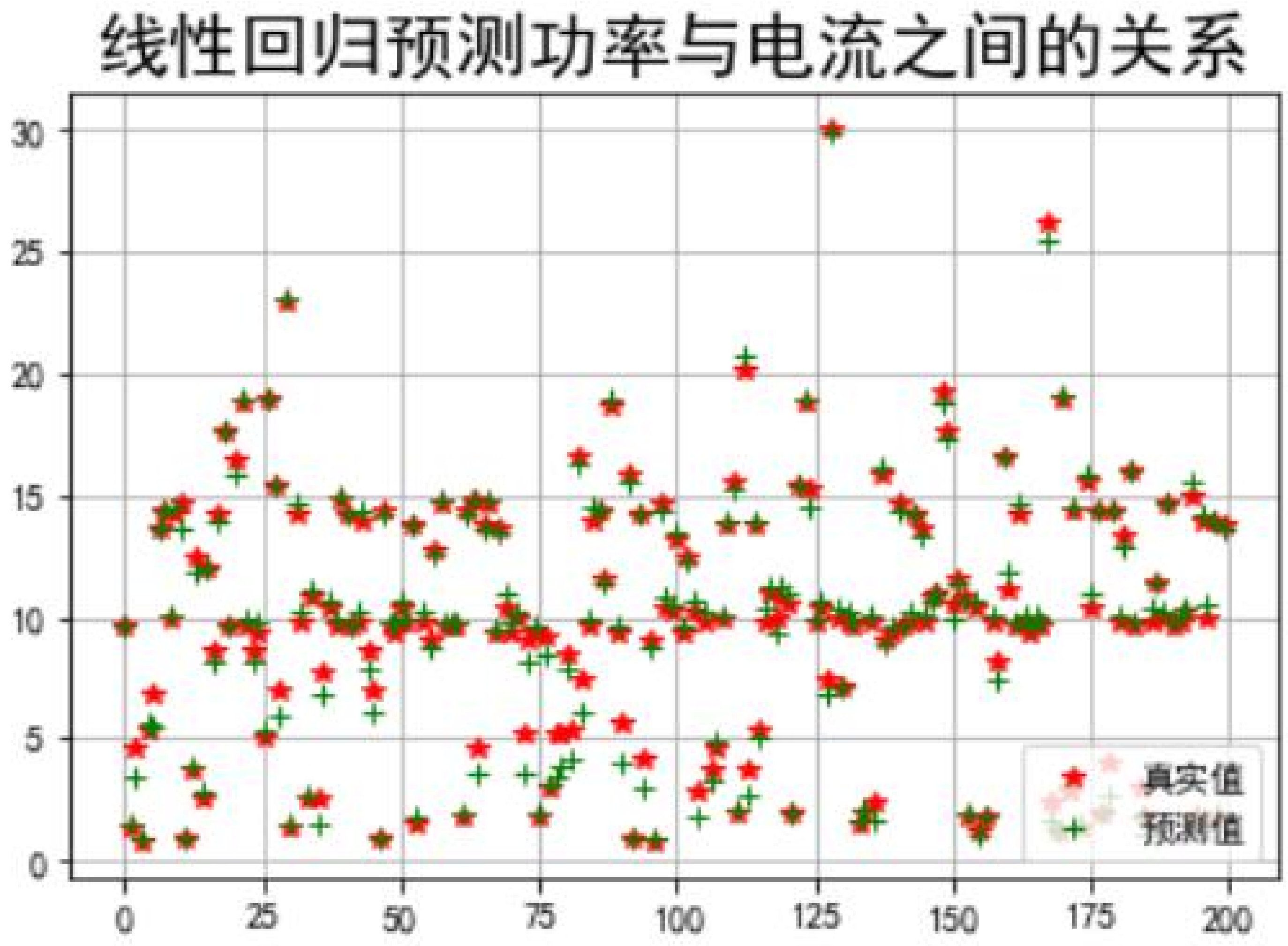
Data Set Characteristics:	Multivariate, Time-Series	**Number of Instances:**	2075259	**Area:**	Physical
Attribute Characteristics:	Real	**Number of Attributes:**	9	**Date Donated**	2012-08-30
Associated Tasks:	Regression, Clustering	**Missing Values?**	Yes	**Number of Web Hits:**	135342

Attribute Information:

1.date: Date in format dd/mm/yyyy
2.time: time in format hh:mm:ss
3.global_active_power: household global minute-averaged active power (in kilowatt)
4.global_reactive_power: household global minute-averaged reactive power (in kilowatt)
5.voltage: minute-averaged voltage (in volt)
6.global_intensity: household global minute-averaged current intensity (in ampere)
7.sub_metering_1: energy sub-metering No. 1 (in watt-hour of active energy). It corresponds to the kitchen, containing mainly a dishwasher, an oven and a microwave (hot plates are not electric but gas powered).
8.sub_metering_2: energy sub-metering No. 2 (in watt-hour of active energy). It corresponds to the laundry room, containing a washing-machine, a tumble-drier, a refrigerator and a light.
9.sub_metering_3: energy sub-metering No. 3 (in watt-hour of active energy). It corresponds to an electric water-heater and an air-conditioner.

```
matrix([[ 4.10551185],
        [ 0.68820978],
        [ 0.35884791]])
```

正规化方程求解家庭用电案例



特征扩展

$$(x^{(1)},x^{(2)},x^{(3)})$$

$$\Downarrow$$

$$(x^{(1)},x^{(2)},x^{(3)},x^{(1)}x^{(1)},x^{(1)}x^{(2)},x^{(1)}x^{(3)},x^{(2)}x^{(2)},x^{(2)}x^{(3)},x^{(3)}x^{(3)},\cdots)$$

电流	功率
1	7
2	36
3	101
4	80
5	238
6	380

$$h(w)=\mathrm{w}_1x^{(1)}+\mathrm{w}_2$$

电流	电流^2	功率
1	1	7
2	4	36
3	9	101
4	16	80
5	25	238
6	36	380

$$h(w)=\mathrm{w}_1x^{(1)}+\mathrm{w}_2x^{(2)}+\mathrm{w}_3$$

电流	电流^2	...	电流^n	功率
1	1	...	XX	7
2	4	...	XX	36
3	9	...	XX	101
4	16	...	XX	80
5	25	...	XX	238
6	36	...	XX	380

$$h(w)=\mathrm{w}_1x^{(1)}+\cdots+\mathrm{w}_nx^{(n)}+w_{n+1}$$

PS：特征扩展*

- 参考薛毅P321牙膏厂的例子

例 6.9 某大型牙膏制造企业为了更好地拓展产品市场，有效地管理库存，公司董事会要求销售部门根据市场调查，找出公司生产的牙膏销售量与销售价格、广告投入等之间的关系，从而预测出在不同价格和广告费用下销售量．为此，销售部研的研究人员收集了过去 *30* 个销售周期 (每个销售周期为 *4* 周) 公司生产的牙膏的销售量、销售价格、投入的广告费用，以及周期其他厂家生产同类牙膏的市场平均销售价格，如表 *6.4* 所示．试根据这些数据建立一个数学模型，分析牙膏销售量与其他因素的关系，为制订价格策略和广告投入策略提供数量依据．

记牙膏销售量为 Y，价格差为 X_1，公司的广告费为 X_2，假设基本模型为线性模型

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon.$$

模型通过 t 检验和 F 检验，并且 $\hat{\sigma}$ 减少， R^2 增加．因此，最终模型选为

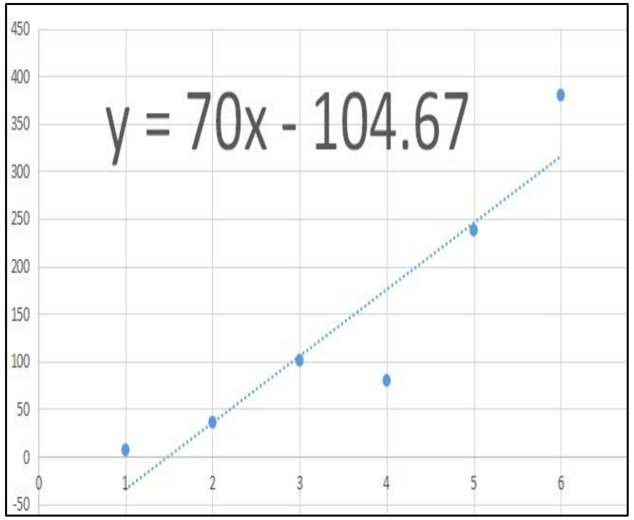
$$Y = 29.1133 + 11.1342X_1 - 7.6080X_2 + 0.6712X_2^2 - 1.4777X_1X_2 + \varepsilon.$$

过拟合现象

功率=电流²·电阻（P=I²·R）

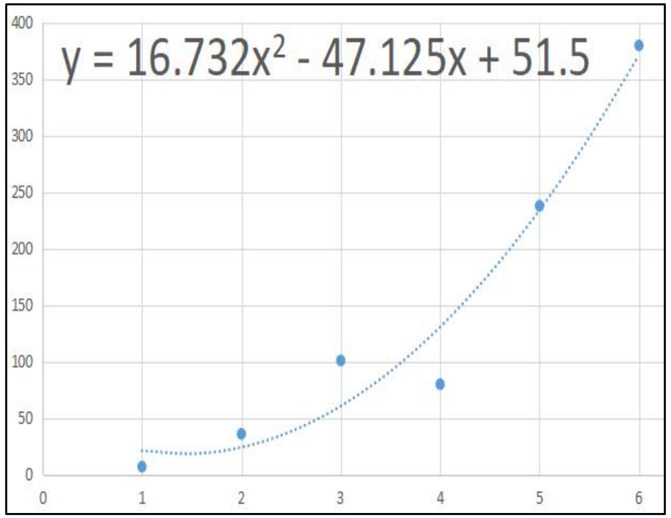
电流	功率
1	7
2	36
3	101
4	80
5	238
6	380

$$h(w)=\mathrm{w}_1x^{(1)}+\mathrm{w}_2$$



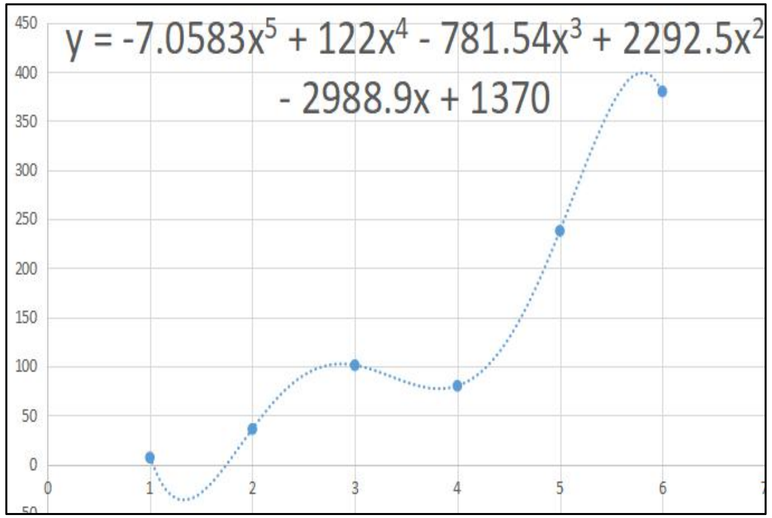
电流	电流²	功率
1	1	7
2	4	36
3	9	101
4	16	80
5	25	238
6	36	380

$$h(w)=\mathrm{w}_1x^{(1)}+\mathrm{w}_2x^{(2)}+\mathrm{w}_3$$



电流	电流²	...	电流ⁿ	功率
1	1	...	XX	7
2	4	...	XX	36
3	9	...	XX	101
4	16	...	XX	80
5	25	...	XX	238
6	36	...	XX	380

$$h(w)=\mathrm{w}_1x^{(1)}+\cdots+\mathrm{w}_nx^{(n)}+w_{n+1}$$



正则化项（Regularization）

$$\underbrace{\underbrace{Obj(w)=\frac{1}{2N}\sum_{i=1}^{N}(w\cdot x-y_i)^2}_{\text{经验风险最小化}}+\lambda R(w)}_{\text{结构风险最小化}}$$

- 经验风险最小化可以理解为：最小化代价函数
- 结构风险最小化可以理解为：最小化目标函数（代价函数+正则化项）

正则化方法

$$Obj(w) = \frac{1}{2N}\sum_{i=1}^{N}(w\cdot x - y_i)^2 + \lambda R(w)$$

- L1正则化
  - 权值向量w中各个元素的绝对值之和:

$$R(w) = \|w\|_1 = |w_1| + |w_2|$$

- L2正则化
  - 权值向量w中各个元素的平方和:

$$R(w) = \frac{1}{2}\|w\|_2^2 = \frac{1}{2}(w_1^2 + w_2^2)$$

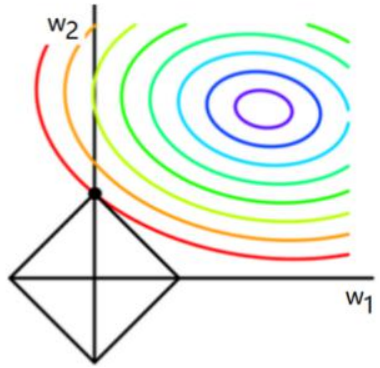
- L1正则化 VS L2正则化
  - L1正则化可以产生稀疏权值矩阵，即产生一个稀疏模型，可以用于特征选择
  - L2正则化可以防止模型过拟合（overfitting）
- 经典面试题
  - 为什么 L1 正则可以产生稀疏模型（很多参数=0），而 L2 正则不会出现很多参数为0的情况?

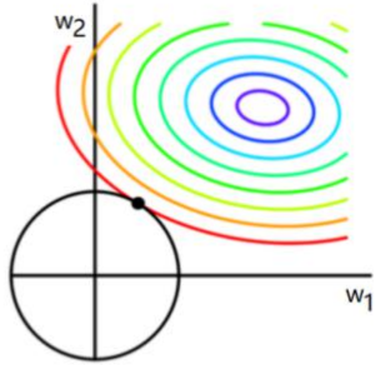
经典面试题

- 加入正则化的目标函数是 $Obj(w) = \frac{1}{2N}\sum_{i=1}^{N}(w\cdot x - y_i)^2 + \lambda R(w)$
- 要让Obj(w)最小，反解出w*，即 $\min_w Obj(w) = \min_w \frac{1}{2N}\sum_{i=1}^{N}(w\cdot x - y_i)^2 + \lambda R(w)$
- An equivalent way to write the problem is：

$$\min_w \frac{1}{2N}\sum_{i=1}^{N}(w\cdot x - y_i)^2$$

$$s.t. \quad R(w) \le t$$



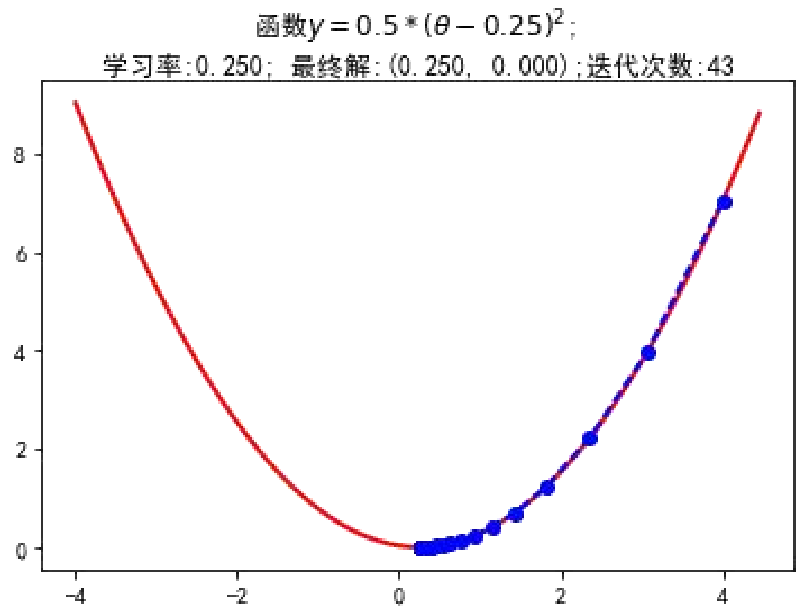


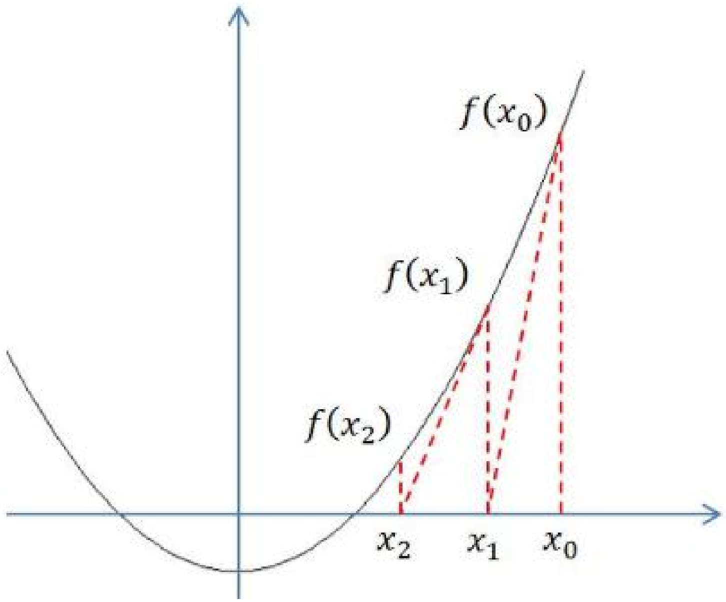
- There is a one-to-one correspondence between the parameters λ and t
- 这就把 w 的解限制在黑色区域内，同时使得经验风险尽可能小，因此取交点就是最优解，从图可以看出，因为L1正则黑色区域是有棱角的，所以更容易在棱角取得交点，从而导致出现参数为0的情况

最优化方法

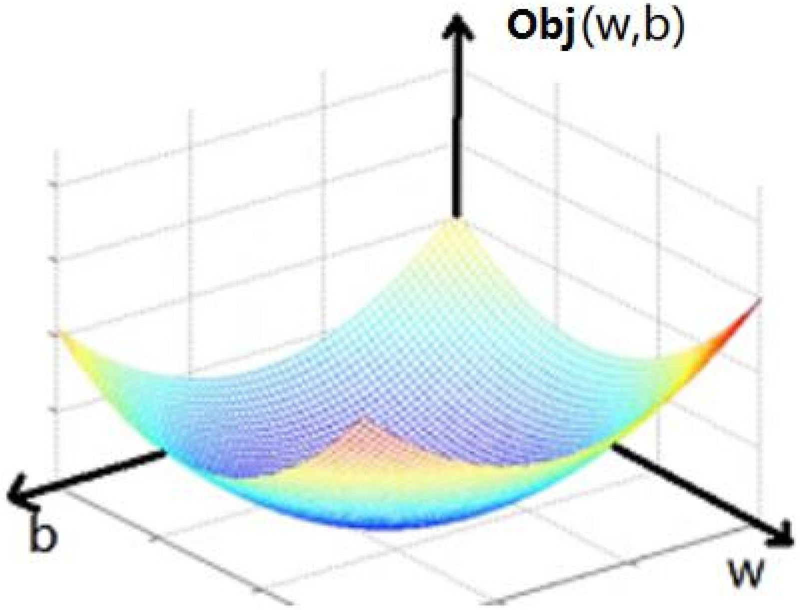
- 前面提到，最小二乘法是基于均方误差，使用最优化方法的一种数学优化技术，这个最优化方法可以具体为以下几种：
- 正规方程法、梯度下降法、牛顿法、拟牛顿法等等

$$\begin{cases} \dfrac{\partial Obj(w)}{\partial w_0} \overset{令}{=} 0 \\ \dfrac{\partial Obj(w)}{\partial w_1} \overset{令}{=} 0 \end{cases}$$





正规方程法



$$Obj(w)=\frac{1}{8}\sum_{i=1}^{4}(w_0+w_1x_i-y_i)^2+1\times\frac{1}{2}\times(w_0^2+w_1^2)$$

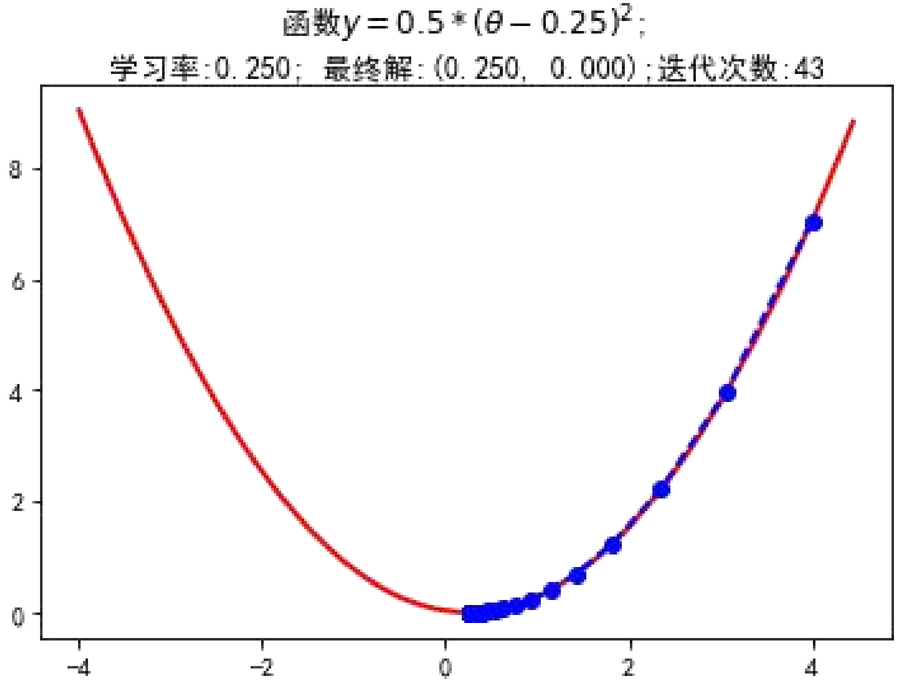
$$\begin{cases}\dfrac{\partial Obj(w)}{\partial w_0}=\dfrac{1}{4}\sum_{i=1}^{4}(w_0+w_1x_i-y_i)+w_0\overset{令}{=}0\\[2ex]\dfrac{\partial Obj(w)}{\partial w_1}=\dfrac{1}{4}\sum_{i=1}^{4}(w_0+w_1x_i-y_i)x_i+w_1\overset{令}{=}0\end{cases}$$

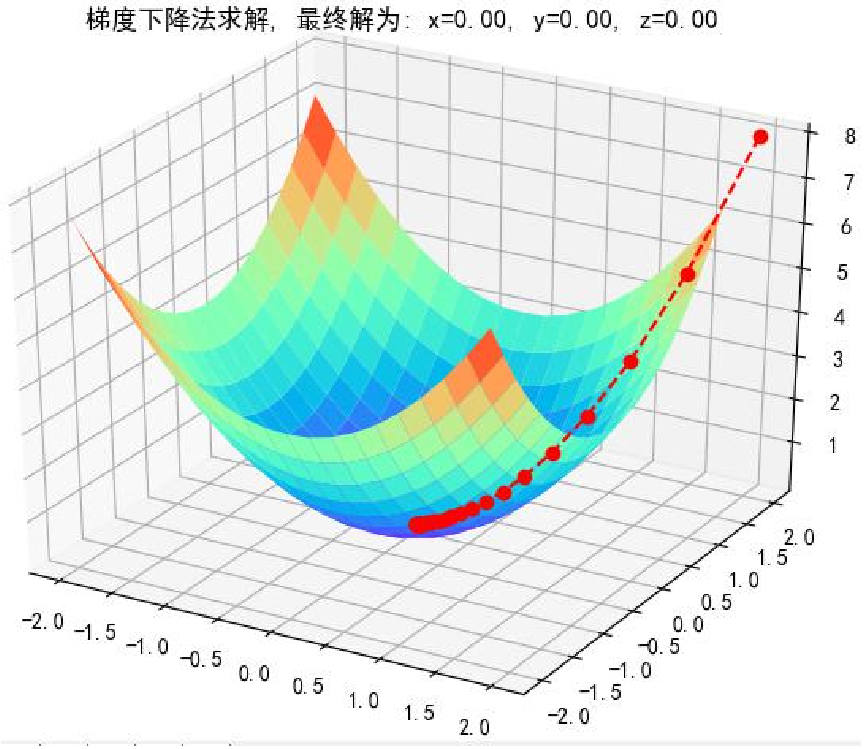
比如解得

$$\begin{cases}w_0=0.9\\w_1=0.96\end{cases}$$

梯度下降法介绍

- 梯度下降法(Gradient Descent， GD)常用于求解**无约束**情况下**凸函数(Convex Function)**的**极小值**，是一种**迭代类型**的算法，因为凸函数只有一个极值点，故求解出来的极小值点就是函数的**最小值点**。



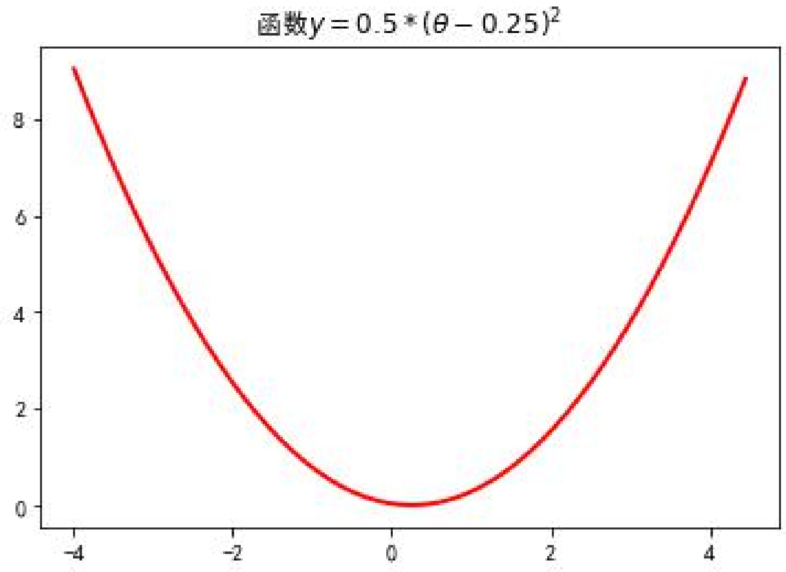


梯度下降法介绍

- 梯度下降法的优化思想是用当前位置负梯度方向作为搜索方向，因为该方向为当前位置的最快下降方向，所以梯度下降法也被称为“最速下降法”。梯度下降法中越接近目标值，变量变化越小。计算公式如下：

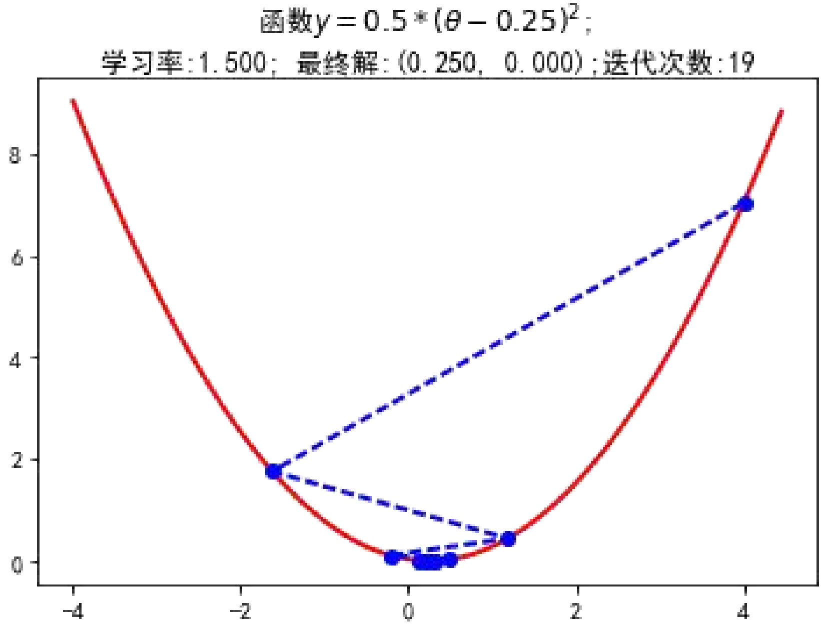
$$\theta^{k+1} = \theta^k - \alpha \nabla f\left(\theta^k\right)$$

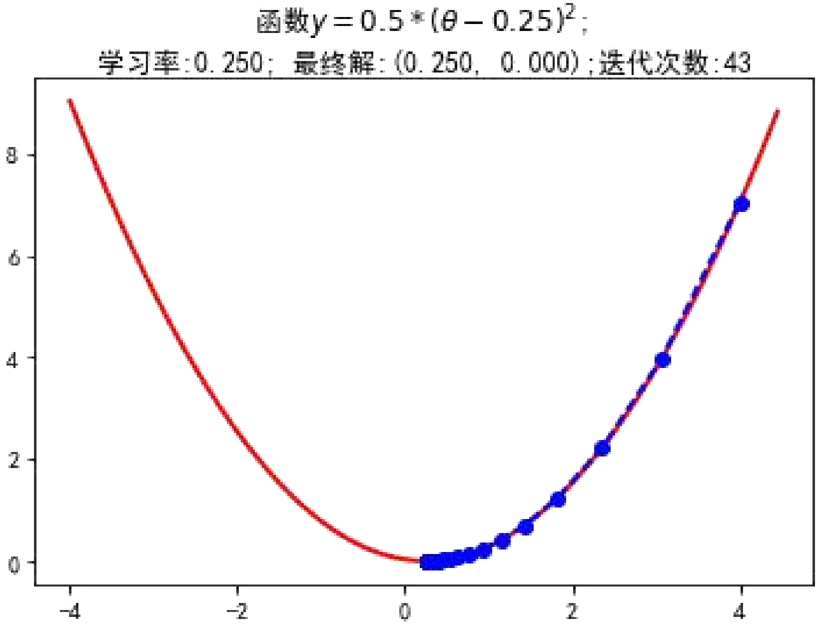
- α被称为**步长**或者**学习率(learning rate)**，表示自变量θ每次迭代变化的大小。
- 收敛条件：当目标函数的函数值变化非常小的时候或者达到最大迭代次数的时候，就结束循环。



编程——梯度下降法

- 用梯度下降法求y=0.5*(x-0.25)^2的最小值





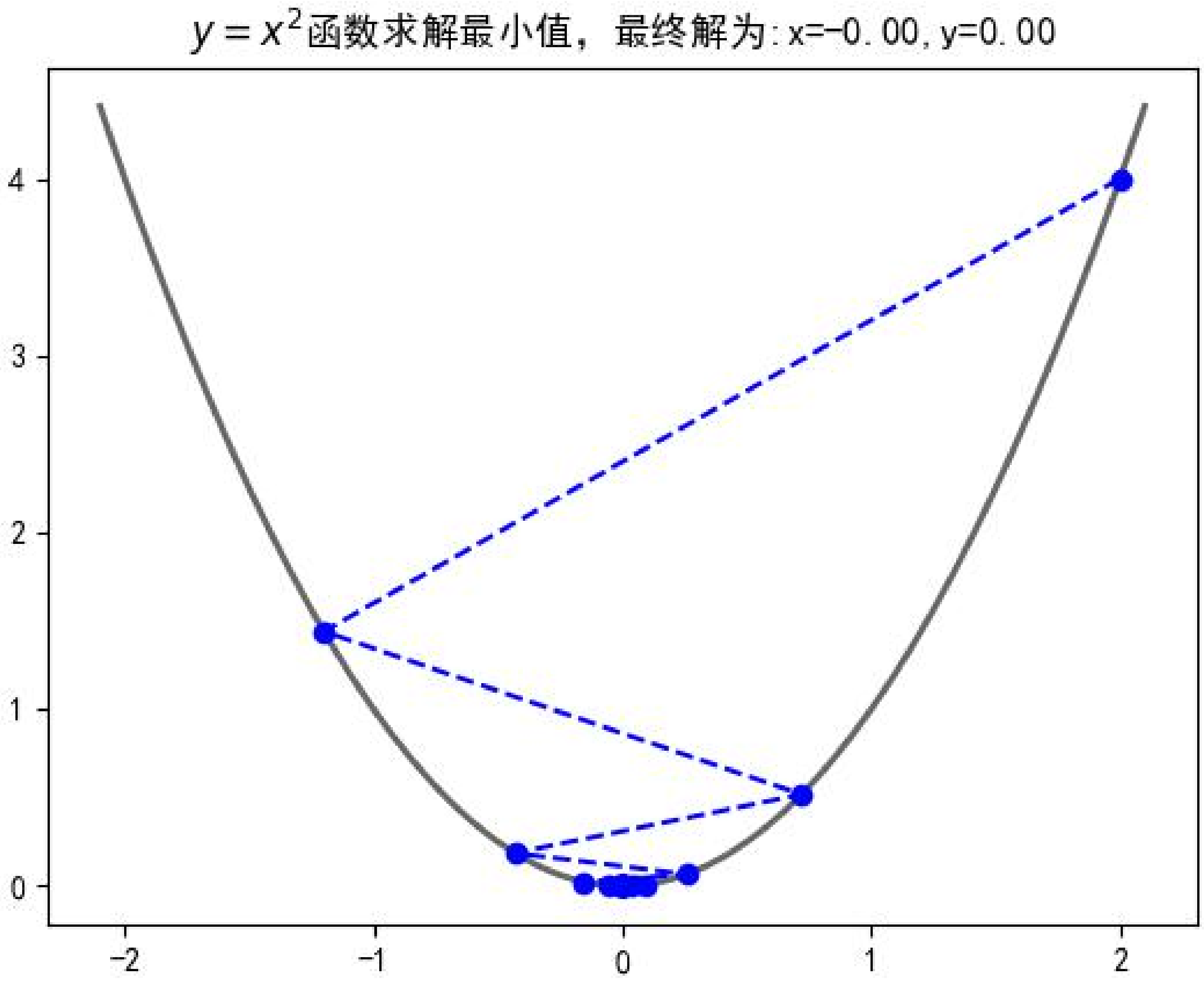

```
-1.625 1.7578125
1.1875 0.439453125
-0.21875 0.10986328125
0.484375 0.0274658203125
0.1328125 0.006866455078125
0.30859375 0.0017166137695312
0.220703125 0.00042915344238281
0.2646484375 0.00010728836059570312
0.24267578125 2.682209014892578e-05
0.253662109375 6.705522537231445e-06
0.2481689453125 1.6763806343078613e-06
```

```
3.0625 3.955078125
2.359375 2.2247314453125
1.83203125 1.2514114379882812
1.4365234375 0.7039189338684082
1.139892578125 0.3959544003009796
0.91741943359375 0.22272435016930103
0.7505645751953125 0.12528244697023183
0.6254234313964844 0.0704713764207554
0.5315675735473633 0.039640149236674915
```

```
0.25891903357825186 3.9774579984992056e-05
0.2566892751836889 2.237320124155803e-05
0.25501695638776667 1.2584925698376392e-05
```

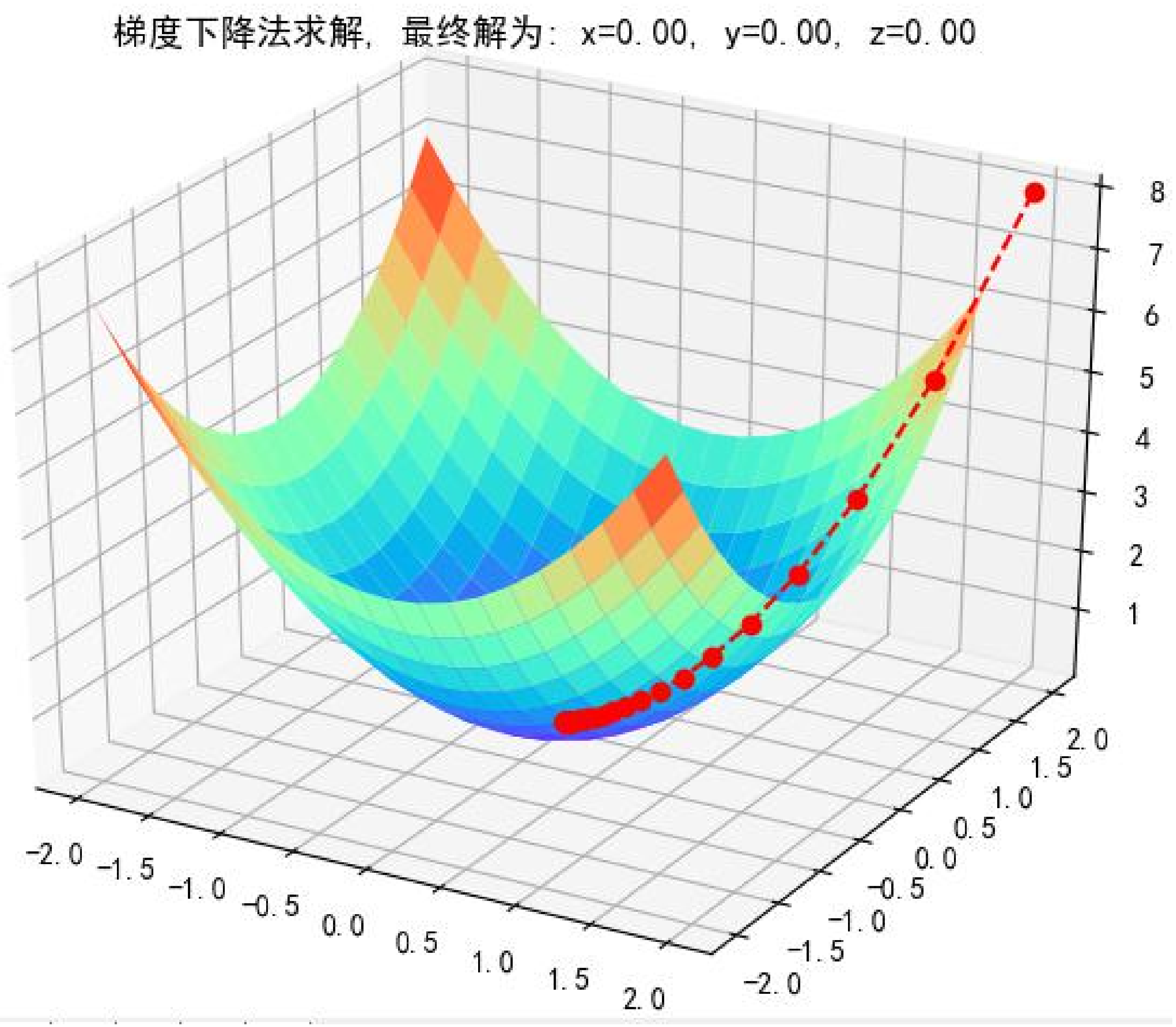
梯度下降案例

$$y=f(x)=x^2$$



梯度下降案例

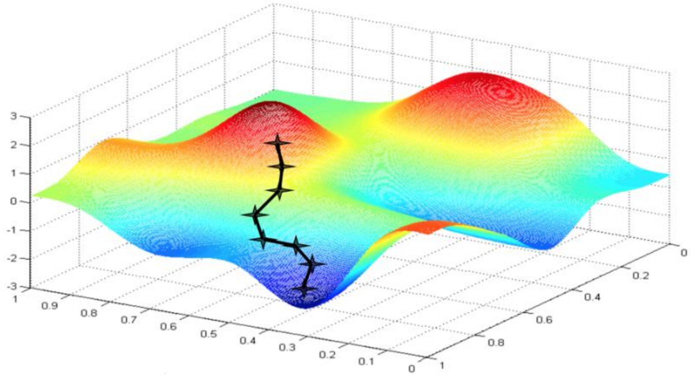
$$z = f(x, y) = x^2 + y^2$$



梯度下降法在线性模型求解中的应用

- 目标函数：

$$Obj(w) = \frac{1}{N}\sum_{i=1}^{N} Loss(w; x_i, y_i) + \lambda R(w)$$



- 目标函数的梯度为：

$$Obj'_w = \frac{1}{N}\sum_{i=1}^{N} \frac{dLoss(w; x_i, y_i)}{dw} + \lambda R'_w$$

- 更新权重：

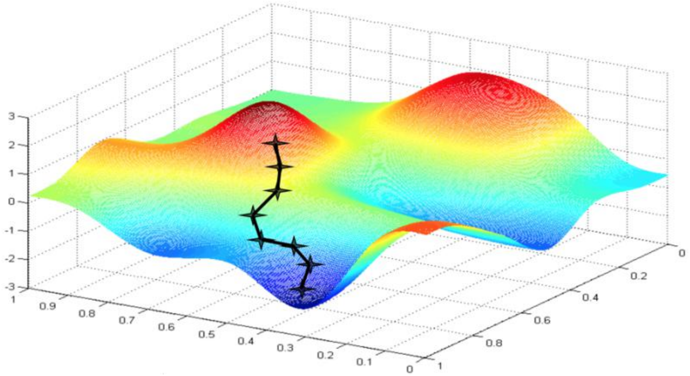
γ是学习率/步长 $w = w - \gamma Obj'_w$

梯度下降法求解Ridge Regression

北风网
IBEIFENG.COM

- 目标函数：

$$Obj(w) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}(w\cdot x_i - y_i)^2 + \lambda\frac{1}{2}\|w\|_2^2$$



- 目标函数的梯度为：

$$Obj'_w = \frac{1}{N}\sum_{i=1}^{N}(w\cdot x_i - y_i)\cdot x_i + \lambda w$$

- 更新权重：

γ是学习率/步长 $w = w - \gamma Obj'_w$

附：梯度下降法求解L2正则化的线性回归模型

$$\frac{1}{2}(w\cdot x_i - y_i)^2 \xRightarrow{\text{对w求导?}} (w\cdot x_i - y_i)\cdot x_i$$

$$\frac{1}{2}\left(\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}\cdot\begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ x_i^{(3)} \end{bmatrix} - y_i\right)^2 \xRightarrow{\text{对w求导}} \left(\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}\cdot\begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ x_i^{(3)} \end{bmatrix} - y_i\right) * \frac{d(w\cdot x_i)}{dw}$$

已知$w\cdot x_i = w_1 x_i^{(1)} + w_2 x_i^{(2)} + w_3 x_i^{(3)}$(标量)

$$\frac{d(w\cdot x_i)}{dw} = \begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ x_i^{(3)} \end{bmatrix} = x_i$$

Loss Function and Regularization summary

	loss function $L(\mathbf{w}; \mathbf{x}, y)$	gradient or sub-gradient
hinge loss	$\max\{0, 1 - y\mathbf{w}^T\mathbf{x}\}, \quad y \in \{-1, +1\}$	$\begin{cases} -y \cdot \mathbf{x} & \text{if } y\mathbf{w}^T\mathbf{x} < 1, \\ 0 & \text{otherwise.} \end{cases}$
logistic loss	$\log(1 + \exp(-y\mathbf{w}^T\mathbf{x})), \quad y \in \{-1, +1\}$	$-y\left(1 - \frac{1}{1+\exp(-y\mathbf{w}^T\mathbf{x})}\right) \cdot \mathbf{x}$
squared loss	$\frac{1}{2}(\mathbf{w}^T\mathbf{x} - y)^2, \quad y \in \mathbb{R}$	$(\mathbf{w}^T\mathbf{x} - y) \cdot \mathbf{x}$

	regularizer $R(\mathbf{w})$	gradient or sub-gradient
zero (unregularized)	0	$\mathbf{0}$
L2	$\frac{1}{2}\|\mathbf{w}\|_2^2$	$\mathbf{w}$
L1	$\|\mathbf{w}\|_1$	$\mathrm{sign}(\mathbf{w})$
elastic net	$\alpha\|\mathbf{w}\|_1 + (1-\alpha)\frac{1}{2}\|\mathbf{w}\|_2^2$	$\alpha\mathrm{sign}(\mathbf{w}) + (1-\alpha)\mathbf{w}$

PS：Loss Function and Regularization summary

$$\frac{1}{2}\|w\|_2^2 = \frac{1}{2}(w_1^2 + w_2^2 + \cdots + w_n^2) \qquad \|w\|_1 = |w_1| + |w_2| + \cdots + |w_n|$$

$$\overset{\text{对}w\text{求导}}{\Longrightarrow} \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_n \end{bmatrix} \qquad \overset{\text{对}w\text{求导}}{\Longrightarrow} \begin{bmatrix} sign(w_1) \\ sign(w_2) \\ \cdots \\ sign(w_n) \end{bmatrix} = sign \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_n \end{bmatrix}$$

梯度下降法 VS 正规方程法

梯度下降	正规方程
需要选择学习率 α	不需要
需要多次迭代	一次运算得出
当特征数量 n 大时也能较好适用	需要计算$(X^TX)^{-1}$ 如果特征数量 n 较大则运算代价大，因为矩阵逆的计算时间复杂度为 $O(n^3)$，通常来说当 n 小于 10000 时还是可以接受的
适用于各种类型的模型	只适用于线性回归，不适合逻辑回归模型等其他模型

梯度下降法大家族

- 批量梯度下降法（Batch Gradient Descent）
  - 有N个样本，求梯度的时候就用了N个样本的梯度数据
  - 优点：准确　　　缺点：速度慢

$$Obj'_w = \frac{1}{N}\sum_{i=1}^{N} Loss'_{w,i} + \lambda R'_w$$

- 随机梯度下降法（Stochastic Gradient Descent）

$$w = w - \gamma Obj'_w$$

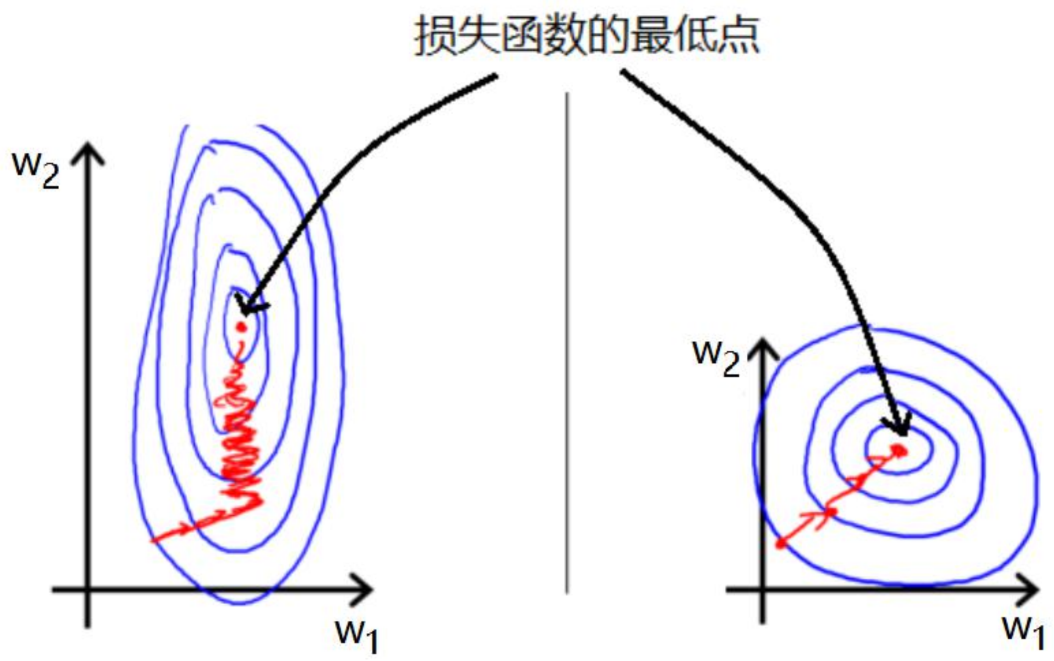
  - 和批量梯度下降法原理类似，区别在于求梯度时没有用所有的N个样本的数据，而是仅仅选取一个样本来求梯度
  - 优点：速度快　　缺点：准确度低
- 小批量梯度下降法（Mini-batch Gradient Descent）
  - 批量梯度下降法和随机梯度下降法的折衷，比如N=100
  - spark中使用的此方法

线性回归大家族

- 线性最小二乘回归
  - 没有使用正则化
- LASSO回归
  - 使用L1正则化
- 岭回归
  - 使用L2正则化

特征缩放

房屋面积（x_1）	卧室数量（x_2）	售价（y）
2104	3	399900
1600	3	329900
2400	3	369000
1416	2	232000
3000	4	539900
1985	4	299900



- 直接求解缺点：
  - x_1特征对应权重会比x_2对应的权重小很多，降低模型可解释性
  - 梯度下降时，最终解被数值大的特征所主导，会影响模型精度与收敛速度
  - 正则化时会不平等看待特征的重要程度（尚未标准化就进行L1/L2正则化是错误的）
- 特征缩放：
  - 归一化或标准化

特征归一化 VS 特征标准化

	归一化	标准化
定义	把数据映射到 [0，1] 区间内	保持特征的原始分布，对特征进行缩放
公式	$f(x)=\frac{x-min}{max-min}$	$f(x)=\frac{x-\mu}{\delta}$ 其中，μ是均值，δ是标准差
相同点	对特征进行缩放，提升求解速度，提升模型精度	
不同点	输出范围在 [0，1] 之间	数据可正可负，但是一般绝对值不会太大，保持原始数据的分布

举例

归一化：

房屋面积	卧室数量	售价
0.434	0.5	399900
0.116	0.5	329900
0.621	0.5	369000
0	0	232000
1	1	539900
0.359	1	299900

标准化：

房屋面积	卧室数量	售价
0.038	-0.242	399900
-0.929	-0.242	329900
0.606	-0.242	369000
-1.282	-1.697	232000
1.757	1.212	539900
-0.190	1.212	299900

房屋面积	卧室数量	售价
2104	3	399900
1600	3	329900
2400	3	369000
1416	2	232000
3000	4	539900
1985	4	299900

注意：当对新数据进行特征缩放时，应使用训练集的缩放规则

离散特征的处理

- 存在“序”关系：
  - 可通过连续化将其转化为连续值
- 不存在“序”关系
  - 如果简单地把类别型变量当作数值型对待，将其映射到不同的数字，这种做法是错误的，原因在于算法会试图从一个没有意义的大小顺序中学习

身高
高
矮
高
矮
高
矮

身高
1
0
1
0
1
0

强度
强
中
弱
强
中
弱

强度
1
0.5
0
1
0.5
0

天气
雨
风
晴
雨
风
晴

雨	风	晴
1	0	0
0	1	0
0	0	1
1	0	0
0	1	0
0	0	1

梳理一下

一元线性回归
最小二乘法（误差度量+最优化方法）
损失函数、代价函数、目标函数
多元线性回归
求解：正规方程法
特征工程：特征扩展
过拟合现象
正则化
求解：梯度下降法
梯度下降法的广义化
梯度下降法 VS 正规方程法
梯度下降法大家族
线性回归大家族
特征工程：特征缩放
特征工程：离散特征的处理
模型评估：评估方法和度量指标
线性模型的优缺点
特征工程API介绍
综合实战

模型评估

- 评估方法（如何切分数据）：
  - 留出法、交叉验证法、留一法、自助法等。
- 度量指标（如何衡量误差）：
  - 回归
    - 均方误差、均方根误差、绝对平均误差等
  - 分类
    - 准确率、精准率、召回率等

评估方法——留出法和交叉验证法

- 留出法（hold-out）：一部分为训练集，一部分为测试集。
  - 应尽量保证数据分布的一致性。
  - 划分比例：7:3左右

训练集（0.7）	测试集（0.3）

- 交叉验证法（k-fold cross validation）：划分为k个互斥子集，用k-1作为训练集，剩下一个为测试集，最终每一个子集都会作为测试集，其余子集作为训练集，共进行k次建模，最终得到测试结果的均值。
  - K取值一般为10
  - 随机取k个互斥子集，进行p次，最后对p个k-fold cv进行取平均，叫作p次k折交叉验证

评估方法——留一法和自助法

- 留一法（Leave-one-out cross validation）：m个样本，令k=m，作为cv的特例。只有一种划分方法，即每个测试集只有一条数据。
  - 优势，每个模型都能很好的反应原始数据集的特性
  - 劣势，计算量在数据量大时会非常大，还不算调参的计算量
- 自助法（Bootstrapping）：对D中的m个数据随机取样，接着将数据放回原数据集继续取样，重复m次，产生一个新的数据集D’。最后用未取到的数据作为测试集。
  - 未取到的数据占比36.8%

$$\lim_{m \mapsto \infty} \left(1 - \frac{1}{m}\right)^m \mapsto \frac{1}{e} \approx 0.368$$

回归任务的误差度量指标

Metric	Definition
Mean Squared Error (MSE)	$MSE = \frac{\sum_{i=0}^{N-1}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{N}$
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=0}^{N-1}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{N}}$
Mean Absolute Error (MAE)	$MAE = \frac{1}{N}\sum_{i=0}^{N-1}\lvert\mathbf{y}_i - \hat{\mathbf{y}}_i\rvert$
Coefficient of Determination (R^2)	$R^2 = 1 - \frac{MSE}{\mathrm{VAR}(\mathbf{y})\cdot(N-1)} = 1 - \frac{\sum_{i=0}^{N-1}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{\sum_{i=0}^{N-1}(\mathbf{y}_i - \bar{\mathbf{y}})^2}$
Explained Variance	$1 - \frac{\mathrm{VAR}(\mathbf{y} - \hat{\mathbf{y}})}{\mathrm{VAR}(\mathbf{y})}$

分类任务的误差度量指标

Metric	Definition
Precision (Postive Predictive Value)	$PPV = \frac{TP}{TP+FP}$
Recall (True Positive Rate)	$TPR = \frac{TP}{P} = \frac{TP}{TP+FN}$
F-measure	$F(\beta) = (1+\beta^2) \cdot \left(\frac{PPV \cdot TPR}{\beta^2 \cdot PPV + TPR}\right)$
Receiver Operating Characteristic (ROC)	$FPR(T) = \int_T^\infty P_0(T)\, dT$ $TPR(T) = \int_T^\infty P_1(T)\, dT$
Area Under ROC Curve	$AUROC = \int_0^1 \frac{TP}{P} d\left(\frac{FP}{N}\right)$
Area Under Precision-Recall Curve	$AUPRC = \int_0^1 \frac{TP}{TP+FP} d\left(\frac{TP}{P}\right)$

线性模型优缺点

- 优点
  - 模型简单，易于解释，w反应了特征的重要程度
  - 算法容易并行，适合大数据的求解
- 缺点
  - 不能拟合非线性数据，需要做特征扩展
  - 采用何种特征组合需要一定的数据敏感性
  - 特征多了：提高模型效果，但是计算成本高，需要数据量大，否则会过拟合
  - 特征少了：有欠拟合的风险

特征工程API介绍

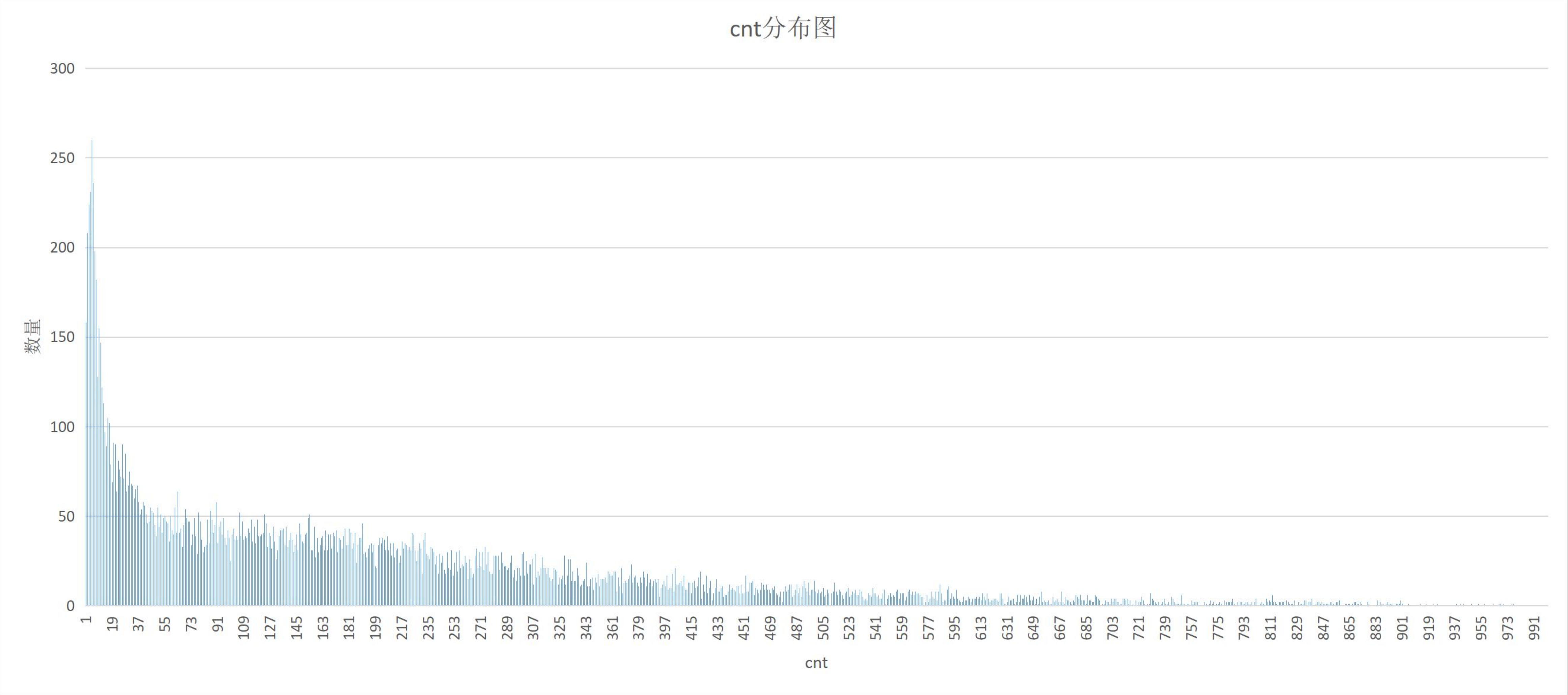
数据收集 → 数据清洗 → 特征工程 → 数据建模

编程——回归算法综合案例之共享单车租赁数量预测

- This dataset contains the hourly and daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system with the corresponding weather and seasonal information.
  - 数据下载 http://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

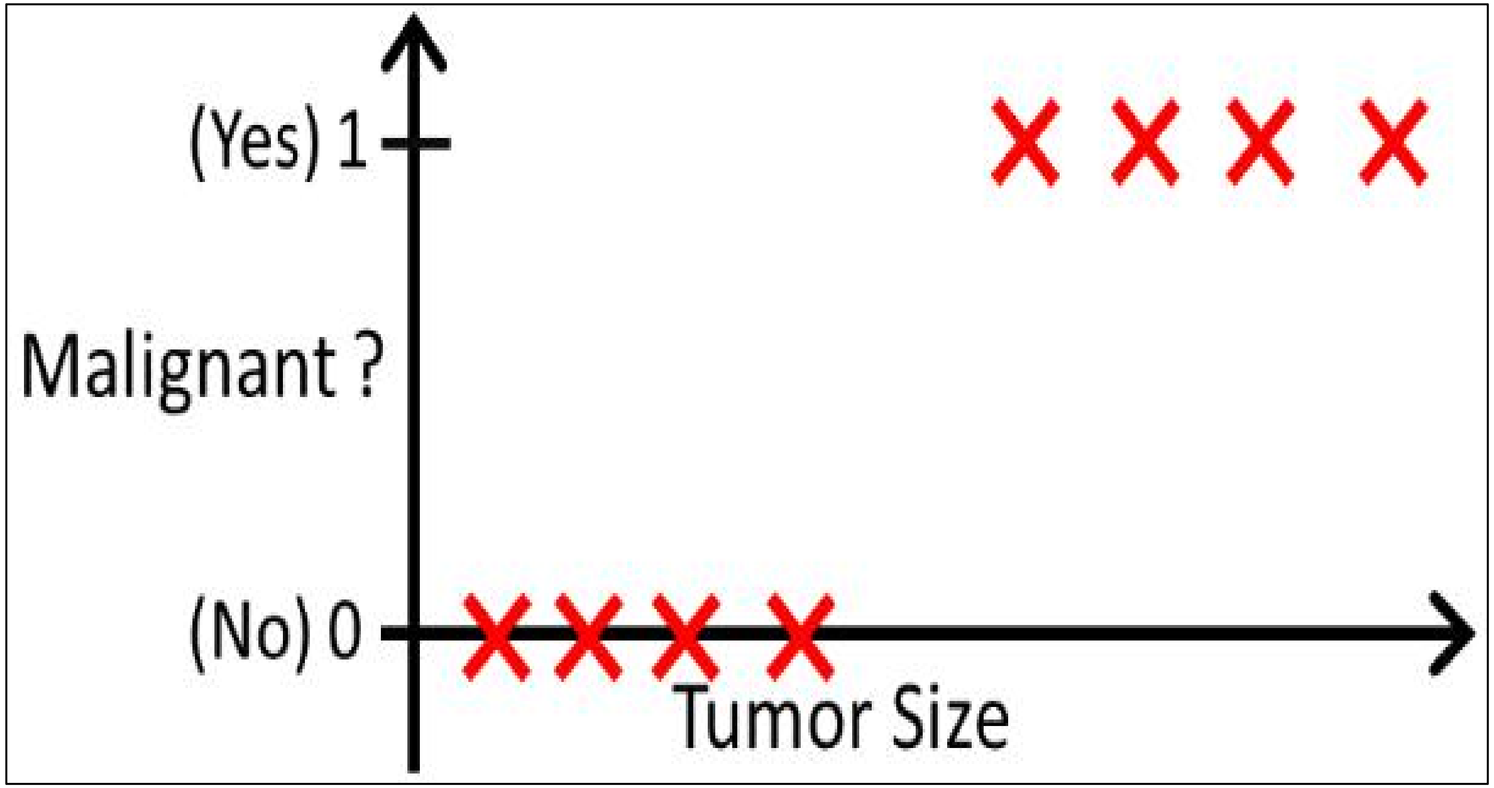
Data Set Characteristics:	Univariate	Number of Instances:	17389	Area:	Social
Attribute Characteristics:	Integer, Real	Number of Attributes:	16	Date Donated	2013-12-20
Associated Tasks:	Regression	Missing Values?	N/A	Number of Web Hits:	232895

编程——回归算法综合案例之共享单车租赁数量预测

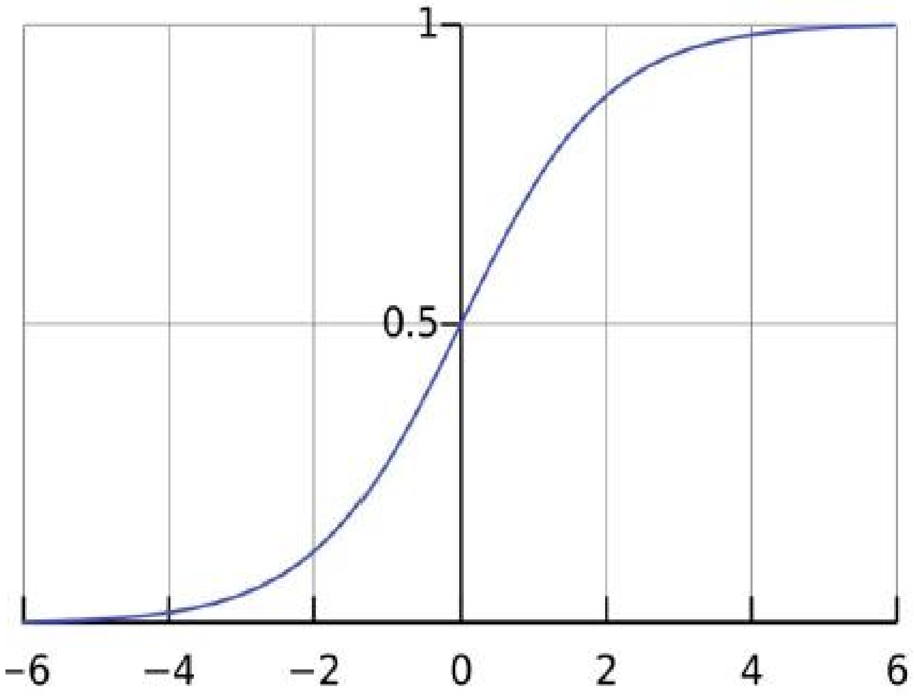


Logistic回归

• 逻辑回归（英语：Logistic regression 或logit regression），即逻辑模型（英语：Logit model，也译作“评定模型”、“分类评定模型”）



sigmoid函数



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

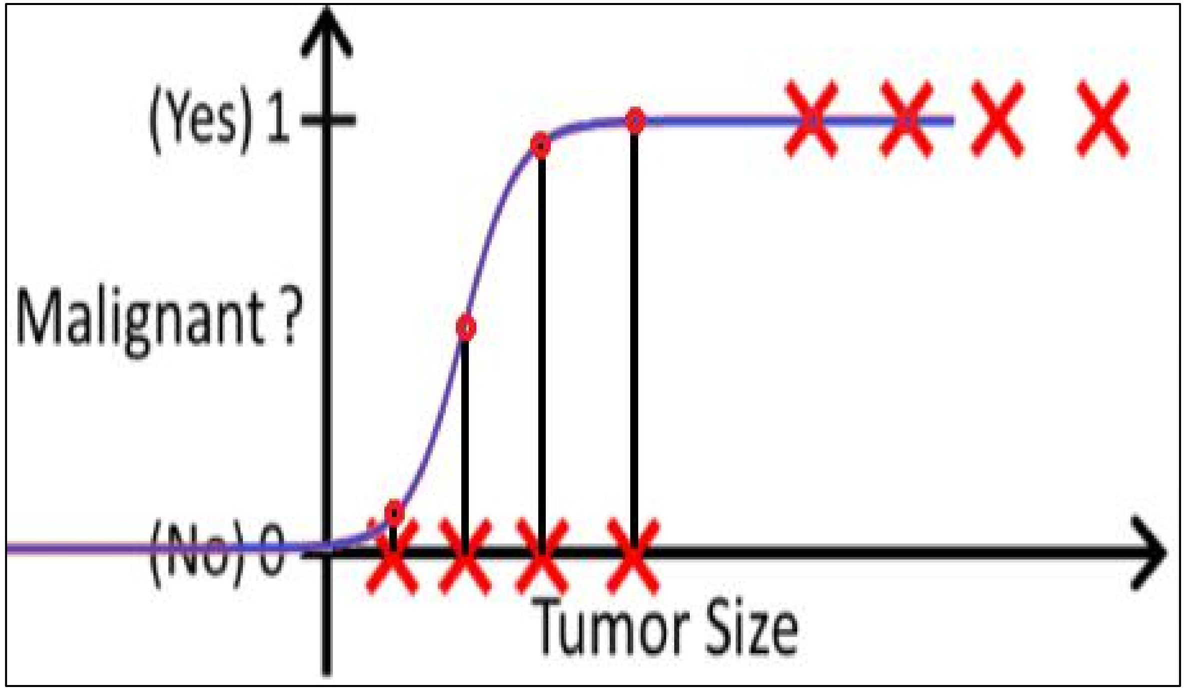
$$g'(z)=\left(\frac{1}{1+e^{-z}}\right)'=\frac{e^{-z}}{\left(1+e^{-z}\right)^{2}}$$

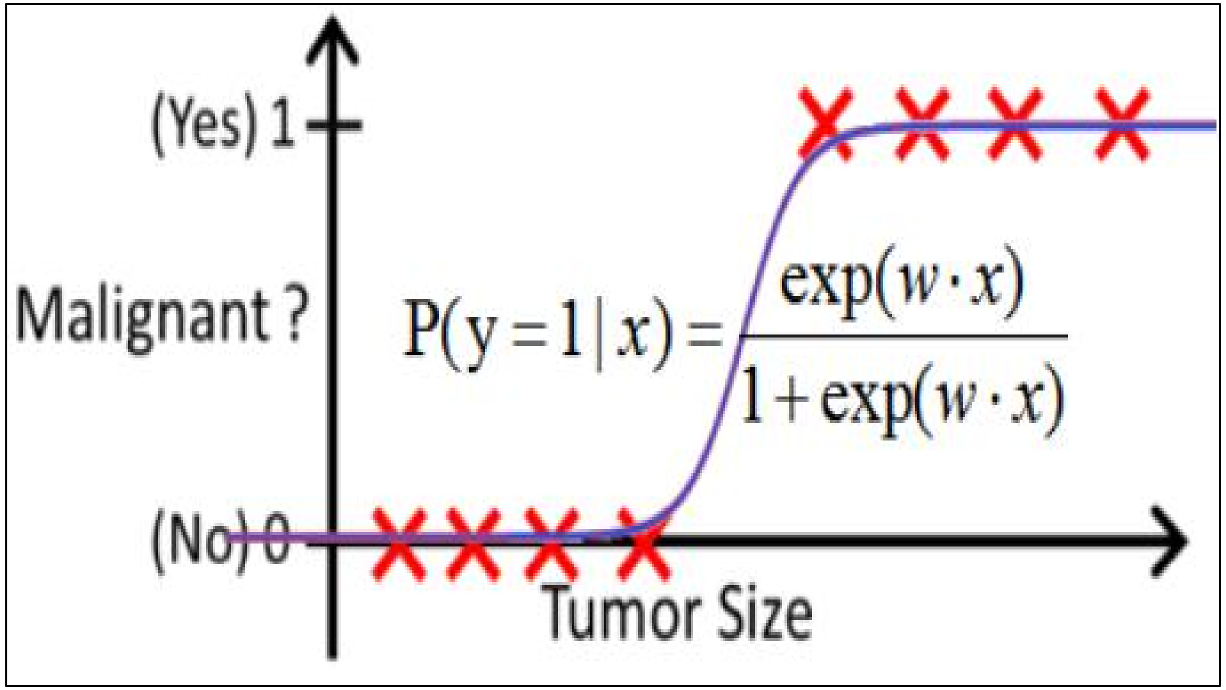
$$=\frac{1}{1+e^{-z}}\cdot\frac{e^{-z}}{1+e^{-z}}=\frac{1}{1+e^{-z}}\cdot\left(1-\frac{1}{1+e^{-z}}\right)$$

$$=g(z)\cdot\left(1-g(z)\right)$$

逻辑回归的求解思路

设1为正例，0为负例





将w·x视为Sigmoid函数的输入，其中w是模型参数，x是特征向量，将Sigmoid函数的输出视为预测为正例的概率。那模型将样本预测为正例的概率为sigmod(w·x)；将样本预测为负例概率为1- sigmod(w·x)

极大化 => 将所有正例预测为正例的概率的累乘 * 将所有负例预测为负例的概率的累乘

逻辑回归代价函数构建思路——极大似然估计

- 极大似然估计：
- 事情已经发生了，当未知参数等于多少时，

果索因。

极大似然估计例题

- 一个暗箱里有三种球（1，2，3），其概率分布律如表所示，进行有放回的抽样，得到了 1 2 2 2 1 2 2 3 1 3，记为$x_1, x_2, \ldots, x_n$，现在想通过极大似然估计的方法，估计θ

X	1	2	3
P{X=?}	0.5θ	0.3-0.4θ	0.7-0.9θ

$$L(\theta)=\prod_{i=1}^{N} P\{X=x_i\}=[P\{X=1\}]^3[P\{X=2\}]^5[P\{X=3\}]^2$$

$$=\prod_{i=1}^{N}[P\{X=1\}]^{\mathrm{I}(x_i=1)}[P\{X=2\}]^{\mathrm{I}(x_i=2)}[P\{X=3\}]^{\mathrm{I}(x_i=3)}$$

$$=(0.5\theta)^3(0.3+0.4\theta)^5(0.7-0.9\theta)^2$$

$$\ln L(\theta)=3\ln 0.5\theta+5\ln(0.3+0.4\theta)+2\ln(0.7-0.9\theta)$$

求导，令$=0$,求出$\theta=0.5598$

使用极大似然估计构建逻辑回归代价函数

- 某样本属于正例的概率可以表示为

$$P(y=1|x)=\frac{\exp(w\cdot x)}{1+\exp(w\cdot x)}$$

- 某样本属于负例的概率可以表示为

$$P(y=0|x)=\frac{1}{1+\exp(w\cdot x)}$$

- 因此，似然函数可以是

$$\prod_{i=1}^{N}[P(y=1|x_i)]^{y_i}[P(y=0|x_i)]^{1-y_i}$$

- 代价函数：

$$Cost(w)=-\frac{1}{N}\sum_{i=1}^{N}[y_i\log P(y=1|x_i)+(1-y_i)\log P(y=0|x_i)]$$

- 使用梯度下降法就能反解出w了

使用极大似然估计构建逻辑回归代价函数另一种写法

- 某样本属于正例的概率可以表示为

$$P(y=1|x)=\frac{\exp(w\cdot x)}{1+\exp(w\cdot x)}$$

- 某样本属于负例的概率可以表示为

$$P(y=-1|x)=\frac{1}{1+\exp(w\cdot x)}$$

- 因此，似然函数可以是

$$\prod_{i=1}^{N}\frac{1}{1+\exp(-y_i w\cdot x_i)}$$

- 代价函数：

$$Cost(w)=\frac{1}{N}\sum_{i=1}^{N}\log\left(1+\exp(-y_i w\cdot x_i)\right)$$

- 使用梯度下降法就能反解出w了

PS：逻辑回归代价函数是凸函数的证明

$$Cost(w)=\frac{1}{N}\sum_{i=1}^{N}\log\left(1+\exp(-y_i w\cdot x_i)\right)$$

$$Cost'(w)=-\frac{1}{N}\sum_{i=1}^{N}y_i\left(1-\frac{1}{1+\exp(-y_i w\cdot x_i)}\right)x_i$$

$$Cost'(w)\text{保留关键位置}=\frac{yx}{1+\exp(-yw\cdot x)}$$

$$Cost''(w)=\frac{-yx}{(1+\exp(-yw\cdot x))^2}\exp(-yw\cdot x)(-yx)>0$$

逻辑回归的目标函数

- 目标函数

$$Obj(w) = \frac{1}{N}\sum_{i=1}^{N}\log\left(1+\exp(-y_i w\cdot x_i)\right)+\lambda\frac{1}{2}\|w\|_2^2$$

- 目标函数的导数

$$Obj'_w = \frac{1}{N}\sum_{i=1}^{N}\frac{\exp(-y_i w\cdot x_i)}{1+\exp(-y_i w\cdot x_i)}(-y_i x_i)+\lambda w$$

$$= \frac{1}{N}\sum_{i=1}^{N}-y_i\left(1-\frac{1}{1+\exp(-y_i w\cdot x_i)}\right)x_i+\lambda w$$

- 更新权重

$$w = w - \gamma Obj'_w$$

PS：Loss Function and Regularization summary

	loss function $L(\mathbf{w}; \mathbf{x}, y)$	gradient or sub-gradient
hinge loss	$\max\{0, 1 - y\mathbf{w}^T\mathbf{x}\}, \quad y \in \{-1, +1\}$	$\begin{cases} -y \cdot \mathbf{x} & \text{if } y\mathbf{w}^T\mathbf{x} < 1, \\ 0 & \text{otherwise.} \end{cases}$
logistic loss	$\log(1 + \exp(-y\mathbf{w}^T\mathbf{x})), \quad y \in \{-1, +1\}$	$-y\left(1 - \frac{1}{1+\exp(-y\mathbf{w}^T\mathbf{x})}\right) \cdot \mathbf{x}$
squared loss	$\frac{1}{2}(\mathbf{w}^T\mathbf{x} - y)^2, \quad y \in \mathbb{R}$	$(\mathbf{w}^T\mathbf{x} - y) \cdot \mathbf{x}$

	regularizer $R(\mathbf{w})$	gradient or sub-gradient
zero (unregularized)	0	$\mathbf{0}$
L2	$\frac{1}{2}\|\mathbf{w}\|_2^2$	$\mathbf{w}$
L1	$\|\mathbf{w}\|_1$	$\mathrm{sign}(\mathbf{w})$
elastic net	$\alpha\|\mathbf{w}\|_1 + (1-\alpha)\frac{1}{2}\|\mathbf{w}\|_2^2$	$\alpha\mathrm{sign}(\mathbf{w}) + (1-\alpha)\mathbf{w}$

特征工程——缺失值处理

- 方法一：
  - 直接删除有缺失的样本
  - 优点：简单　　　　　　　　　　　　　　缺点：浪费资源，无法处理有大量缺失值的任务
- 方法二：
  - 用平均值、中值、分位数、众数、随机值等替代
  - 优点：简单　　　　　　　　　　　　　　缺点：人为增加了噪声
- 方法三：
  - 用其他变量做预测模型来算出缺失变量。
  - 看似更符合实际，但是如果其他变量和缺失变量无关，则预测的结果无意义
- 方法四：
  - 最精确的做法，把变量映射到高维空间。
  - 比如性别，有男、女、缺失三种情况，则映射成3个变量：是否男、是否女、是否缺失。

特征工程——类别不平衡处理

- 欠采样
  - 去除一些反例（假设反例多），使得正、反例数目接近，然后再学习
  - 优点：速度快　　　缺点：可能会丢失一些重要信息
- 过采样
  - 增加一些正例，使得正、反例数目接近，然后再学习
  - 优点：保持数据信息　　　　缺点：可能会过拟合

- 代价敏感学习
  - 给某类样本更高的权重，比如，正例是反例的一半，那么正例的权重就是反例的2倍，在sklearn中由class_weight指定
  - 优点：速度快、降低过拟合风险　　　缺点：需要算法支持带权学习

分类模型评估

		真实值		总数
		p	n	
预测输出	p'	真阳性（TP）	伪阳性（FP）	P'
	n'	伪阴性（FN）	真阴性（TN）	N'
总数		P	N	

假设我们手上有60个正样本，40个负样本，我们想找出所有的正样本，模型查找出50个，其中只有40个是真正的正样本

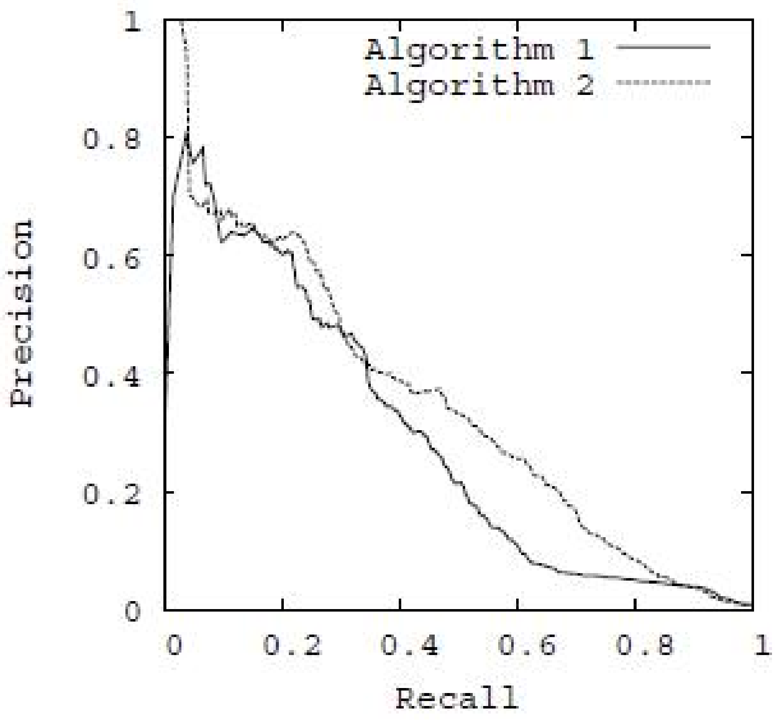
- TP: 将正类预测为正类数 40
- FN: 将正类预测为负类数 20 （漏了10个；错了10个）
- FP: 将负类预测为正类数 10 （错了10个）
- TN: 将负类预测为负类数 30 （40个负例中有10个预测为正：40-10）
- 准确率(accuracy) = 预测对的/所有 = (TP+TN)/(TP+FN+FP+TN) = 70%
- 精确率(precision)、查准类= TP/(TP+FP) = 80%
  - 命中敌人的炮弹数（选出正例的个数） / 发射的炮弹数（选出样本的个数） 过滤垃圾邮件
- 召回率(recall)、查全率 = TP/(TP+FN) = 2/3
  - 命中敌人的炮弹数（选出正例的个数） / 敌人的总数（真实值为正例的个数） 爱国者拦截导弹

分类模型评估

- 精确率(precision)、查准类= TP/(TP+FP)
- 召回率(recall)、查全率 = TP/(TP+FN)
- PR曲线
- F1值就是精确率和召回率的调和均值

$$\frac{2}{F_1} = \frac{1}{P} + \frac{1}{R}$$

		真实值		总数
		p	n	
预测输出	p'	真阳性 (TP)	伪阳性 (FP)	P'
	n'	伪阴性 (FN)	真阴性 (TN)	N'
总数		P	N	

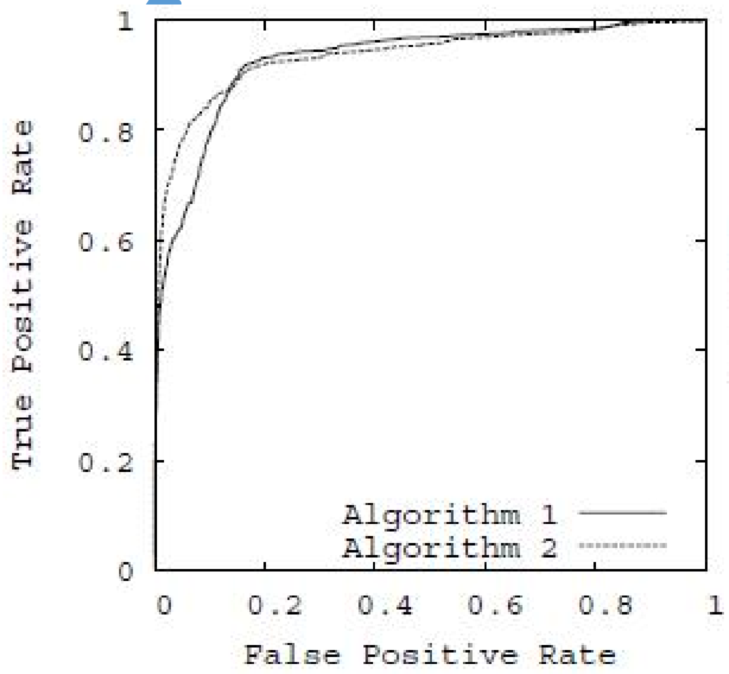


PR曲线

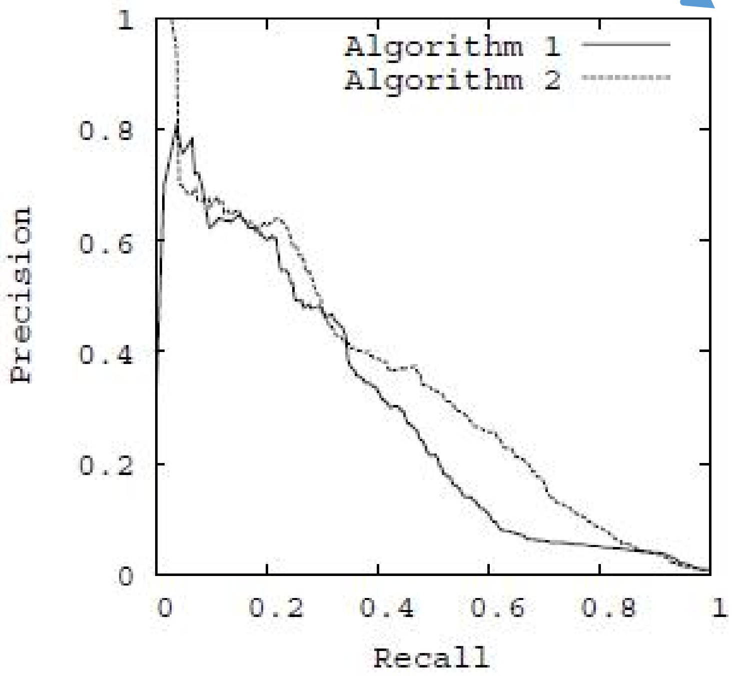
分类模型评估

		真实值		总数
		p	n	
预测输出	p'	真阳性(TP)	伪阳性(FP)	P'
	n'	伪阴性(FN)	真阴性(TN)	N'
总数		P	N	

- TPR：在所有实际为阳性的样本中，被正确地判断为阳性之比率。TPR=TP/(TP+FN)（和召回率一样)
- FPR：在所有实际为阴性的样本中，被错误地判断为阳性之比率。FPR=FP/(FP+TN)
- ROC曲线
- AUC是ROC曲线下的面积



ROC曲线



PR曲线

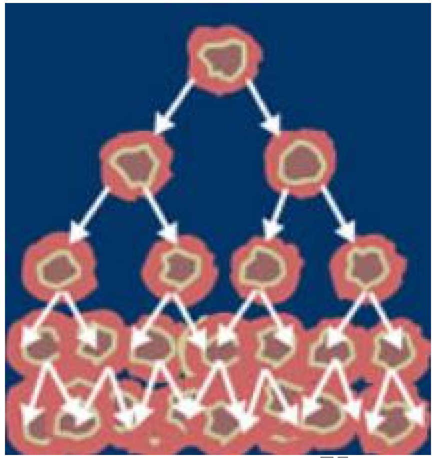
逻辑回归模型优缺点

- 优点
  - 模型简单，易于解释
  - 算法容易并行，适合大数据的求解
- 缺点
  - 不能拟合非线性数据
  - 采用何种特征组合需要一定的数据敏感性
  - 特征多了：提高模型效果，但是计算成本高，需要数据量大，否则会过拟合
  - 特征少了：有欠拟合的风险
  - 删除无用特征需要进行多重共线性判断（人力成本）

编程——逻辑回归综合案例之乳腺癌分类

- 基于病理数据进行乳腺癌预测(良性2/恶性4)，使用Logistic算法构建模型
  - 数据来源：http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29
  - 实验采用 UCI 数据集中的 Wisconsin 医学院的 William H.Wolberg 博士提供的乳腺 癌 的 数 据 样 本 。所有数据来自真实临床案例，每个案例有 10 个属性。其中前九个属性是检 测指标， 每个属性值用 1 到 10 的整数表示， 1 表示检测指标最正常， 10 表示最不正常。 第十个属性是分类属性，指示该肿瘤是否为恶性。

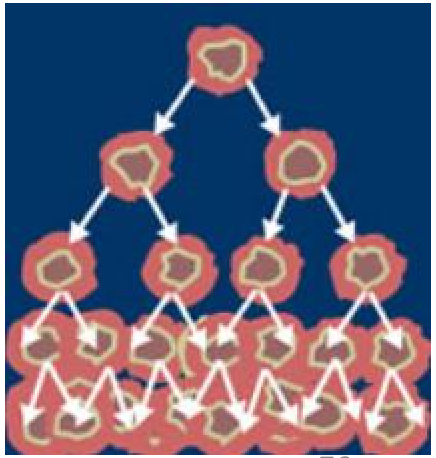
Data Set Characteristics:	Multivariate	Number of Instances:	699	Area:	Life
Attribute Characteristics:	Integer	Number of Attributes:	10	Date Donated	1992-07-15
Associated Tasks:	Classification	Missing Values?	Yes	Number of Web Hits:	388932



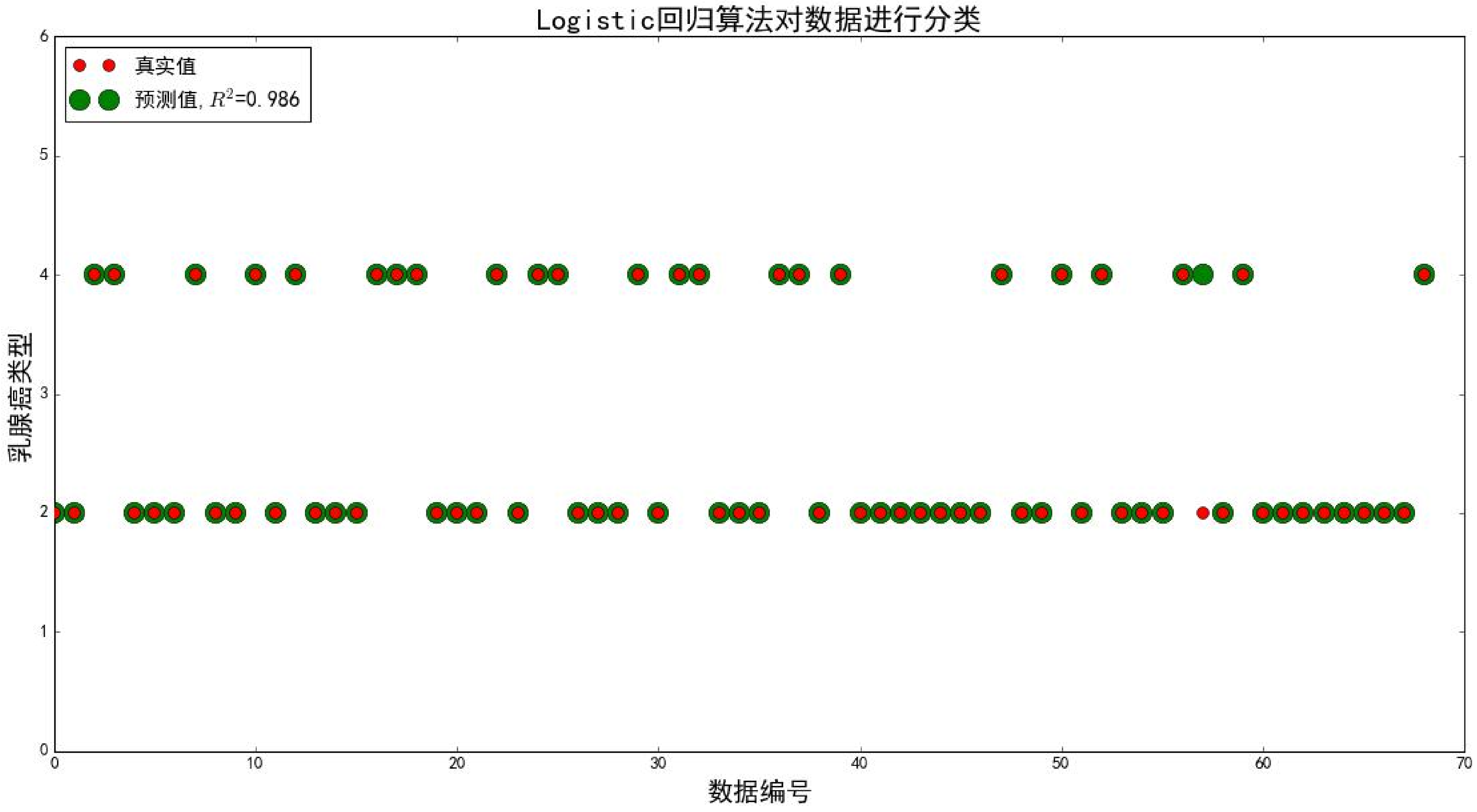
编程——模型保存与加载

- 基于病理数据进行乳腺癌预测(良性2/恶性4)，使用Logistic算法构建模型
  - 数据来源：http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29
  - 实验采用 UCI 数据集中的 Wisconsin 医学院的 William H.Wolberg 博士提供的乳腺 癌 的 数 据 样 本 。所有数据来自真实临床案例，每个案例有 10 个属性。其中前九个属性是检 测指标， 每个属性值用 1 到 10 的整数表示， 1 表示检测指标最正常， 10 表示最不正常。 第十个属性是分类属性，指示该肿瘤是否为恶性。

Data Set Characteristics:	Multivariate	**Number of Instances:**	699	**Area:**	Life
Attribute Characteristics:	Integer	**Number of Attributes:**	10	**Date Donated**	1992-07-15
Associated Tasks:	Classification	**Missing Values?**	Yes	**Number of Web Hits:**	388932



逻辑回归综合案例之乳腺癌分类



Softmax回归*

- 在 Softmax回归中，我们解决的是多分类问题；当K=2时，softmax 回归会退化为 logistic 回归

$$\mathrm{P}(\mathrm{y}=k\mid x)=\frac{\exp(w_{\mathrm{k}}\cdot x)}{1+\sum_{k=1}^{K-1}\exp(w_{\mathrm{k}}\cdot x)},k=1,2,\ldots,K-1$$

$$\mathrm{P}(\mathrm{y}=K\mid x)=\frac{1}{1+\sum_{k=1}^{K-1}\exp(w_{\mathrm{k}}\cdot x)}$$

- 似然函数和对数似然函数分别为：

$$\prod_{i=1}^{N}[\mathrm{P}(\mathrm{y}=1|x_i)]^{\mathrm{I}(y_i=1)}[\mathrm{P}(\mathrm{y}=2|x_i)]^{\mathrm{I}(y_i=2)}\cdots[\mathrm{P}(\mathrm{y}=K|x_i)]^{\mathrm{I}(y_i=K)}$$

$$\sum_{i=1}^{N}\sum_{j=1}^{K}I(y_i=j)\log\mathrm{P}(\mathrm{y}=j|x_i)$$

作业：使用Softmax对葡萄酒质量分类

- 基于葡萄酒数据进行葡萄酒质量预测模型构建，使用Softmax算法构建模型，并获取Softmax算法构建的模型效果(注意：分成11类)
  - 数据来源：http://archive.ics.uci.edu/ml/datasets/Wine+Quality

Attribute Information:

For more information, read [Cortez et al., 2009].
Input variables (based on physicochemical tests):
1 - fixed acidity
2 - volatile acidity
3 - citric acid
4 - residual sugar
5 - chlorides
6 - free sulfur dioxide
7 - total sulfur dioxide
8 - density
9 - pH
10 - sulphates
11 - alcohol
Output variable (based on sensory data):
12 - quality (score between 0 and 10)

非挥发性酸度
挥发性酸度
柠檬酸
残留糖
氯化物
游离二氧化硫
总二氧化硫
浓度
pH值
硫酸盐
酒精度
质量（得分在 0 和 10 之间）

```
7.4;0.7;0;1.9;0.076;11;34;0.9978;3.51;0.56;9.4;5
7.8;0.88;0;2.6;0.098;25;67;0.9968;3.2;0.68;9.8;5
7.8;0.76;0.04;2.3;0.092;15;54;0.997;3.26;0.65;9.8;5
11.2;0.28;0.56;1.9;0.075;17;60;0.998;3.16;0.58;9.8;6
7.4;0.7;0;1.9;0.076;11;34;0.9978;3.51;0.56;9.4;5
7.4;0.66;0;1.8;0.075;13;40;0.9978;3.51;0.56;9.4;5
7.9;0.6;0.06;1.6;0.069;15;59;0.9964;3.3;0.46;9.4;5
7.3;0.65;0;1.2;0.065;15;21;0.9946;3.39;0.47;10;7
7.8;0.58;0.02;2;0.073;9;18;0.9968;3.36;0.57;9.5;7
7.5;0.5;0.36;6.1;0.071;17;102;0.9978;3.35;0.8;10.5;5
```

北风网
IBEIFENG.COM

PS：逻辑回归为什么叫回归?

- $p=\mathrm{sigmod}(w\cdot x)$为样本属于正例的概率，逻辑回归模型可以理解为使用线性模型w·x去拟合对数几率（logit）

$$p=\frac{1}{1+e^{-w\cdot x}}$$

$$e^{-w\cdot x}=\frac{1}{p}-1=\frac{1-p}{p}$$

$$-w\cdot x=\log\frac{1-p}{p}$$

$$w\cdot x=\log\frac{p}{1-p}$$