华中科技大学
电子信息与通信学院
School of Electronic Information and Communications

机器学习导论

– Introduction to Machine Learning

第 02 章：回归模型

(Chapter 02: Regression Model)

华中科技大学电信学院

王邦　博士, 教授博导

wangbang@hust.edu.cn

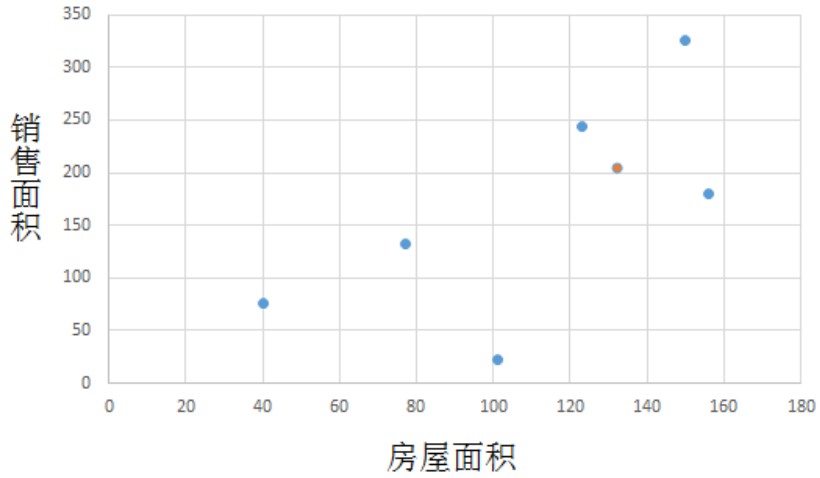
第 02 章：回归模型

华科电信 王邦编写

王邦编写 华科电信

问题引入 (1)

房价预测问题：已知一些不同面积住房的销售房价如右表所示。对于一个新的待售住房，其面积为 $132m^2$，预测其可能的销售价格。



面积（m^2）	销售价格（万元）
123	244
150	325
156	180
102	220
40	76
77	132
132	**?**

销售价格表

我们可将该问题表示为一个机器学习的问题，包括数据和模型两个部分。其中，数据集可进一步分为**训练集**和**测试集**：

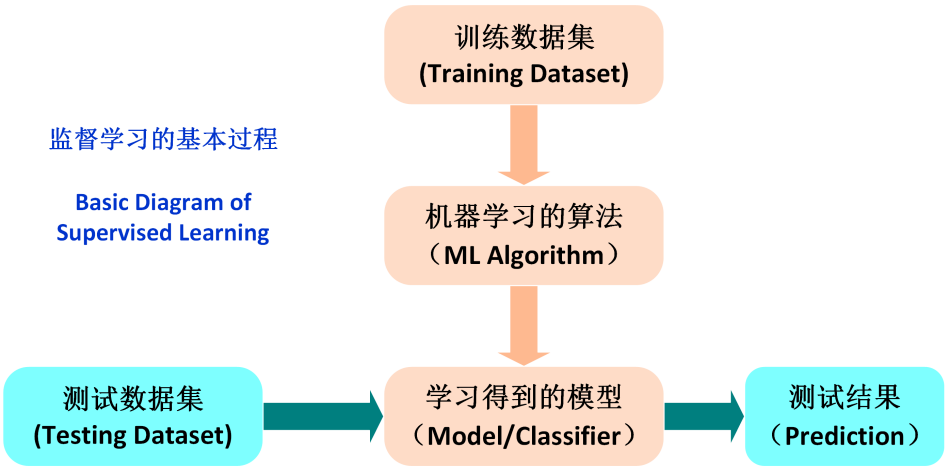
- **训练集 (training set)**：包含 m 个样本的训练集 D（本例 $m = 6$）,$(\boldsymbol{x}_i, y_i) \in D$ 表示 D 中第 i 个样本及其取值/标签，如本例 $(x_2, y_2) = (150, 325)$。此外，样本 $\boldsymbol{x}$ 可以由一个 d 维特征向量 $(x_1, ..., x_d)$ 来描述（本例只有一个属性“面积”，$d = 1$）和一个 1 维的取值 y_i。
- **测试集 (testing set)**：给定了包含 m_t 个样本的训练集 D_t（本例 $m_t = 1$），如本例测试样本为 $\boldsymbol{x}_t = 132$。测试样本的取值/标签 y_t 未知，是我们预测的目标。

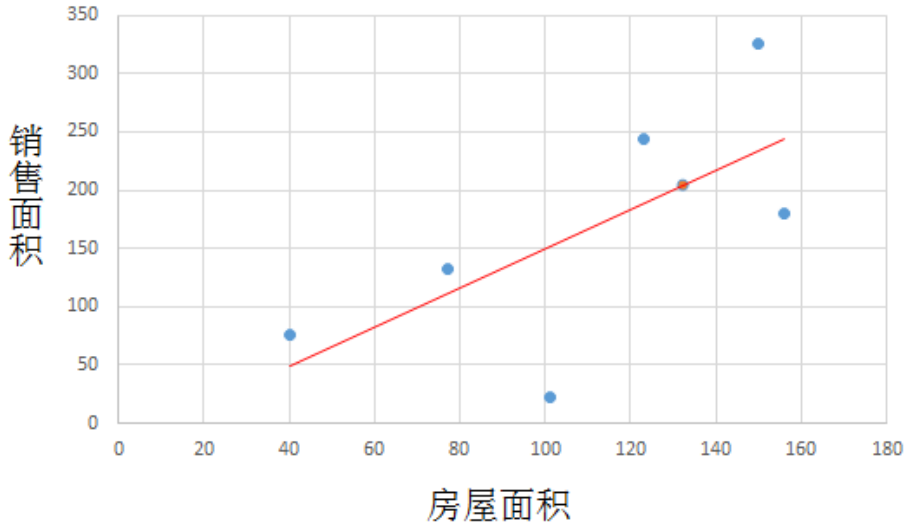
问题引入 (2)

模型包括训练和测试两个过程：

- **训练：** 通过训练集 D 拟合出函数 $h(\boldsymbol{x})$ (例如利用图中的线性函数)，使 $h(\boldsymbol{x})$ 在训练集 D 在某种评价指标 J 上全局最优或局部最优，同时对测试样本集 D_t 有预测能力。
- **测试：** 对于任意测试样本 $x_t \in D_t$ 通过函数 $h(\boldsymbol{x})$ 预测出 $\hat{y}_t = h(\boldsymbol{x}_t)$

如何获得预测函数函数 $h(\boldsymbol{x})$?





王邦编写 华科电信

线性模型 (Linear Model)

- 样本/数据 (sample): $\boldsymbol{x} = (x_1, x_2, ..., x_d)$，其中 x_i 是 $\boldsymbol{x}$ 在第 i 个属性上的取值。
- 回归函数/预测函数：$h(\boldsymbol{x})$

线性模型 (linear model)

$$h(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + ... + w_d x_d + b \tag{1}$$

写成向量形式：

$$h(\boldsymbol{x}) = \boldsymbol{w}^{\mathsf{T}} \boldsymbol{x} + b, \tag{2}$$

其中 $\boldsymbol{w} = (w_1, w_2, ..., w_d)$.

- **目标：** 通过对数据集的学习，确定线性回归函数 $h(\cdot)$，即：确定参数 $\boldsymbol{w}$ 和 b 。

代价函数 (cost function)

- **训练集:** $\{(\boldsymbol{x}_j, y_j)\}_{j=1}^m$，其中 y_j 为数据真实值。
- **代价函数：** 能够衡量模型**预测值**$h(\boldsymbol{x}_i)$ 与**真实值**y_i 之间差异的函数。代价函数不是唯一的，可以由具体的问题来确定；
- 代价函数越小，说明模型和参数越符合训练样本。代价函数取得最小值时，称之为通过训练得到了最优参数。

均方误差代价函数： (Rooted Mean Square Error, RMSE)

$$J(\boldsymbol{w}, b) = \frac{1}{2m} \sum_{j=1}^{m} (h(\boldsymbol{x}_j) - y_j)^2 \tag{3}$$

- **最优参数** $(\boldsymbol{w}^*, b^*) = \arg\min_{(\boldsymbol{w}, b)} J(\boldsymbol{w}, b)$

王邦编写 华科电信

最小二乘法 (1)

- 均方误差的几何意义：对应了常用的欧几里得距离或简称为欧氏距离 (Euclidean distance)。
- 基于均方误差最小化来进行模型求解的方法称为最小二乘法 (least square method)。
- 在线性回归中，最小二乘法就是试图找到一条直线，使得所有样本到直线上的欧式距离之和最小。

当 $d = 1$ 时：令 $E_{(w,b)} = \sum_{j=1}^{m}(y_j - wx_j - b)^2$

$$\frac{\partial E_{(w,b)}}{\partial w} = 2(w\sum_{j=1}^{m}x_j^2 - \sum_{j=1}^{m}(y_j - b)x_i) \tag{4}$$

$$\frac{\partial E_{(w,b)}}{\partial b} = 2(mb - \sum_{j=1}^{m}(y_j - wx_j)) \tag{5}$$

最小二乘法 (2)

- 令 $\frac{\partial E_{(w,b)}}{\partial w} = 0$ 以及 $\frac{\partial E_{(w,b)}}{\partial b} = 0$, 得到。

当 $d = 1$ 时：最优参数 w^* 和 b^* 的闭式 (closed form) 解

$$w^* = \frac{\sum_{j=1}^{m} y_j(x_j - \bar{x})}{\sum_{j=1}^{m} x_j^2 - \frac{1}{m}(\sum_{j=1}^{m} x_j)^2} \tag{6}$$

$$b^* = \frac{1}{m}\sum_{j=1}^{m}(y_j - w^* x_j) \tag{7}$$

其中 $\bar{x} = \frac{1}{m}\sum_{j=1}^{m} x_j$ 为 x 的均值。

最小二乘法 (3)

- 多元线性回归(multivariate linear regression)：当数据集的每个样本由 d 个属性描述时，需要学习

  $h(\boldsymbol{x}_j) = \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_j + b$ 使得 $h(\boldsymbol{x}_j) \approx y_j$

- 令 $\boldsymbol{y} = (y_1; y_2; \ldots; y_m)$；令 $\hat{\mathbf{w}} = (\mathbf{w}; \mathbf{b})$，构建一个 $m \times (d+1)$ 大小的矩阵 $\boldsymbol{X}$，其中每一行对应一个样本。该行前 d 个元素对应样本的 d 个属性，最后一个元素置为 1，即

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^{\mathsf{T}} & 1 \\ \boldsymbol{x}_2^{\mathsf{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathsf{T}} & 1 \end{pmatrix}$$

- 最优参数：$\hat{\boldsymbol{w}}^* = \arg\min_{\hat{\boldsymbol{w}}} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})^{\mathsf{T}} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})$

最小二乘法 (4)

- 令 $E_{\hat{\boldsymbol{w}}} = (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})^{\mathsf{T}}(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})$, 对 $\hat{\boldsymbol{w}}$ 求导得到

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\mathbf{X}^{\mathsf{T}}(\boldsymbol{X}\hat{\boldsymbol{w}} - \boldsymbol{y}). \tag{8}$$

- 当 $\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X}$ 为**满秩矩阵** (full-rank matrix) 或**正定矩阵** (positive definite matrix)，可得

$$\hat{\boldsymbol{w}}^* = (\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathsf{T}}\boldsymbol{y} \tag{9}$$

  其中 $(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}$ 是矩阵 $(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})$ 的逆矩阵。令 $\hat{\boldsymbol{x}}_j = (x_j; 1)$，则最终学得的多元线性回归模型为

$$h(\hat{\boldsymbol{x}}_j) = \hat{\boldsymbol{x}}_j^{\mathsf{T}}(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X})^{-1}\boldsymbol{X}\boldsymbol{y} \tag{10}$$

- 当 $\boldsymbol{X}^{\mathsf{T}}\boldsymbol{X}$ 不是满秩矩阵时，可以解出多个 $\hat{\boldsymbol{w}}$ 均能使得均方误差最小化。此时，常见做法是引入**正则化项** (regularization).

最小二乘法例子：一元线性回归

对于问题引入中的例子，我们如何构建一个线性函数 $h = wx + b$ 来对样本 $x_t = 132$ 进行预测呢？

我们可通过最小二乘法得到预测函数 $h(\boldsymbol{x})$，即最小化均方误差 $E(w, b)$。对如下的特征维度 $d = 1$ 的房价数据，

$E(w, b) = \sum_{j=1}^{m}(y_j - wx_j - b)^2$，我们可通过公式 (6)(7) 直接求得 w^*, b^*。

面积（m^2）	销售价格（万元）
123	244
150	325
156	180
102	220
40	76
77	132
132	**?**

$\bar{x} = \frac{1}{m}\sum_{j=1}^{m} x_j = \frac{123+150+156+102+40+77}{6} = 108$

$\sum_{j=1}^{m} x_j^2 = 123^2+150^2+156^2+102^2+40^2+77^2 = 79898$

$x_j - \bar{x}$:$\{15, 42, 48, -6, -68, -31\}$

$\sum_{j=1}^{m} y_j(x_j - \bar{x}) = 244 \times 15 + 325 \times 42 + 180 \times 48 - 220 \times 6 - 76 \times 68 - 132 \times 31 = 15370$

$w^* = \frac{\sum_{j=1}^{m} y_j(x_j - \bar{x})}{\sum_{j=1}^{m} x_j^2 - m\bar{x}^2} = \frac{15370}{79898 - 6 \times 108^2} = 1.5503$

$b^* = \frac{1}{m}\sum_{j=1}^{m}(y_j - w^* x_j) = (224 - 1.5503 \times 123 + 325 - 1.5503 \times 150 + 180 - 1.5503 \times 156 +$

$220 - 1.5503 \times 102 + 76 - 1.5503 \times 40 + 132 - 1.5503 \times 77)/6 = 28.7343$

由此得到预测函数 $h(x) = w^* x + b^* = 1.5503x + 28.7343$

从而 $y_t = h(x_t) = 1.5503 \times 132 + 28.7343 = 204.6396$

最小二乘法例子：多元线性回归

如果每个样本有两个属性“面积”和“所在地块平均地价”，则应该得到预测函数 $h(\boldsymbol{x})$，并对样本 $\boldsymbol{x}_t = \{132, 2634\}$ 进行预测。

$x_{(1)}$: 面积（m^2）	$x_{(2)}$: 所在地块平均地价（元/平）	销售价格（万元）
123	2400	244
150	2824	325
156	1706	180
102	3016	220
40	2432	76
77	2264	132

我们可通过式 $\hat{w}^* = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$ 求得最小化均方误差的 $\hat{w}^*$

$$\mathbf{X} = \begin{bmatrix} 123 & 2400 & 1 \\ 150 & 2824 & 1 \\ 156 & 1704 & 1 \\ 102 & 3016 & 1 \\ 40 & 2432 & 1 \\ 77 & 2264 & 1 \end{bmatrix} \quad \mathbf{X}^\mathsf{T}\mathbf{X} = \begin{bmatrix} 79898 & 1563864 & 648 \\ 1563864 & 36775168 & 14640 \\ 648 & 14640 & 6 \end{bmatrix}$$

由于 $\mathbf{X}^\mathsf{T}\mathbf{X}$ 是正定矩阵，可由公式 (9) 得到 $\hat{w}^*$

$$\hat{w}^* = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} = \begin{bmatrix} 1.7411 \\ 0.1096 \\ -259.3303 \end{bmatrix}$$

从而得到二元回归模型: $h(\hat{x}) = \hat{x}^\mathsf{T}\hat{w}^* = 1.7411x_1 + 0.1096x_2 - 259.3303$

由 $x_1 = 132, x_2 = 2634$ 可得 $h(\hat{x}_t) = 259.1813$

梯度下降法 (1)

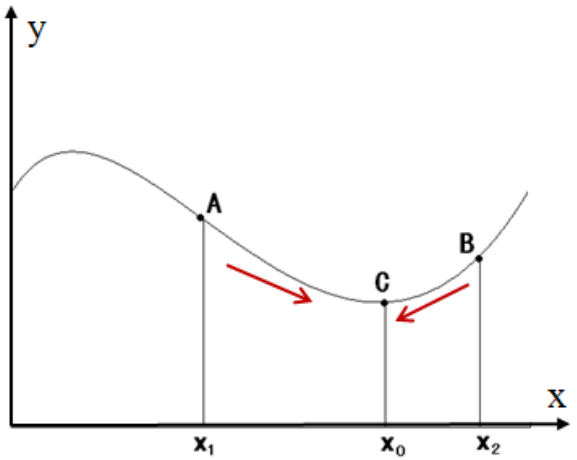
梯度下降法 (Gradient Descent Method)，又称为最速下降法。1847 年由著名数学家**柯西** Cauchy 提出。

基本思想：假设我们爬山，如果想最快的上到山顶，那么我们应该从山势最陡的地方上山。也就是山势变化最快的地方上山。同样，如果从任意一点出发，需要**最快搜索到函数最大值**，那么我们也应该从**函数变化最快的方向**搜索。

函数变化最快的方向是什么呢？函数的梯度。如果需要找的是函数极小点，那么应该从负梯度的方向寻找，该方法称之为梯度下降法。

一元函数的梯度为该函数的倒数：$\nabla f(x) = f'(x)$

要搜索极小值 C 点，在 A 点必须向 x 增加方向搜索，此时与 A 点梯度方向相反；在 B 点必须向 x 减小方向搜索，此时与 B 点梯度方向相反。总之，搜索极小值，必须向负梯度方向搜索。



梯度下降法 (2)

梯度下降法的一般步骤：考虑函数 $y = f(\boldsymbol{x}) = f(x_1, x_2, ..., x_d)$，给定初始参数 $\boldsymbol{x}^0 = (x_1^0, x_2^0, ..., x_d^0)$。

1. 设定迭代步长 α 以及参数变化量 ϵ，以及迭代次数 $t = 0$
2. 计算当前位置处的各个偏导数：$\nabla f_{x_i}(x_i^t) = \frac{\partial y}{\partial x_i}(x_i^t), \quad i = 1, ..., d$
3. 设置 $t = t + 1$，更新当前函数的各个参数值：

$$x_i^t = x_i^{t-1} - \alpha \nabla f_{x_i}(x_i^t), \quad i = 1, ..., d$$

4. 计算当前参数的变化量 $\| \boldsymbol{x}^t - \boldsymbol{x}^{t-1} \|_\ell$，如果小于 ϵ，则退出；否则，返回第 2 步。

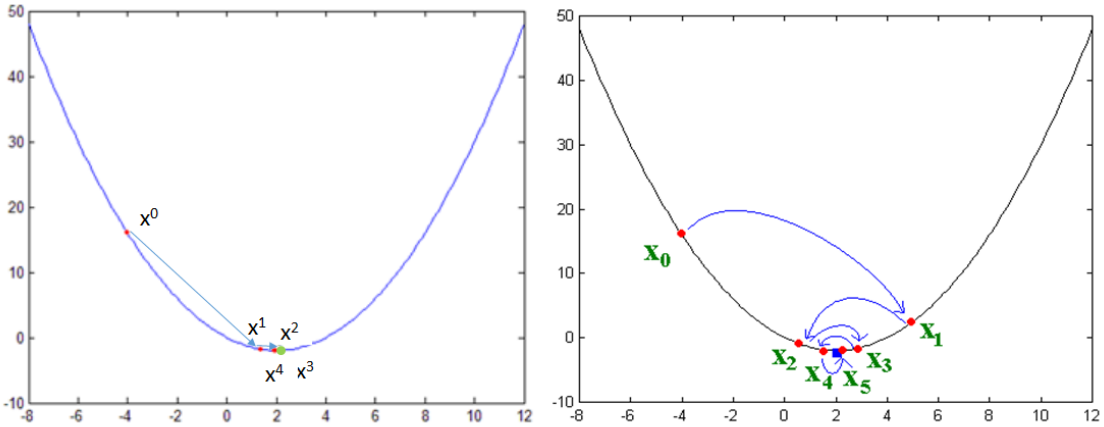
注意：$\| \cdot \|_\ell$ 计算两个向量的 ℓ 范数，也可以采用其他的计算参数变化量的函数。参数初始值 $\boldsymbol{x}^0$ 以及步长 α 的选择都会影响参数搜索结果。

梯度下降法 (3)

梯度下降法举例：对函数 $f(x) = \frac{x^2}{2} - 2x$，给定初始出发点 $x^0 = -4$，利用梯度下降法求其极值。

(1) 设置 $\alpha = 0.9$ 和 $\epsilon = 0.01$；

(2) 设置 $\alpha = 1.5$ 和 $\epsilon = 0.01$；



解： $f'(x) = x - 2$

(1) $x^0 = -4$；$x^1 = x^0 - \alpha f'(x^0) = -4 - 0.9 \times (-6) = 1.4$, $|x^1 - x^0| = 5.4 > \epsilon$；

$x^2 = 1.94, |x^2 - x^1| = 0.54$；$x^3 = 1.994, |x^3 - x^2| = 0.054$；$x^4 = 1.9994, |x^4 - x^3| = 0.0054 < \epsilon$。

终止迭代。

(2) $x^0 = -4$；$x^1 = x^0 - \alpha f'(x^0) = -4 - 1.5 \times (-6) = 5.0$, $|x^1 - x^0| = 9.0 > \epsilon$；

$x^2 = 2.75, |x^2 - x^1| = 2.25$；$x^3 = 1.625, |x^3 - x^2| = 1.125$；$x^4 = 2.1875, |x^4 - x^3| = 0.5625$

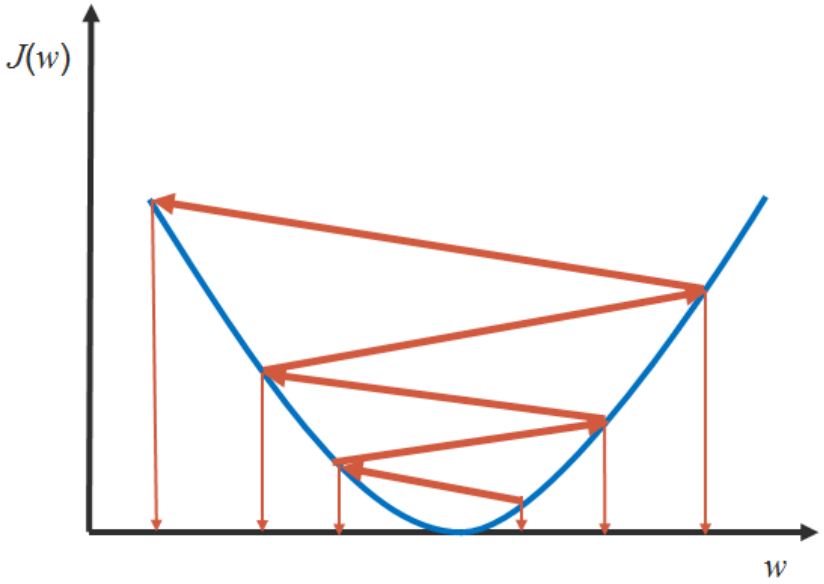
...

梯度下降法 (4)

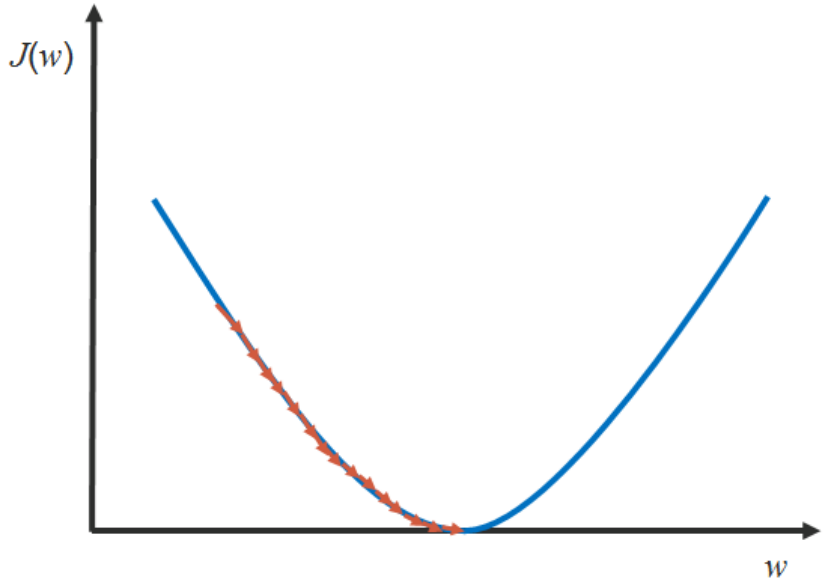
在线性回归问题中，假设 $d = 1, b = 0$。此时的代价函数是关于 w 的二次函数

$$J(w, 0) = \frac{1}{2}(h(x) - y)^2 = \frac{1}{2}(wx - y)^2$$

学习速率 α 过大时，可能造成不收敛。



学习速率 α 过小时，收敛速度可能很慢。



注意：根据问题的不同，应设置不同的学习速度，如果代价函数很平滑（梯度很小），可以设置稍大的 α 以尽快收敛，若很陡峭（梯度很大），可以设置稍小的 α 以不至于跳出最优解。

梯度下降法 (5)

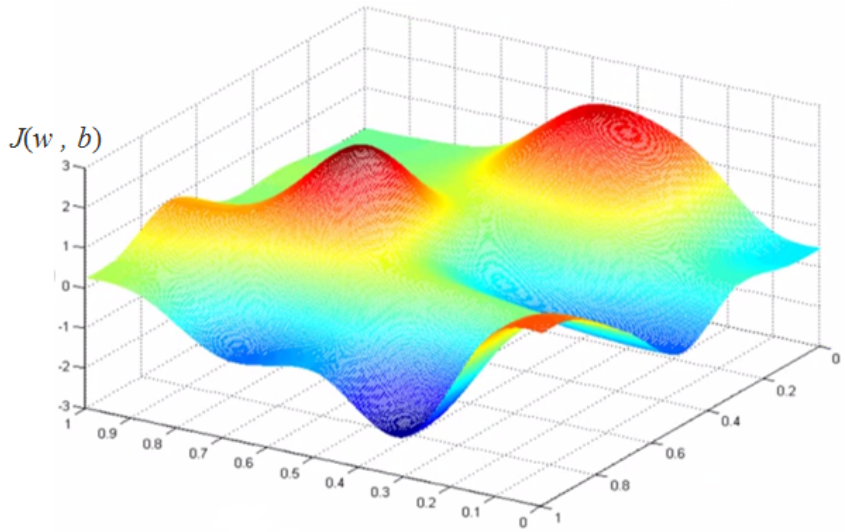
在线性回归问题中，假设 $d = 1$,b 为带估计参数。此时的代价函数是关于 (w, b) 的二元函数，在第 $t + 1$ 次迭代时，更新公式为：

$$b^{t+1} = b^t - \alpha \frac{\partial J(w,b)}{\partial b}(w^t, b^t), \quad w^{t+1} = w^t - \alpha \frac{\partial J(w,b)}{\partial w}(w^t, b^t)$$

推广至一般情况，对 d 个属性描述的样本 $\boldsymbol{x} = (x_1, ..., x_d)$，在第 $t + 1$ 步时迭代公式为：

$$\begin{aligned} b^{t+1} &= b^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial b}(\boldsymbol{w}^t, b^t) \\ w_1^{t+1} &= w_1^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial w_1}(\boldsymbol{w}^t, b^t) \\ &\cdots \\ w_d^{t+1} &= w_d^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial w_d}(\boldsymbol{w}^t, b^t) \end{aligned}$$

$d = 1$ 时，估计参数 (w, b)，此时代价函数为一个曲面。



梯度下降法 (5)

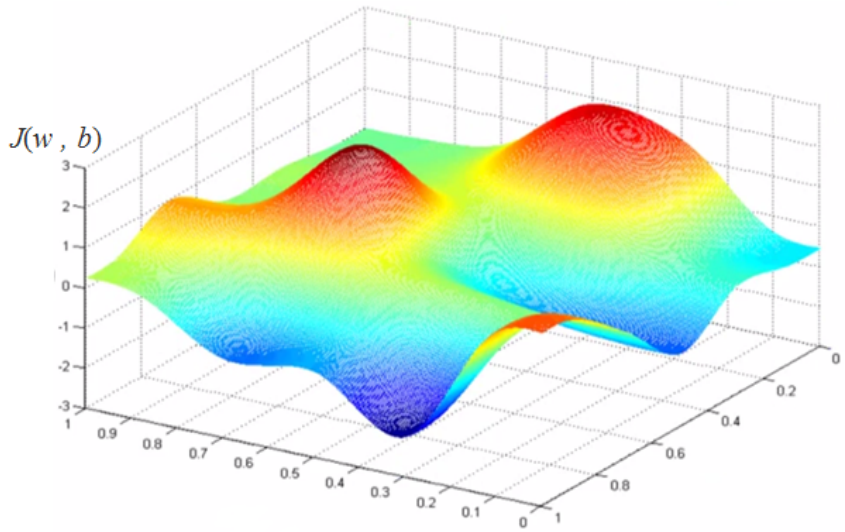
在线性回归问题中，假设 $d = 1$,b 为带估计参数。此时的代价函数是关于 (w, b) 的二元函数，在第 $t + 1$ 次迭代时，更新公式为：

$$b^{t+1} = b^t - \alpha \frac{\partial J(w,b)}{\partial b}(w^t, b^t), \quad w^{t+1} = w^t - \alpha \frac{\partial J(w,b)}{\partial w}(w^t, b^t)$$

推广至一般情况，对 d 个属性描述的样本 $\boldsymbol{x} = (x_1, ..., x_d)$，在第 $t + 1$ 步时迭代公式为：

$$\begin{aligned}
b^{t+1} &= b^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial b}(\boldsymbol{w}^t, b^t) \\
w_1^{t+1} &= w_1^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial w_1}(\boldsymbol{w}^t, b^t) \\
&\cdots \\
w_d^{t+1} &= w_d^t - \alpha \frac{\partial J(\boldsymbol{w},b)}{\partial w_d}(\boldsymbol{w}^t, b^t)
\end{aligned}$$

$d = 1$ 时，估计参数 (w, b)，此时代价函数为一个曲面。



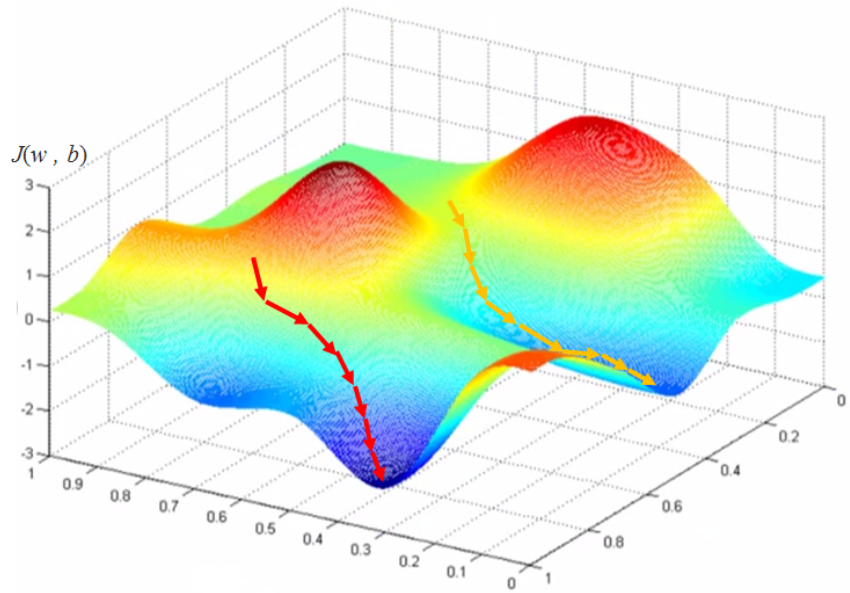
全局最优与局部最优

全局最优： 在解搜索空间，若某个解与所有其他解相比是最优的，就可以被称为全局最优。

局部最优： 在解搜索空间，某个解在局部搜索空间是最优的，但不一定在全部解搜索空间最优。

- 代价函数可能有多个局部最优值，通常局部最优值不对应全局最优值。
- 起始点即对应着 $(\boldsymbol{w}^0, b^0)$ 的初始化值。从不同的“起始点”出发时，梯度下降会收敛到不同的局部最优位置。
- 可以多次随机初始化初始点 $(\boldsymbol{w}^0, b^0)$，取多次迭代结果中的最小值作为最后输出解。

从不同初始点进行搜索，可能得到不同的局部最优解。



王邦编写 华科电信

广义线性模型

广义线性模型(generalized linear model): 考虑单调可微函数 $g(\cdot)$，称之为“联系函数”(link function)。定义

$$y = g^{-1}(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x} + b). \tag{11}$$

对数线性回归 (log-linear regression)：

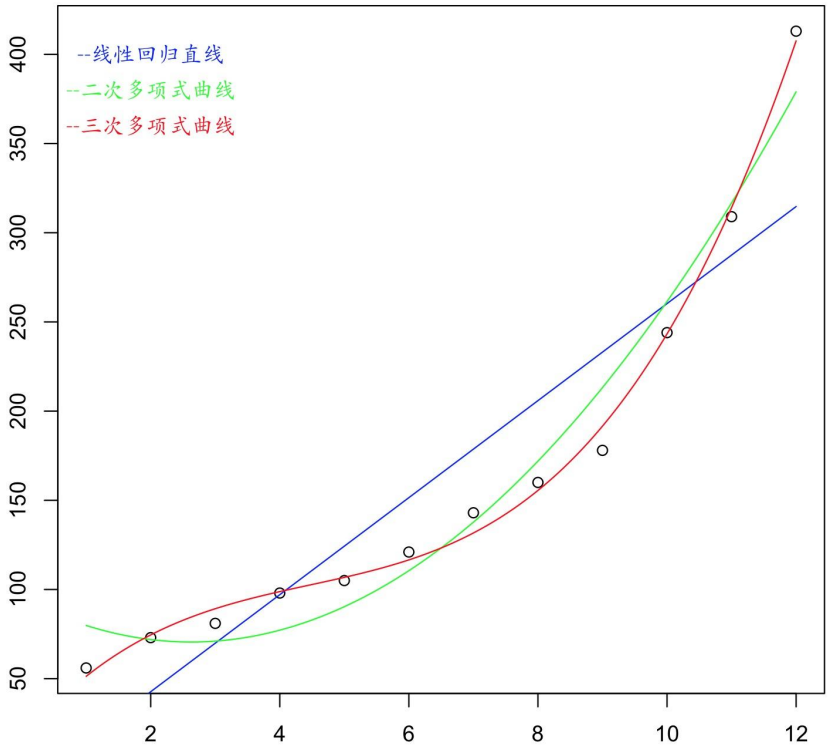
$g(\cdot) = e^{\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}+b}$

$$\ln y = \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x} + b \tag{12}$$

多项式回归 (Polynomial regression)：例如，令 $x_1 = x, x_2 = x^2, x_3 = x^3, \ldots$，设

$$h(\boldsymbol{x}) = w_1 x + w_2 x^2 + w_3 x^2 + b \tag{13}$$

当多项式的幂次选取适宜时，预测函数 $h(\boldsymbol{x})$ 能较好的拟合类似的非线性问题。



王邦编写 华科电信

对数几率模型 (1)

回归问题：预测结果通常是在某区间内的一个**连续**值。

- 根据最近三年的房屋交易情况，预测某套房屋的成交价格;
- 根据历史气象记录，预测某地某时的降水概率；

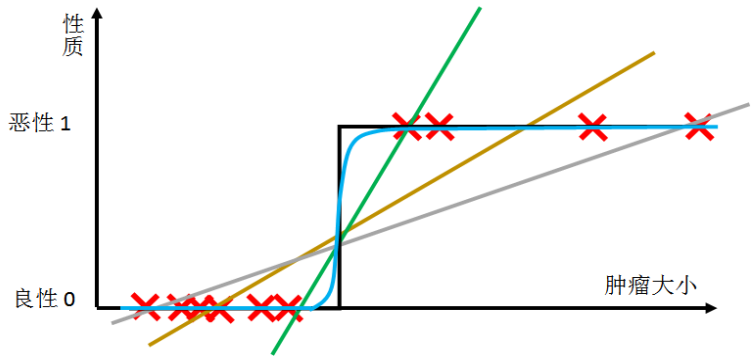
分类问题：预测结果通常为在一个可数集合中的**离散值**（类别标号）。

- 根据以往收到邮件的特征，判定新邮件是不是垃圾邮件。**是**或**否**？
- 根据以往病例，判定某个病人身患肿瘤的情况。**良性**或**恶性**？

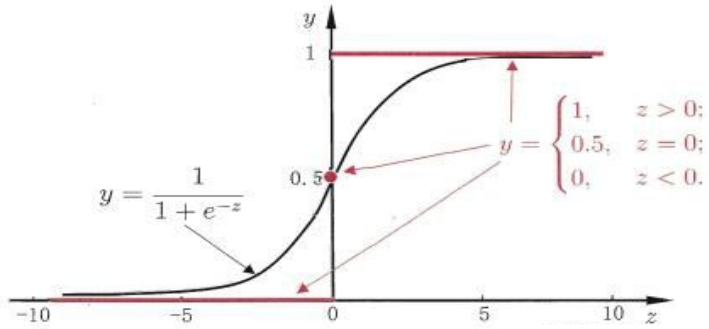
两者联系：可以通过一个单调可微函数将分类任务的类别标记与线性回归模型的预测值联系起来。

例子（二分类任务）：假设肿瘤的性质（良性或恶性）可以通过肿瘤大小 x 判断。即肿瘤性质的输出为 $y \in \{0, 1\}$，而线性回归模型产生的预测值 $z = wx + b$ 是实数值。因此，我们需要将 z 转换为 $0/1$ 值。

对数几率模型 (2)



肿瘤大小与肿瘤性质判断



单位阶跃函数与对数几率函数

单位阶跃函数 (unit-step function): $y = \begin{cases} 1, & z > \epsilon \\ 0, & \text{otherwise.} \end{cases}$。该函数在 $z = \epsilon$ 处不连续，不能作为广义线性回归的联系函数 $g(\cdot)$.

Sigmoid 函数 (又称之为对数几率函数，logistic function)：

$$y = g(z) = \frac{1}{1 + e^{-z}} \tag{14}$$

Sigmoid 函数单调可微，其值域为 $(0, 1)$，将 z 值转化为一个接近 0 或 1 的 y 值。且输出值在 $z = 0$ 附近变化很陡，适用于分类问题。

对数几率模型 (3)

利用 Sigmoid 函数作为预测函数 $h(\boldsymbol{x})$，得到：

$$h(\boldsymbol{x}) = \frac{1}{1 + e^{-(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}+b)}}.$$

其预测值 $h(\boldsymbol{x})$ 仍为连续区间 $(0, 1)$ 上的某个值。处理方法是确定一个阈值 ϵ（通常取 $\epsilon = 0.5$），使得

$$\begin{aligned} h(\boldsymbol{x}) \geq \epsilon \qquad & \text{predict: } y = 1 \\ h(\boldsymbol{x}) < \epsilon \qquad & \text{predict: } y = 0 \end{aligned}$$

若将 $h(\boldsymbol{x})$ 视作样本 $\boldsymbol{x}$ 作为正例的可能性，则 $1 - h(\boldsymbol{x})$ 是其作为反例的可能性，即：

$$\begin{aligned} p(y = 1|\boldsymbol{x}) &= h(\boldsymbol{x}) \\ p(y = 0|\boldsymbol{x}) &= 1 - h(\boldsymbol{x}) \end{aligned}$$

对数几率模型 (4)

几率(odds)：$h(\boldsymbol{x})$ 与 $1-h(\boldsymbol{x})$ 的比值

$$\frac{h(\boldsymbol{x})}{1-h(\boldsymbol{x})}$$

反映了 $\boldsymbol{x}$ 作为正例的相对可能性。对几率取对数得到对数几率(log odds, 亦称 logit)，即

$$\ln \frac{h(\boldsymbol{x})}{1-h(\boldsymbol{x})}$$

注意：预测函数 $h(\boldsymbol{x})=\frac{1}{1+e^{-(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}+b)}}$ 实际上在用线性回归模型的预测结果去逼近真实标记的对数几率。因此，其对应的模型称之为“对数几率回归”(logistic regression, 亦称 logit regression)。需要指出的是，虽然其称之为" 回归"，但实际是一种**分类**方法。因其能直接对分类可能性进行建模，无需事先假设数据分布，可以避免假设分布不准确带来的问题。

对数几率模型 (5)

对**训练集**: $\{(\boldsymbol{x}_j, y_j)\}_{j=1}^m$，如何确定 $h(\boldsymbol{x}) = \frac{1}{1+e^{-(\boldsymbol{w}^\top \boldsymbol{x}+b)}}$ 的参数 $\boldsymbol{w}$ 和 b。

代价函数用于衡量模型**预测值**$h(\boldsymbol{x}_j)$ 与**真实值**y_j 之间的差异。利用**极大似然法**设置代价函数，最大化**对数似然**(log-likelihood)：

$$\ell(\boldsymbol{x}, b) = \sum_{j=1}^{m} \ln p(y_j|\boldsymbol{x}_j; \boldsymbol{w}, b) \tag{15}$$

在正例和反例两种情况下，$p(y_j|\boldsymbol{x}_j; \boldsymbol{w}, b) = \begin{cases} h(\boldsymbol{x}_j), & y_j = 1 \\ 1 - h(\boldsymbol{x}_j), & y_j = 0 \end{cases}$

代价函数：

$$J(\boldsymbol{w}, b) = -\sum_{j=1}^{m} [y_j \ln h(\boldsymbol{x}_j) + (1 - y_j) \ln(1 - h(\boldsymbol{x}_j))]$$

优化目标：$\min_{(\boldsymbol{w},b)} J(\boldsymbol{w}, b)$；优化方法：梯度下降法、牛顿法等。

对数几率回归例子 (1)

如果要预测住房 $\boldsymbol{x}_t = [132, 2634]$ 所在小区的档次呢？

面积（m^2）	所在地块平均地价（元/平）	所在小区档次
123	2400	中档
150	2824	高档
156	1706	中档
102	3016	高档
40	2432	中档
77	2264	高档

采用梯度下降法训练模型：$h(x) = \frac{1}{1+exp(-(\mathbf{w}^T\mathbf{x}+b))}$
数据预处理阶段：

- 我们令 $y = 1$ 表示高档，$y = 0$ 表示中档；
- 采用线性映射 $\tilde{x}^a = \frac{x^a - x^a_{min}}{x^a_{max} - x^a_{min}}$ 归一化：将“面积”归一化为 $[0.7155, 0.9483, 1.0000, 0.5345, 0, 0.3190]$，“地价”归一化为 $[0.5305, 0.8537, 0, 1.0000, 0.5549, 0.4268]$。

对数几率回归例子 (2)

训练过程：

1. 以随机值初始化 $\mathbf{w}$，如初始化为均匀分布 $U(-0.1, 0.1)$，本例中，$\mathbf{w}$ 是 2 维列向量
2. 由 $h(x)$ 计算出每个数据 $x_i \in D$ 的预测结果 $\tilde{y}_i = h(x_i)$；
3. 由 $J(\mathbf{w}, b) = -\sum_{i=1}^{m} (y_i \ln(h(\mathbf{x}_i)) + (1 - y_i) \ln(1 - h(\mathbf{x}_i)))$ 及当前预测结果 $\tilde{y}$ 计算代价函数 J
4. 求代价函数 $J(\mathbf{w}, b)$ 关于 $\forall w \in \mathbf{w}, b$ 的梯度 $g_w = \frac{dJ(w)}{dw}, g_b = \frac{dJ(b)}{db}$，并以 $w = w - \alpha g_w$ 更新权值，其中 α 为常数步长如 $\alpha \approx 0.01$
5. 重复步骤 $2 \sim 4$，直到 $J(\mathbf{w}, b)$ 收敛，得到最终的 w^*, b^*

测试过程：将 x_t 由训练过程得到的 x_{max}, x_{min} 采用线性映射进行归一化，之后带入 $h(x)$，若 $h(x_t) > 0.5$，则认为是“高档”，否则是中档。

注意：使用梯度下降法时，需要进行特征归一化，即将特征缩放到同一尺度，比如，我们采用线性映射 $\tilde{x}^a = \frac{x^a - x^a_{min}}{x^a_{max} - x^a_{min}}$ 将训练集中的属性 a 归一化到区间 $[0, 1]$。

线性回归与对数几率回归比较

	线性回归	对数几率回归
适用问题	回归	分类
预测函数	$h(\boldsymbol{x}) = \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x} + b$	$h(\boldsymbol{x}) = \frac{1}{1+e^{-(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}+b)}}$
代价函数	$J(\boldsymbol{x}, b) = \frac{1}{2m}\sum_{j=1}^{m}(h(\boldsymbol{x}_j) - y_j)^2$	$J(\boldsymbol{x}, b) = -\sum_{j=1}^{m}[y_j \ln h(\boldsymbol{x}_j) + (1-y_j)\ln(1-h(\boldsymbol{x}_j))]$
优化目标	$\min_{(\boldsymbol{w},b)} J(\boldsymbol{w}, b)$	$\min_{(\boldsymbol{w},b)} J(\boldsymbol{w}, b)$
优化方法	最小二乘法、梯度下降法、牛顿法等	梯度下降法、牛顿法等

线性回归和对数几率回归既有区别又联系紧密。

- 区别在于，线性回归中标签为一连续值（回归），而对数几率回归中标签为离散值（分类）。其本质上都属于线性模型（对数几率回归为广义线性模型）。
- 在某些情况下：如要预测某地的降水概率，这是一个回归问题，应用线性回归。
- 若定义降水概率小于 20% 为“好天气”，否则为“坏天气”，预测天气的“好坏”，这是一个分类问题，应用对数几率回归

华科电信 王邦编写

多分类学习 (1)

多分类问题： 考虑 N 个类别 $\mathcal{Y} = \{C_1, C_2, ..., C_N\}$，其中 c_n 为第 n 个类别。给定数据集 $D = \{(\boldsymbol{x}_j, y_j)\}_{j=1}^m$，$y_j \in \mathcal{Y}$，对新样本 $\boldsymbol{x}$ 分为 $\mathcal{Y}$ 中的某一个类。

思路： 将多分类任务拆为若干个二分类任务求解。先对问题进行进行拆分，然后为拆出的每个二分类任务训练一个分类器；在测试时，对这些分类器的预测结果进行**集成**以获得最终的多分类结果。

拆分策略： “一对一” (One vs. One，简称 OvO)；“一对其余” (One vs. Rest，简称 OvR) 和 “多对多” (Many vs. Many，简称 MvM)。

OvO 策略: 将 N 个类别两两配对，从而产生 $N(N-1)/2$ 个二分类任务。例如将标记为类别 C_i 和 C_j 训练数据用于训练一个二分类器；测试阶段样本 $\boldsymbol{x}$ 通过该二分类器标记为类别 C_i **或**C_j；这样该样本经过 $N(N-1)/2$ 这样的二分类器得到 $N(N-1)/2$ 个分类结果；最终结果可通过**投票产生：**即被预测类别最多的类别作为最后分类结果。

多分类学习 (2)

OvR 策略：：每次将一个类的训练数据作为正例，所有其余训练数据作为反例来训练一个二分类器；这样共得到 N 个二分类器。在测试时，如果仅有一个分类器预测为正类，则对应的类别标记为最终分类结果；如果有多个分类器预测为正类，则通常考虑各分类器的**预置置信度**，选择置信度最大的类别标记为分类结果。如 logistic 回归中比较第 n 个二分类器输出 $h_n(\boldsymbol{x})$ 的大小。

	OvO 策略	OvR 策略
分类器数目	$N(N-1)/2$	N
存储开销和测试时间	较大	较少
训练时间	较少	较长
预测性能	取决于具体数据分布，多数情况下接近	

多分类学习 (3)

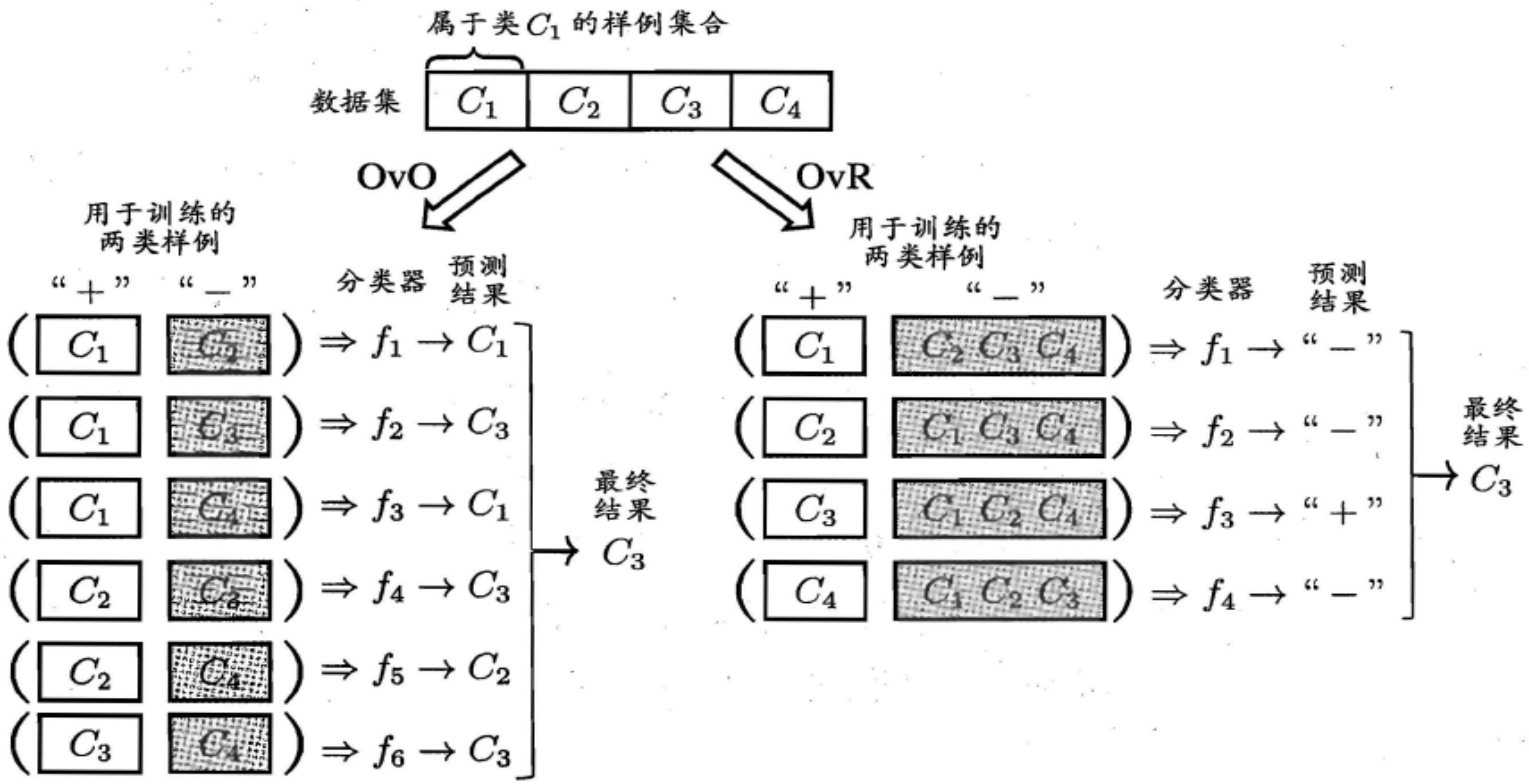


图 3.4 OvO 与 OvR 示意图 (来源：周志华，《机器学习》，清华大学出版社)

OvO：对 6 个二分类器的分类结果投票，最终输出类别 C_3。

OvR：4 个二分类器的分类结果只有一个为正类，则最终输出该二分类器分类结果 C_3。

多分类学习 (4)

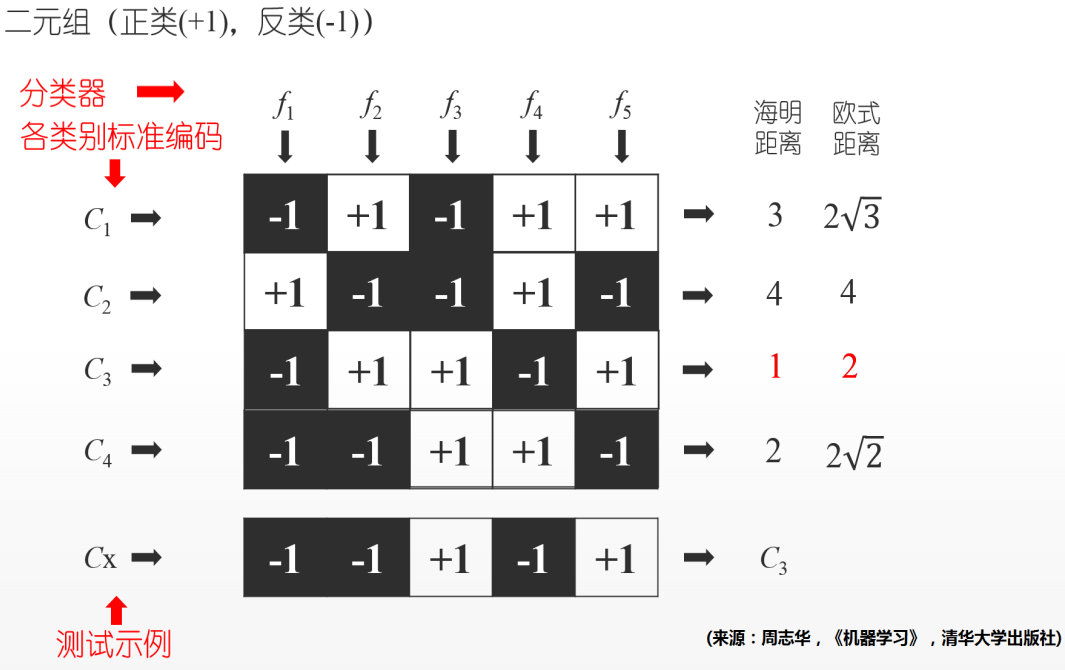
MvM 策略：每次将若干个类作为正类，若干个其他类作为反类。OvO 和 OvR 可看做是 MvM 的特例。MvM 的正，反类构造必须有特殊的设计，例如常见的 MvM 技术：**纠错输出编码** (Error Correcting Output Codes, ECOC)。

- **编码**：对 N 个类别做 M 次划分，每次将一部分作为正类，其余作为反类，共形成 M 个二分类训练集；产生 M 个分类器。
- **解码**：M 个分类器分别对测试样本预测，预测标记组成一个**编码**。将这个**预测编码**与每个类别各自的**标准编码**进行比较，返回**距离最小**的类别作为预测结果。

类别划分通过“编码矩阵”(coding matrix) 指定，常见的编码矩阵主要有二元码和三元码。

多分类学习 (5)

右图 ECOC 二元码矩阵：分类器 f_2 将 C_1 类和 C_3 类的训练数据作为正类，C_2 和 C_4 类的训练数据作为反类；在解码阶段，各分类器的分类结果联合起来形成了测试数据的编码，该编码与各类所对应的编码进行比较，将距离最小的编码所对应的类别作为最后输出分类。右图中，最后输出分类为 C_3。



纠错输出码：：ECOC 编码对分类器的错误有一定的容忍和修正能力。类别 C_3 标准编码：$(-1, +1, +1, -1, +1)$，因为 f_2 分类器预测出差导致错误编码：$(-1, -1, +1, -1, +1)$，但基于该错误编码仍能产生正确的最终分类结果 C_3。

一般来说，对于同一个学习任务，ECOC 编码越长，纠错能力越强，然而编码越长，对应计算、存储开销都会增大；另一方面，对有限类别数，可能的类别组合是有限的，码长超过一定范围后就失去了意义。

王邦编写 华科电信

类别不平衡问题 (1)

类别不平衡问题：(class imbalance) 指分类任务中不同类别的训练数据的数目差别很大。通常情况下，正例样本较少，负例样本较多。

> **例子：** 如 1000 个肿瘤病人中，998 个为良性，只有 2 个为恶性。那么学习方法只需返回一个永远将新样本预测为良性的分类器，就能达到 99.8% 的精度；然而这样的分类器往往没有价值。

常见处理方法：

- 欠采样(undersampling)：去除一些反例使得正、反例数目接近。**优点：**丢弃了很多反例，获得较小的数据集，时间开销小。**缺点：**若随机丢弃反例，可能丢失一些重要信息。
- 过采样(oversampling)：增加一些正例使得正、反例数目接近。**优点：**保存了原有的特征信息。**缺点：**训练集较大，时间开销大。若重复采样，易造成严重的过拟合。
- 阈值移动(threshold-moving)：

类别不平衡问题 (2)

阈值移动：直接基于原训练集进行学习，但是在用训练好的分类器进行预测时，依据正负类别比例进行判决比例缩放。

回顾线性分类器 $y = h(\boldsymbol{x}) = \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x} + b$ 对新样本 $\boldsymbol{x}$ 进行分类时，实际上是在用预测出的 y 值与一个阈值比较。例如 $y = h(\boldsymbol{x}) > \epsilon$ 时，判别为正例；否则为反例。y 实际上表达了正例的可能性，**几率** $\frac{y}{1-y}$ 则反映了正例可能性与反例可能性之比值。当所研究的问题正、反例数目大致相等时，设置阈值为 $\epsilon = 0.5$ 表明分类器认为正、反例可能性也相等；即分类器决策规则为：

若 $\frac{y}{1-y} > 1$，则判断为正例。

当训练集中正、反例数目不同时，令 m^+ 表示正例数目，m^- 表示反例个数，则**观察几率是** $\frac{m^+}{m^-}$。**通常假设训练集是真实样本的总体无偏采样，因此观测几率就代表了真实几率**。故若分类器的预测几率高于观测几率则应判为正例，即：

若 $\frac{y}{1-y} > \frac{m^+}{m^-}$，则判断为正例。

令 $\frac{y'}{1-y'} = \frac{y}{1-y} \times \frac{m^-}{m^+}$，$\epsilon' = 0.5$。即利用正负类别比例进行判决比例缩放。

王邦编写 华科电信

小结

本节主要讲解了回归模型及其应用，主要包括以下内容：

- 线性模型及其应用；
- 优化方法，包括最小二乘法，梯度下降法；
- 广义线性模型：利用函数变换方法扩展线性模型；
- 对数几率模型及其应用：Sigmoid 函数；
- 多分类学习问题：包括 OvO, OvR, MvM 等。
- 类别不平衡问题：包括欠采样，过采样和阈值移动方法等。

华科电信 王邦编写