第一次作业

1、请给出 Fisher 线性判别分析的主要计算步骤和分类决策规则

主要计算步骤:

输入：数据集 $D=(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)$,其中，任意样本 x_i 为 n 维向量，$y_i\in\{C_1,C_2,\ldots,C_k\}$,降维到的维度为 d .

输出：降维后的数据集 D' .

1）计算类内散度矩阵 S_w

2）计算类间散度矩阵 S_b

3）计算矩阵 $S_w^{-1}S_b$

4）计算矩阵 $S_w^{-1}S_b$ 的特征值与特征向量，按从小到大的顺序选取前 d 个特征值和对应的 d 个特征向量 $(w_1,w_2,\ldots,w_d)$ ，得到投影矩阵 W .

5）对样本集中的每一个样本特征 x_i ，转化为新的样本 $z_i=W^Tx_i$

6）得到输出样本集 $D'=\{(z_1,y_1),(z_2,y_2),\ldots,(z_m,y_m)\}$

分类决策规则: 根据投影点(project point)分类, 设置阈值 y0, 如果 y>y0, 则为 1; 如果 y<y0, 则为 0

2、从概率角度，试证明在噪声ε~$N(0,\sigma^2)$的条件下，最小二乘法等价于极大似然估计。

首先假设线性回归**模型**具有如下形式:

$$f(\mathbf{x})=\sum_{j=1}^{d}x_jw_j+\epsilon=\mathbf{x}\mathbf{w}^{\intercal}+\epsilon$$

其中 $\mathbf{x}\in\mathbb{R}^{1\times d}$ ， $\mathbf{w}\in\mathbb{R}^{1\times d}$ ， **误差** $\epsilon\in\mathbb{R}$ 。

当前已知 $\mathbf{X}=(\mathbf{x}_1\cdots\mathbf{x}_n)^{\intercal}\in\mathbb{R}^{n\times d}$ ， $\mathbf{y}\in\mathbb{R}^{n\times 1}$ ，怎样求 $\mathbf{w}$ 呢?

策略1. 假设 $\epsilon_i\sim\mathcal{N}(0,\sigma^2)$ ，也就是说 $\mathbf{y}_i\sim\mathcal{N}(\mathbf{x}_i\mathbf{w}^{\intercal},\sigma^2)$ ，那么用最大似然估计推导:

$$\begin{aligned}\arg\max_{\mathbf{w}}L(\mathbf{w})&=\ln\prod_{i=1}^{n}\frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{1}{2}(\frac{\mathbf{y}_i-\mathbf{x}_i\mathbf{w}^{\intercal}}{\sigma})^2)\\&=-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(\mathbf{y}_i-\mathbf{x}_i\mathbf{w}^{\intercal})^2-n\ln\sigma\sqrt{2\pi}\\\arg\min_{\mathbf{w}}f(\mathbf{w})&=\sum_{i=1}^{n}(\mathbf{y}_i-\mathbf{x}_i\mathbf{w}^{\intercal})^2=\|\mathbf{y}-\mathbf{X}\mathbf{w}^{\intercal}\|_2^2\end{aligned}$$

这不就是最小二乘么。

3.7 令码长为 9，类别数为 4，试给出海明距离意义下理论最优的 ECOC二元码并证明之。

答：

原书对很多地方解释没有解释清楚，把原论文看了一下《Solving Multiclass Learning Problems via Error-Correcting Output Codes》。

先把几个涉及到的理论解释一下。

首先原书中提到：

> 对同等长度的编码，理论上来说，任意两个类别之间的编码距离越远，则纠错能力越强。因此，在码长较小时可根据这个原则计算出理论最优编码。

其实这一点在论文中也提到，“假设任意两个类别之间最小的海明距离为 d ，那么此纠错输出码最少能矫正 $\left[\frac{d-1}{2}\right]$ 位的错误。

Class	Code Word							
	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
c_0	0	0	0	0	1	1	1	1
c_1	0	0	1	1	0	0	1	1
c_2	0	1	0	1	0	1	0	1

拿上图论文中的例子解释一下，上图中，所有类别之间的海明距离都为4，假设一个样本正确的类别为 $c1$ ，那么codeword应该为 ‘0 0 1 1 0 0 1 1’ ，若此时有一个分类器输出错误，变成 ‘0 0 0 1 0 0 1 1’ ，那么此时距离最近的仍然为 $c1$ ，若有两个分类输出错误如 ‘0 0 0 0 0 0 1 1’ ，此时与 $c1, c2$ 的海明距离都为2，无法正确分类。即任意一个分类器将样本分类错误，最终结果依然正确，但如果有两个以上的分类器错误，结果就不一定正确了。这是 $\left[\frac{d-1}{2}\right]$ 的由来。

此外，原论文中提到，一个好的纠错输出码应该满足两个条件：

1. 行分离。任意两个类别之间的codeword距离应该足够大。
2. 列分离。任意两个分类器 f_i, f_j 的输出应相互独立，无关联。这一点可以通过使分类器 f_i 编码与其他分类编码的海明距离足够大实现，且与其他分类**编码的反码的海明距离也足够大（有点绕。）。**

第一点其实就是原书提到的，已经解释过了，说说第二点：

> 如果两个分类器的编码类似或者完全一致，很多算法（比如C4.5）会有相同或者类似的错误分类，如果这种同时发生的错误过多，会导致纠错输出码失效。（翻译原论文）

个人理解就是：若增加两个类似的编码，那么当误分类时，就从原来的1变成3，导致与真实类别的codeword海明距离增长。极端情况，假设增加两个相同的编码，此时任意两个类别之间最小的海明距离不会变化依然为 d ，而纠错输出码输出的codeword与真实类别的codeword的海明距离激增（从1变成3）。所以如果有过多同时发出的错误分类，会导致纠错输出码失效。

另外，两个分类器的编码也不应该互为反码，因为很多算法（比如C4.5，逻辑回归）对待0-1分类其实是对称的，即将0-1类互换，最终训练出的模型是一样的。也就是说两个编码互为补码的分类器是会同时犯错的。同样也会导致纠错输出码失效。

当然当类别较少时，很难满足上面这些条件。如上图中，一共有三类，那么只有 $2^3 = 8$ 中可能的分类器编码（ $f_0 - f7$ ），其中后四种（ $f_4 - f_7$ ）是前四种的反码，都应去除，再去掉全为0的 f_0 ，就只剩下三种编码选择了，所以很难满足上述的条件。事实上，对于 k 种类别的分类，再去除反码和全是0或者1的编码后，就剩下 $2^k - 1$ 中可行的编码。

原论文中给出了构造编码的几种方法。其中一个是：

2.3.1 Exhaustive Codes

When $3 \le k \le 7$, we construct a code of length $2^{k-1} - 1$ as follows. Row 1 is all ones. Row 2 consists of 2^{k-2} zeroes followed by $2^{k-2} - 1$ ones. Row 3 consists of 2^{k-3} zeroes, followed by 2^{k-3} ones, followed by 2^{k-3} zeroes, followed by $2^{k-3} - 1$ ones. In row i, there are alternating runs of 2^{k-i} zeroes and ones. Table 6 shows the exhaustive code for a five-class problem. This code has inter-row Hamming distance 8; no columns are identical or complementary.

回到题目上，在类别为4时，其可行的编码有7种，按照上述方法有：

	f0	f1	f2	f3	f4	f5	f6
c1	1	1	1	1	1	1	1
c2	0	0	0	0	1	1	1
c3	0	0	1	1	0	0	1
c4	0	1	0	1	0	1	0

当码长为9时，那么 f_6 之后加任意两个编码，即为最优编码，因为此时再加任意的编码都是先有编码的反码，此时，类别之间最小的海明距离都为4，不会再增加。

4、根据下表数据，基于信息增益（即，ID3 算法）构造决策树。

属性				决策目标
天气	温度	湿度	刮风	是否打篮球
晴天	高	中	否	否
晴天	高	中	是	否
阴天	高	高	否	是
小雨	高	高	否	是
小雨	低	高	否	否
晴天	中	中	是	是
阴天	中	高	是	否

注意：该题不止这一种决策树的划分，由于后面有**信息增益**相同的情况，每个人的选择不一样，所以该题不止一种决策树的划分。

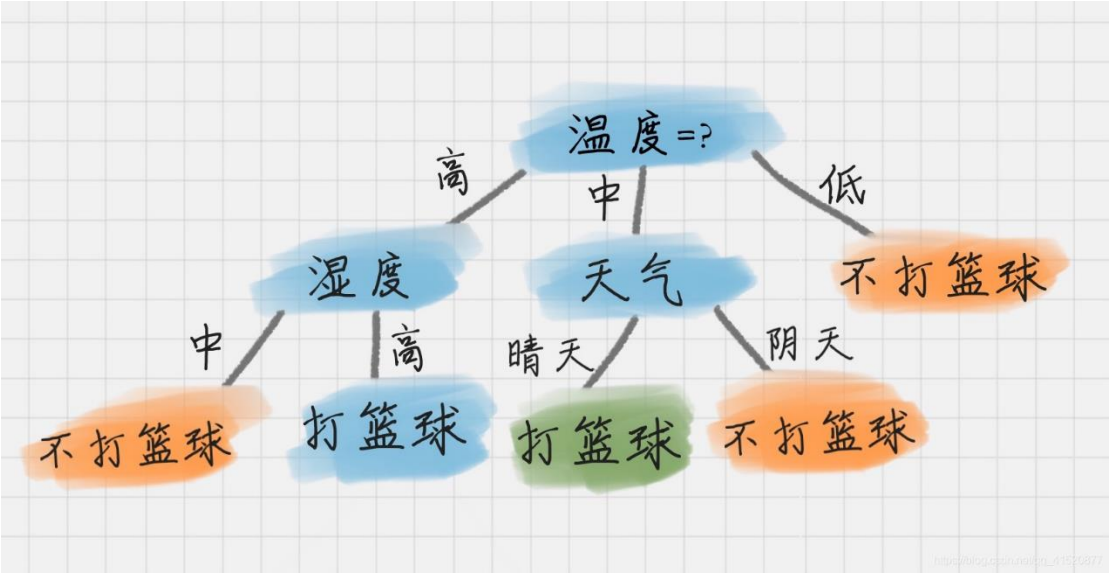
因为我们用 ID3 中的信息增益来构造决策树，所以要计算每个节点的信息增益。
天气作为属性节点的信息增益为，Gain(D，天气)=0.985-0.965=0.020。。
同理我们可以计算出其他属性作为根节点的信息增益，它们分别为 ：

Gain(D，温度)=0.128
Gain(D，湿度)=0.020
Gain(D，刮风)=0.020

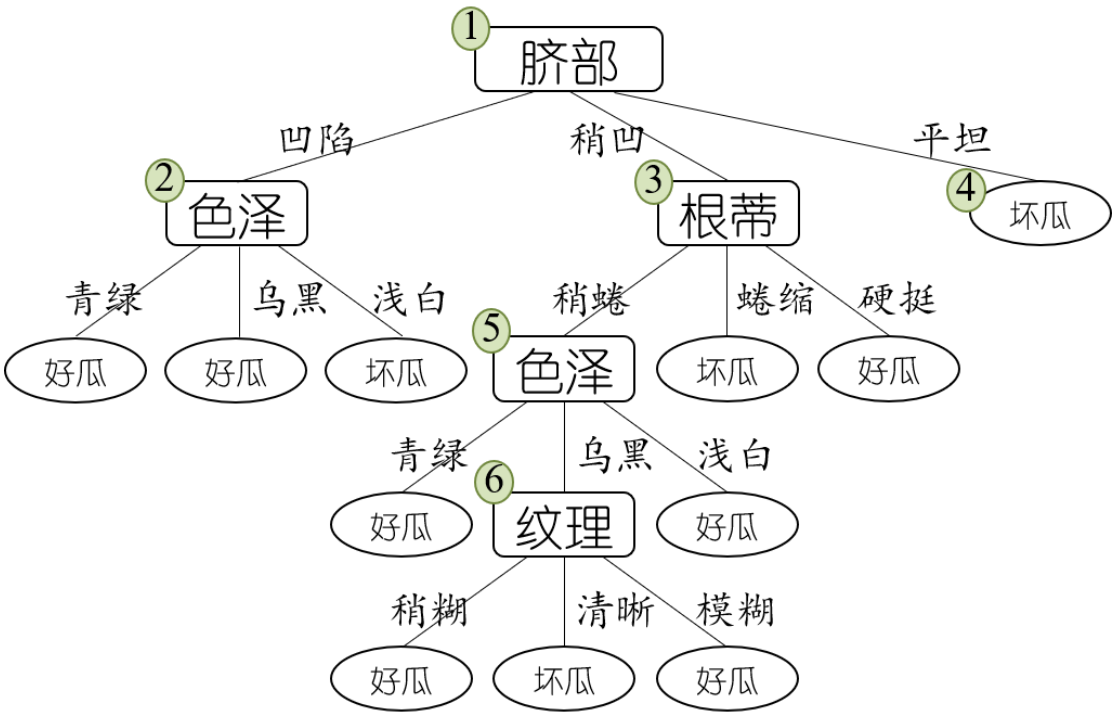
我们能看出来温度作为属性的信息增益最大。因为 ID3 就是要将信息增益最大的节点作为父节点，这样可以得到纯度高的决策树，所以我们将温度作为根节点。

紧接着，温度高时，计算湿度和天气的信息增益相同，任选其一作为分支即可。

其决策树状图分裂为下图所示：



5.（简答题，5.0 分）请从预剪枝和后剪枝的角度分别分析下图（图 4.5）中节点⑤是否应该为叶子节点，并说明原因。



预剪枝：剪枝前后精度相同，禁止划分，即剪枝，是叶节点
后剪枝：剪枝前后精度相同，不剪枝，不是叶节点

6、计算题，5.0 分）

（1）根据下表数据（表 4.4），计算“纹理”属性的信息增益；

答：Gain (D, texture) = 0.424

（2） 已知“纹理”属性取得了最大的信息增益，请画出其分支结构，并写出样本权值。

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	–	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	–	是
3	乌黑	蜷缩	–	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	–	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	–	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	–	稍凹	硬滑	是
9	乌黑	–	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	–	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	–	否
12	浅白	蜷缩	–	模糊	平坦	软粘	否
13	–	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	–	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	–	沉闷	稍糊	稍凹	硬滑	否

答：清晰 7/15、稍糊 5/15、模糊 3/15

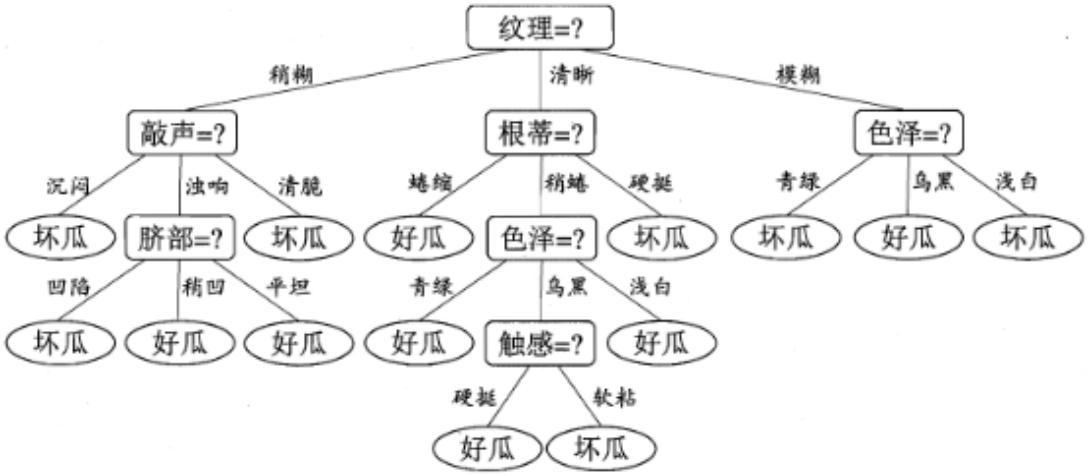


图 4.9　在西瓜数据集 2.0α 上基于信息增益生成的决策树

第二次作业

Q1:

作业

大连理工大学 人工智能学院 School of Artificial Intelligence, Dalian University of Technology

1. （习题8.1）一个抛硬币时一个硬币A面朝上的概率为p，B面朝上的概率则为1-p。我们抛n次硬币，那么A面朝上次数的期望值为np。那么进一步我们可以知道，A面朝上的次数不超过k次的概率能够被下面的表达式完全确定：

$$\mathbb{P}(H(n) \leq k) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}$$

这里的H(n)为抛n次硬币其A面朝上的次数。

对ε>0，k=(p-ε)n，有霍夫丁不等式：$\mathbb{P}(H(n) \leq (p-\varepsilon)n) \leq \exp\left(-2\varepsilon^2 n\right)$

试推导出ppt第8页公式（8.3），即 $P(H(\boldsymbol{x}) \neq f(\boldsymbol{x})) = \sum_{k=0}^{\lfloor T/2 \rfloor} \binom{T}{k} (1-\epsilon)^k \epsilon^{T-k} \leq \exp\left(-\frac{1}{2} T (1-2\epsilon)^2\right)$

A1:

公式 (8.3)

$$P(H(\boldsymbol{x}) \neq f(\boldsymbol{x})) = \sum_{k=0}^{\lfloor T/2 \rfloor} \binom{T}{k} (1-\epsilon)^k \epsilon^{T-k} \leqslant \exp\left(-\frac{1}{2} T (1-2\epsilon)^2\right)$$

[推导]：由基分类器相互独立，假设随机变量 X 为 T 个基分类器分类正确的次数，因此 $\mathrm{X} \sim \mathrm{B}(\mathrm{T}, 1-\epsilon)$，设 x_i 为每一个分类器分类正确的次数，则 $x_i \sim \mathrm{B}(1, 1-\epsilon)$ $i = 1\,2\,3 \ldots \mathrm{T}$，那么有

$$\mathrm{X} = \sum_{i=1}^{\mathrm{T}} x_i$$

$$\mathbb{E}(X) = \sum_{i=1}^{\mathrm{T}} \mathbb{E}(x_i) = (1-\epsilon) T$$

证明过程如下：

$$\begin{aligned} P(H(x) \neq f(x)) &= P(X \leq \lfloor T/2 \rfloor) \\ &\leqslant P(X \leq T/2) \\ &= P\left[X - (1-\epsilon)T \leqslant \frac{T}{2} - (1-\epsilon)T\right] \\ &= P\left[X - (1-\epsilon)T \leqslant -\frac{T}{2}(1-2\epsilon)\right] \\ &= P\left[\sum_{i=1}^{\mathrm{T}} x_i - \sum_{i=1}^{\mathrm{T}} \mathbb{E}(x_i) \leqslant -\frac{T}{2}(1-2\epsilon)\right] \\ &= P\left[\frac{1}{\mathrm{T}} \sum_{i=1}^{\mathrm{T}} x_i - \frac{1}{\mathrm{T}} \sum_{i=1}^{\mathrm{T}} \mathbb{E}(x_i) \leqslant -\frac{1}{2}(1-2\epsilon)\right] \end{aligned}$$

根据 Hoeffding 不等式知

$$P\left(\frac{1}{m}\sum_{i=1}^{m} x_i - \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}(x_i) \leqslant -\delta\right) \leqslant \exp\left(-2m\delta^2\right)$$

令 $\delta = \frac{(1-2\epsilon)}{2}, m = T$ 得

$$P(H(\boldsymbol{x}) \neq f(\boldsymbol{x})) = \sum_{k=0}^{\lfloor T/2 \rfloor} \binom{T}{k} (1-\epsilon)^k \epsilon^{T-k} \leqslant \exp\left(-\frac{1}{2} T (1-2\epsilon)^2\right)$$

Q2:

例 8.1 给定如表 8.1 所示训练数据. 假设弱分类器由 $x<v$ 或 $x>v$ 产生，其阈值 v 使该分类器在训练数据集上分类误差率最低. 试用 AdaBoost 算法学习一个强分类器.

表 8.1 训练数据表

序号	1	2	3	4	5	6	7	8	9	10
x	0	1	2	3	4	5	6	7	8	9
y	1	1	1	−1	−1	−1	1	1	1	−1

解 初始化数据权值分布

$$D_1=(w_{11},w_{12},\cdots,w_{110})$$
$$w_{1i}=0.1,\quad i=1,2,\cdots,10$$

对 $m=1$，

（a）在权值分布为 D_1 的训练数据上，阈值 v 取 2.5 时分类误差率最低，故基本分类器为

$$G_1(x)=\begin{cases}1, & x<2.5\\ -1, & x>2.5\end{cases}$$

（b）$G_1(x)$ 在训练数据集上的误差率 $e_1=P(G_1(x_i)\neq y_i)=0.3$.

（c）计算 $G_1(x)$ 的系数：$\alpha_1=\frac{1}{2}\log\frac{1-e_1}{e_1}=0.4236$.

（d）更新训练数据的权值分布：

$$D_2=(w_{21},\cdots,w_{2i},\cdots,w_{210})$$
$$w_{2i}=\frac{w_{1i}}{Z_1}\exp(-\alpha_1 y_i G_1(x_i)),\quad i=1,2,\cdots,10$$
$$D_2=(0.0715,0.0715,0.0715,0.0715,0.0715,0.0715,$$
$$0.1666,0.1666,0.1666,0.0715)$$
$$f_1(x)=0.4236G_1(x)$$

分类器 $\mathrm{sign}[f_1(x)]$ 在训练数据集上有 3 个误分类点.

对 $m=2$，

（a）在权值分布为 D_2 的训练数据上，阈值 v 是 8.5 时分类误差率最低，基本分类器为

$$G_2(x)=\begin{cases}1, & x<8.5\\ -1, & x>8.5\end{cases}$$

（b）$G_2(x)$ 在训练数据集上的误差率 $e_2=0.2143$.

（c）计算 $\alpha_2=0.6496$.

（d）更新训练数据权值分布：

$$D_3=(0.0455,\ 0.0455,\ 0.0455,\ 0.1667,\ 0.1667,\ 0.1667,$$
$$0.1060,\ 0.1060,\ 0.1060,\ 0.0455)$$
$$f_2(x)=0.4236G_1(x)+0.6496G_2(x)$$

分类器 $\mathrm{sign}[f_2(x)]$ 在训练数据集上有 3 个误分类点.

对 $m=3$，

（a）在权值分布为 D_3 的训练数据上，阈值 v 是 5.5 时分类误差率最低，基本分类器为

$$G_3(x)=\begin{cases}1, & x>5.5\\ -1, & x<5.5\end{cases}$$

（b）$G_3(x)$ 在训练样本集上的误差率 $e_3=0.1820$.

（c）计算 $\alpha_3=0.7514$.

（d）更新训练数据的权值分布：

$$D_4=(0.125,0.125,0.125,0.102,0.102,0.102,0.065,0.065,0.065,0.125)$$

于是得到：

$$f_3(x)=0.4236G_1(x)+0.6496G_2(x)+0.7514G_3(x)$$

分类器 $\mathrm{sign}[f_3(x)]$ 在训练数据集上误分类点个数为 0.

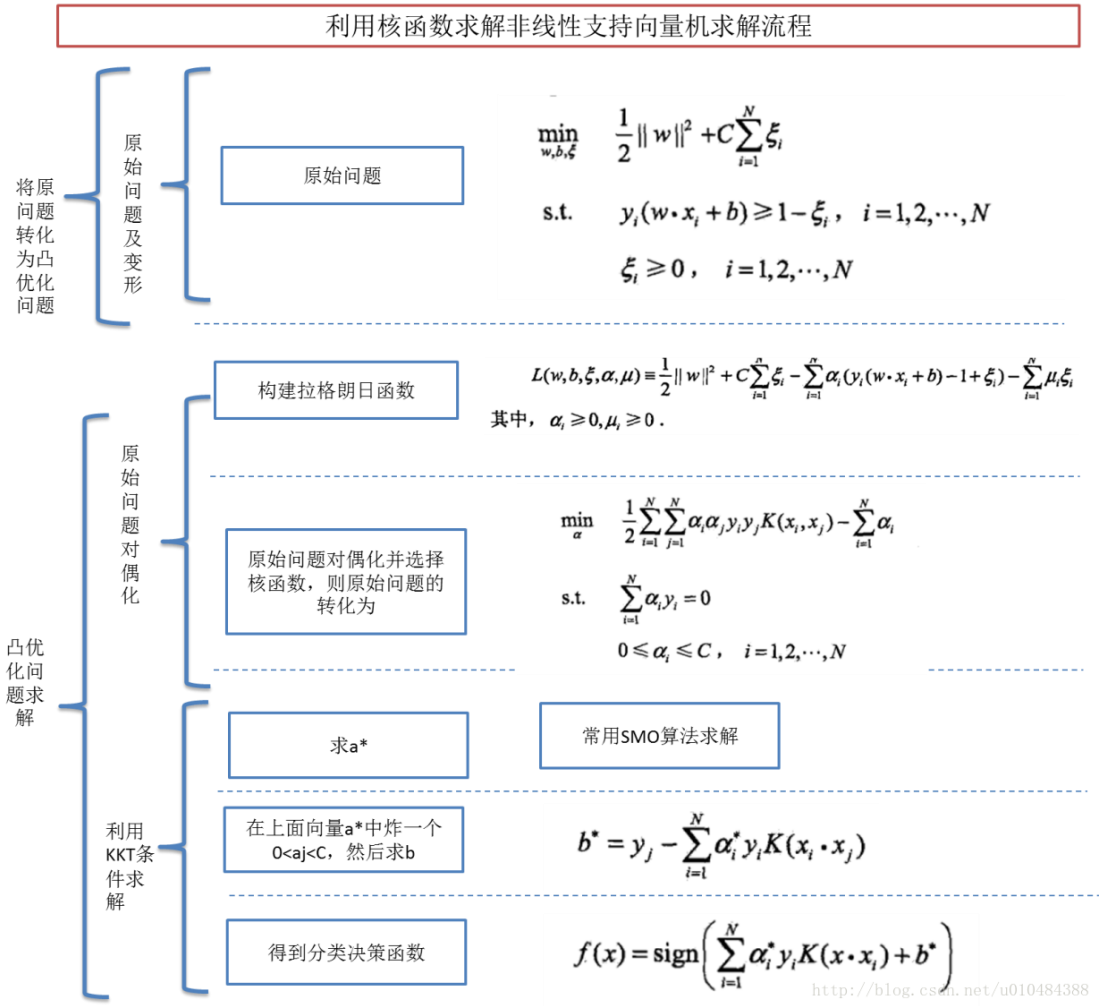
于是最终分类器为

$$G(x)=\mathrm{sign}[f_3(x)]=\mathrm{sign}[0.4236G_1(x)+0.6496G_2(x)+0.7514G_3(x)]$$

第三次作业

Q1：(简答题, 5 分) 总结非线性 SVM 算法输入：输出：计算步骤：

A1：



Q2：(简答题, 5 分) SVM 对噪声是否敏感，并分析其原因。

A2：SVM 的决策只基于少量的支持向量，若噪音样本出现在支持向量中，容易对决策造成影响，所以 SVM 对噪音敏感。

Q3：（简答题, 0.1 分）试使用核技巧推广 LDA，产生“KLDA”。【选做，但核技巧并非不考！】

A3：

Kernel trick with LDA [edit]

To extend LDA to non-linear mappings, the data, given as the ℓ points $\mathbf{x}_i$, can be mapped to a new feature space, F, via some function ϕ. In this new feature space, the function that needs to be maximized is[1]

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_B^{\phi}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_W^{\phi}\mathbf{w}},$$

where

$$\mathbf{S}_B^{\phi} = \left(\mathbf{m}_2^{\phi} - \mathbf{m}_1^{\phi}\right)\left(\mathbf{m}_2^{\phi} - \mathbf{m}_1^{\phi}\right)^{\mathrm{T}}$$

$$\mathbf{S}_W^{\phi} = \sum_{i=1,2}\sum_{n=1}^{l_i}\left(\phi(\mathbf{x}_n^i) - \mathbf{m}_i^{\phi}\right)\left(\phi(\mathbf{x}_n^i) - \mathbf{m}_i^{\phi}\right)^{\mathrm{T}},$$

and

$$\mathbf{m}_i^{\phi} = \frac{1}{l_i}\sum_{j=1}^{l_i}\phi(\mathbf{x}_j^i).$$

Further, note that $\mathbf{w} \in F$. Explicitly computing the mappings $\phi(\mathbf{x}_i)$ and then performing LDA can be computationally expensive, and in many cases intractable. For example, F may be infinitely dimensional. Thus, rather than explicitly mapping the data to F, the data can be implicitly embedded by rewriting the algorithm in terms of dot products and using the kernel trick in which the dot product in the new feature space is replaced by a kernel function, $k(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{y})$.

LDA can be reformulated in terms of dot products by first noting that $\mathbf{w}$ will have an expansion of the form[5]

$$\mathbf{w} = \sum_{i=1}^{l}\alpha_i\phi(\mathbf{x}_i).$$

Then note that

$$\mathbf{w}^{\mathrm{T}}\mathbf{m}_i^{\phi} = \frac{1}{l_i}\sum_{j=1}^{l}\sum_{k=1}^{l_i}\alpha_j k\left(\mathbf{x}_j, \mathbf{x}_k^i\right) = \alpha^{\mathrm{T}}\mathbf{M}_i,$$

where

$$(\mathbf{M}_i)_j = \frac{1}{l_i}\sum_{k=1}^{l_i}k(\mathbf{x}_j, \mathbf{x}_k^i).$$

The numerator of $J(\mathbf{w})$ can then be written as:

$$\mathbf{w}^{\mathrm{T}}\mathbf{S}_B^{\phi}\mathbf{w} = \mathbf{w}^{\mathrm{T}}\left(\mathbf{m}_2^{\phi} - \mathbf{m}_1^{\phi}\right)\left(\mathbf{m}_2^{\phi} - \mathbf{m}_1^{\phi}\right)^{\mathrm{T}}\mathbf{w} = \alpha^{\mathrm{T}}\mathbf{M}\alpha, \qquad \text{where} \qquad \mathbf{M} = (\mathbf{M}_2 - \mathbf{M}_1)(\mathbf{M}_2 - \mathbf{M}_1)^{\mathrm{T}}.$$

Similarly, the denominator can be written as

$$\mathbf{w}^{\mathrm{T}}\mathbf{S}_W^{\phi}\mathbf{w} = \alpha^{\mathrm{T}}\mathbf{N}\alpha, \qquad \text{where} \qquad \mathbf{N} = \sum_{j=1,2}\mathbf{K}_j(\mathbf{I} - \mathbf{1}_{l_j})\mathbf{K}_j^{\mathrm{T}},$$

with the $n^{\text{th}}, m^{\text{th}}$ component of $\mathbf{K}_j$ defined as $k(\mathbf{x}_n, \mathbf{x}_m^j)$, $\mathbf{I}$ is the identity matrix, and $\mathbf{1}_{l_j}$ the matrix with all entries equal to $1/l_j$. This identity can be derived by starting out with the expression for $\mathbf{w}^{\mathrm{T}}\mathbf{S}_W^{\phi}\mathbf{w}$ and using the expansion of $\mathbf{w}$ and the definitions of $\mathbf{S}_W^{\phi}$ and $\mathbf{m}_i^{\phi}$

$$\begin{aligned}
\mathbf{w}^{\mathrm{T}}\mathbf{S}_W^{\phi}\mathbf{w} &= \left(\sum_{i=1}^{l}\alpha_i\phi^{\mathrm{T}}(\mathbf{x}_i)\right)\left(\sum_{j=1,2}\sum_{n=1}^{l_j}\left(\phi(\mathbf{x}_n^j) - \mathbf{m}_j^{\phi}\right)\left(\phi(\mathbf{x}_n^j) - \mathbf{m}_j^{\phi}\right)^{\mathrm{T}}\right)\left(\sum_{k=1}^{l}\alpha_k\phi(\mathbf{x}_k)\right) \\
&= \sum_{j=1,2}\sum_{i=1}^{l}\sum_{n=1}^{l_j}\sum_{k=1}^{l}\left(\alpha_i\phi^{\mathrm{T}}(\mathbf{x}_i)\left(\phi(\mathbf{x}_n^j) - \mathbf{m}_j^{\phi}\right)\left(\phi(\mathbf{x}_n^j) - \mathbf{m}_j^{\phi}\right)^{\mathrm{T}}\alpha_k\phi(\mathbf{x}_k)\right) \\
&= \sum_{j=1,2}\sum_{i=1}^{l}\sum_{n=1}^{l_j}\sum_{k=1}^{l}\left(\alpha_i k(\mathbf{x}_i, \mathbf{x}_n^j) - \frac{1}{l_j}\sum_{p=1}^{l_j}\alpha_i k(\mathbf{x}_i, \mathbf{x}_p^j)\right)\left(\alpha_k k(\mathbf{x}_k, \mathbf{x}_n^j) - \frac{1}{l_j}\sum_{q=1}^{l_j}\alpha_k k(\mathbf{x}_k, \mathbf{x}_q^j)\right) \\
&= \sum_{j=1,2}\left(\sum_{i=1}^{l}\sum_{n=1}^{l_j}\sum_{k=1}^{l}\left(\alpha_i\alpha_k k(\mathbf{x}_i, \mathbf{x}_n^j)k(\mathbf{x}_k, \mathbf{x}_n^j) - \frac{2\alpha_i\alpha_k}{l_j}\sum_{p=1}^{l_j}k(\mathbf{x}_i, \mathbf{x}_n^j)k(\mathbf{x}_k, \mathbf{x}_p^j) + \frac{\alpha_i\alpha_k}{l_j^2}\sum_{p=1}^{l_j}\sum_{q=1}^{l_j}k(\mathbf{x}_i, \mathbf{x}_p^j)k(\mathbf{x}_k, \mathbf{x}_q^j)\right)\right) \\
&= \sum_{j=1,2}\left(\sum_{i=1}^{l}\sum_{n=1}^{l_j}\sum_{k=1}^{l}\left(\alpha_i\alpha_k k(\mathbf{x}_i, \mathbf{x}_n^j)k(\mathbf{x}_k, \mathbf{x}_n^j) - \frac{\alpha_i\alpha_k}{l_j}\sum_{p=1}^{l_j}k(\mathbf{x}_i, \mathbf{x}_n^j)k(\mathbf{x}_k, \mathbf{x}_p^j)\right)\right) \\
&= \sum_{j=1,2}\alpha^{\mathrm{T}}\mathbf{K}_j\mathbf{K}_j^{\mathrm{T}}\alpha - \alpha^{\mathrm{T}}\mathbf{K}_j\mathbf{1}_{l_j}\mathbf{K}_j^{\mathrm{T}}\alpha \\
&= \alpha^{\mathrm{T}}\mathbf{N}\alpha.
\end{aligned}$$

With these equations for the numerator and denominator of $J(\mathbf{w})$, the equation for J can be rewritten as

$$J(\alpha) = \frac{\alpha^{\mathrm{T}}\mathbf{M}\alpha}{\alpha^{\mathrm{T}}\mathbf{N}\alpha}.$$

Then, differentiating and setting equal to zero gives

$$(\alpha^{\mathrm{T}}\mathbf{M}\alpha)\mathbf{N}\alpha = (\alpha^{\mathrm{T}}\mathbf{N}\alpha)\mathbf{M}\alpha.$$

Since only the direction of $\mathbf{w}$, and hence the direction of α, matters, the above can be solved for α as

$$\alpha = \mathbf{N}^{-1}(\mathbf{M}_2 - \mathbf{M}_1).$$

Note that in practice, $\mathbf{N}$ is usually singular and so a multiple of the identity is added to it [1]

$$\mathbf{N}_{\epsilon} = \mathbf{N} + \epsilon\mathbf{I}.$$

Given the solution for α, the projection of a new data point is given by[1]

$$y(\mathbf{x}) = (\mathbf{w} \cdot \phi(\mathbf{x})) = \sum_{i=1}^{l}\alpha_i k(\mathbf{x}_i, \mathbf{x}).$$

Q4:

1. 已知训练数据中的正例点为 $x_1=(3,3)^T$，$x_2=(4,3)^T$ 负例点为 $x_3=(1,1)^T$，试分别从原问题和对偶问题的角度求线性可分支持向量机。

A4:

解：① 从原问题的角度构造最优化问题。

$$\min_{w,b} \ \frac{1}{2}(w_1^2+w_2^2)$$
$$\text{s.t.} \ \ 3w_1+3w_2+b\geq 1 \quad ①$$
$$4w_1+3w_2+b\geq 1 \quad ②$$
$$-w_1-w_2-b\geq 1 \quad ③$$

③×3+①式有 $b\leq -2$ ④

①式可写为 $w_1+w_2\geq \frac{1-b}{3}$，由b最大值为-2.

则 w_1+w_2 的最小值为1

则原问题最优解为：当 $w_1=w_2=\frac{1}{2}$ 时，$\frac{1}{2}(w_1^2+w_2^2)=\frac{1}{4}$ 取最小

此时 $\frac{1-b}{3}\leq w_1+w_2=1\leq -1-b \Rightarrow b=-2$.

则得到最大间隔分离超平面为：

$$\frac{1}{2}x^{(1)}+\frac{1}{2}x^{(2)}-2=0$$

决策函数 $f(x)=\text{sign}\left(\frac{1}{2}x^{(1)}+\frac{1}{2}x^{(2)}-2\right)$.

其中，支持向量为 $(3,3)^T$，$(1,1)^T$ 为支持向量。

② 从对偶问题角度构造拉格朗日函数

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i\cdot x_j)-\sum_{i=1}^{N}\alpha_i$$
$$=\frac{1}{2}(18\alpha_1^2+25\alpha_2^2+2\alpha_3^2+42\alpha_1\alpha_2-12\alpha_1\alpha_3-14\alpha_2\alpha_3)-(\alpha_1+\alpha_2+\alpha_3) \quad ①$$
$$\text{s.t.} \ \ \alpha_1+\alpha_2-\alpha_3=0 \quad ②$$
$$\alpha_i\geq 0, \ i=1,2,3$$

②式改写为 $\alpha_3=\alpha_1+\alpha_2$ ③，代入①式有

目标函数 $S(\alpha_1,\alpha_2)=\frac{1}{2}(18\alpha_1^2+25\alpha_2^2+2\alpha_1^2+4\alpha_1\alpha_2+2\alpha_2^2+42\alpha_1\alpha_2-12\alpha_1^2-12\alpha_1\alpha_2-14\alpha_2^2-14\alpha_1\alpha_2)$

$=4\alpha_1^2+\frac{13}{2}\alpha_2^2+10\alpha_1\alpha_2-2\alpha_1-2\alpha_2 \qquad -2\alpha_1-2\alpha_2$

$$\begin{cases}\frac{\partial S(\alpha_1,\alpha_2)}{\partial \alpha_1}=8\alpha_1+10\alpha_2-2=0\\ \frac{\partial S(\alpha_1,\alpha_2)}{\partial \alpha_2}=13\alpha_2+10\alpha_1-2=0\end{cases}$$

解得：$\alpha_1=\frac{3}{2}$ $\alpha_2=-1$

由于 $\alpha_2\geq 0$，不满足条件，则最小值应在边界上.

1) $\alpha_1=0$ 时 $\frac{\partial S(0,\alpha_2)}{\partial \alpha_2}=0\Rightarrow \alpha_2=\frac{2}{13}$ $\Rightarrow S(0,\frac{2}{13})=-\frac{2}{13}$

2) $\alpha_2=0$ 时 $\frac{\partial S(\alpha_1,0)}{\partial \alpha_1}=0\Rightarrow \alpha_1=\frac{1}{4}$ $\Rightarrow S(\frac{1}{4},0)=-\frac{1}{4}$.

$\therefore \alpha^*=(\frac{1}{4},0,\frac{1}{4})^T$

$\therefore w^*=\sum_{i=1}^{3}\alpha_i^* y_i x_i=(\frac{1}{4},0,\frac{1}{4})^T\cdot\begin{pmatrix}3&3&-1\\3&4&-1\end{pmatrix}=\begin{pmatrix}\frac{1}{2}\\ \frac{1}{2}\end{pmatrix}$

将 $\alpha_3=\frac{1}{4}$，$(x_1,x_2)=(1,1)$ 代入

$b^*=-1-\frac{1}{4}\cdot 1\cdot(3,3)^T(1,1)-0-\frac{1}{4}\cdot(-1)\cdot(1,1)^T(1,1)$

$=-1-\frac{6}{4}+\frac{2}{4}=-2$.

$\therefore$ 最大间隔分离超平面：$\frac{1}{2}x^{(1)}+\frac{1}{2}x^{(2)}-2=0$. $f(x)=\text{sign}\left(\frac{1}{2}x^{(1)}+\frac{1}{2}x^{(2)}-2\right)$.

第四次作业

Q1：(计算题, 5 分)使用极大似然法估算如下数据集中“敲声”属性的类条件概率。

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

A1：

$D_{好瓜,浊响}=6$, $D_{坏瓜,浊响}=4$	$P(浊响\|好瓜)=\frac{6}{8}, P(浊响\|坏瓜)=\frac{4}{9}$
$D_{好瓜,沉闷}=2$, $D_{坏瓜,沉闷}=3$	$P(沉闷\|好瓜)=\frac{2}{8}, P(沉闷\|坏瓜)=\frac{3}{9}$
$D_{好瓜,清脆}=0$, $D_{坏瓜,清脆}=2$	$P(清脆\|好瓜)=\frac{0}{8}, P(清脆\|坏瓜)=\frac{2}{9}$

Q2：(论述题, 5 分)以上一题中的数据集为训练数据，构建带拉普拉斯修正的朴素贝叶斯分类器，并对如下样本进行判别。

编号	色泽	根蒂	敲声	纹理	脐部	触感
测 1	青绿	蜷缩	浊响	清晰	凹陷	硬滑

A2：

$P(好瓜=是)=\frac{8+1}{17+2}=\frac{9}{19}\approx 0.474$　　$P(坏瓜=是)=\frac{9+1}{17+2}=\frac{10}{19}\approx 0.526$

$P(青绿\|是)=\frac{4}{11}\approx 0.364$	$P(青绿\|否)=\frac{4}{12}\approx 0.333$
$P(蜷缩\|是)=\frac{6}{11}\approx 0.546$	$P(蜷缩\|否)=\frac{4}{12}\approx 0.333$
$P(浊响\|是)=\frac{7}{11}\approx 0.636$	$P(浊响\|否)=\frac{5}{12}\approx 0.417$
$P(清晰\|是)=\frac{8}{11}\approx 0.727$	$P(清晰\|否)=\frac{3}{12}\approx 0.250$
$P(凹陷\|是)=\frac{6}{11}\approx 0.546$	$P(凹陷\|否)=\frac{3}{12}\approx 0.250$
$P(硬滑\|是)=\frac{7}{10}\approx 0.700$	$P(硬滑\|否)=\frac{7}{11}\approx 0.636$

因为：$P(好瓜)\approx 0.0166$　>　$P(坏瓜)\approx 0.00097$

所以：该瓜是好瓜

Q3：（论述题，5.0 分）证明：二分类任务中，当两类数据满足高斯分布且协方差相同时，LDA 产生贝叶斯最优分类器。

A3：

首先看一下贝叶斯最优分类器：在书中p148中解释了对于最小化分类错误率的贝叶斯最优分类器可表示为：$h^*(x) = \arg\max_{c \in y} P(c|x)$，由贝叶斯定理即转换为：$h^*(x) = \arg\max_{c \in y} P(x|c)P(c)$。

那么在数据满足高斯分布时有：$h^*(x) = \arg\max_{c \in y} P(x|c)P(c) = \arg\max_{c \in y} log(f(x|c)P(c))$

$$= \arg\max_{c \in y} log(\frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} exp(-\frac{1}{2}(x-\mu_c)^T\Sigma^{-1}(x-\mu_c))) + log(P(c))$$

$$= \arg\max_{c \in y} -\frac{1}{2}(x-\mu_c)^T\Sigma^{-1}(x-\mu_c) + log(P(c))$$

$$= \arg\max_{c \in y} x^T\Sigma^{-1}\mu_c - \frac{1}{2}\mu_c^T\Sigma^{-1}\mu_c + log(P(c))$$

在二分类任务中，贝叶斯决策边界可表示为

$$g(x) = x^T\Sigma^{-1}\mu_1 - x^T\Sigma^{-1}\mu_0 - (\frac{1}{2}\mu_1^T\Sigma^{-1}\mu_1 - \frac{1}{2}\mu_0^T\Sigma^{-1}\mu_0) + log(\frac{P(1)}{P(0)})$$

$$= x^T\Sigma^{-1}(\mu_1 - \mu_0) - \frac{1}{2}(\mu_1 + \mu_0)^T\Sigma^{-1}(\mu_1 - \mu_0) + \log(\frac{P(1)}{P(0)})$$

再看看线性判别分析：

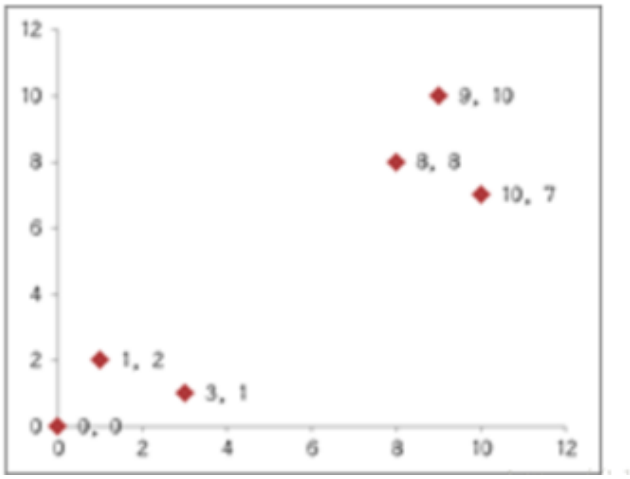
书中p62给出式3.39，其投影界面可等效于 $w = (\Sigma_0 + \Sigma_1)^{-1}(\mu_1 - \mu_0)$，注意为了和上面的推导一致，这里和书中给出的差了一个负号，但 w 位置没有改变，只是改变了方向而已。在两类别方差相同时有：$w = \frac{1}{2}\Sigma^{-1}(\mu_1 - \mu_0)$，两类别在投影面连线的中点可为 $\frac{1}{2}(\mu_1 + \mu_0)^T w = \frac{1}{4}(\mu_1 + \mu_0)^T\Sigma^{-1}(\mu_1 - \mu_0)$，那么线性判别分析的决策边界可表示为

$g(x) = x^T\Sigma^{-1}(\mu_1 - \mu_0) - \frac{1}{2}(\mu_1 + \mu_0)^T\Sigma^{-1}(\mu_1 - \mu_0)$。

推导到这里发现贝叶斯最优分类器和线性判别分析的决策边界只相差 $log(\frac{P(1)}{P(0)})$，在题目左边小字中有提及，“假设同先验”，所以 $log(\frac{P(1)}{P(0)}) = 0$，于是得证。

Q4：（计算题，5.0 分）假设 6 个二维数据点如下表所示，初始聚类中心为 P1 和 P2，请使用 K-means 算法将所有点聚为两类。

	X	Y
P1	0	0
P2	1	2
P3	3	1
P4	8	8
P5	9	10
P6	10	7



A4:

簇 C1={P1,P2,P3}　　均值向量为$\mu_1 = \left(\frac{4}{3}, 1\right)$

簇 C2={P4,P5,P6}　　均值向量为$\mu_2 = \left(9, \frac{25}{3}\right)$

Q5：(简答题, 5 分) 分析 AGNES 算法使用最小距离和最大距离的区别。

A5:

最小距离：$d_{\min}(C_i, C_j) = \min_{\boldsymbol{x} \in C_i, \boldsymbol{z} \in C_j} \operatorname{dist}(\boldsymbol{x}, \boldsymbol{z})$,

最大距离：$d_{\max}(C_i, C_j) = \max_{\boldsymbol{x} \in C_i, \boldsymbol{z} \in C_j} \operatorname{dist}(\boldsymbol{x}, \boldsymbol{z})$,

平均距离：$d_{\text{avg}}(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{\boldsymbol{x} \in C_i} \sum_{\boldsymbol{z} \in C_j} \operatorname{dist}(\boldsymbol{x}, \boldsymbol{z})$.

（1）**最大距离由两个簇的最远样本决定，**可以认为是所有类别先生成一个能包围所有类内样本的最小圆，然后所有圆同时慢慢扩大相同的半径，哪个类圆能完全包围另一个类则停止，并合并这两个类。由于此时的圆已经包含另一个类的全部样本，所以称为全连接。

（2）**最小距离由两个簇的最近样本决定，**则是扩大时遇到第一个非自己类的点就停止，并合并这两个类。由于此时的圆只包含另一个类的一个点，所以称为单连接。