

第3章：线性模型

1. 周志华, 《机器学习》, 清华大学出版社, 2016, P₅₄
2. 李航著, 《统计学习方法》第2版, 清华大学出版社, 2019, P₉₁
3. 安德烈-布可夫著, 《机器学习精讲》, 人民邮电出版社, 2020, (第3章)
4. 《机器学习实战：基于Scikit-Learn、Keras和TensorFlow》(原书第2版),
Aurelien Geron著, 王静源等译, 机械工业出版社, 2020, P₁₀₈ 编程实践
5. 周志华《机器学习》教材数学公式推导的细节, 参见《南瓜书PumpkinBook》:

在线阅读地址: <https://datawhalechina.github.io/pumpkin-book>

Pdf版下载地址: <https://github.com/datawhalechina/pumpkin-book/releases>

目录

□ 线性回归

- 最小二乘法

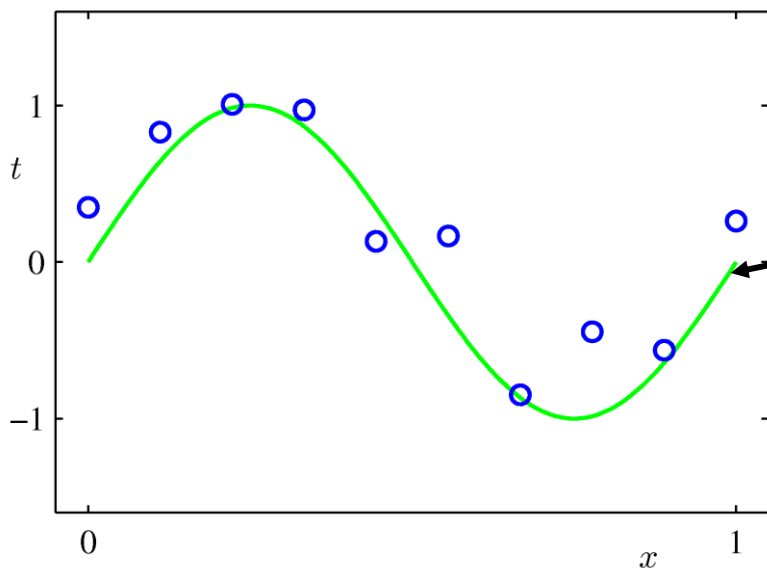
□ 二分类任务

- 对数几率回归
- 线性判别分析

多项式拟合问题

□ 多项式曲线拟合问题

问题：给定10个训练数据点(图中蓝色的小圆圈)，其由 $\sin(2x)$ 函数(绿色曲线)加上随机噪声扰动得到。每个数据点由输入变量 x 的观测以及对应的目标变量 t 组成。



绿色曲线是用来生成数据的 $\sin(2x)$ 函数

□ 用多项式函数拟合给定的观测数据点

多项式阶数

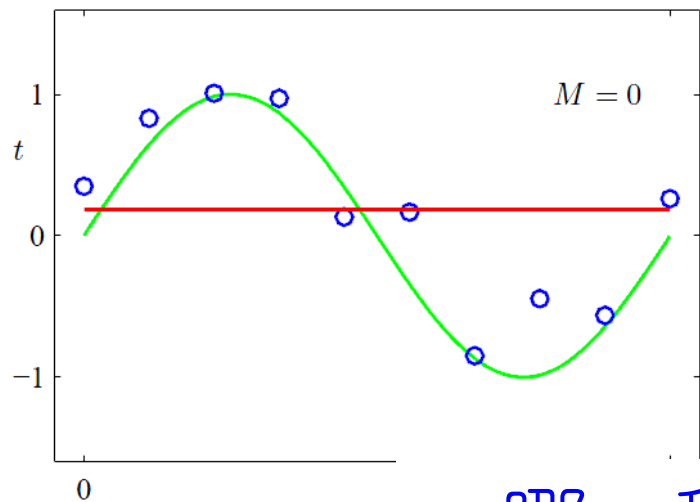
M阶多项式函数 $y(x, \mathbf{w}) = w_0 + w_1x + w_2x^2 + \dots + w_Mx^M = \sum_{j=0}^M w_jx^j$

↑
系数未知 $\mathbf{w} = [w_0, w_1, \dots, w_M]^\top$

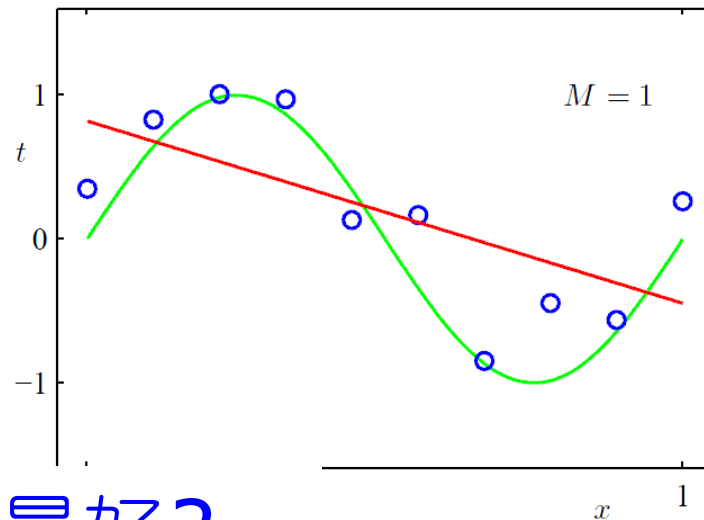
多项式曲线拟合问题

红色为拟合曲线，绿色为真实曲线

$$y = w_0$$

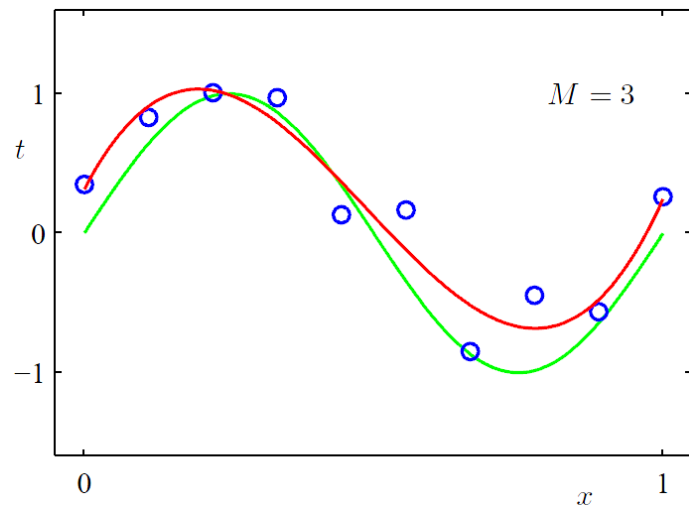


$$y = w_0 + w_1x$$

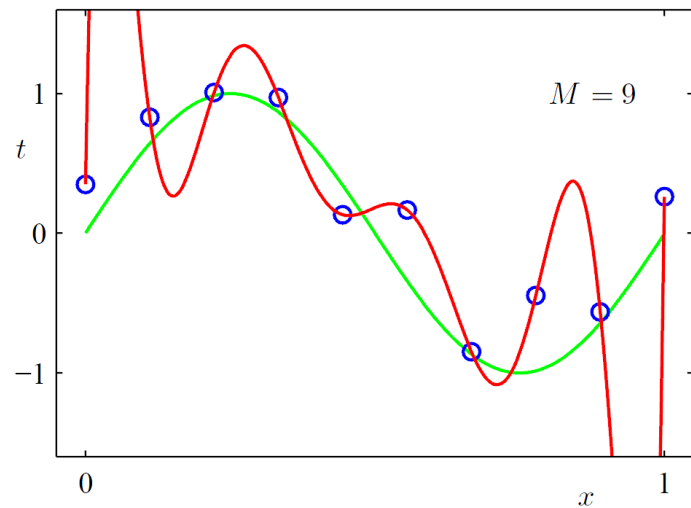


哪一种拟合最好？

$$y = w_0 + w_1x + w_2x^2 + w_3x^3$$

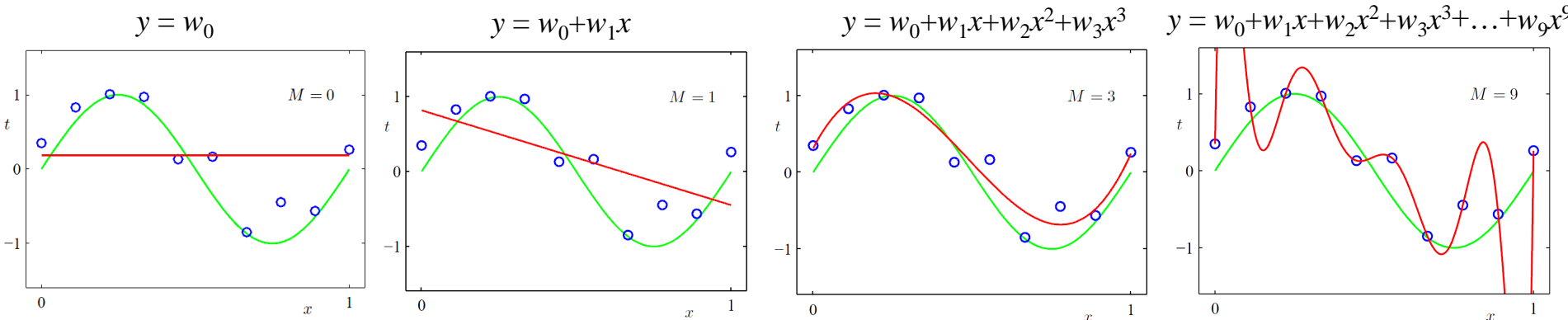


$$y = w_0 + w_1x + w_2x^2 + w_3x^3 + \dots + w_9x^9$$



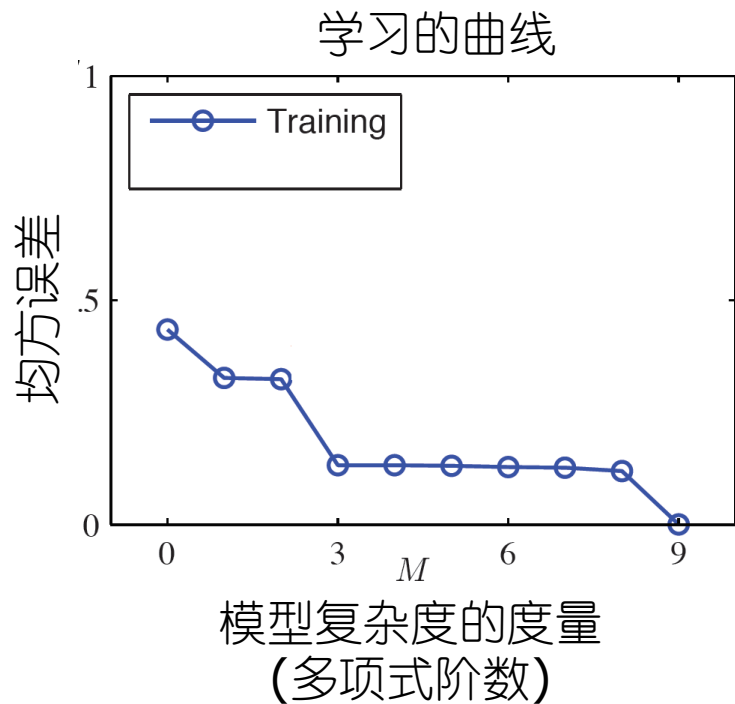
假设空间: M 阶多项式 $f(x)$

红色为拟合曲线, 绿色为真实曲线



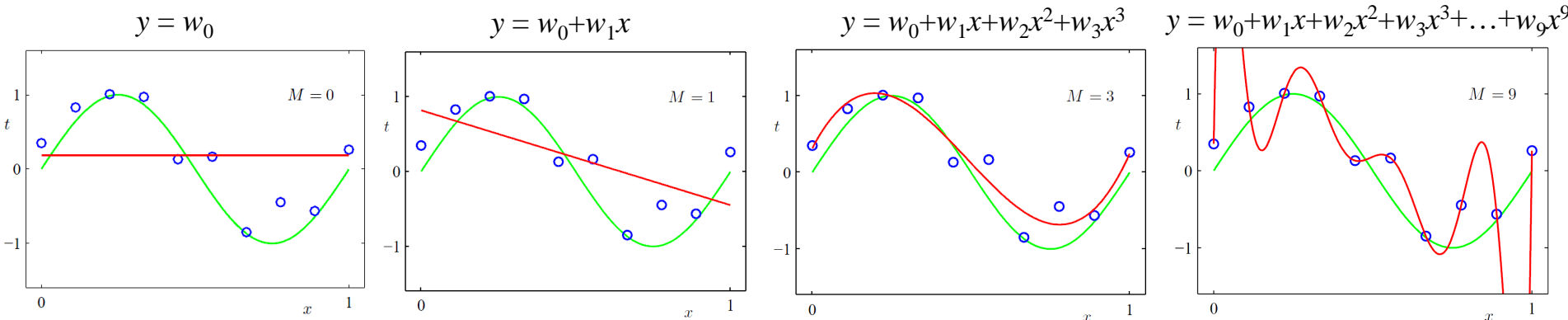
- 给定10个观测数据点 (x_i, y_i) , $i = 1, 2, \dots, 10$
- 观测数据点 x_i 上的预测值为多项式函数 $f(x_i)$
- 我们使用代价函数 (cost function) $L(y, f(x))$ 来度量误差.
- 对于回归问题, 代价函数的普遍选择是均方误差

$$E = \frac{1}{10} \sum_{i=1}^{10} (y_i - f(x_i))^2$$



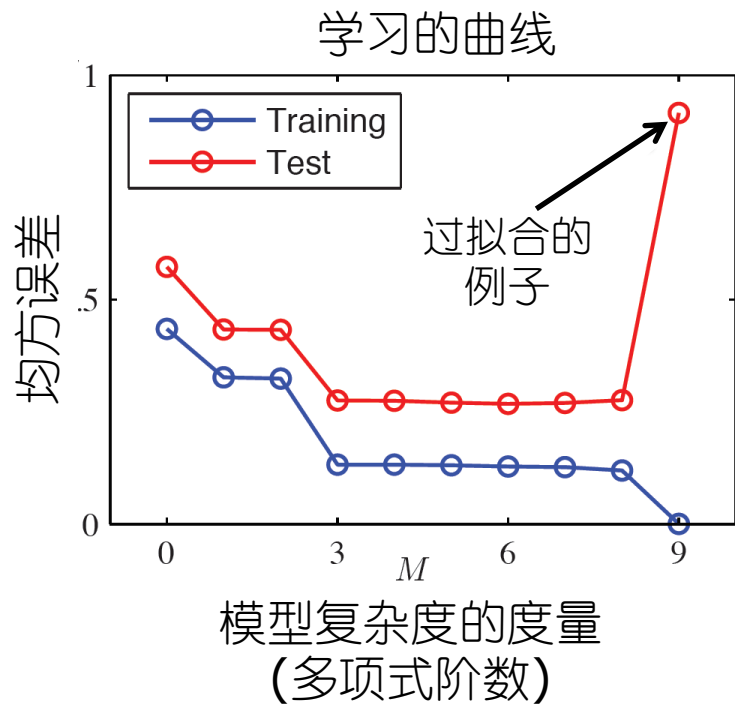
假设空间: M 阶多项式 $f(x)$

红色为拟合曲线, 绿色为真实曲线



- 给定10个观测数据点 (x_i, y_i) , $i = 1, 2, \dots, 10$
- 观测数据点 x_i 上的预测值为多项式函数 $f(x_i)$
- 我们使用代价函数 (cost function) $L(y, f(x))$ 来度量误差.
- 对于回归问题, 代价函数的普遍选择是均方误差

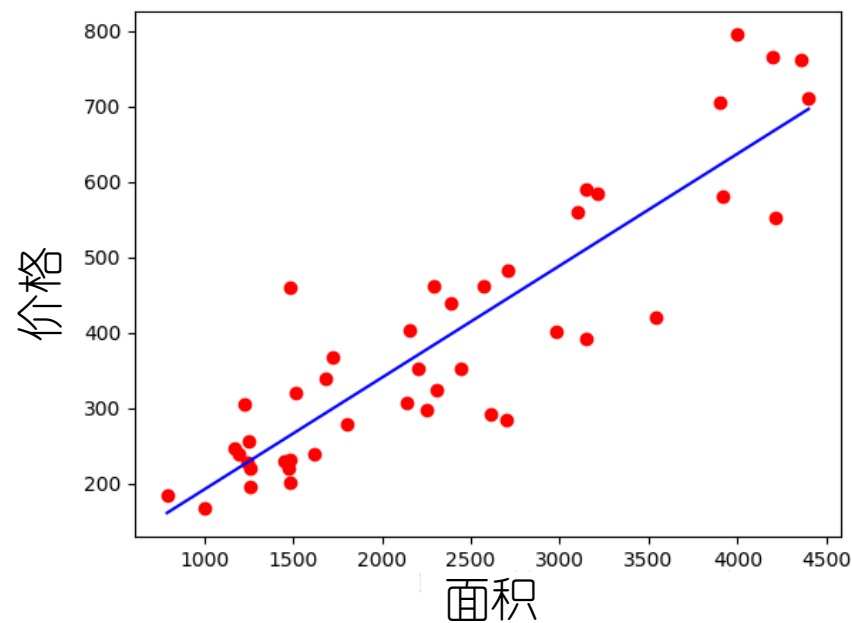
$$E = \frac{1}{10} \sum_{i=1}^{10} (y_i - f(x_i))^2$$



- 给定一组数据点 $(x_1, y_1), \dots, (x_n, y_n)$, 根据这些数据点研究 x 和 y 之间关系的分析方法就是回归。
- 线性回归(Linear Regression)是利用数理统计中回归分析, 来确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法
- 如果以一个线性函数 $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$ 来描述两者之间的关系, 则称为线性回归。如果以一个逻辑函数来描述两者之间的关系, 则称为逻辑回归。

房价与房屋面积关系的线性拟合

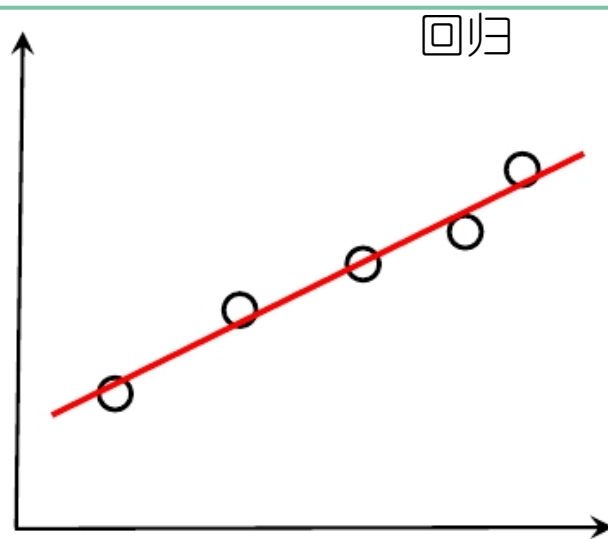
- 给定房屋面积与交易价格的数据表(右端), 通过回归方程拟合的直线(蓝色)与原有数据点(红色)的关系如下图
- 根据该直线方程, 由房屋的面积来预测房屋的交易价格



单一特征

编号	房屋面积/ft ²	交易价格/万	编号	房屋面积/ft ²	交易价格/万
1	1000	168	26	2700	285
2	792	184	27	2612	292
3	1260	197	28	2705	482
4	1262	220	29	2570	462
5	1240	228	30	2442	352
6	1170	248	31	2387	440
7	1230	305	32	2292	462
8	1255	256	33	2308	325
9	1194	240	34	2252	298
10	1450	230	35	2202	352
11	1481	202	36	2157	403
12	1475	220	37	2140	308
13	1482	232	38	4000	795
14	1484	460	39	4200	765
15	1512	320	40	3900	705
16	1680	340	41	3544	420
17	1620	240	42	2980	402
18	1720	368	43	4355	762
19	1800	280	44	3150	392
20	4400	710	45	3025	320
21	4212	552	46	3450	350
22	3920	580	47	4402	820
23	3212	585	48	3454	425
24	3151	590	49	890	272
25	3100	560			

基本形式



□ 线性模型一般形式

$$f(\mathbf{x}) = w_1x_1 + w_2x_2 + \dots + w_dx_d + b$$

$\mathbf{x} = [x_1, x_2, \dots, x_d]^T$ 是由属性描述的样本，其中 x_i 是 \mathbf{x} 在第 i 个属性上的取值 (即属性值)

□ 向量形式

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$$

其中 $\mathbf{w} = (w_1; w_2; \dots; w_d) = [w_1, w_2, \dots, w_d]^T$

线性模型优点

- 形式简单、易于建模
- 可解释性(权重系数 \mathbf{w} 直观地表达了各属性在预测中的重要性)
- 非线性模型的基础
 - 引入层级结构或高维映射
- 一个例子
 - 综合考虑色泽、根蒂和敲声来判断西瓜好不好
 - 其中根蒂的系数最大，表明根蒂最要紧；而敲声的系数比色泽大，说明敲声比色泽更重要

$$f_{\text{好瓜}}(\mathbf{x}) = 0.2 \cdot x_{\text{色泽}} + 0.5 \cdot x_{\text{根蒂}} + 0.3 \cdot x_{\text{敲声}} + 1$$

线性回归

□ 给定数据集 $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$ m 为样本个数

其中 $\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id})$, $y_i \in \mathbb{R}$ d 为每个样本的特征维度

□ 线性回归 (linear regression) 目的

- 学得一个线性模型以尽可能准确地预测实值输出标记

□ 离散属性处理 (见周志华机器学习p54)

- 有“序”关系
 - 连续化为连续值
- 无“序”关系
 - 有k个属性值, 则转换为k维向量

线性回归

□ 单一属性(特征)的线性回归目标

$$f(x_i) = wx_i + b \quad \text{使得} \quad f(x_i) \simeq y_i \quad (x_i, y_i) \in D, \quad x_i \in \mathbb{R}$$

□ 对回归任务, 代价(损失)函数(cost/loss function)普遍使用均方误差

□ 基于均方误差最小化进行模型求解的方法称为“最小二乘法”

□ 参数/模型估计: 最小二乘法 (least square method)

$$(w^*, b^*) = \arg \min_{(w, b)} \frac{1}{m} \sum_{i=1}^m (f(x_i) - y_i)^2$$

代价函数取得
最小值时对应
的参数 (w^*, b^*)

$$= \arg \min_{(w, b)} \frac{1}{m} \sum_{i=1}^m (wx_i + b - y_i)^2$$

$$= \arg \min_{(w, b)} \sum_{i=1}^m (wx_i + b - y_i)^2$$

arg: argument (参数) 的简写

min: minimum (最小值) 的简写

argument=实参(actual parameter), parameter=形参(formal parameter)

线性回归 - 最小二乘法

□ 最小化均方误差

$$E_{(w,b)} = \sum_{i=1}^m (y_i - wx_i - b)^2$$

□ 分别对 w 和 b 求偏导，可得

$$\frac{\partial E_{(w,b)}}{\partial w} = 2 \left(w \sum_{i=1}^m x_i^2 - \sum_{i=1}^m (y_i - b) x_i \right)$$

$$\frac{\partial E_{(w,b)}}{\partial b} = 2 \left(mb - \sum_{i=1}^m (y_i - wx_i) \right)$$

线性回归 - 最小二乘法

□ 得到封闭形式 (closed-form) 的解

$$w = \frac{\sum_{i=1}^m y_i (x_i - \bar{x})}{\sum_{i=1}^m x_i^2 - \frac{1}{m} \left(\sum_{i=1}^m x_i \right)^2}$$

$$b = \frac{1}{m} \sum_{i=1}^m (y_i - wx_i)$$

其中均值

$$\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i$$

多元线性回归

□ 给定数据集(多个属性)

$$D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$$

$$\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id}) \quad (d \text{ 为属性维数})$$

$$= (x_{i1}, x_{i2}, \dots, x_{id})^T$$

$$= [x_{i1}, x_{i2}, \dots, x_{id}]^T$$

$$y_i \in \mathbb{R} \quad i = 1, 2, \dots, m$$

□ 多元线性回归目标

$f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i + b$ 使得预测结果 $f(\mathbf{x}_i)$ 逼近标记 y_i , 即 $f(\mathbf{x}_i) \simeq y_i$

其中 $\mathbf{w} \in \mathbb{R}^d$

多元线性回归

□ 把 \mathbf{w} 和 b 吸收为向量形式 $\hat{\mathbf{w}} = (\mathbf{w}; b) = [\mathbf{w}, b]^\top$, 数据集 D 表示为

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 & & & & \\ x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1^\top & 1 \\ \mathbf{x}_2^\top & 1 \\ \vdots & \vdots \\ \mathbf{x}_m^\top & 1 \end{pmatrix} = \begin{bmatrix} \hat{\mathbf{x}}_1^\top \\ \hat{\mathbf{x}}_2^\top \\ \vdots \\ \hat{\mathbf{x}}_m^\top \end{bmatrix} \in \mathbb{R}^{m \times (d+1)}$$

$$\text{其中 } \hat{\mathbf{x}}_i = [x_{i1}, x_{i2}, \dots, x_{id}, 1]^\top \quad \mathbf{y} = (y_1; y_2; \dots; y_m)$$

□ 多元线性回归的最小二乘模型

$$\begin{aligned} (\mathbf{w}^*, b^*) &= \arg \min_{\mathbf{w}, b} \sum_{i=1}^m (f(\mathbf{x}_i) - y_i)^2 = \arg \min_{\mathbf{w}, b} \sum_{i=1}^m [(\mathbf{w}^\top \mathbf{x}_i + b) - y_i]^2 \\ \Rightarrow \hat{\mathbf{w}}^* &= \arg \min_{\hat{\mathbf{w}}} \sum_{i=1}^m (\hat{\mathbf{w}}^\top \hat{\mathbf{x}}_i - y_i)^2 = \arg \min_{\hat{\mathbf{w}}} \sum_{i=1}^m (\hat{\mathbf{x}}_i^\top \hat{\mathbf{w}} - y_i)^2 \\ &= \arg \min_{\hat{\mathbf{w}}} (\mathbf{X} \hat{\mathbf{w}} - \mathbf{y})^\top (\mathbf{X} \hat{\mathbf{w}} - \mathbf{y}) \end{aligned}$$

多元线性回归 - 最小二乘法

□ 最小二乘法 (least square method)

$$\begin{aligned}\hat{\boldsymbol{w}}^* &= \arg \min_{\hat{\boldsymbol{w}}} (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y})^\top (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}) \\ &= \arg \min_{\hat{\boldsymbol{w}}} \|\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}\|_2^2 \quad (2\text{范数的平方}) \\ &= \arg \min_{\hat{\boldsymbol{w}}} \hat{\boldsymbol{w}}^\top \mathbf{X}^\top \mathbf{X} \hat{\boldsymbol{w}} - 2\hat{\boldsymbol{w}}^\top \mathbf{X}^\top \boldsymbol{y} + \boldsymbol{y}^\top \boldsymbol{y}\end{aligned}$$

令 $E_{\hat{\boldsymbol{w}}} = (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y})^\top (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y})$, 对 $\hat{\boldsymbol{w}}$ 求偏导得到

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\mathbf{X}^\top (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}) \quad \leftarrow \text{如何得到?}$$

令上式为零可得 $\hat{\boldsymbol{w}}$ 最优解的闭式解. 然而, 麻烦来了: 涉及矩阵求逆!

多元线性回归 - 满秩讨论

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

□ 若 $\mathbf{X}^T\mathbf{X}$ 是满秩矩阵或正定矩阵(矩阵可逆), 则

$$\hat{\mathbf{w}}^* = (\mathbf{X}^T\mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

其中 $(\mathbf{X}^T\mathbf{X})^{-1}$ 是 $\mathbf{X}^T\mathbf{X}$ 的逆矩阵, 线性回归模型为

$$f(\hat{\mathbf{x}}_i) = \hat{\mathbf{w}}^T \hat{\mathbf{x}}_i = [(\mathbf{X}^T\mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}]^T \hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i^T (\mathbf{X}^T\mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

□ 若 $\mathbf{X}^T\mathbf{X}$ 不是满秩矩阵, 可解出多个 $\hat{\mathbf{w}}$

● 引入正则化 (regularization) (参见《机器学习》6.4节, 11.4节)

$$\min_{\hat{\mathbf{w}}} \|\mathbf{X}\hat{\mathbf{w}} - \mathbf{y}\|^2 + \lambda \|\hat{\mathbf{w}}\|^2 \quad \Rightarrow \quad \hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

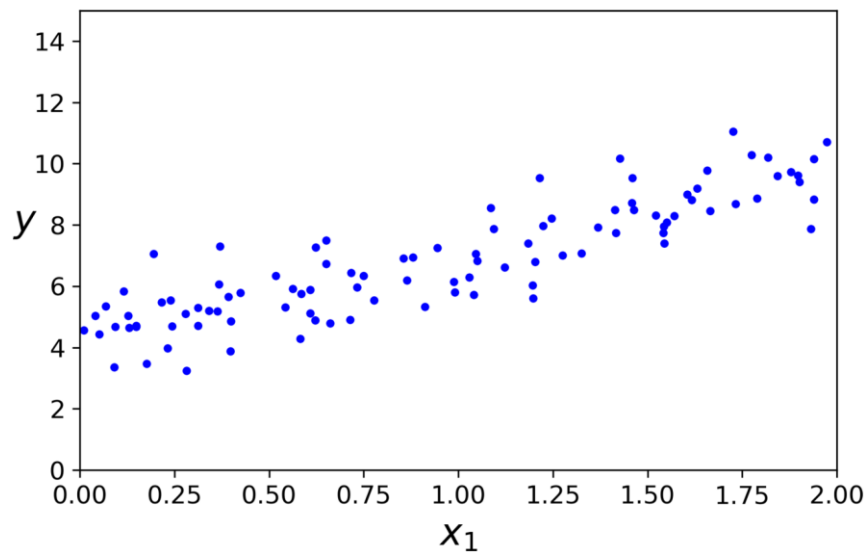
正则化项

● 根据归纳偏好选择解 (参见《机器学习》1.4节)

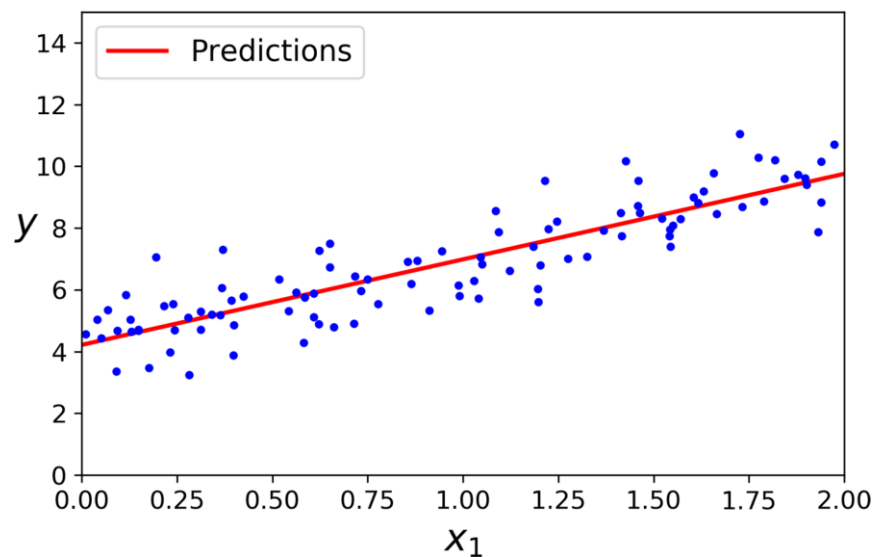
\mathbf{I} 是单位矩阵

线性回归实例预测结果

$$f(\hat{\mathbf{x}}_i) = \hat{\mathbf{w}}^\top \hat{\mathbf{x}}_i = [(\mathbf{X}^\top \mathbf{X})^\top \mathbf{X}^\top \mathbf{y}]^\top \hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i^\top (\mathbf{X}^\top \mathbf{X})^\top \mathbf{X}^\top \mathbf{y}$$



随机生成的线性数据



线性回归模型预测

对数线性回归(log-linear regression)

□ 输出标记的对数为线性模型逼近的目标

- 对于样例 (x, y) , 若希望线性模型的预测值逼近真实标记 y , 则得线性回归模型 $y = \mathbf{w}^T \mathbf{x} + b$

- 有时上述原始线性回归可能并不能满足需求, 例如: 样本对应标记 y 值并不是线性变化, 而是在指数尺度上变化(如右图黑色部分)

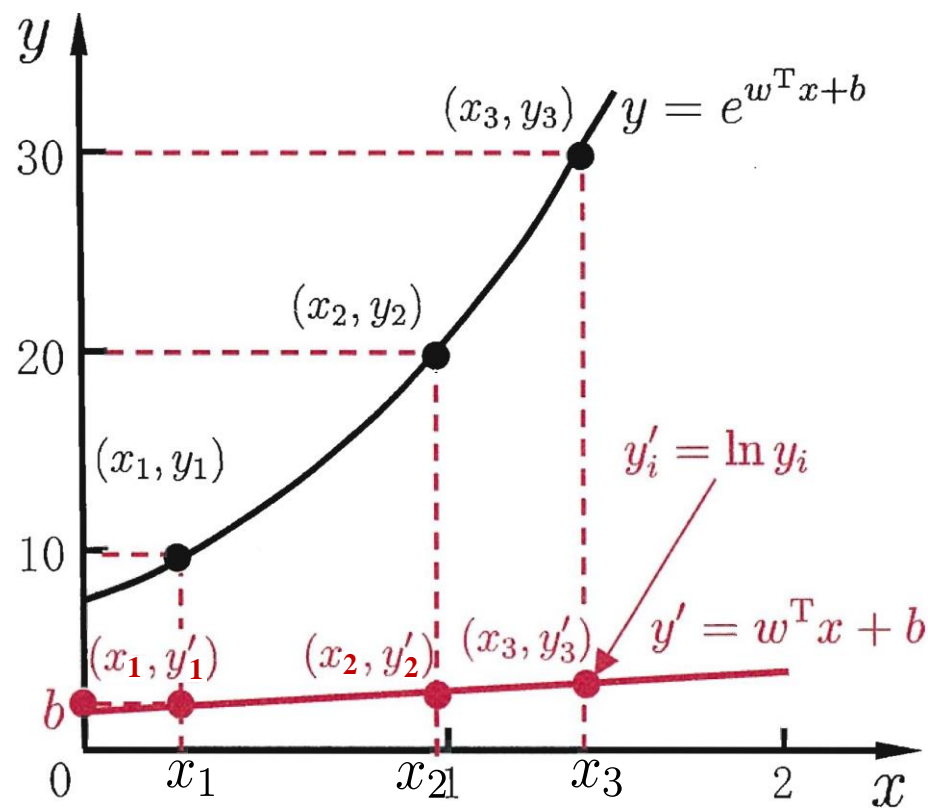
- 让模型预测值逼近 y 的衍生物



- 让线性模型 $\mathbf{w}^T \mathbf{x} + b$ 去逼近 $\ln y$

- 若令 $\ln y = \mathbf{w}^T \mathbf{x} + b$, 则得对数线性回归

- 实际是在用 $e^{\mathbf{w}^T \mathbf{x} + b}$ 逼近 y



对数线性回归示意图

线性回归 - 广义线性模型

□ 一般形式

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

$g(\cdot)$ 称为联系函数 (link function), g^{-1} 称为 g 的反函数

- g 单调可微函数 ($g(\cdot)$ 连续且充分光滑)

- 对数线性回归是 $g(\cdot) = \ln(\cdot)$ 时广义线性模型的特例,
即令 $g(\cdot) = \ln(\cdot)$, 则得对数线性回归 $\ln y = \boldsymbol{w}^T \boldsymbol{x} + b$

回归模型用于二分类任务

□ 对于二分类任务, 输出标记为 $y \in \{0, 1\}$, 即我们更倾向于选择介于0和1之间的概率, 而线性回归的预测结果 $z = \mathbf{w}^T \mathbf{x} + b$ 一般是一个连续值, 因此, 我们需要将 $z = \mathbf{w}^T \mathbf{x} + b$ 做某种变换, 将其转换到输出介于0和1之间的值(类似于概率范围)。

□ 线性回归模型产生的实际预测值 $z = \mathbf{w}^T \mathbf{x} + b$ } 找 z 和 y 的
期望的输出标记 $y \in \{0, 1\}$ } 联系函数

□ 寻找联系函数将线性回归模型输出与分类标记信息联系起来

□ 最理想的函数——单位阶跃函数

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

预测值大于零就判为正例, 小于零就判为反例, 预测值为临界值零则可任意判别

回归模型用于二分类任务

单位阶跃函数缺点

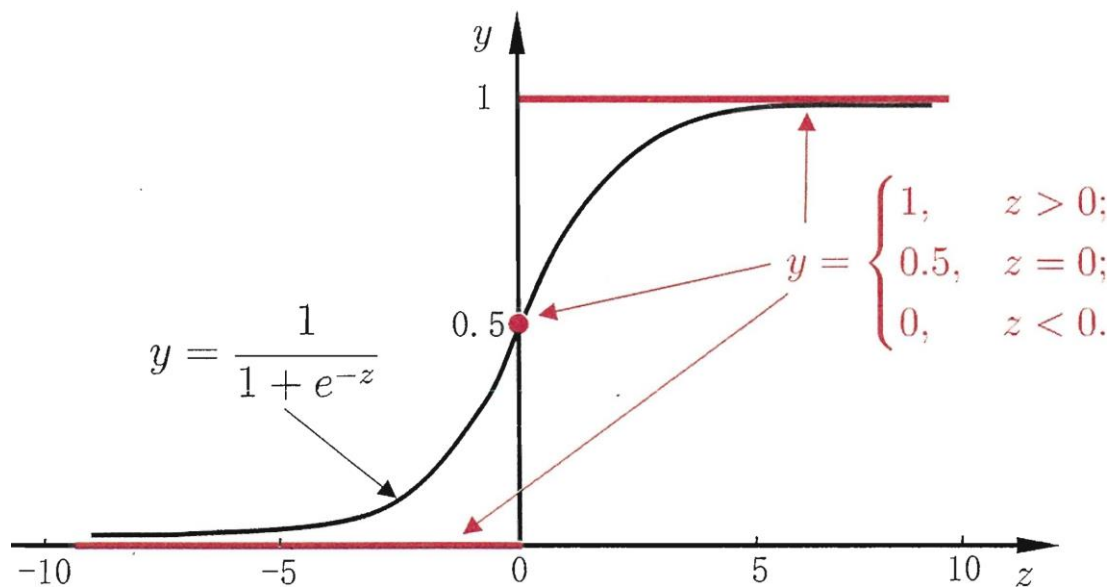
- 不连续 ($x=0$ 时导数不存在)

替代函数——对数几率函数 (logistic function)

- 单调可微、任意阶可导

$$y = \frac{1}{1 + e^{-z}} \quad \Longrightarrow \quad \text{将 } z \text{ 值转化为一个接近0或1的 } y \text{ 值}$$

对数几率函数
(logistic function)
简称“对率函数”，
也称 Sigmoid 函数



对数几率函数定义域 $(-\infty, +\infty)$, 值域 $(0, 1)$

单位阶跃函数与对数几率函数的比较

几率和对数几率的定义

- **几率(odds)定义**：几率是指该事件发生的概率与该事件不发生的概率的比值。即如果事件发生概率是 p ，那么该事件的几率为 $\frac{p}{1-p}$ 。(可以想象，当几率大于1时，说明该事件发生的概率大，几率小于1时，说明该事件发生的概率小；几率变化范围为 $(0, +\infty)$)
- 几率的概念推广叫**对数几率(log odds)或logit函数**： $\log \frac{p}{1-p}$ (可以想象，当对数几率大于0时，说明该事件发生的概率大；对数几率小于0时，说明该事件发生的概率小)

对数几率回归

□ 把对数几率函数作为联系函数 $g^{-1}(\cdot)$

$$y = \frac{1}{1 + e^{-z}} \xrightarrow{z = \mathbf{w}^T \mathbf{x} + b} y = \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x} + b)}}$$

□ 对数几率 (log odds)

- 样本作为正例的相对可能性的对数

称比值 $\frac{y}{1-y}$ 为几率

$$\ln \left(\frac{y}{1-y} \right) = \mathbf{w}^T \mathbf{x} + b$$

将 y 视为样本 x 作为正例的可能性
(y 表示为 x 被分到正例的概率)

$1-y$ 是看作样本 x 为反例的概率

- 上式实质上是用线性回归模型的预测结果来逼近真实标记的对数几率，因此这个模型被称为对数几率回归
- 虽然名字是回归，但实际上是一种分类学习方法

□ 对数几率回归优点

- 无需事先假设数据分布
- 可得到“类别”的近似概率预测
- 可直接应用现有数值优化算法求取最优解

对数几率回归的概率解释

- 把样本 \mathbf{x} 的类别 y 看作有 0 和 1 两种取值的随机变量, 故只需判断 $p(y = 1|\mathbf{x})$ 和 $p(y = 0|\mathbf{x})$ 之间的大小关系, 再将 \mathbf{x} 归为概率较大的一类
- 我们说 y 可以视为样本 \mathbf{x} 为正例的可能性 (y 不是仅仅确定的标签), 从概率角度, 若将 y 看作类后验概率估计, 可以有 $y = p(y = 1|\mathbf{x})$, 则 $1 - y = p(y = 0|\mathbf{x})$

$$\ln \frac{y}{1-y} = \mathbf{w}^T \mathbf{x} + b \quad \text{可写为} \quad \ln \frac{p(y = 1 | \mathbf{x})}{p(y = 0 | \mathbf{x})} = \mathbf{w}^T \mathbf{x} + b$$

则由 $p(y = 1|\mathbf{x}) + p(y = 0|\mathbf{x}) = 1$ 可得

$$p(y = 1|\mathbf{x}) = \frac{e^{\mathbf{w}^T \mathbf{x} + b}}{1 + e^{\mathbf{w}^T \mathbf{x} + b}} \quad \text{记为} \quad h_{\boldsymbol{\beta}}(\mathbf{x})$$

$$p(y = 0|\mathbf{x}) = \frac{1}{1 + e^{\mathbf{w}^T \mathbf{x} + b}} \quad \text{记为} \quad 1 - h_{\boldsymbol{\beta}}(\mathbf{x})$$

其中 $\boldsymbol{\beta} = (\mathbf{w}; b) = [\mathbf{w}, b]^T$

对数几率回归模型求解 - 最大似然法

□ 似然函数的意义。对于 $p(\mathbf{x}|\theta)$ 或者是 $p(\mathbf{x};\theta)$ ：

- 当 \mathbf{x} 是变量， θ 是已知量的时候，上述式子表示的是一个**概率函数**，即概率分布的参数取值为 θ 时，不同样本 \mathbf{x} 出现的概率。
- 当 \mathbf{x} 是已知量， θ 为变量的时候，上述式子表示的是一个**似然函数**，即概率分布的参数取不同值时，某个样本 \mathbf{x} 出现的概率。
- **似然函数**就是在已经有观测样本时，寻找最符合当前数据分布的参数

□ 最大似然估计中采样需满足一个很重要的假设：所有的采样都是独立同分布的。

□ $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ 为独立同分布的采样，定义**似然函数** L 为混合密度函数 (m 个样本同时出现)：

$$\begin{aligned} L(\beta) &= p(y_1|\mathbf{x}_1;\beta) \times p(y_2|\mathbf{x}_2;\beta) \times \dots \times p(y_m|\mathbf{x}_m;\beta) \\ &= \prod_{i=1}^m p(y_i|\mathbf{x}_i;\beta) = \prod_{i=1}^m \underbrace{(h_{\beta}(\mathbf{x}_i))^{y_i} (1 - h_{\beta}(\mathbf{x}_i))^{1-y_i}}_{\text{如何得到?}} \end{aligned}$$

最大似然估计思想：找到一组参数使得所有观测样本的联合概率最大化

对数几率回归模型求解 - 最大似然法求 w 和 b

正确分类概率


$$p(y_i|\mathbf{x}_i;\boldsymbol{\beta}) = \begin{cases} p(y=1|\mathbf{x}_i;\boldsymbol{\beta}), & \text{if } y_i = 1, \\ p(y=0|\mathbf{x}_i;\boldsymbol{\beta}) = 1 - p(y=1|\mathbf{x}_i;\boldsymbol{\beta}) & \text{if } y_i = 0, \end{cases}$$
$$= (p(y=1|\mathbf{x}_i;\boldsymbol{\beta}))^{y_i} \times (1 - p(y=1|\mathbf{x}_i;\boldsymbol{\beta}))^{1-y_i}$$
$$= (h_{\boldsymbol{\beta}}(\mathbf{x}_i))^{y_i} \times (1 - h_{\boldsymbol{\beta}}(\mathbf{x}_i))^{1-y_i}$$

$$\ln p(y_i|\mathbf{x}_i;\boldsymbol{\beta}) = \begin{cases} \ln p(y=1|\mathbf{x}_i;\boldsymbol{\beta}), & \text{if } y_i = 1, \\ \ln p(y=0|\mathbf{x}_i;\boldsymbol{\beta}) = \ln[1 - p(y=1|\mathbf{x}_i;\boldsymbol{\beta})] & \text{if } y_i = 0, \end{cases}$$
$$= y_i \ln p(y=1|\mathbf{x}_i;\boldsymbol{\beta}) + (1 - y_i) \ln p(y=0|\mathbf{x}_i;\boldsymbol{\beta})$$

其中 $\boldsymbol{\beta} = (\mathbf{w}_i; b)$.

□ 最大似然法 (maximum likelihood) (《机器学习》第7章)

- 给定数据集 $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, 最大化样本属于其真实标记的概率
- 最大化对数似然函数

$$\ell(\mathbf{w}, b) = \ln L(\boldsymbol{\beta}) = \sum_{i=1}^m \ln p(y_i|\mathbf{x}_i; \mathbf{w}, b)$$


概率中分号表示分号后的 \mathbf{w}, b 是待估参数, 它们确定的, 只是当前未知。它们不是随机变量。

对数几率回归模型求解 - 极大似然法

□ 对数似然函数 $\ln L$: (一般将相乘转换为相加, 对数函数单调递增不改变最大值位置)

$$\ln L(\boldsymbol{\beta}) = \sum_{i=1}^m \left[y_i \ln h_{\boldsymbol{\beta}}(\mathbf{x}_i) + (1 - y_i) \ln(1 - h_{\boldsymbol{\beta}}(\mathbf{x}_i)) \right]$$

$$\max_{\boldsymbol{\beta}} \ln L(\boldsymbol{\beta}) \Leftrightarrow \min_{\boldsymbol{\beta}} -\ln L(\boldsymbol{\beta}) \quad (\text{最大化转化为求最小化})$$

□ 对率回归模型最大化“对数似然”等价于最小化负对数似然

$$\begin{aligned} \ell(\boldsymbol{\beta}) &= -\ln L(\boldsymbol{\beta}) \\ &= \sum_{i=1}^m \left[y_i \ln h_{\boldsymbol{\beta}}(\mathbf{x}_i) + (1 - y_i) \ln(1 - h_{\boldsymbol{\beta}}(\mathbf{x}_i)) \right] \\ &= \sum_{i=1}^m \left[-y_i \boldsymbol{\beta}^T \hat{\mathbf{x}}_i + \ln \left(1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i} \right) \right] \quad (\text{动手推导}) \end{aligned}$$

对数几率回归模型求解 - 最大似然法

□ 转化为最小化负对数似然函数求解

- 令 $\boldsymbol{\beta} = (\mathbf{w}; b)$, $\hat{\mathbf{x}} = (\mathbf{x}; 1)$, 则 $\mathbf{w}^T \mathbf{x} + b$ 可简写为 $\boldsymbol{\beta}^T \hat{\mathbf{x}}$

- 再令

$$p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta}) = p(y = 1 \mid \hat{\mathbf{x}}; \boldsymbol{\beta})$$

$$p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta}) = p(y = 0 \mid \hat{\mathbf{x}}; \boldsymbol{\beta}) = 1 - p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta})$$

则似然项可重写为

$$p(y_i \mid \mathbf{x}_i; \mathbf{w}_i, b) = y_i p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta})$$

- 故等价形式为要最小化

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^m \left(-y_i \boldsymbol{\beta}^T \hat{\mathbf{x}}_i + \ln \left(1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i} \right) \right)$$

对数几率回归-牛顿法求解

□ 求解

$$\beta^* = \arg \min_{\beta} \ell(\beta)$$

□ 牛顿法第 $t+1$ 轮迭代解的更新公式

$$\beta^{t+1} = \beta^t - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \right)^{-1} \frac{\partial \ell(\beta)}{\partial \beta}$$

其中关于 β 的一阶、二阶导数分别为

$$\frac{\partial \ell(\beta)}{\partial \beta} = - \sum_{i=1}^m \hat{\mathbf{x}}_i (y_i - p_1(\hat{\mathbf{x}}_i; \beta)) \quad (\text{动手推导})$$

$$\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} = \sum_{i=1}^m \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T p_1(\hat{\mathbf{x}}_i; \beta) (1 - p_1(\hat{\mathbf{x}}_i; \beta)) \quad (\text{动手推导})$$

高阶可导连续凸函数，梯度下降法/牛顿法

[Convex optimization, Boyd and Vandenberghe, 2004]

推导也可参考南瓜书：<https://datawhalechina.github.io/pumpkin-book/#/chapter3/chapter3>

对数几率回归-梯度下降法求解

□ 关于参数 β 的更新公式

$$\beta^{t+1} = \beta^t - \alpha \frac{\partial \ell(\beta)}{\partial \beta}$$

其中 α 是学习率. 关于 β 的一阶偏导数为

$$\frac{\partial \ell(\beta)}{\partial \beta} = - \sum_{i=1}^m \hat{\mathbf{x}}_i (y_i - p_1(\hat{\mathbf{x}}_i; \beta))$$

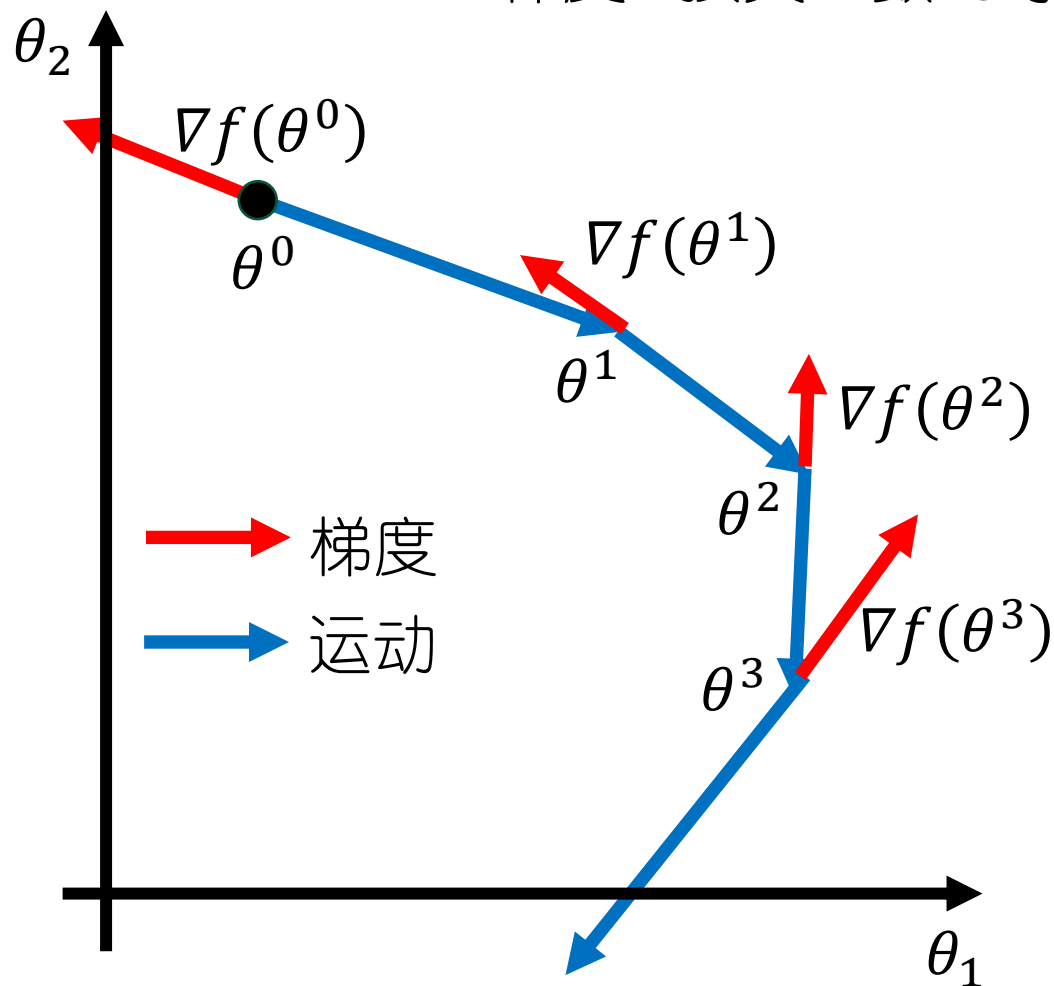
高等数学基本结论：多元函数的值沿其梯度的负方向下降最快

高阶可导连续凸函数，梯度下降法/牛顿法

[Convex optimization, Boyd and Vandenberghe, 2004]

优化：梯度降(Gradient Descent)

梯度：损失函数的等高线的法线方向



起点位置 θ^0 (考虑单个参数)

计算 θ^0 的梯度

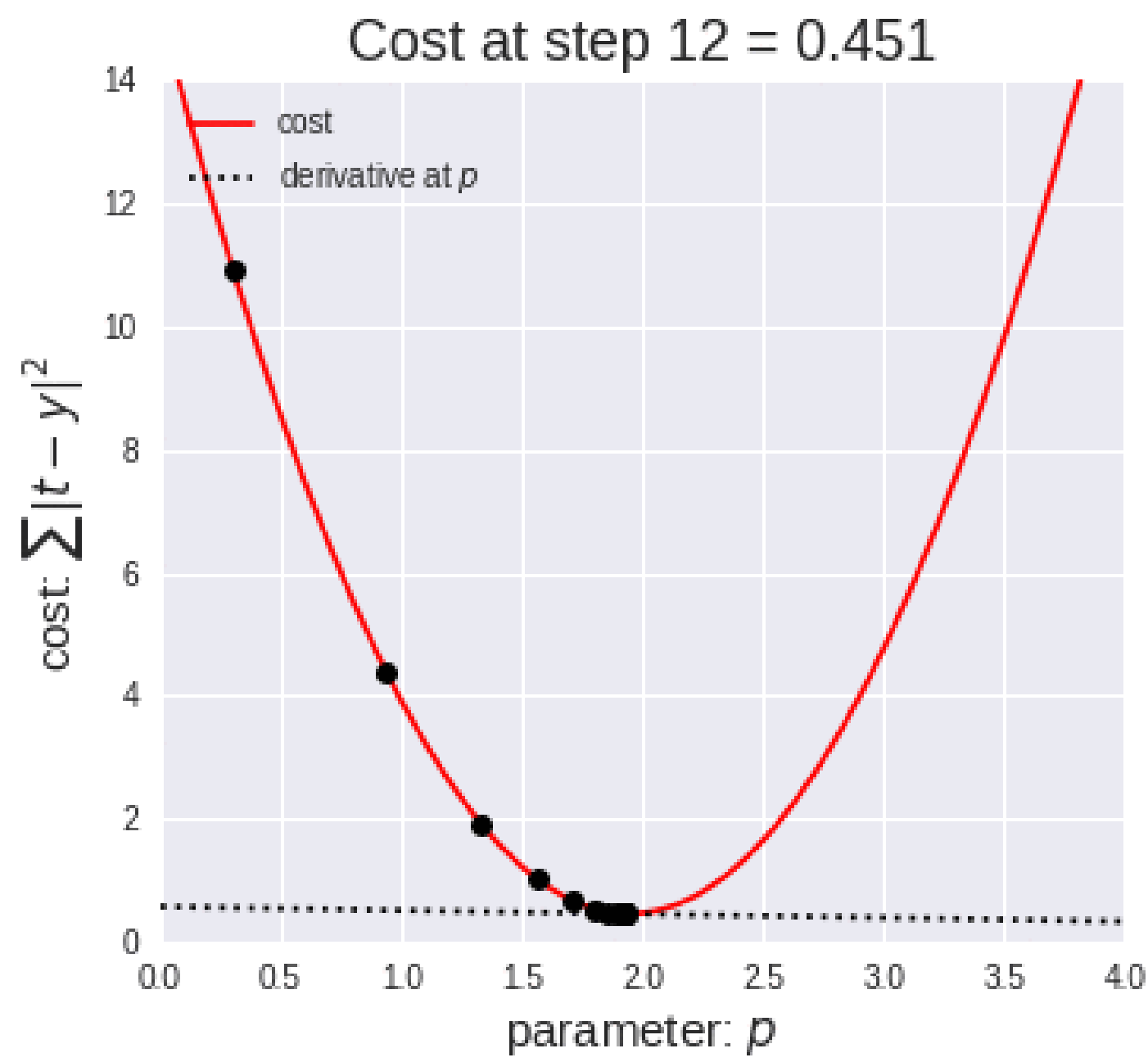
倒向 $\theta^1 = \theta^0 - \eta \nabla f(\theta^0)$

计算 θ^1 处的梯度

倒向 $\theta^2 = \theta^1 - \eta \nabla f(\theta^1)$

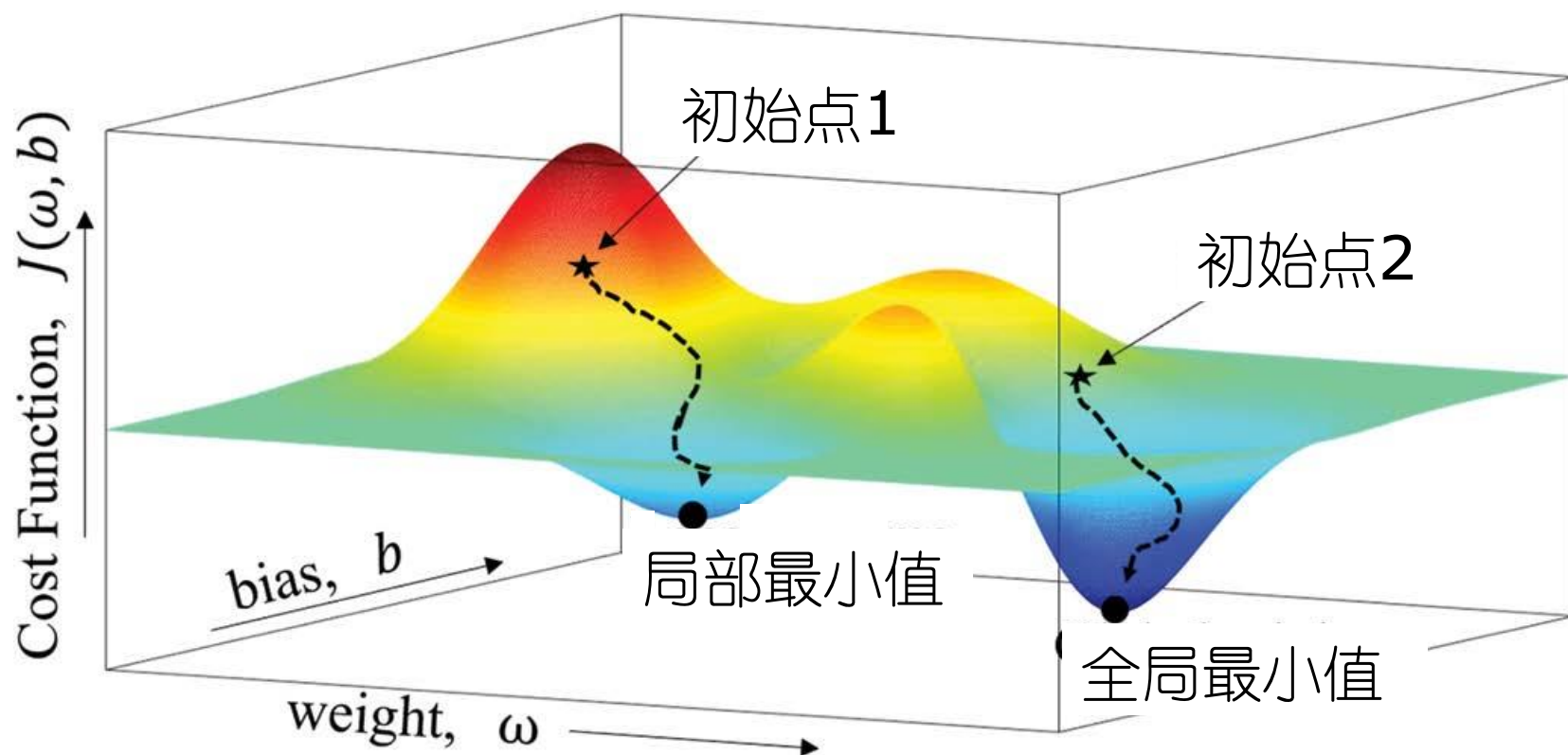
⋮

梯度下降法在一维空间示意图



梯度下降法在二维空间示意图

- 初始点：可以人为的设定，也可以随机设定。初始点选取影响全局收敛性。需要考虑全局收敛性，最好是多设几个初始值迭代。
- 初始点离最优点越近，收敛越快。



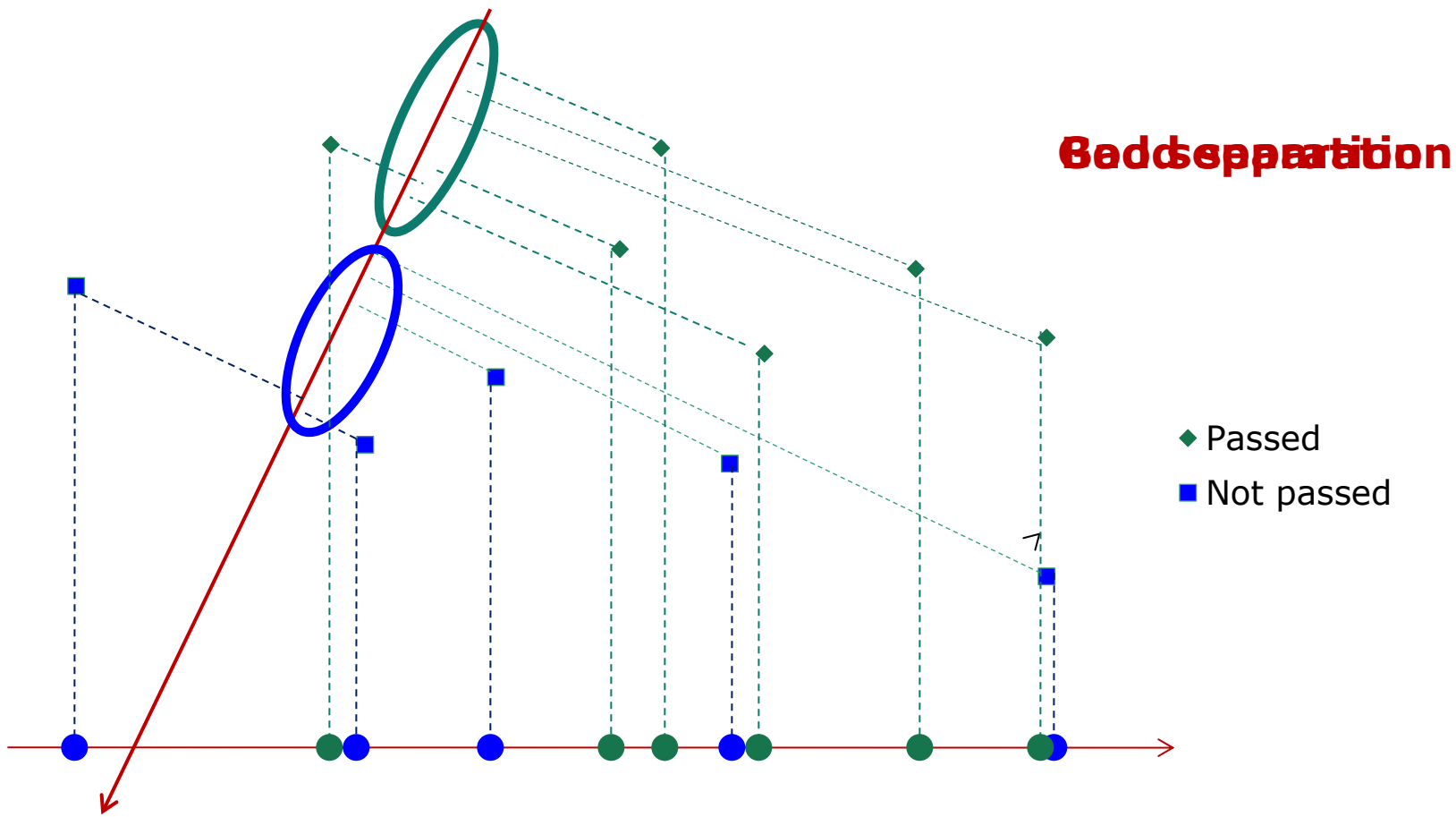
初始点 1 将收敛到局部极小值, 初始点2将收敛到全局最小值

课程作业-对率回归

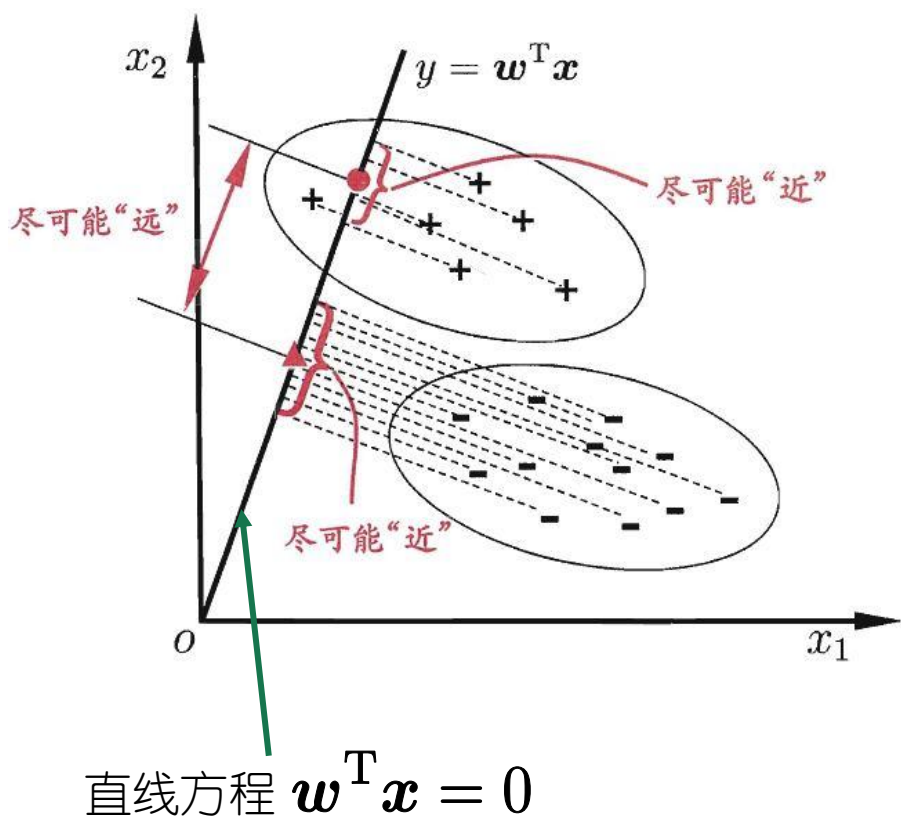
- 使用Python实现牛顿法和梯度下降法优化求解对数几率回归
- 具体细节和要求见 学在西电 “资料” 栏中 “作业要求” 文件夹或QQ附件文件

线性判别分析-二分类任务

■



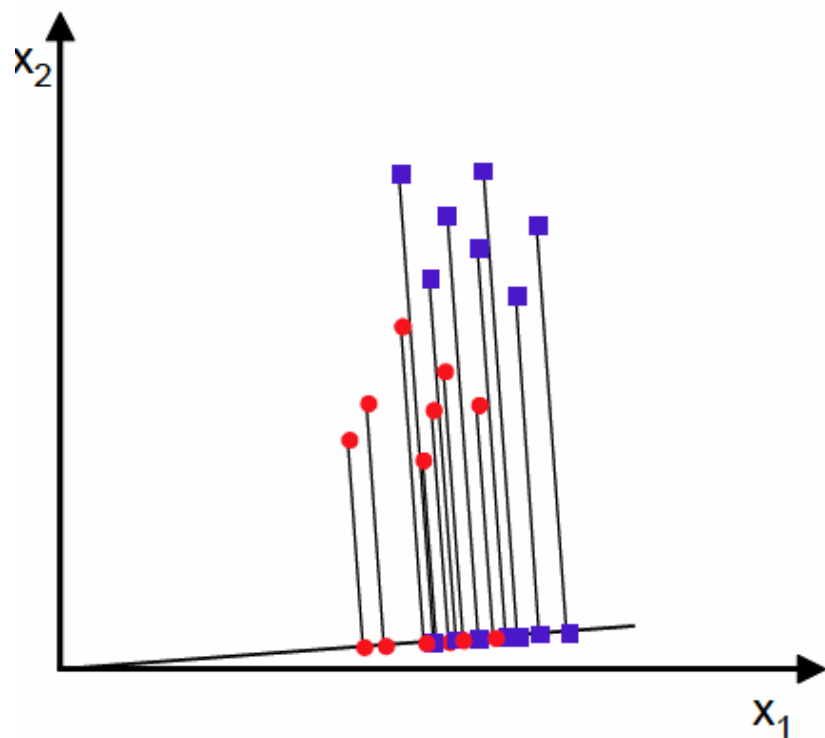
线性判别分析-二分类任务



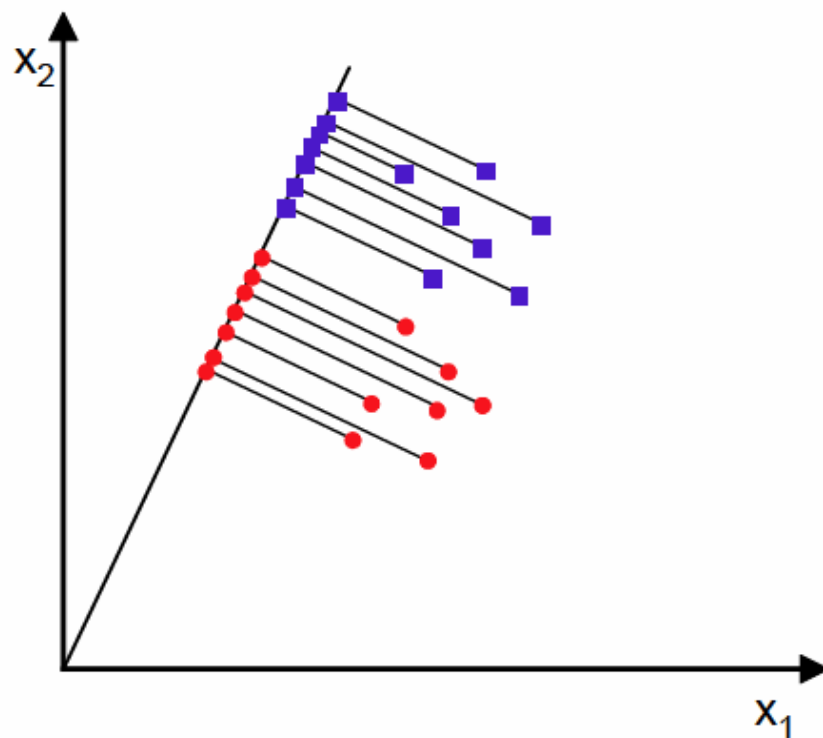
- 线性判别分析 (Linear Discriminant Analysis, LDA) 的基本思想：给定训练样例集，设法将样例投影到一条**适当选择**的直线上，使得同类样例的投影点尽可能接近、异类样例的投影点中心尽可能远离。
- 目标：“投影后类内方差小，类间距离大”
- 对新样本进行分类时，将其投影到同样的直线上，根据投影点位置来确定新样本类别
- 由于将样例投影到一条直线（低维空间），因此也被视为一种“监督降维”技术(降维参考第10章)，也就是说它的数据集的每个样本是有类别输出的
- 高维空间使某些解析和计算方法难以实现，低维空间给解析和计算带来很多方便

线性判别分析

- LDA的目标是在保留尽可能多的类区分信息同时进行降维.
- 给定 d -维向量 x_1, x_2, \dots, x_m , 其中属于 ω_0 类的有 n_1 个, 属于 ω_1 类的有 n_2 个. 在所有可能的投影 $y = \mathbf{w}^\top \mathbf{x}$ 中, 求能达到最大可分效果的投影.

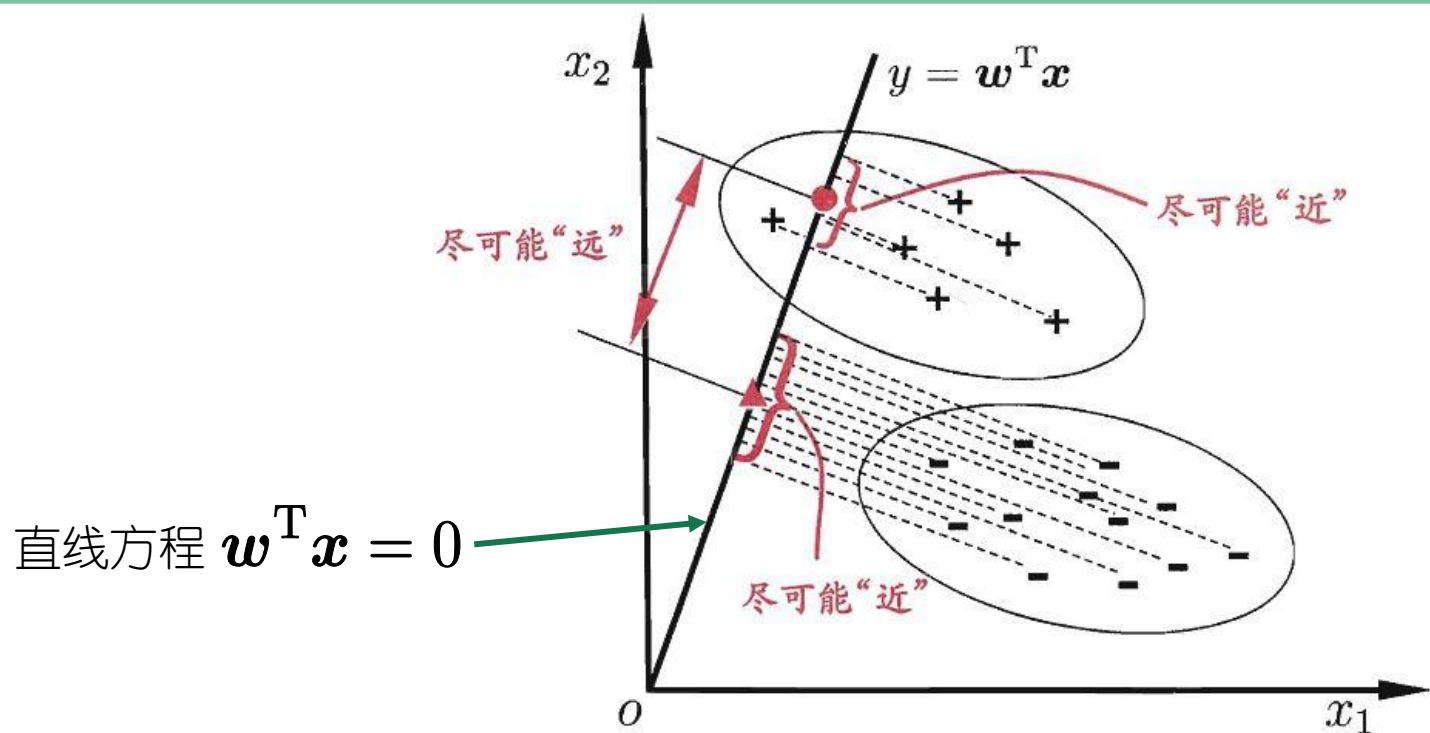


投影后可分效果差



投影后可分效果好

线性判别分析的思想



- 假设我们有两类数据分为“+”（正类）和“-”（负类），如上图所示，这些数据特征是二维的，我们希望将这些数据投影到一维的一条直线，让同一种类别数据的投影点尽可能的接近，而“+”和“-”数据中心之间的距离尽可能的大。
- 我们将这个最佳的投影向量称为 w ，那么样例 x 到方向向量 w 上的投影可以用下式来计算

$$y = w^T x$$

- 下面我们讨论如何确定最佳的投影直线方向 w ，以达到最好的分类效果。因此，我们需要找到一种两个投影点之间可分离的度量。

线性判别分析的目标

我们有一组 m 个 d 维的样本 $\mathbf{x}_1, \dots, \mathbf{x}_m$, 它们分属于示例集合 X_i 中两个不同的类别 ω_0 和 ω_1 , 即 $X_0 = \{\mathbf{x}_i | \omega(\mathbf{x}_i) = \omega_0\}$, $X_1 = \{\mathbf{x}_i | \omega(\mathbf{x}_i) = \omega_1\}$, 且 $|X_0| = n_0$, $|X_1| = n_1$.

对 \mathbf{x} 中的各个成分作线性组合, 得到 $y = \mathbf{w}^T \mathbf{x}$, 这样 n 个样本 $\mathbf{x}_1, \dots, \mathbf{x}_m$ 就产生了 n 个投影结果 y_1, \dots, y_n , 相应的属于集合 Y_0 和 Y_1 , 即 $Y_i = \mathbf{w}^T X_i$ ($i = 0, 1$). 如果 $\boldsymbol{\mu}_i$ 为 d 维样本均值, 则

$$\boldsymbol{\mu}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in X_i} \mathbf{x}, \quad i = 0, 1,$$

则投影后的点的样本均值为

$$\tilde{\boldsymbol{\mu}}_i = \frac{1}{n_i} \sum_{y \in Y_i} y = \frac{1}{n_i} \sum_{\mathbf{x} \in X_i} \mathbf{w}^T \mathbf{x} = \mathbf{w}^T \boldsymbol{\mu}_i$$

也就恰好是原样本均值 $\boldsymbol{\mu}_i$ 的投影.

线性判别分析的目标

投影后的点的样本均值之差为

$$|\tilde{\mu}_0 - \tilde{\mu}_1| = |\mathbf{w}^T(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)|.$$

投影后样本空间的样本方差为

$$s_i^2 = \frac{1}{n_i} \sum_{y \in Y_i} (y - \tilde{\mu}_i)^2.$$

定义投影后第 i 类的类别 ω_i 的类内散度为

$$\tilde{s}_i^2 = \sum_{y \in Y_i} (y - \tilde{\mu}_i)^2,$$

则 $\frac{1}{n}(\tilde{s}_0^2 + \tilde{s}_1^2)$ 就是全部数据的总体的方差的估计. $\tilde{s}_0^2 + \tilde{s}_1^2$ 称为投影样本的总体类内散度. Fisher 线性可分性准则要求在投影 $y = \mathbf{w}^T \mathbf{x}$ 下, 准则函数

$$J(\mathbf{w}) = \frac{|\tilde{\mu}_0 - \tilde{\mu}_1|^2}{\tilde{s}_0^2 + \tilde{s}_1^2}$$

最大化.

线性判别分析的目标

定义类间散度矩阵 $\mathbf{S}_b = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T$, 可证明其为对称半正定的. 投影后两类样本均值之差展开为

$$|\tilde{\mu}_0 - \tilde{\mu}_1|^2 = \mathbf{w}^T (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^T \mathbf{w} = \mathbf{w}^T \mathbf{S}_b \mathbf{w}.$$

原样本空间 X_0 和 X_1 的协方差矩阵

$$\mathbf{C}_0 = \frac{1}{n_0} \sum_{x \in X_0} (\mathbf{x} - \boldsymbol{\mu}_0)(\mathbf{x} - \boldsymbol{\mu}_0)^T$$

$$\mathbf{C}_1 = \frac{1}{n_1} \sum_{x \in X_1} (\mathbf{x} - \boldsymbol{\mu}_1)(\mathbf{x} - \boldsymbol{\mu}_1)^T$$

定义原样本空间 X_0 和 X_1 中的类内散度矩阵

$$\boldsymbol{\Sigma}_0 = \sum_{x \in X_0} (\mathbf{x} - \boldsymbol{\mu}_0)(\mathbf{x} - \boldsymbol{\mu}_0)^T,$$

$$\boldsymbol{\Sigma}_1 = \sum_{x \in X_1} (\mathbf{x} - \boldsymbol{\mu}_1)(\mathbf{x} - \boldsymbol{\mu}_1)^T.$$

则总类内散度矩阵 $\mathbf{S}_w = \boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1$, 可证明其是对称半正定的.

线性判别分析的目标

投影后第 i 内的类内散度为:

$$\begin{aligned}\tilde{s}_i^2 &= \sum_{y \in Y_i} (y - \tilde{\mu}_i)^2 \\ &= \sum_{y \in Y_i} (\boldsymbol{w}^T \boldsymbol{x} - \boldsymbol{w}^T \boldsymbol{\mu}_i)^2 \\ &= \sum_{\boldsymbol{x} \in X_i} \boldsymbol{w}^T (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x} - \boldsymbol{\mu}_i)^T \boldsymbol{w} = \boldsymbol{w}^T \boldsymbol{\Sigma}_i \boldsymbol{w}.\end{aligned}$$

故散度矩阵总和可写为 $\tilde{s}_0^2 + \tilde{s}_1^2 = \boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}$. 所以

$$J(\boldsymbol{w}) = \frac{|\tilde{\mu}_0 - \tilde{\mu}_1|^2}{\tilde{s}_0^2 + \tilde{s}_1^2} = \frac{\boldsymbol{w}^T \boldsymbol{S}_b \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}}.$$

线性判别分析的目标

给定数据集 $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \{0, 1\}$

第 i 类示例的集合 X_i

第 i 类示例的均值向量 $\boldsymbol{\mu}_i$

第 i 类示例的协方差矩阵 $\boldsymbol{\Sigma}_i$

两类样本的中心在直线上的投影: $\mathbf{w}^T \boldsymbol{\mu}_0$ 和 $\mathbf{w}^T \boldsymbol{\mu}_1$

两类样本的协方差: $\mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w}$ 和 $\mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}$

同类样例的投影点尽可能接近 $\rightarrow \mathbf{w}^T \mathbf{S}_w \mathbf{w} = \mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w} + \mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}$ 尽可能小

异类样例的投影点尽可能远离 $\rightarrow \mathbf{w}^T \mathbf{S}_b \mathbf{w} = \|\mathbf{w}^T \boldsymbol{\mu}_0 - \mathbf{w}^T \boldsymbol{\mu}_1\|_2^2$ 尽可能大

于是, 最大化目标函数

$$J(\mathbf{w}) = \frac{\|\mathbf{w}^T \boldsymbol{\mu}_0 - \mathbf{w}^T \boldsymbol{\mu}_1\|_2^2}{\mathbf{w}^T \boldsymbol{\Sigma}_0 \mathbf{w} + \mathbf{w}^T \boldsymbol{\Sigma}_1 \mathbf{w}} = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

线性判别分析-二分类任务

□ 最大化广义瑞利商 (generalized Rayleigh quotient)

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \quad (\mathbf{w} \text{ 成倍缩放不影响 } J \text{ 值})$$

$J(\mathbf{w})$ 是两类均值差的一种度量, 且由类内散度矩阵进行归一化

- 令 $\mathbf{w}^T \mathbf{S}_w \mathbf{w} = c \neq 0$, 最大化广义瑞利商等价形式为

$$\min_{\mathbf{w}} -\mathbf{w}^T \mathbf{S}_b \mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^T \mathbf{S}_w \mathbf{w} = c$$

- 运用拉格朗日乘子法(见《机器学习》教材P₄₀₃页)

$$L(\mathbf{w}, \lambda) = -\mathbf{w}^T \mathbf{S}_b \mathbf{w} + \lambda(\mathbf{w}^T \mathbf{S}_w \mathbf{w} - c)$$

$$\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = \frac{\partial L(\mathbf{w}, \lambda)}{\partial \mathbf{w}} = -2\mathbf{S}_b \mathbf{w} + 2\lambda \mathbf{S}_w \mathbf{w} = 0$$

$$\longrightarrow \mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

线性判别分析-二分类任务

□ 同向向量

$$\underbrace{S_b w}_{\text{同向向量}} = \underbrace{(\mu_0 - \mu_1)}_{\text{同向向量}} \underbrace{(\mu_0 - \mu_1)^T w}_{\text{标量}} = \lambda' (\mu_0 - \mu_1)$$

某个未知的 λ'

$$S_w^{-1} \lambda' (\mu_0 - \mu_1) = \lambda w \quad \longrightarrow \quad w = \lambda' \lambda^{-1} S_w^{-1} (\mu_0 - \mu_1)$$

当样本数大于样本特征维数, 即 $m > d$, S_w 通常非奇异

□ m维空间到一维空间投影轴的最佳方向

$$w^* = S_w^{-1} (\mu_0 - \mu_1) \quad (\text{因 } w \text{ 与大小无关, 只与方向有关})$$

□ J(w) 最大值

$$(\mu_0 - \mu_1)^T S_w^{-1} (\mu_0 - \mu_1)$$

□ 最佳投影变换为

$$y = (\mu_0 - \mu_1)^T S_w^{-1} x$$

线性判别分析-二分类任务

□ 求解. 实践中通常是进行奇异值分解 (《机器学习》(周志华)附录A)

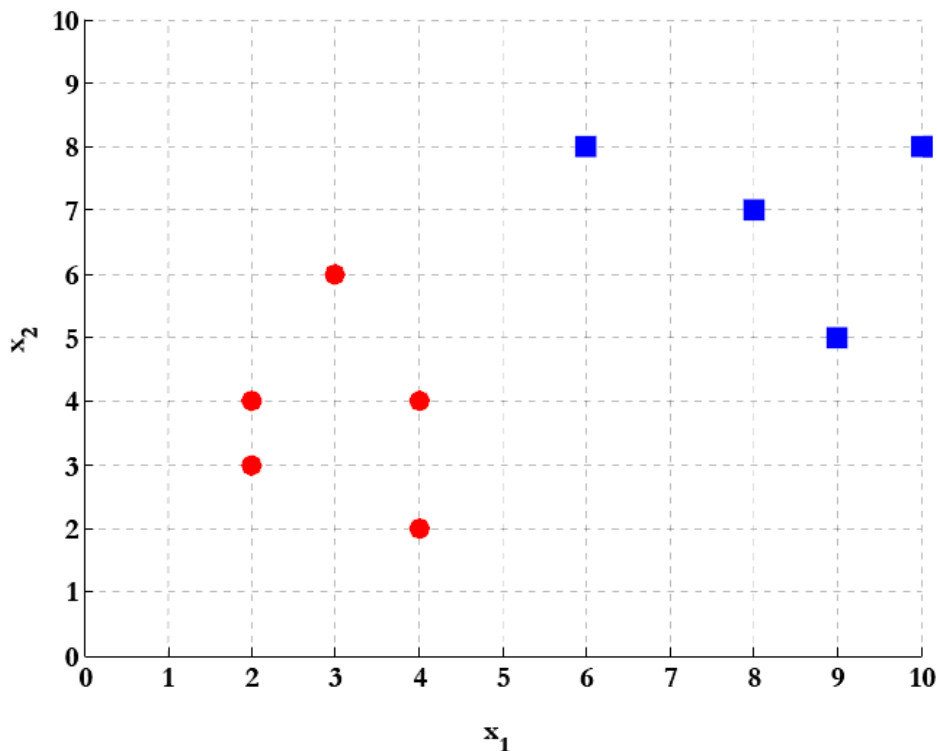
$$\mathbf{S}_w = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad w = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T(\mu_0 - \mu_1)$$

线性判别分析-二分类任务例子

由两类二维数据计算线性判别分析(LDA)投影向量

第一类采样数据 ω_1 : $\mathbf{X}_1=(x_1,x_2)=\{(4,2),(2,4),(2,3),(3,6),(4,4)\}$ (红色点)

第二类采样数据 ω_2 : $\mathbf{X}_2=(x_1,x_2)=\{(9,10),(6,8),(9,5),(8,7),(10,8)\}$ (蓝色点)



```
% samples for class 1  
X1 = [4,2;  
      2,4;  
      2,3;  
      3,6;  
      4,4];
```

```
% samples for class 2  
X2 = [9,10;  
      6,8;  
      9,5;  
      8,7;  
      10,8];
```

线性判别分析-二分类任务例子

两个类的均值为：

$$\mu_1 = \frac{1}{N_1} \sum_{x \in \omega_1} x = \frac{1}{5} \left[\begin{pmatrix} 4 \\ 2 \end{pmatrix} + \begin{pmatrix} 2 \\ 4 \end{pmatrix} + \begin{pmatrix} 2 \\ 3 \end{pmatrix} + \begin{pmatrix} 3 \\ 6 \end{pmatrix} + \begin{pmatrix} 4 \\ 4 \end{pmatrix} \right] = \begin{pmatrix} 3 \\ 3.8 \end{pmatrix}$$

$$\mu_2 = \frac{1}{N_2} \sum_{x \in \omega_2} x = \frac{1}{5} \left[\begin{pmatrix} 9 \\ 10 \end{pmatrix} + \begin{pmatrix} 6 \\ 8 \end{pmatrix} + \begin{pmatrix} 9 \\ 5 \end{pmatrix} + \begin{pmatrix} 8 \\ 7 \end{pmatrix} + \begin{pmatrix} 10 \\ 8 \end{pmatrix} \right] = \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}$$

```
% class means
```

```
Mu1 = mean(X1) ' ;
```

```
Mu2 = mean(X2) ' ;
```

线性判别分析-二分类任务例子

第一类样本的类内散度矩阵为：

$$\begin{aligned} S_1 &= \sum_{x \in \omega_1} (x - \mu_1)(x - \mu_1)^T = \left[\begin{pmatrix} 4 \\ 2 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 2 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 \\ &\quad + \left[\begin{pmatrix} 2 \\ 3 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 3 \\ 6 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 4 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 \\ &= \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix} \end{aligned}$$

```
% covariance matrix of the first class  
S1 = cov(X1);
```

线性判别分析-二分类任务例子

第二类样本的类内散度矩阵为：

$$\begin{aligned} S_2 &= \sum_{x \in \omega_2} (x - \mu_2)(x - \mu_2)^T = \left[\begin{pmatrix} 9 \\ 10 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 6 \\ 8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 \\ &\quad + \left[\begin{pmatrix} 9 \\ 5 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 8 \\ 7 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[\begin{pmatrix} 10 \\ 8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 \\ &= \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix} \end{aligned}$$

线性判别分析-二分类任务例子

总类内散度矩阵为：

$$\begin{aligned} S_w = S_1 + S_2 &= \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix} + \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix} \\ &= \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix} \end{aligned}$$

```
% within-class scatter matrix  
Sw = S1 + S2 ;
```

线性判别分析-二分类任务例子

类间散度矩阵为：

$$\begin{aligned} S_B &= (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \\ &= \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right] \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^T \\ &= \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix} \begin{pmatrix} -5.4 & -3.8 \end{pmatrix} \\ &= \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} \end{aligned}$$

```
% between-class scatter matrix  
SB = (Mu1-Mu2) * (Mu1-Mu2) ' ;
```

线性判别分析-二分类任务例子

直接计算 w ：

$$\begin{aligned}w^* &= S_W^{-1}(\mu_1 - \mu_2) = \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}^{-1} \left[\begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right] \\&= \begin{pmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{pmatrix} \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix} \\&= \begin{pmatrix} 0.9088 \\ 0.4173 \end{pmatrix}\end{aligned}$$

线性判别分析-二分类任务例子

或者LDA投影转化为以下广义特征值问题的解：

$$S_W^{-1}S_B w = \lambda w$$

$$\Rightarrow |S_W^{-1}S_B - \lambda I| = 0$$

$$\Rightarrow \left| \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}^{-1} \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

$$\Rightarrow \left| \begin{pmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{pmatrix} \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

$$\Rightarrow \left| \begin{pmatrix} 9.2213 - \lambda & 6.489 \\ 4.2339 & 2.9794 - \lambda \end{pmatrix} \right|$$

$$= (9.2213 - \lambda)(2.9794 - \lambda) - 6.489 \times 4.2339 = 0$$

$$\Rightarrow \lambda^2 - 12.2007\lambda = 0 \Rightarrow \lambda(\lambda - 12.2007) = 0$$

$$\Rightarrow \lambda_1 = 0, \lambda_2 = 12.2007$$

线性判别分析-二分类任务例子

因此：

$$\begin{pmatrix} 9.2213 & 6.489 \\ 4.2339 & 2.9794 \end{pmatrix} w_1 = 0 \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

$$\begin{pmatrix} 9.2213 & 6.489 \\ 4.2339 & 2.9794 \end{pmatrix} w_2 = \underbrace{12.2007}_{\lambda_2} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

$$w_1 = \begin{pmatrix} -0.5755 \\ 0.8178 \end{pmatrix} \text{ 和 } w_2 = \begin{pmatrix} 0.9088 \\ 0.4173 \end{pmatrix} = w^*$$

```
% computing the LDA projection
invSw = inv(Sw);

invSw_by_SB = invSw * SB;

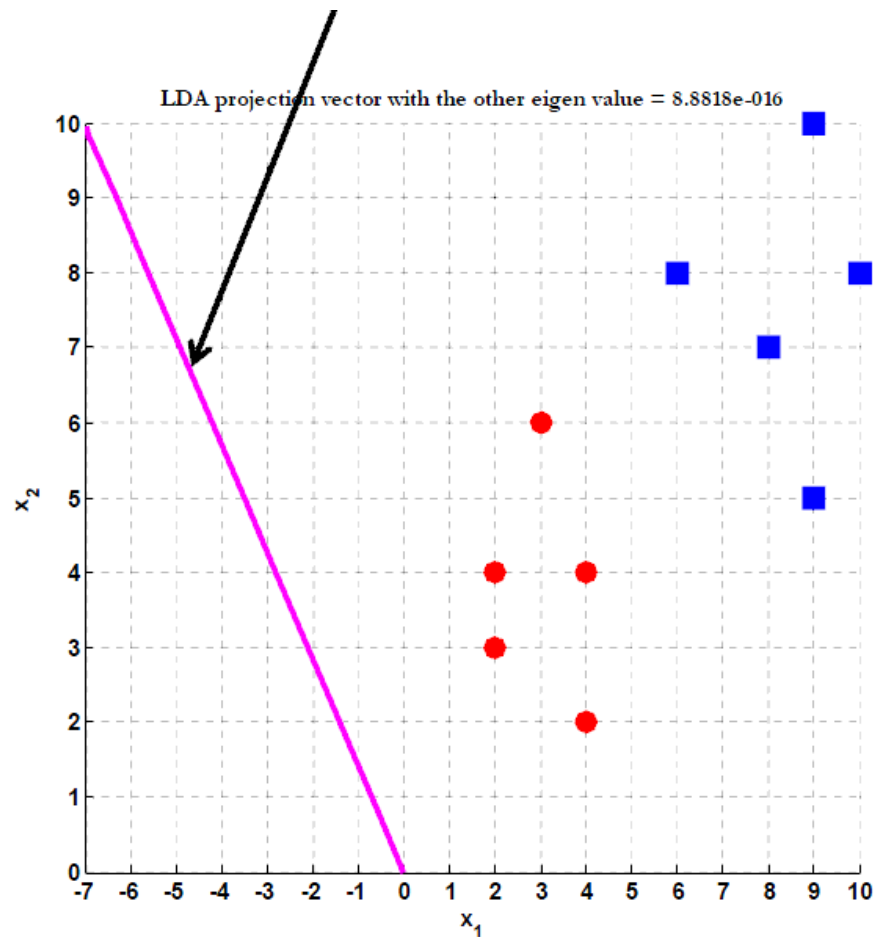
% getting the projection vector
[V,D] = eig(invSw_by_SB)

% the projection vector
W = V(:,1);
```

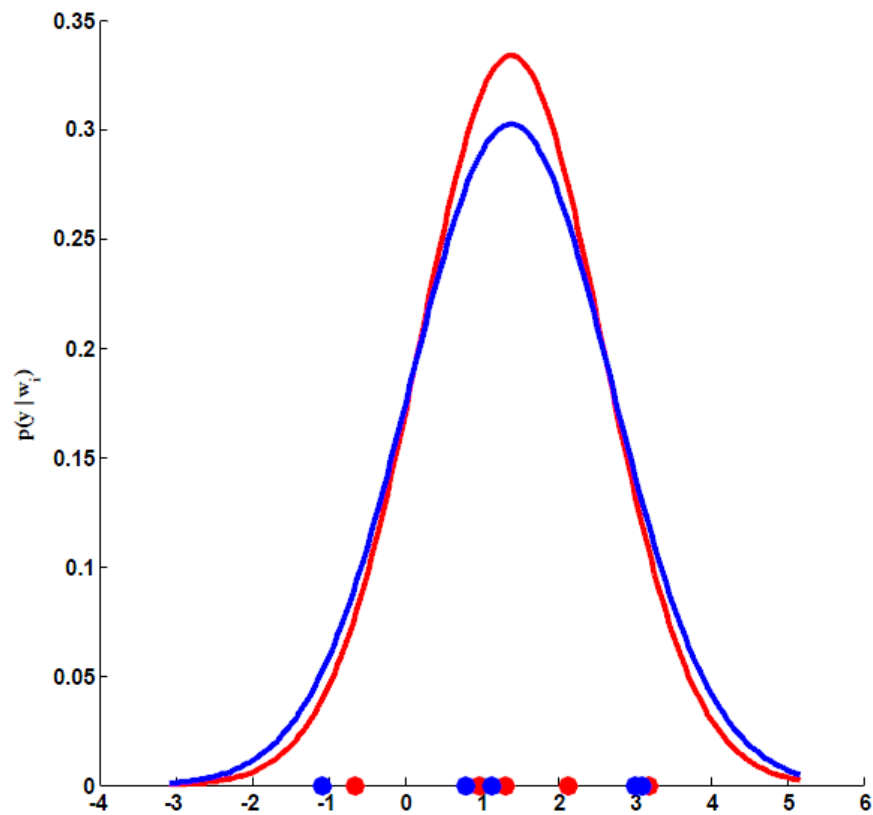
最优投影是使得 $\lambda = J(w)$ 达到最大时的 λ 对应的 w

线性判别分析-二分类任务例子

最小特征值 w_1 对应的投影向量



类PDF: 利用最小特征值=8.8818e-016的LDA投影

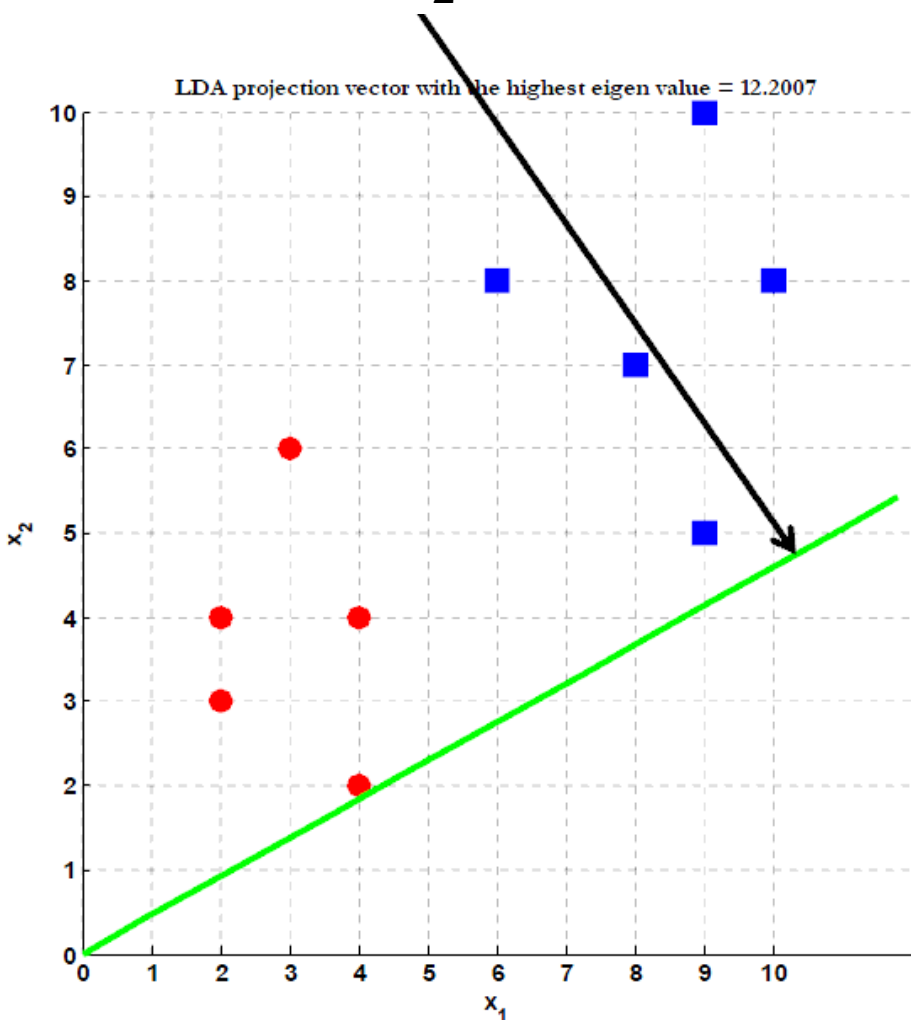


利用这种投影向量导致糟糕的两
类之间可分性

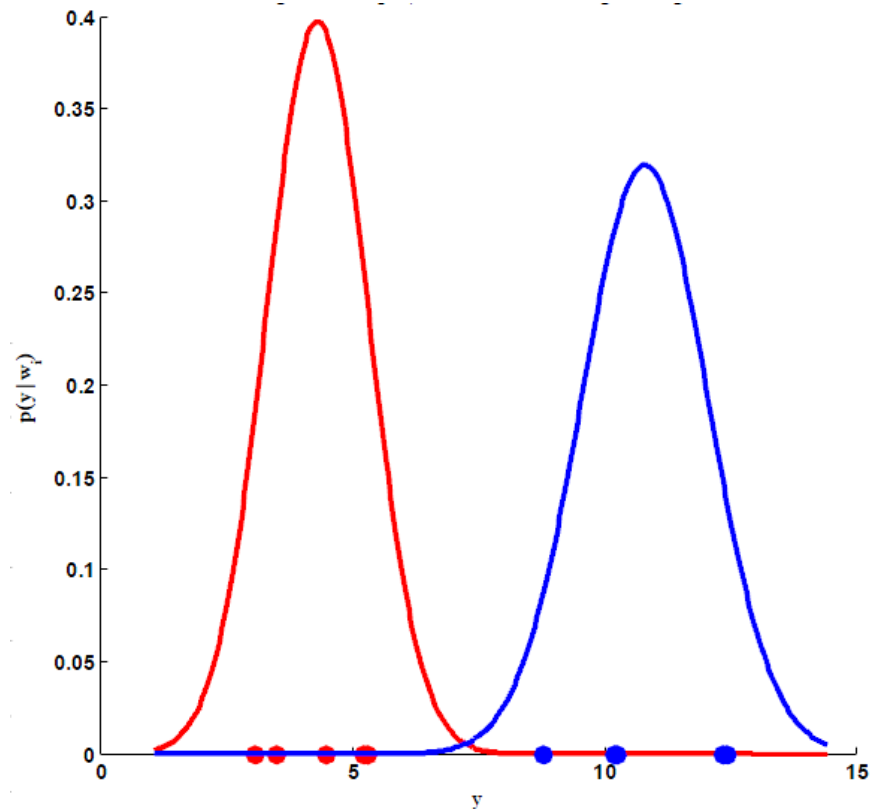
`mvnpdf(range, mean_vec, sigma)`

线性判别分析-二分类任务例子

最大特征值 w_2 对应的投影向量



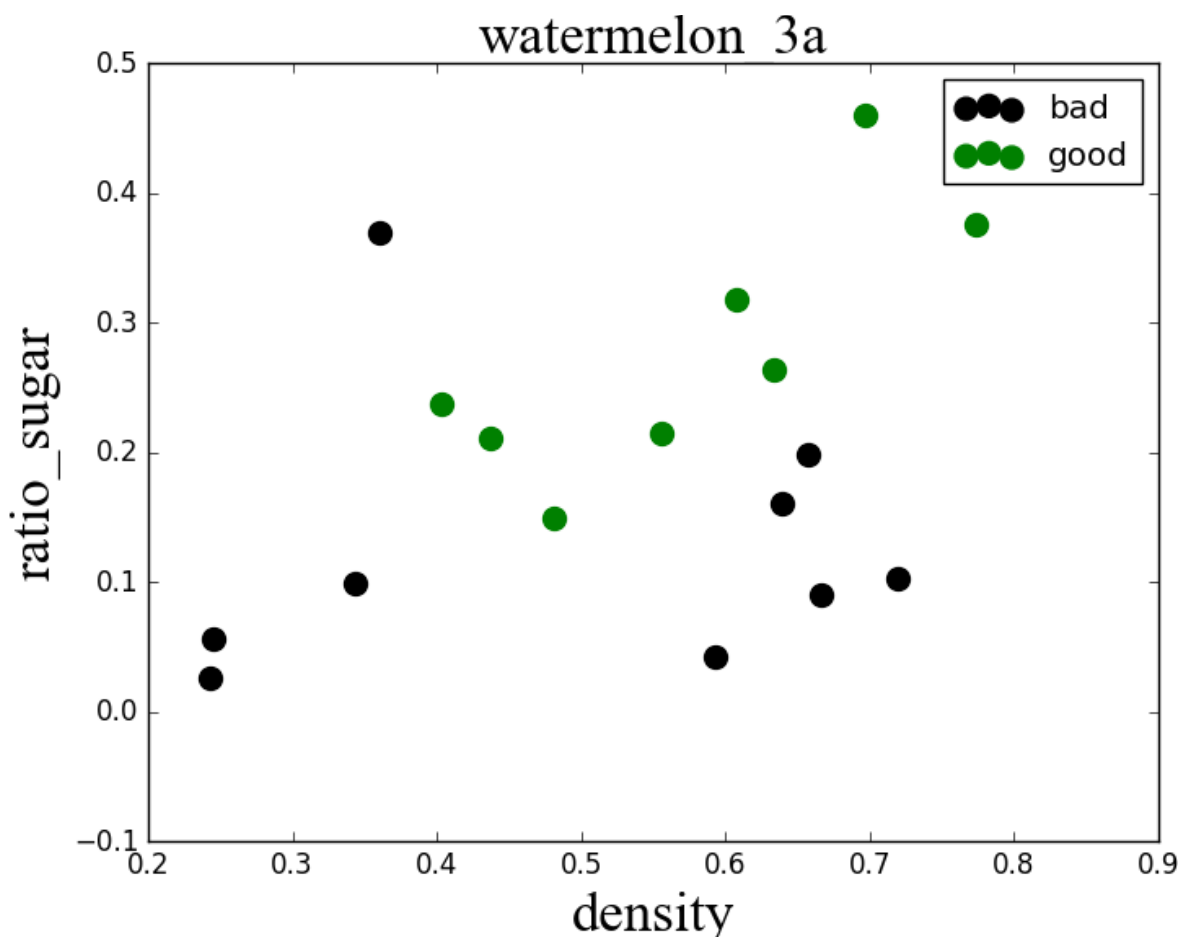
类PDF: 利用最大特征值=12.2007的LDA投影



利用这种投影向量导致很好的两类之间可分性

`mvnpdf(range, mean_vec, sigma)`

线性判别分析-西瓜数据二分类任务例子

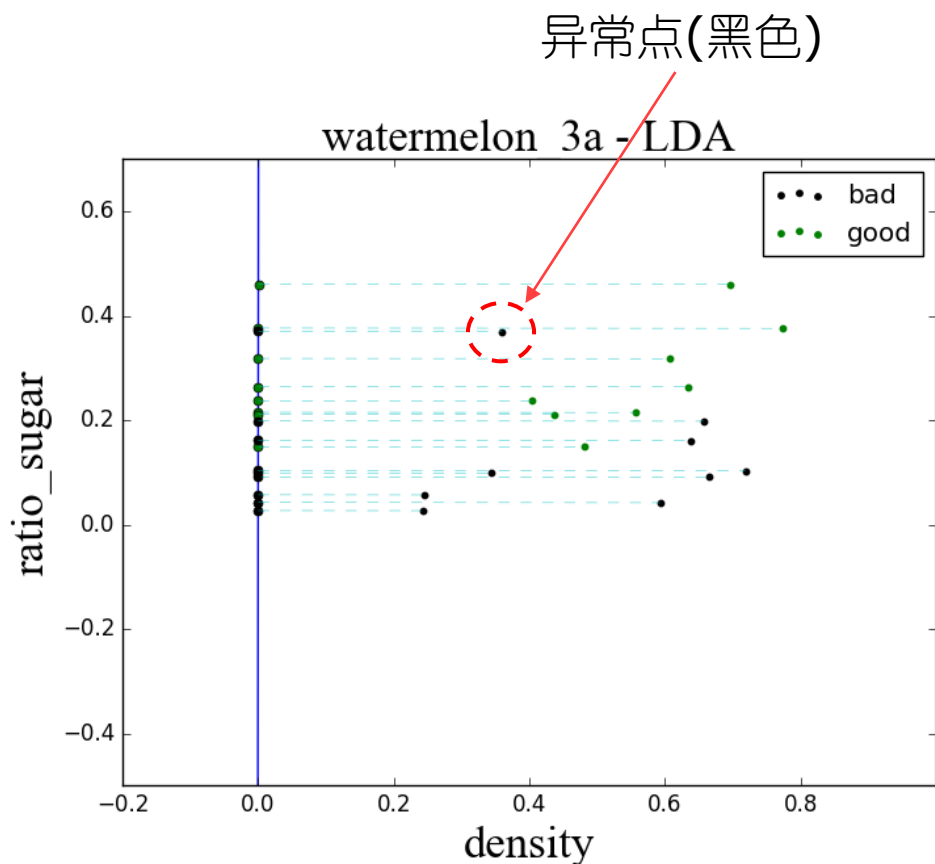


西瓜数据集3.0 α 可视化结果

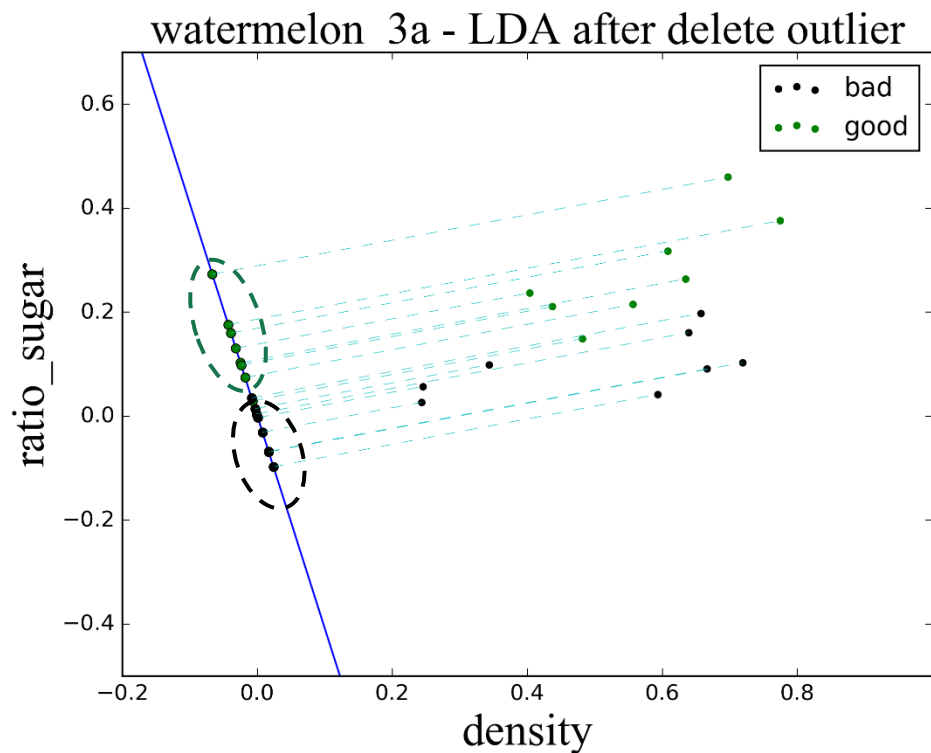
表 4.5 西瓜数据集 3.0 α

编号	密度	含糖率	好瓜
1	0.697	0.460	是
2	0.774	0.376	是
3	0.634	0.264	是
4	0.608	0.318	是
5	0.556	0.215	是
6	0.403	0.237	是
7	0.481	0.149	是
8	0.437	0.211	是
9	0.666	0.091	否
10	0.243	0.267	否
11	0.245	0.057	否
12	0.343	0.099	否
13	0.639	0.161	否
14	0.657	0.198	否
15	0.360	0.370	否
16	0.593	0.042	否
17	0.719	0.103	否

线性判别分析-二分类任务例子



LDA在**含有异常点**的西瓜数据集3.0 α 上的分类结果 (存在异常点的情况)



LDA在**去除异常点**后的西瓜数据集3.0 α 上的分类结果

- 编程实现线性判别分析，并给出西瓜数据集3.0 α 上的结果