线性回归
Linear Regression






目录(CONTENT)

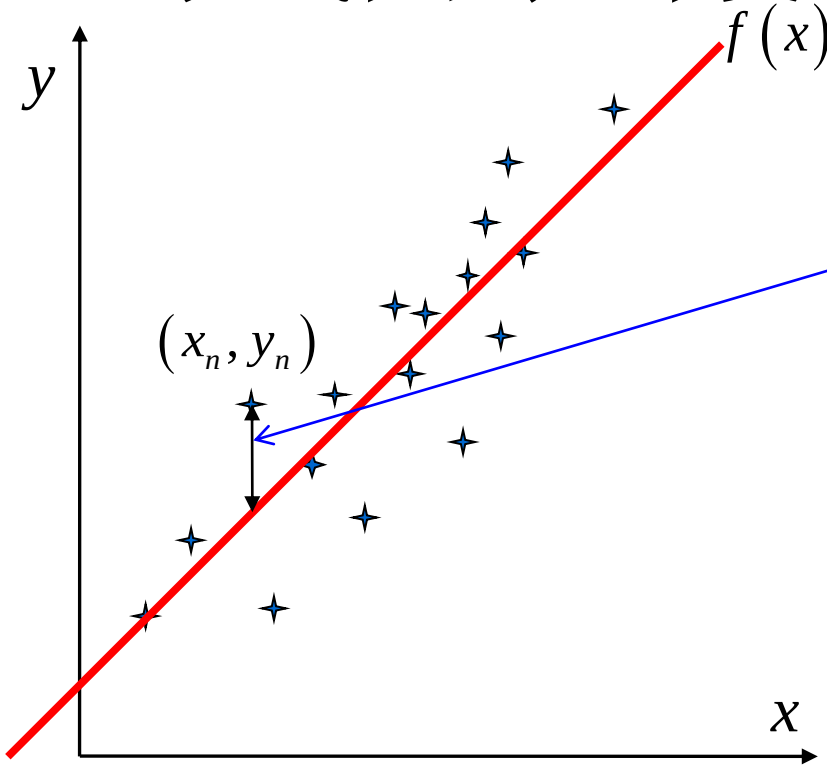
大连理工大学 Dalian University Of Technology

线性回归

Linear Regression

线性回归-->一元线性回归

■ 一元线性回归/单变量线性回归



$$y_1 \approx wx_1 + b$$

$$y_2 \approx wx_2 + b$$

$$\ldots$$

$$y_N \approx wx_N + b$$

- 回归误差：

$$e_n = y_n - (wx_n + b)$$

- 均方和误差：

$$\sum_{n=1}^{N} e_n^2 = \sum_{n=1}^{N} (y_n - wx_n - b)^2$$

  - ✓ 欧式距离
  - ✓ 能量...

- 最小二乘法

(Least Square Method)

$$(w^*, b^*) = \arg\min_{w,b} \sum_{n=1}^{N} (y_n - wx_n - b)^2$$

线性回归-->多元线性回归

大连理工大学 Dalian University Of Technology

■ 多元线性回归 (Multivariate Linear Regression)

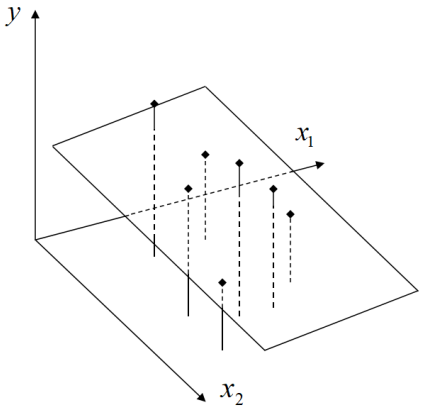
$$y_1 \approx x_1 w + b \qquad y_1 \approx x_{1,1}w_1 + x_{1,2}w_2 + \ldots + x_{1,D}w_D + b$$

$$y_2 \approx x_2 w + b \quad \Rightarrow \quad y_2 \approx x_{2,1}w_1 + x_{2,2}w_2 + \ldots + x_{2,D}w_D + b$$

一元到多元

$$\ldots \qquad \ldots$$

$$y_N \approx x_N w + b \qquad y_N \approx x_{N,1}w_1 + x_{N,2}w_2 + \ldots + x_{N,D}w_D + b$$



一般形式

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2$$

多元最小二乘法

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{\mathbf{y}} \approx \underbrace{\begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,D} & 1 \\ x_{2,1} & x_{2,2} & \cdots & x_{2,D} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,D} & 1 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_D \\ b \end{bmatrix}}_{\mathbf{w}}$$

线性回归-->最小二乘法故事

大连理工大学 Dalian University Of Technology

1801年，意大利天文学家朱赛普·皮亚齐发现了第一颗小行星谷神星。经过40天的跟踪观测后，由于谷神星运行至太阳背后，使得皮亚齐失去了谷神星的位置。随后全世界的科学家利用皮亚齐的观测数据开始寻找谷神星，但是根据大多数人计算的结果来寻找谷神星都没有结果。时年24岁的高斯也计算了谷神星的轨道。奥地利天文学家海因里希·奥尔伯斯根据高斯计算出来的轨道重新发现了谷神星。

高斯使用的最小二乘法的方法发表于1809年他的著作《天体运动论》中。

法国科学家勒让德于1806年独立发明“最小二乘法”，但因不为世人所知而默默无闻。

勒让德曾与高斯为谁最早创立最小二乘法原理发生争执。

线性回归-->问题求解/优化

大连理工大学 Dalian University Of Technology

■ 问题求解/优化

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2$$

- **闭合解/解析解 (Closed-form Solution/ Analytic Solution):**

$$J(\mathbf{w}) = \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 = (\mathbf{y} - \mathbf{X}\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w})$$

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = 2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) = 2\mathbf{X}^T\mathbf{X}\mathbf{w} - 2\mathbf{X}^T\mathbf{y} = \mathbf{0}$$

$$\mathbf{w}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \quad \Rightarrow \quad \mathbf{w}^* = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

✓ 正则化
✓ 岭回归

- **梯度下降法 (Gradient Descent):**

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = 2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \quad \Rightarrow \quad \mathbf{w} \leftarrow \mathbf{w} - \eta\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}}$$

https://www.bilibili.com/video/av41473299?from=search&seid=8346799062692305739

线性回归-->最小二乘几何解释

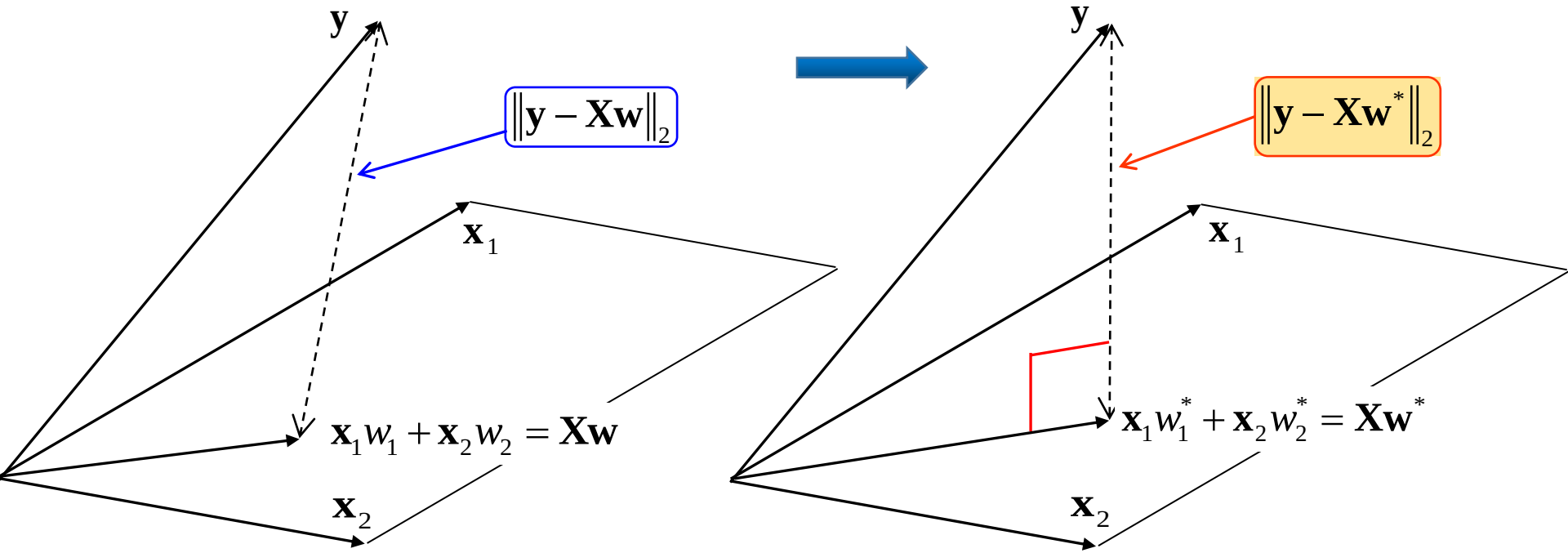
大连理工大学 Dalian University Of Technology

■ 子空间X上的正交投影

$$\mathbf{e}=\mathbf{y}-\mathbf{X}\mathbf{w}=\begin{bmatrix} y_1 \\ : \\ y_N \end{bmatrix}-\left(\begin{bmatrix} x_{1,1} \\ \dots \\ x_{N,1} \end{bmatrix}w_1+\begin{bmatrix} x_{1,2} \\ \dots \\ x_{N,2} \end{bmatrix}w_2+\dots+\begin{bmatrix} x_{1,D} \\ \dots \\ x_{N,D} \end{bmatrix}w_D\right)$$

$$=\mathbf{y}-\left(\mathbf{x}_1 w_1+\mathbf{x}_2 w_2+\dots+\mathbf{x}_d w_D\right)$$

$$\mathbf{w}^*=\arg\min_{\mathbf{w}}\left\|\mathbf{y}-\mathbf{X}\mathbf{w}\right\|_2^2$$

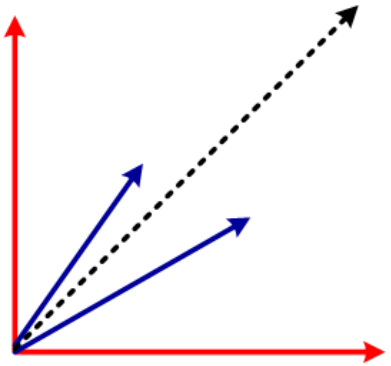


线性回归-->最小二乘几何解释

大连理工大学 Dalian University Of Technology

■ 子空间

$$\underbrace{\begin{bmatrix}1\\0\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix},\begin{bmatrix}0\\0\\1\end{bmatrix}}_{\text{完备正交基}} \rightarrow \underbrace{\begin{bmatrix}1\\1\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix},\begin{bmatrix}1\\0\\1\end{bmatrix}}_{\text{完备基}}$$

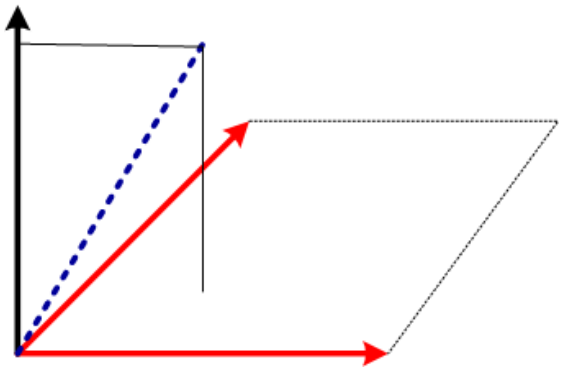


列秩（rank）

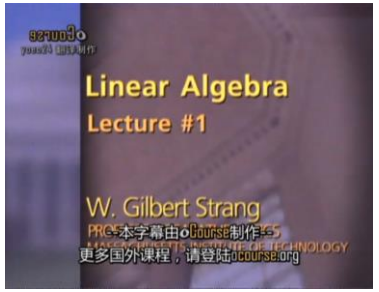
子空间

正交补空间

$$\underbrace{\begin{bmatrix}1\\0\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix},\begin{bmatrix}0\\0\\1\end{bmatrix}}_{\mathbf{a}_1x_1+\mathbf{a}_2x_2+\mathbf{a}_3x_3} \rightarrow \underbrace{\begin{bmatrix}1\\0\\0\end{bmatrix},\begin{bmatrix}0\\1\\0\end{bmatrix}}_{\mathbf{a}_1x_1+\mathbf{a}_2x_2} \rightarrow \begin{bmatrix}0\\0\\1\end{bmatrix}$$



MIT公开课《线性代数》：
https://www.bilibili.com/video/av24368594



线性回归-->最小二乘概率解释

大连理工大学 Dalian University Of Technology

高斯噪声假设

$$\mathbf{e}=\mathbf{y}-\mathbf{X}\mathbf{w} \quad \Rightarrow \quad \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} - \begin{bmatrix} x_{1,1}w_1 + x_{1,2}w_2 + \ldots + x_{1,D}w_D \\ \vdots \\ x_{N,1}w_1 + x_{N,2}w_2 + \ldots + x_{N,D}w_D \end{bmatrix}$$

- 零均值等方差高斯噪声假设

$$e_n \sim \mathcal{N}\left(0,\sigma^2\right) \Rightarrow p\left(e_n\right)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{e_n^2}{2\sigma^2}\right)$$

- 最大似然估计

$$p\left(e_1,e_2,\ldots,e_N\right)=\prod_{n=1}^{N}p\left(e_n\right)=\left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left(-\sum_{n=1}^{N}e_n^2 \Big/ 2\sigma^2\right)$$

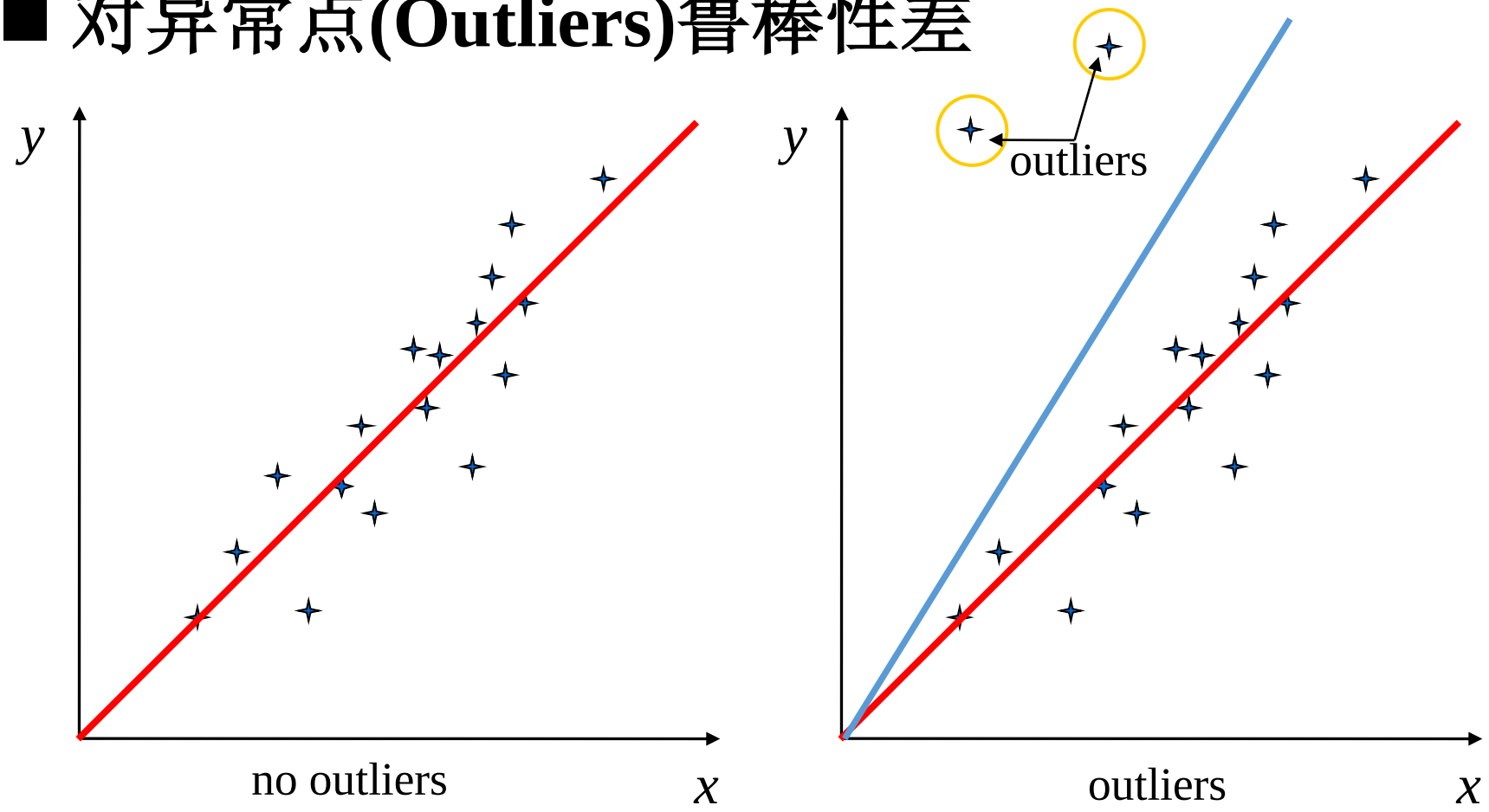
- 概率解释

$$\max_{\mathbf{w}} p\left(e_1,e_2,\ldots,e_N;\mathbf{w}\right) \Leftrightarrow \min_{\mathbf{w}} \sum_{n=1}^{N} e_n^2 \Leftrightarrow \min_{\mathbf{w}} \left\|\mathbf{y}-\mathbf{X}\mathbf{w}\right\|_2^2$$

线性回归-->最小二乘鲁棒性

大连理工大学 Dalian University Of Technology

■ 对异常点(Outliers)鲁棒性差



- 随机取样一致 (Random Sample Consensus, RANSAC)
- 鲁棒回归 (Robust Regression)

线性回归-->建模非线性

大连理工大学 Dalian University Of Technology

■ 线性的含义

$$y = wx + b$$ 线性？ 

$$y = w_1 x + w_2 x^2 + w_3 x^3 + b$$ 线性？ 

$$x_1 = x,\ x_2 = x^2,\ x_3 = x^3$$

$$y = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$$ 线性？ 

- 线性并不指对输入变量的线性，而是指对参数空间的线性。也就说对于输入来说，完全可以先对其进行非线性变换，再进行线性组合。从这个角度来说，线性模型完全具有描述非线性的能力。
- 通用非线性化方法：核学习方法 **(Kernel-based Learning Algorithms)**

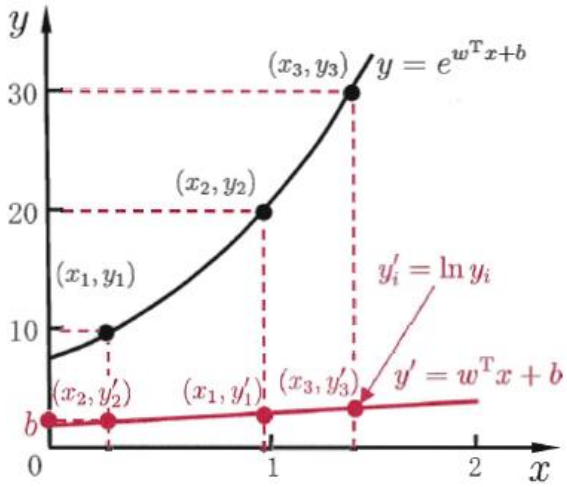
线性回归-->广义线性模型

■ 广义线性模型 (Generalized Linear Model)

- 对数线性回归 (Log-Linear Regression)：

$$y = \exp(wx + b) \qquad \log y = wx + b$$

形式上仍是线性回归，实质上已经是求取输入空间到输出空间的非线性函数映射



摘自《机器学习》，清华大学出版社，周志华著，P56图3.1

- 广义线性模型 (Generalized Linear Model)：

$$y = g^{-1}(\mathbf{w}^T\mathbf{x}) \qquad y' = g(y) = \mathbf{w}^T\mathbf{x}$$

单调可微函数

02

PART TWO

正则化
Regularization

正则化

一般的**目标函数**包含两部分：

1. 数据项 (Data Term):

回归/分类的目标，比如：误差尽可能小；

分类尽可能准确等。

2. 正则化项 (Regularization Term):

对参数空间的限制/对解额外属性的追求（比如稀疏）

数据项　　正则化项

$$O(\mathbf{x}) = D(\mathbf{x}) + \lambda R(\mathbf{x})$$

$$D(\mathbf{x}) \geq 0, R(\mathbf{x}) \geq 0$$

正则化

大连理工大学 Dalian University Of Technology

■ 岭回归 (Ridge Regression):

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_2^2$$

$$Data\ Term: \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2$$

$$Regularization\ Term: \|\mathbf{x}\|_2^2$$

$$trade-off: \lambda$$

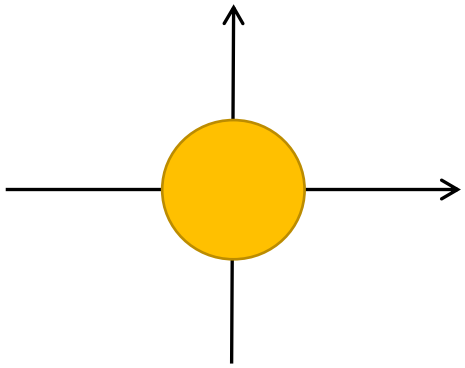
➢ 正则化是对解空间的一种限制

$$\min_{\mathbf{x}} D(\mathbf{x}) + \lambda R(\mathbf{x})$$

$$\Leftrightarrow$$

$$\min_{\mathbf{x}} D(\mathbf{x})$$

$$s.t.\ R(\mathbf{x}) \le M$$

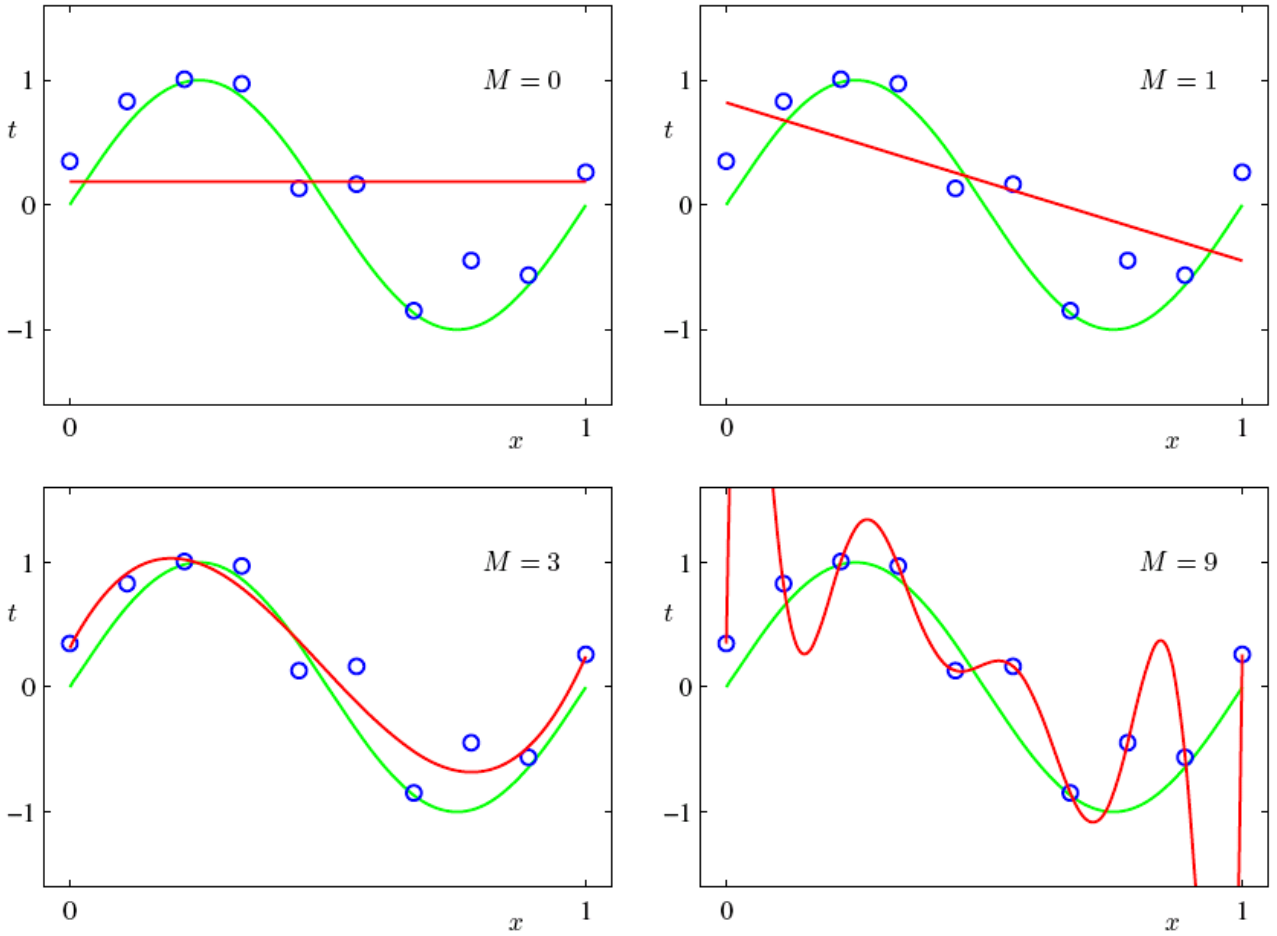


正则化

■ 举例

最小二乘模型：$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2$

$y = w_0 + w_1 x + \ldots + w_M x^M$；　　M ??−−*hyperparameter*超参数



正则化

- 举例

Table 1.1 Table of the coefficients $\mathbf{w}^\star$ for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M=0$	$M=1$	$M=6$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

正则化

大连理工大学 Dalian University Of Technology

■ 举例

➢增大样本可以避免过拟合，但是。。。

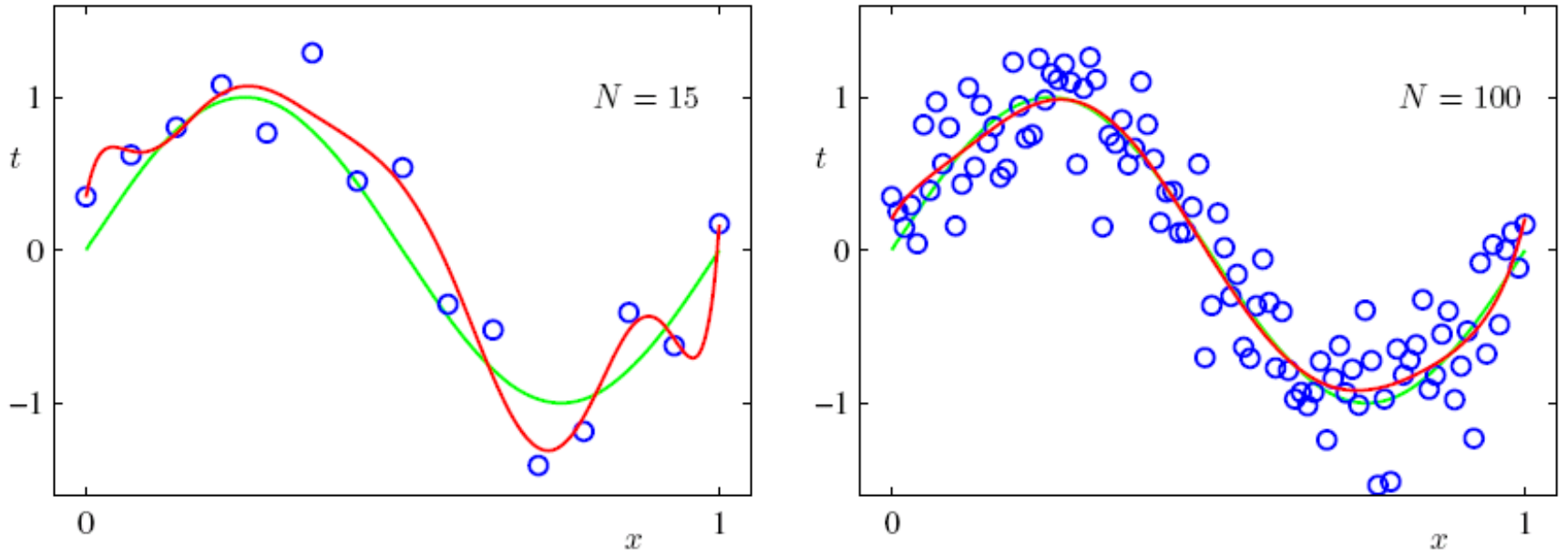


Figure 1.6 Plots of the solutions obtained by minimizing the sum-of-squares error function using the $M = 9$ polynomial for $N = 15$ data points (left plot) and $N = 100$ data points (right plot). We see that increasing the size of the data set reduces the over-fitting problem.

正则化

大连理工大学 Dalian University Of Technology

■ 举例

岭回归：$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \frac{1}{2}\lambda\|\mathbf{x}\|_2^2$

➢正则化，控制模型的复杂度

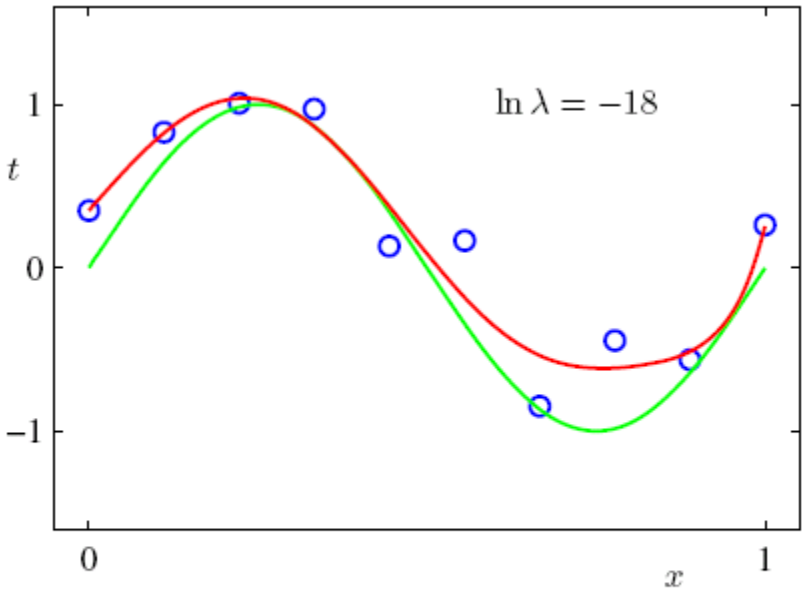


Figure 1.7 Plots of $M = 9$ polynomials fitted to the da function (1.4) for two values of the regularization param case of no regularizer, i.e., $\lambda = 0$, corresponding to $\ln \lambda =$

	$\ln\lambda = -\infty$	$\ln\lambda = -18$	$\ln\lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

正则化

大连理工大学 Dalian University Of Technology

- 不同正则化追求解的不同属性

addition, the second term is a regularization term. Different kinds of regularization lead to different properties of the coding coefficients (ℓ_1-regularization (2(a)) encourages sparsity [29], $\ell_{2,1}$-regularization (2(b)) promotes group sparsity, and ℓ_2-regularization (2(c)) does not lead to sparsity but makes the solution simple and stable [35]). After obtaining the coding

2. Code a test sample $\mathbf{y}$ by a linear representation (2(a), 2(b) or 2(c)),

$$2(a)\ \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2+\lambda\|\mathbf{x}\|_1 \quad \text{(SR [29])}$$

$$2(b)\ \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2+\lambda\|\mathbf{x}\|_{2,1} \quad \text{(GSR [17])}$$

$$2(c)\ \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2+\lambda\|\mathbf{x}\|_2^2 \quad \text{(CR [35])}$$

to obtain the corresponding coding coefficients.

- Dong Wang, Huchuan Lu, Minghsuan Yang, Kernel Collaborative Face Recognition, Pattern Recognition, 48(10):3025-3037, 2015.

正则化

- 正则化项可以看作参数的先验 (从贝叶斯估计角度)

观测数据： $\mathbf{y}$

参数/系数： $\mathbf{x}$

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{x},\mathbf{y})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\mathbf{x})\,p(\mathbf{x})}{p(\mathbf{y})} \propto p(\mathbf{y}|\mathbf{x})\,p(\mathbf{x})$$

Posteriori *Likelihood* *prior*

(1) *MAP (Maximum a Posteriori)*:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg\max_{\mathbf{x}} p(\mathbf{x},\mathbf{y}) = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})\,p(\mathbf{x})$$

(2) *MLE (Maximum Likelihood Estimation)*:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})$$

$ps: p(\mathbf{x}) = const$, *uniform prior*

The coefficient x can be obtained by maximizing the posteriori probability $p(\mathbf{x}|\mathbf{y})$, which is also equivalent to maximizing the joint likelihood probability $p(\mathbf{x},\mathbf{y})$. Assume there is a uniform prior, the coefficient x is estimated by $\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}) = \arg\max_{\mathbf{x}} p(\mathbf{e})$, which is the maximum likelihood estimation (MLE).

正则化

- MLE(Maximum Likelihood Estimation)与OLS(Ordinary Least-Squares):

$$\mathbf{y} = \mathbf{A}\mathbf{x}$$

$$error / residual : \mathbf{e} = \mathbf{y} - \mathbf{A}\mathbf{x}$$

$$Assume\ that\ \mathbf{e} = [e_1, e_2, ..., e_N]\ follow\ i.i.d\ Gaussian\ distribution\ N(0, \sigma^2)$$

$$\mathbf{x} = [x_1, x_2, ..., x_d]\ follow\ \underline{some}\ uniform\ prior.$$

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x})\cancel{p(\mathbf{x})}$$

$$= p(\mathbf{e}) = \prod_{n=1}^{N} p(e_n)$$

$$= \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{e_n^2}{2\sigma^2}\right]$$

$$= \left\{\prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{\sigma^2}\right)\right\} \exp\left(-\frac{1}{2}\|\mathbf{e}\|_2^2\right)$$

$$\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) \Leftrightarrow \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2$$

正则化

- MAP(Maximum A Posterior)与RR(Ridge Regression):

$\mathbf{y} = \mathbf{Ax}$ *error / residual* : $\mathbf{e} = \mathbf{y} - \mathbf{Ax}$

Assume that $\mathbf{e} = [e_1, e_2, ..., e_N]$ *follow i.i.d Gaussian distribution* $N(0, \sigma^2)$

$\mathbf{x} = [x_1, x_2, ..., x_D]$ *also follow i.i.d Gaussian distribution* $N(0, \sigma_1^2)$

$$p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x})\,p(\mathbf{x}) = p(\mathbf{e})\,p(\mathbf{x}) = \left[\prod_{n=1}^{N} p(e_n)\right]\left[\prod_{d=1}^{D} p(x_d)\right]$$

$$= \left\{\prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{e_n^2}{2\sigma^2}\right]\right\}\left\{\prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left[-\frac{x_d^2}{2\sigma_1^2}\right]\right\}$$

$$\propto \left\{\prod_{n=1}^{N} \exp\left[-\frac{e_n^2}{2\sigma^2}\right]\right\}\left\{\prod_{d=1}^{D} \exp\left[-\frac{x_d^2}{2\sigma_1^2}\right]\right\}$$

$$= \exp\left(-\frac{\|\mathbf{e}\|_2^2}{2\sigma^2}\right)\exp\left(-\frac{\|\mathbf{x}\|_2^2}{2\sigma_1^2}\right) = \exp\left(\frac{1}{\sigma^2}\right)\exp\left[-\frac{\|\mathbf{e}\|_2^2}{2} - \frac{\sigma^2}{2\sigma_1^2}\|\mathbf{x}\|_2^2\right]$$

$$\propto \exp\left[-\frac{1}{2}\|\mathbf{y} - \mathbf{Ax}\|_2^2 - \frac{\lambda}{2}\|\mathbf{x}\|_2^2\right] \qquad \left(ps: \lambda = \frac{\sigma^2}{\sigma_1^2}\right)$$

$$\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) \Leftrightarrow \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{Ax}\|_2^2 + \frac{\lambda}{2}\|\mathbf{x}\|_2^2$$

正则化

大连理工大学 Dalian University Of Technology

- Ordinary Least-Squares v.s. Ridge Regression:

1.*Objective Function*:

$$OLS: \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2$$

$$RR: \min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \frac{\lambda}{2}\|\mathbf{x}\|_2^2$$

2.*Solution*:

$$OLS: \mathbf{x} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{y}$$

$$RR: \mathbf{x} = \left(\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I}\right)^{-1}\mathbf{A}^T\mathbf{y}$$

3.*Probability*:

$OLS: MLE$

$RR: MAP$

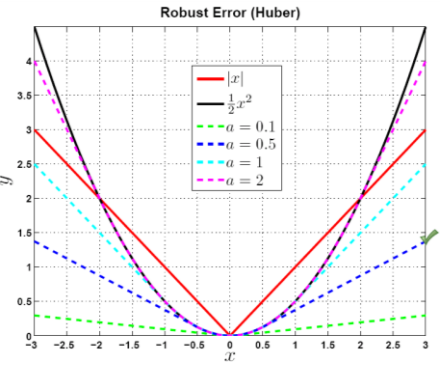
- 吴恩达《机器学习》-正则化 https://www.bilibili.com/video/av55276229
- L1&L2正则化详解 https://www.bilibili.com/video/av77106463?from=search&seid=4369320229005019988
- 什么是L1 L2正则化? https://www.bilibili.com/video/av16009446?from=search&seid=4369320229005019988

线性回归归纳

大连理工大学 Dalian University Of Technology

$$\mathbf{e} = \mathbf{y} - \mathbf{X}\mathbf{w} \quad \Rightarrow \quad \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} - \begin{bmatrix} x_{1,1}w_1 + x_{1,2}w_2 + \ldots + x_{1,D}w_D \\ \vdots \\ x_{N,1}w_1 + x_{N,2}w_2 + \ldots + x_{N,D}w_D \end{bmatrix}$$

$$\rho(\mathbf{y} - \mathbf{X}\mathbf{w}) + \lambda R(\mathbf{w})$$



$$\frac{1}{2}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2$$

$$\|\mathbf{y} - \mathbf{X}\mathbf{w}\|_1$$

Huber

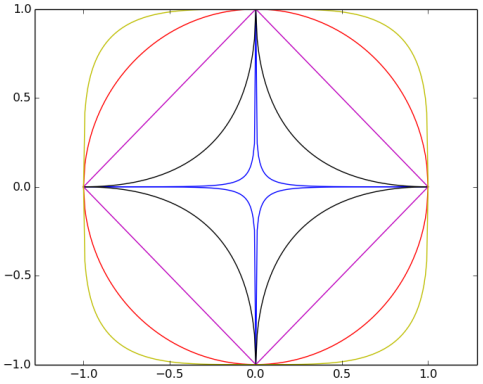
$$\vdots$$

$$\frac{1}{2}\|\mathbf{w}\|_2^2$$

$$\|\mathbf{w}\|_1$$

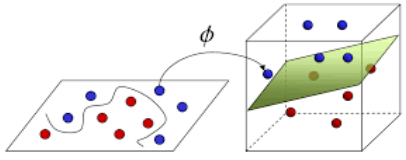
$$\|\mathbf{w}\|_p := \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

$$\vdots$$



$$\mathbf{y} \rightarrow \phi(\mathbf{y})$$

$$\mathbf{X} \rightarrow \phi(\mathbf{X})$$



逻辑斯蒂回归

Logistic Regression

逻辑斯蒂回归-->引言

大连理工大学 Dalian University Of Technology

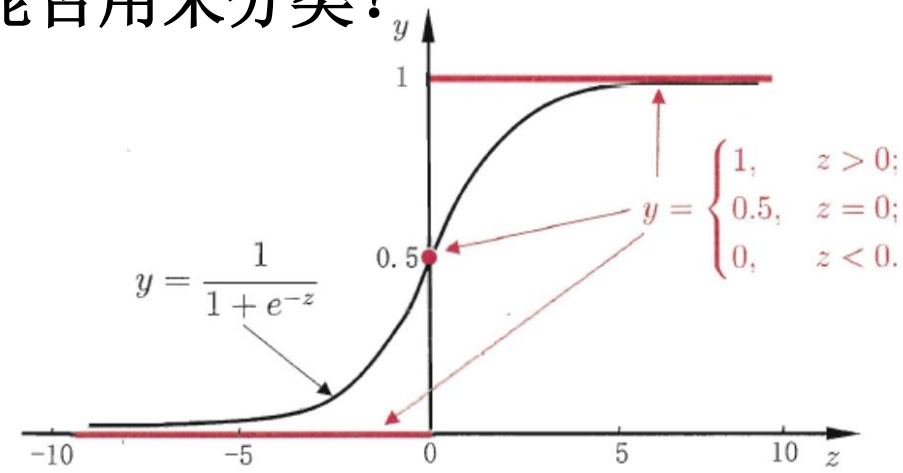
■ 逻辑斯蒂回归 (Logistic Regression)

- 线性回归的任务是预测，能否用来分类？

$$z = \mathbf{w}^T \mathbf{x}$$

$$y = \begin{cases} 0, & z < 0 \\ 0.5, & z = 0 \\ 1, & z > 0 \end{cases}$$

$$y = \frac{1}{1+e^{-z}}$$



逻辑斯蒂函数

- 广义线性回归特列

$$\log \frac{y}{1-y} = \mathbf{w}^T \mathbf{x}$$

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

对数几率

《机器学习》，清华大学出版社，周志华著，**P58**

逻辑斯蒂回归-->概率解释

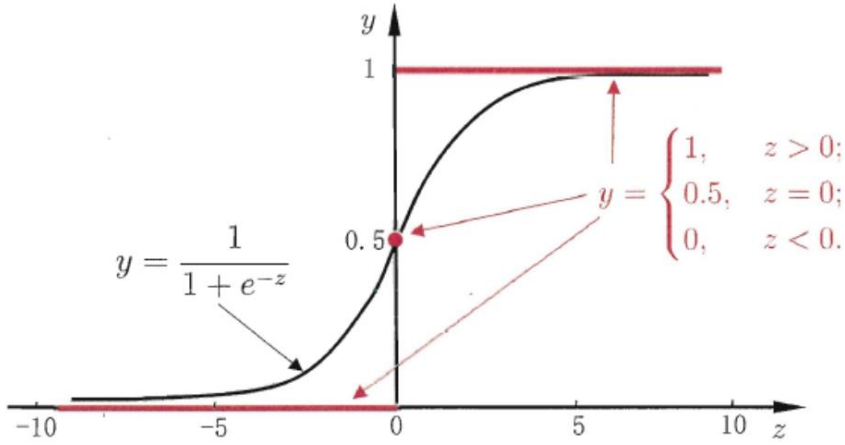
大连理工大学 Dalian University Of Technology

■ 概率解释

➢ 分类概率 $y=\dfrac{1}{1+\exp\left(-\mathbf{w}^T\mathbf{x}\right)}$

$$p_1\left(\mathbf{x};\mathbf{w}\right)=p\left(y=1\middle|\mathbf{x};\mathbf{w}\right)=\frac{\exp\left(\mathbf{w}^T\mathbf{x}\right)}{1+\exp\left(\mathbf{w}^T\mathbf{x}\right)}$$

$$p_0\left(\mathbf{x};\mathbf{w}\right)=p\left(y=0\middle|\mathbf{x};\mathbf{w}\right)=\frac{1}{1+\exp\left(\mathbf{w}^T\mathbf{x}\right)}$$



➢ 贝叶斯决策

$$\log\frac{p\left(y=1\middle|\mathbf{x};\mathbf{w}\right)}{p\left(y=0\middle|\mathbf{x};\mathbf{w}\right)}=\mathbf{w}^T\mathbf{x}\quad\Rightarrow\begin{cases}\mathbf{w}^T\mathbf{x}>0 & \rightarrow\quad y=1\\ \mathbf{w}^T\mathbf{x}<0 & \rightarrow\quad y=0\end{cases}$$

➢ 判决模型

- ✓ 从生成模型和判决模型角度，逻辑斯蒂回归算法属于判决模型；
- ✓ 直接对分类的可能性进行建模，无需事先假设数据分布，避免数据分布假设不准确所带来的问题；
- ✓ 并且逻辑斯蒂回归不仅输出分类类别，而且能得到概率近似。

逻辑斯蒂回归-->极大似然推导

大连理工大学 Dalian University Of Technology

■ 极大似然推导

➢ 伯努利分布

y_n	1	0
p_n	$p^1(\mathbf{x}_n;\mathbf{w})$	$p^0(\mathbf{x}_n;\mathbf{w})$

$$p^1(\mathbf{x}_n;\mathbf{w})+p^0(\mathbf{x}_n;\mathbf{w})=1$$

➢ 最大似然估计 **(i.i.d, independent and identically distributed)**

$$p(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N;\mathbf{w})=\prod_{n=1}^{N}\left[p^1(\mathbf{x}_n;\mathbf{w})\right]^{y_n}\left[p^0(\mathbf{x}_n;\mathbf{w})\right]^{1-y_n}$$

$$\max_{\mathbf{w}} p(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N;\mathbf{w}) \Leftrightarrow \min_{\mathbf{w}}\left[-\log p(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N;\mathbf{w})\right]$$

➢ 目标函数

$$L(\mathbf{w})=-\log p(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_N;\mathbf{w})$$

$$=-\sum_{n=1}^{N}\left[y_n\log p^1(\mathbf{x}_n;\mathbf{w})+(1-y_n)\log p^0(\mathbf{x}_n;\mathbf{w})\right]$$

$$=\sum_{n=1}^{N}\left[-y_n\mathbf{w}^T\mathbf{x}_n+\log\left(1+\exp\left(\mathbf{w}^T\mathbf{x}_n\right)\right)\right]$$

逻辑斯蒂回归-->问题求解/优化

■ 问题求解/优化

➢ 优化问题

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} L(\mathbf{w})$$

$$L(\mathbf{w}) = \sum_{n=1}^{N}\left[-y_n \mathbf{w}^T \mathbf{x}_n + \log\left(1+\exp\left(\mathbf{w}^T \mathbf{x}_n\right)\right)\right]$$

➢ 一阶和二阶导数

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} = -\sum_{n=1}^{N}\left(y_n - \frac{\exp\left(\mathbf{w}^T \mathbf{x}_n\right)}{1+\exp\left(\mathbf{w}^T \mathbf{x}_n\right)}\right)\mathbf{x}_n = -\sum_{n=1}^{N}\left[y_n - p^1\left(\mathbf{x}_n;\mathbf{w}\right)\right]\mathbf{x}_n$$

$$\frac{\partial^2 L(\mathbf{w})}{\partial \mathbf{w}\partial \mathbf{w}^T} = \sum_{n=1}^{N}\frac{\exp\left(\mathbf{w}^T \mathbf{x}_n\right)}{\left(1+\exp\left(\mathbf{w}^T \mathbf{x}_n\right)\right)^2}\mathbf{x}_n\mathbf{x}_n^T = \sum_{n=1}^{N} p^1\left(\mathbf{x}_n;\mathbf{w}\right)p^0\left(\mathbf{x}_n;\mathbf{w}\right)\mathbf{x}_n\mathbf{x}_n^T$$

➢ 迭代求解

$$\mathbf{w} \leftarrow \mathbf{w} - \eta\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}}$$

梯度下降法

Gradient Descent Method

$$\mathbf{w} \leftarrow \mathbf{w} - \left(\frac{\partial^2 L(\mathbf{w})}{\partial \mathbf{w}\partial \mathbf{w}^T}\right)^{-1}\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}}$$

牛顿法

Newton Method

逻辑斯蒂回归-->多类分类

大连理工大学 Dalian University Of Technology

■ 多类分类

➢ 多元逻辑回归 (Multi-nomial Logistic Regression)

y_n	1	2	···	C
p_n	$p^1(\mathbf{x}_n;\mathbf{w}_1)$	$p^2(\mathbf{x}_n;\mathbf{w}_2)$	···	$p^C(\mathbf{x}_n;\mathbf{w}_C)$

$$p^c(\mathbf{x}_n;\mathbf{w}_c)=\frac{\exp\left(\mathbf{w}_c^T\mathbf{x}_n\right)}{1+\sum_{k=1}^{C-1}\exp\left(\mathbf{w}_k^T\mathbf{x}_n\right)},\ c=1,2,...,C-1;\quad p^C(\mathbf{x}_n;\mathbf{w}_C)=\frac{1}{1+\sum_{k=1}^{C-1}\exp\left(\mathbf{w}_k^T\mathbf{x}_n\right)}$$

➢ Softmax回归

$$p^c(\mathbf{x}_n;\mathbf{w}_c)=\frac{\exp\left(\mathbf{w}_c^T\mathbf{x}_n\right)}{\sum_{k=1}^{C}\exp\left(\mathbf{w}_k^T\mathbf{x}_n\right)},\ c=1,2,...,C$$

$$\mathbf{W}=\left[\mathbf{w}_1,\mathbf{w}_2,...,\mathbf{w}_C\right]$$

$$\mathbf{y}_n=\left[\delta(y_n=1),\delta(y_n=2),...,\delta(y_n=C)\right]^T$$

$$h\left(\mathbf{W}^T\mathbf{x}\right)=\frac{\exp\left(\mathbf{W}^T\mathbf{x}\right)}{\mathbf{1}^T\exp\left(\mathbf{W}^T\mathbf{x}\right)}$$

$$L(\mathbf{W})=-\frac{1}{N}\sum_{n=1}^{N}(\mathbf{y}_n)^T\log h\left(\mathbf{W}^T\mathbf{x}_n\right)$$

交叉熵损失函数
(Cross Entropy Loss)

- https://www.bilibili.com/video/av71259996

04

PART Four

多分类学习

Multiclass Classification

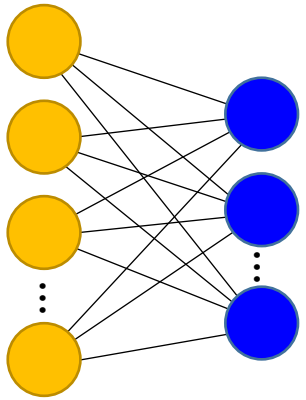
多分类学习-->引言

大连理工大学 Dalian University Of Technology

■ 二分类-->多分类

- 通常开始讨论分类问题时大多讨论两类问题（如线性回归、SVM、Adaboost等），但是实际问题往往是多类的。
- 线性回归中可以将输出由一个值（[0,1]或[-1,1]）扩展一个指示向量(Indicator Vector)来处理多类问题：

$$\mathbf{y}_n = \left[\delta(y_n = 1), \delta(y_n = 2), ..., \delta(y_n = C)\right]^T$$



■ 通用方案

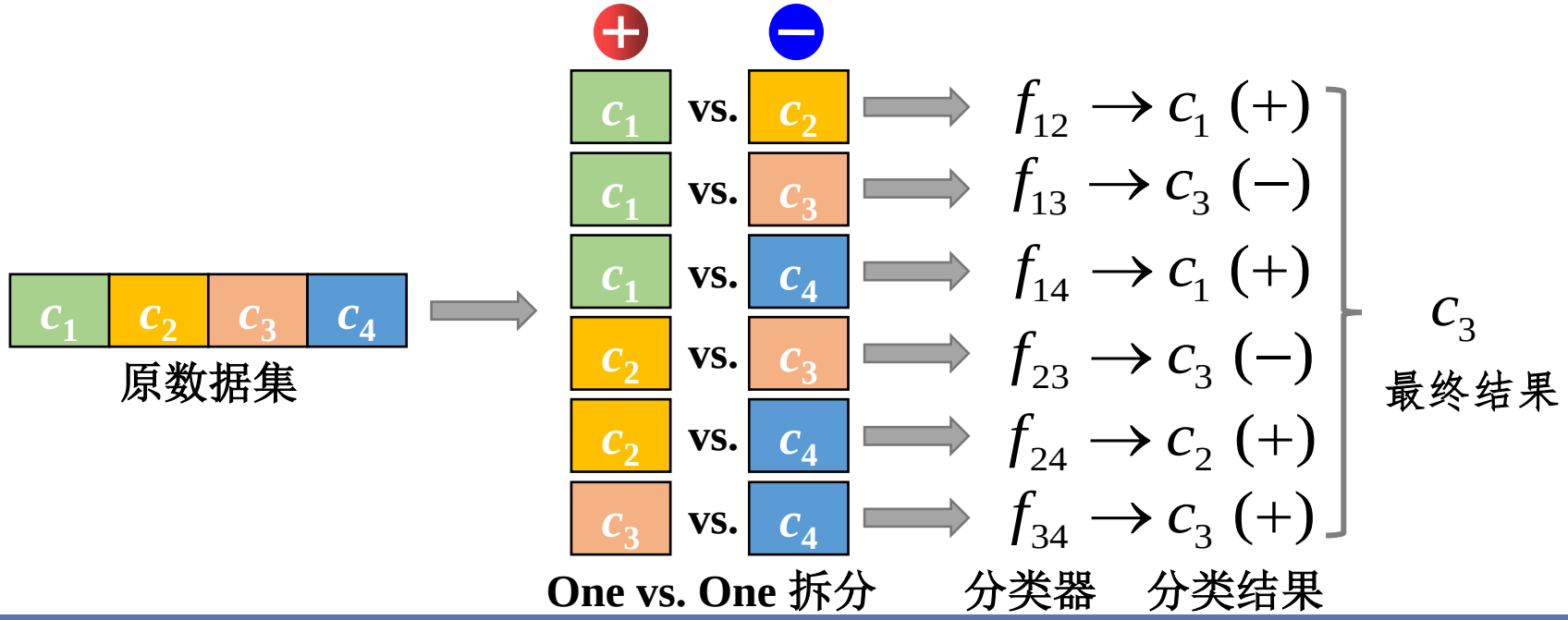
- 存在利用二分类学习器解决多分类问题的通用方案？
  - ✓ 一对一 **(One vs. One, OvO)**
  - ✓ 一对其余 **(One vs. Rest, OvR)**
  - ✓ 多对多 **(Many vs. Many, MvM)**

多分类学习-->一对一

大连理工大学 Dalian University Of Technology

■ 一对一 (One vs. One, OvO)

- 训练：将C类分类问题拆分为$C(C-1)/2$个二分类问题，训练$C(C-1)/2$个分类器 (每个分类器区分两个类别)；
- 测试：利用所有分类器对待测试样本进行分类，获得$C(C-1)/2$个分类结果，通过投票获得最终结果。

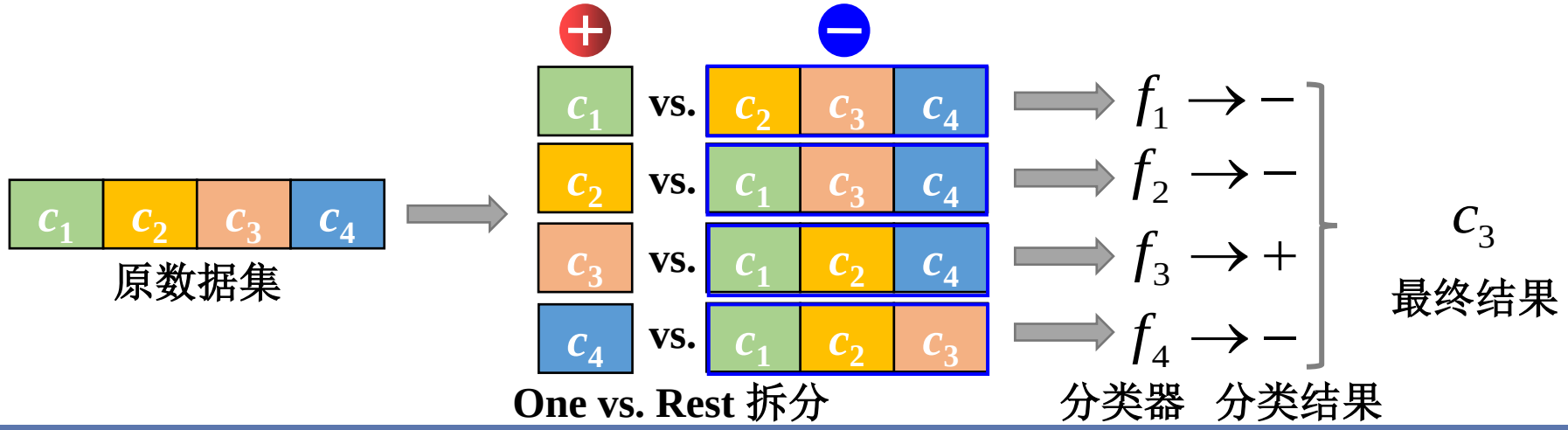


多分类学习-->一对其余/一对多

大连理工大学 Dalian University Of Technology

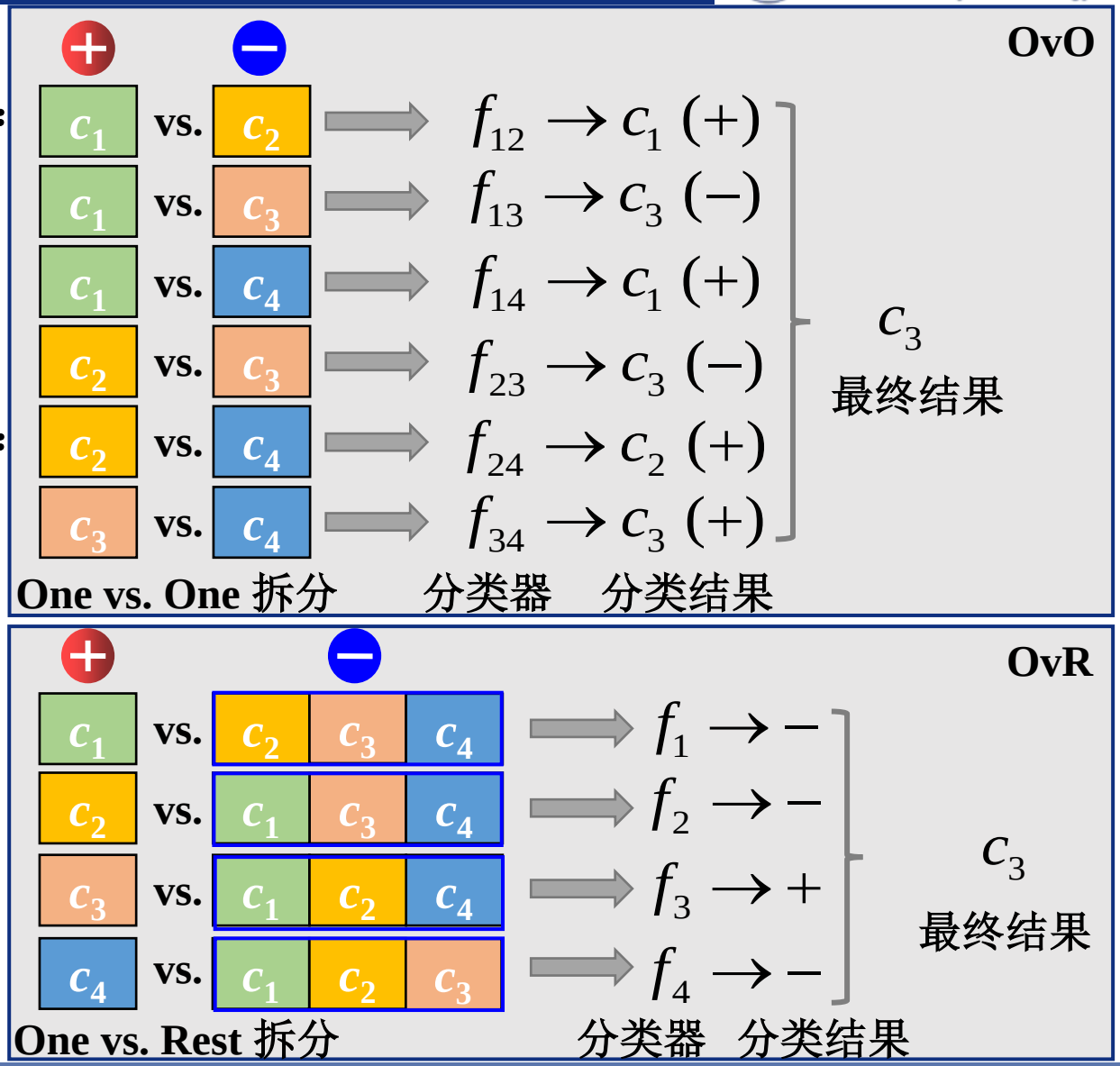
■ 一对其余 (One vs. Rest, OvR)

- 训练：将C类分类问题拆分为C个二分类问题，训练C个分类器 (每次将一类作为正类，其余作为负类)；
- 测试：利用所有分类器对待测试样本进行分类，获得C个分类结果。若结果仅有一个结果为正类，则对应的类别标记作为最终分类结果。若存在多个正类，则需要用其他方法判定 (如分类置信度)。



多分类学习-->OvO vs. OvR

- OvO的存储开销和测试时间开销通常比OvR大：OvR只需训练C个分类器，而OvO需训练$C(C-1)/2$个分类器。
- 类别多时，OvO的训练时间开销通常比OvR小：训练时，OvR的每个分类器均使用全部训练样本，而OvO的每个分类器仅用到两个类样本；
- 预测性能差不多：至于预测性能，则取决于具体的数据分布，在多数情形下两者差不多。



多分类学习-->多对多

■ 多对多 (Many vs. Many, MvM)

- 基本思想：每次将若干类作为正类，其余类作为负类。

  纠错输出编码 (Error-Correcting Output Codes, ECOC)；

- 编码：对*C*个类做*M*次划分，每次划分将一部分类划为正类，另一部分划为负类，从而形成一个二分类训练集。这样一共有*M*个训练集，可以训练出*M*个分类器；

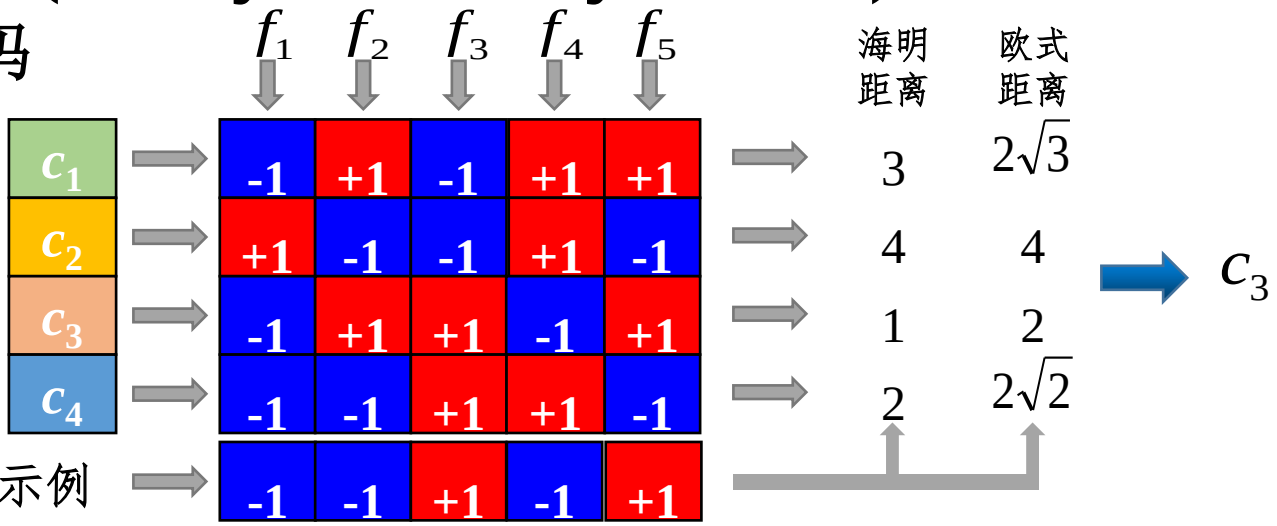
- 解码：*M*个分类器分别对测试样本进行分类，这些分类标记组成一个编码，将这个分类编码与每个类别各自的编码进行比较，返回距离最小的类作为最终预测结果。

多分类学习-->多对多

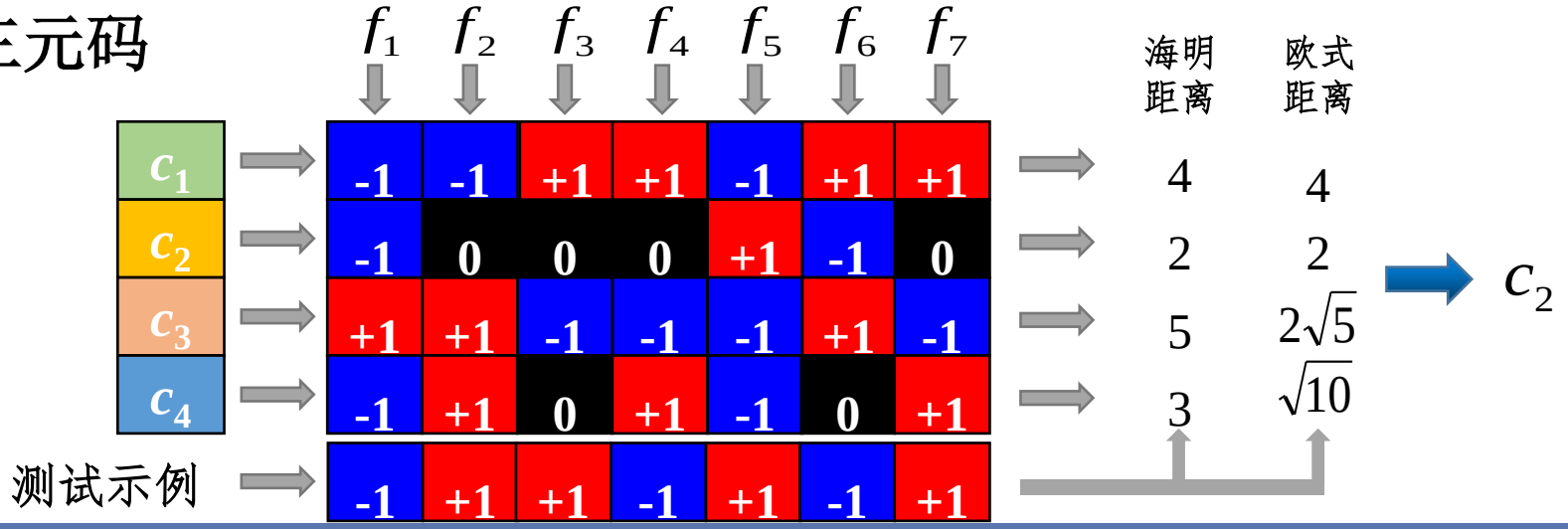
大连理工大学 Dalian University Of Technology

■ 多对多 (Many vs. Many, MvM)

➢ 二元码



➢ 三元码



05

PART FIVE

类别不平衡问题

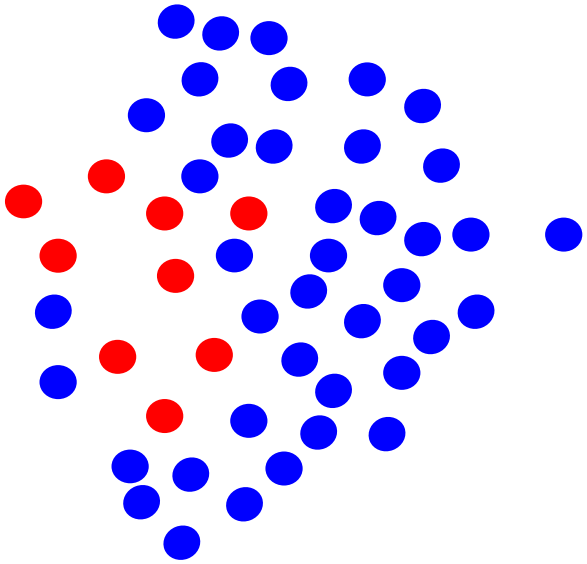
Class-Imbalance

类别不平衡问题-->引言

大连理工大学 Dalian University Of Technology

■ 类别不平衡 (Class-Imbalance)

- 分类任务中不同类别的训练样本数目差别很大的情况
  - ✓ 以二分类为例
  - ✓ 假设正样本很少，负样本很多
  - ✓ 类别严重不平衡时，分类器学习将过分关注数目多的类别



- 再缩放策略 (rescaling）
  - ✓ 阈值移动 (threshold-moving)
  - ✓ 过采样 (oversampling)
  - ✓ 欠采样 (undersampling)
- 阈值移动 (threshold-moving)

N^+ : 正样本总数, N^- : 负样本总数

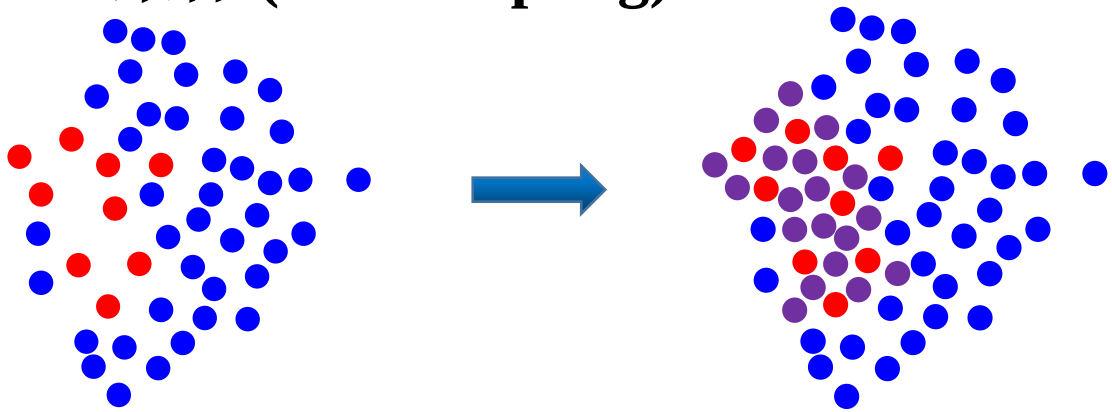
$$\frac{p}{1-p} > 1 \quad \Rightarrow \quad \frac{p}{1-p} > \frac{N^+}{N^-} \quad \Rightarrow \quad \frac{p'}{1-p'} = \frac{p}{1-p}\frac{N^-}{N^+} > 1$$

类别不平衡问题-->引言

大连理工大学 Dalian University Of Technology

■ 类别不平衡 (Class-Imbalance)

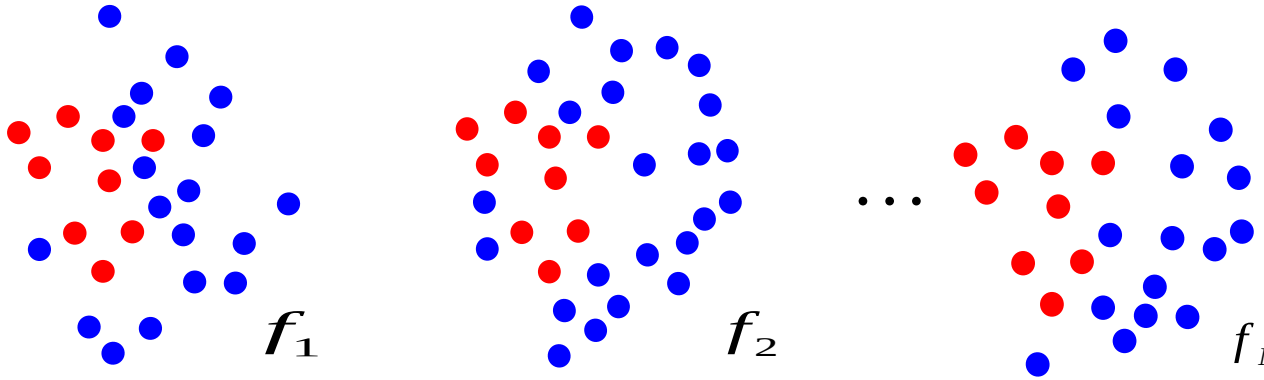
- ➢ 过采样 (oversampling):



✓ 样本复制
✓ 样本插值
✓ 样本生成 (GAN)

[1] Chawla N V, Bowyer K W, et al. **SMOTE: Synthetic Minority Over-Sampling Technique**. ***JAIR***, 2002.

- ➢ 欠采样 (undersampling)



● 集成学习

$$f = \frac{1}{M}\sum_{m=1}^{M} f_M$$

✓ EasyEnsemble[2]
✓ BalanceCascade[2]

[2] Xu-Ying Liu, Jianxin Wu, Zhi-Hua Zhou. **Exploratory Undersampling for Class-Imbalance Learning**. ***IEEE TSMCB***, 2009.

类别不平衡问题-->加权损失

大连理工大学 Dalian University Of Technology

■ 加权损失函数 (Weighted Loss Function)

- 一般损失函数

$$L(\mathbf{X};\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} L(\mathbf{x}_n, y_n; \mathbf{w}) = \sum_{n=1}^{N} \frac{1}{N} L(\mathbf{x}_n, y_n; \mathbf{w})$$

$$= \sum_{n=1}^{N^+} \frac{1}{N} L(\mathbf{x}_n, y_n; \mathbf{w}) + \sum_{n=1}^{N^-} \frac{1}{N} L(\mathbf{x}_n, y_n; \mathbf{w})$$

N：样本总数，N^+：正样本总数，N^-：负样本总数 $\left(N = N^+ + N^-\right)$

- 加权损失函数

$$L(\mathbf{X};\mathbf{w}) = \sum_{n=1}^{N} \rho_n L(\mathbf{x}_n, y_n; \mathbf{w})$$

$$= \sum_{n=1}^{N^+} \rho_n^+ L(\mathbf{x}_n, y_n; \mathbf{w}) + \sum_{n=1}^{N^-} \rho_n^- L(\mathbf{x}_n, y_n; \mathbf{w})$$

$$\rho^+ = \frac{N^-}{N}, \rho^- = \frac{N^+}{N} \quad \Rightarrow \quad \frac{\rho^+}{\rho^-} = \frac{N^-}{N^+}$$

类别不平衡问题-->Focal Loss

大连理工大学 Dalian University Of Technology

■ Focal Loss[1]

➢ 交叉熵损失

$$\mathrm{CE}(p_t) = -\log(p_t) \qquad p_t = \begin{cases} p & if\ y=1 \\ 1-p & otherwise \end{cases}$$

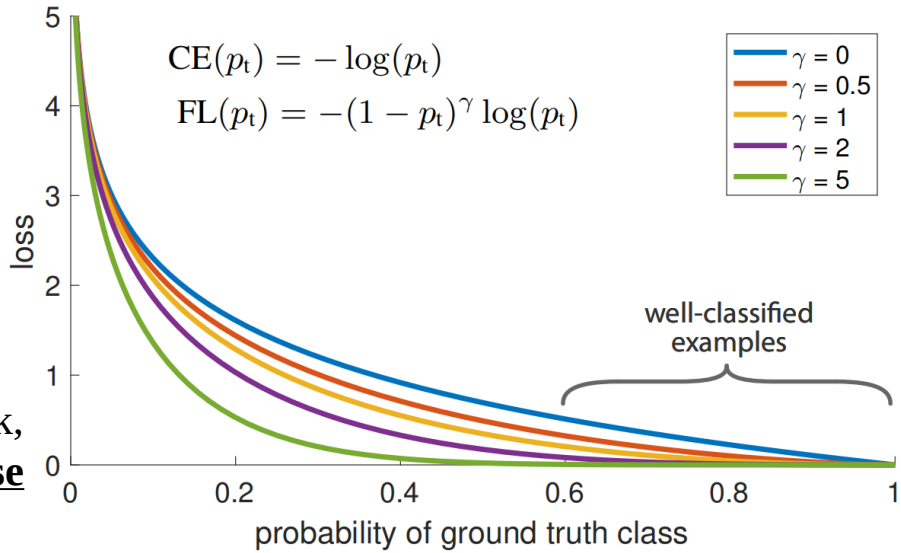
$$L = \frac{1}{N}\sum_{n=1}^{N} CE\left(p_n^t\right) = -\frac{1}{N}\sum_{n=1}^{N} y_n \log(p_n) + (1-y_n)\log(1-p_n)$$

➢ Focal Loss[1] ⟸ 解决目标检测问题中训练集正负样本极度不平衡情况

$$\mathrm{FL}(p_t) = -(1-p_t)^{\gamma}\log(p_t)$$

- ✓ 自适应样本加权
- ✓ 容易分类样本权重降低

  $p_t \to 1 \Rightarrow (1-p_t)^{\gamma} \to 0$
- ✓ 促进分类器学习更关注困难样本



[1] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollar. **Focal Loss for Dense Object Detection**, ***ICCV***, 2017.

类别不平衡问题-->级联分类器

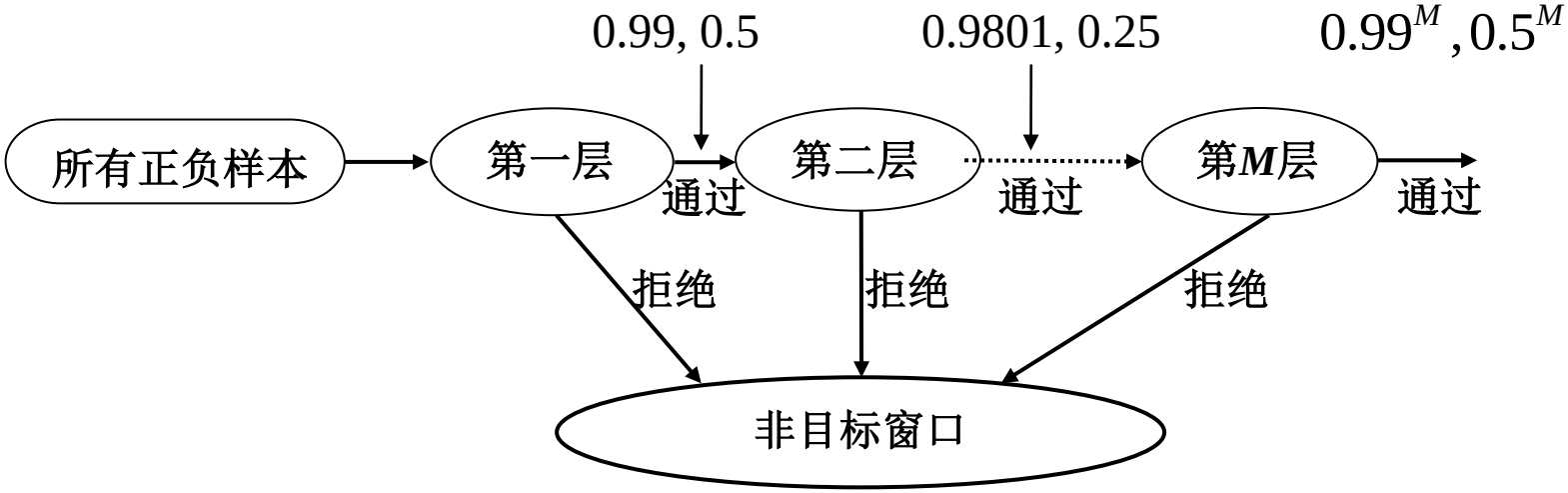
大连理工大学 Dalian University Of Technology

■ 级联分类器 (Cascaded Classifier)[1]

- 人脸检测问题
  - ✓ 正样本（人脸）
  - ✓ 负样本（非人脸）
  - ✓ 正样本远少于负样本

[1] Paul A. Viola, Michael J. Jones: **Robust Real-Time Face Detection**. ***IJCV,*** 2004.

- 设每层的正确率(TPR)为99%，误检率(FPR)为50%



- 假设M =10，则可保证在正确率达到90.4%，而误检率为0.1%

大连理工大学
Dalian University Of Technology

PART SIX

小结

Summary

小结

■小结

- ➢线性回归
  - ✓最小二乘法 (目标函数, 优化求解, 几何意义, 概率意义)
  - ✓最小二乘法存在问题 (Outliers)，线性理解，广义线性回归
- ➢正则化
  - ✓ 正则化->过拟合
- ➢逻辑斯蒂回归
  - ✓从回归到分类, 逻辑斯蒂函数, 概率解释
  - ✓目标函数, 优化求解, 多类逻辑斯蒂回归
- ➢多分类学习
  - ✓直接建立多类模型 (指示向量)
  - ✓通用策略 (一对一, 一对其余, 多对多)
- ➢类别不平衡问题
  - ✓再缩放 (阈值移动, 欠采样, 过采样)　　级联分类器
  - ✓加权损失函数 (正负样本重加权, Focal Loss)
  - ✓代价敏感损失函数 (参见最小风险贝叶斯决策)