1.

1) 最小错误率决策：应当寻找最大的后验概率，来使错误率最小.

$$\boldsymbol{P(error|x) = 1 - P(wi|x) \quad if\ you\ decide\ wi}$$

- Minimum error decision: Maximum a posteriori (MAP)

Decide ω_i if $P(\omega_i|\mathrm{x}) > P(\omega_j|\mathrm{x})$ for all $j \neq i$

最小风险决策:通过计算不同类别的最小风险决策,选择最小的 R(αi|x)所在的类别.

贝叶斯最小风险决策：

Condition risk

$$R(\alpha_i|\mathrm{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|\mathrm{x})$$

Minimum risk decision (Bayes decision)

$$\arg\min_i R(\alpha_i \mid x)$$

2)

- Formulation (Problem 13, Chapter 2)
  - C+1 classes

$$\lambda(\alpha_i \mid \omega_j) = \begin{cases} 0, & i = j \\ \lambda_s, & i \neq j \\ \lambda_r, & \text{reject} \end{cases} \quad \lambda_r < \lambda_s$$

替换错误s

$$R(\alpha_i|\mathrm{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|\mathrm{x})$$

$$\Rightarrow R_i(\mathbf{x}) = \begin{cases} \lambda_s[1 - P(\omega_i \mid \mathbf{x})], & i = 1, \ldots, c \\ \lambda_r, & \text{reject} \end{cases}$$

这里不是公式，只是一个事实阐述

$$\arg\min_i R_i(\mathbf{x}) = \begin{cases} \arg\max_i P(\omega_i \mid \mathbf{x}), & \text{if } \max_i P(\omega_i \mid \mathbf{x}) > 1 - \lambda_r / \lambda_s \\ \text{reject}, & \text{otherwise} \end{cases}$$

2.

1)

Case 2: $\Sigma_i = \Sigma$

$$g_i(\mathrm{x}) = -\frac{1}{2}(\mathrm{x} - \mu_i)^t \Sigma_i^{-1}(\mathrm{x} - \mu_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

$$\Rightarrow g_i(\mathrm{x}) = -\frac{1}{2}(\mathrm{x} - \mu_i)^t \Sigma^{-1}(\mathrm{x} - \mu_i) + \ln P(\omega_i)$$

– 展开二次式 $(\mathrm{x} - \mu_i)^t \Sigma^{-1}(\mathrm{x} - \mu_i)$

线性判别函数！ $g_i(\mathrm{x}) = \mathrm{w}_i^t \mathrm{x} + w_{i0}$

$$\mathrm{w}_i = \Sigma^{-1}\mu_i \quad w_{i0} = -\frac{1}{2}\mu_i^t \Sigma^{-1}\mu_i + \ln P(\omega_i)$$

– 二类决策面 $g_i(\mathrm{x}) = g_j(\mathrm{x})$

协方差任意，不一定垂直

$$\Rightarrow \mathrm{w}^t(\mathrm{x} - \mathrm{x}_0) = 0 \quad \mathrm{w} = \Sigma^{-1}(\mu_i - \mu_j)$$

$$\mathrm{x}_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\ln[P(\omega_i)/P(\omega_j)]}{(\mu_i - \mu_j)^t \Sigma^{-1}(\mu_i - \mu_j)}(\mu_i - \mu_j)$$

二类决策面带入具体的数值进行计算得

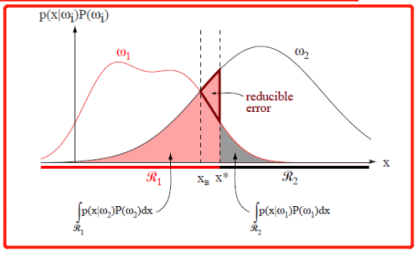
X2=2*x1

- **2类的情况**

$$\begin{aligned} P(error) &= P(\mathbf{x}\in\mathcal{R}_2,\omega_1)+P(\mathbf{x}\in\mathcal{R}_1,\omega_2) \\ &= P(\mathbf{x}\in\mathcal{R}_2|\omega_1)P(\omega_1)+P(\mathbf{x}\in\mathcal{R}_1|\omega_2)P(\omega_2) \\ &= \int_{\mathcal{R}_2} p(\mathbf{x}|\omega_1)P(\omega_1)\,d\mathbf{x}+\int_{\mathcal{R}_1} p(\mathbf{x}|\omega_2)P(\omega_2)\,d\mathbf{x}. \end{aligned}$$

- **一般情况**

$$\begin{aligned} P(correct) &= \sum_{i=1}^{c} P(\mathbf{x}\in\mathcal{R}_i,\omega_i) \\ &= \sum_{i=1}^{c} P(\mathbf{x}\in\mathcal{R}_i|\omega_i)P(\omega_i) \\ &= \sum_{i=1}^{c}\int_{\mathcal{R}_i} p(\mathbf{x}|\omega_i)P(\omega_i)\,d\mathbf{x} \end{aligned}$$



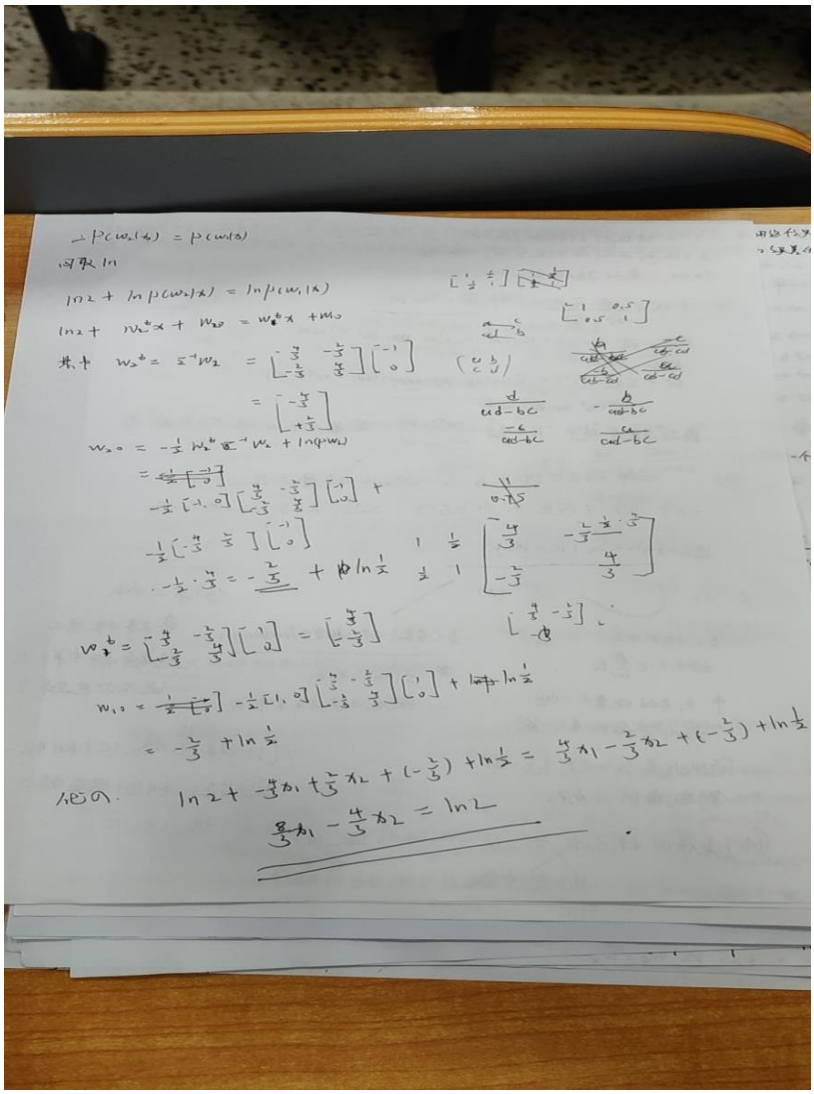
决策面为x_B时为最小错误率分类

$R2$ 与 R1 的范围由判别面方程可求，概率密度和先验可以带入具体数值。

2)

利用风险公式　后验概率*损失　列出来，然后列个等式就好了

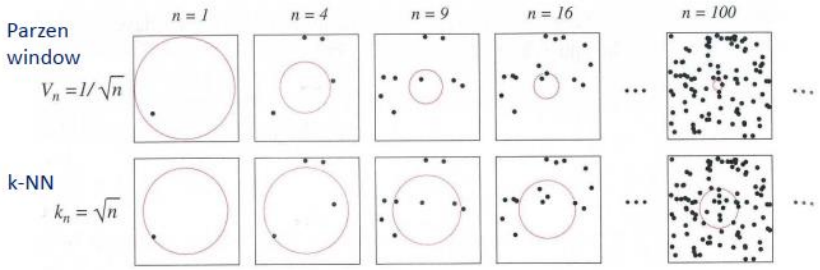
可以取对数进行约简计算



3.

1)

- 非参数概率密度估计
  - Parzen window: 固定局部区域体积V，k变化
  - k-nearest neighbor: 固定局部样本数k，V变化

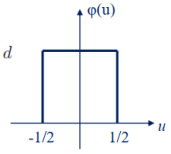


2)

Parzen Window

- 窗函数：hypercube

$$\varphi(\mathbf{u}) = \begin{cases} 1 & |u_j| \le 1/2 \quad j = 1, \ldots, d \\ 0 & \text{otherwise.} \end{cases}$$



  - 满足条件

$$\varphi(\mathbf{x}) \ge 0 \qquad \int \varphi(\mathbf{u})\, d\mathbf{u} = 1$$

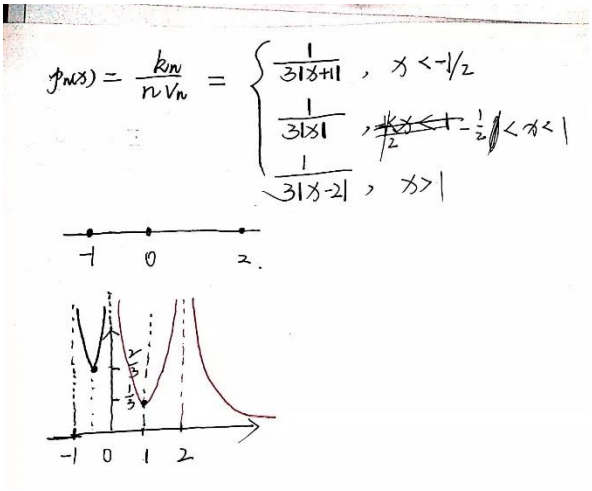
  - 以x为中心、体积为 $V_n = h_n^d$ 的局部区域内样本数

$$k_n = \sum_{i=1}^{n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right)$$

xi 训练样本 x中心 hn 窗口宽度

  - 概率密度估计k_n/nV_n 除以n再除以体积

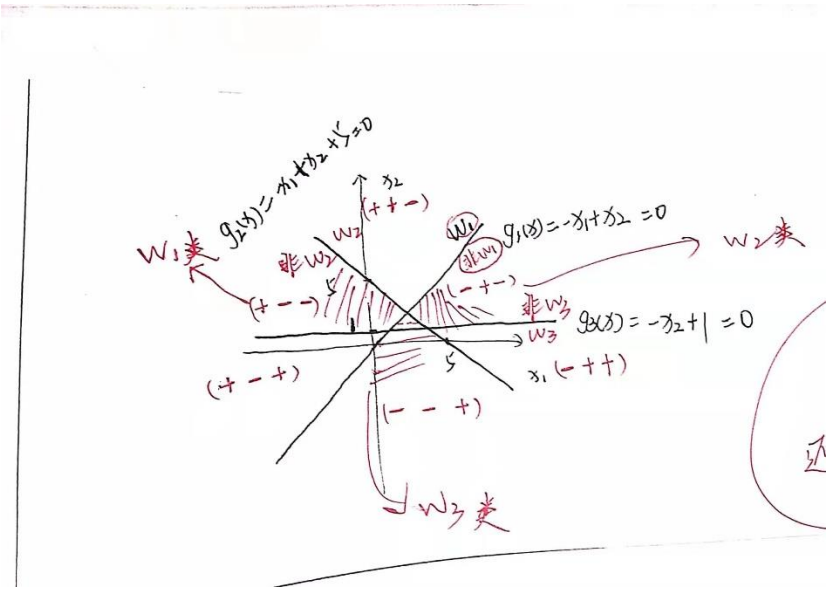
$$p_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right)$$



-无穷的时候趋近于 0，忘记画

4.

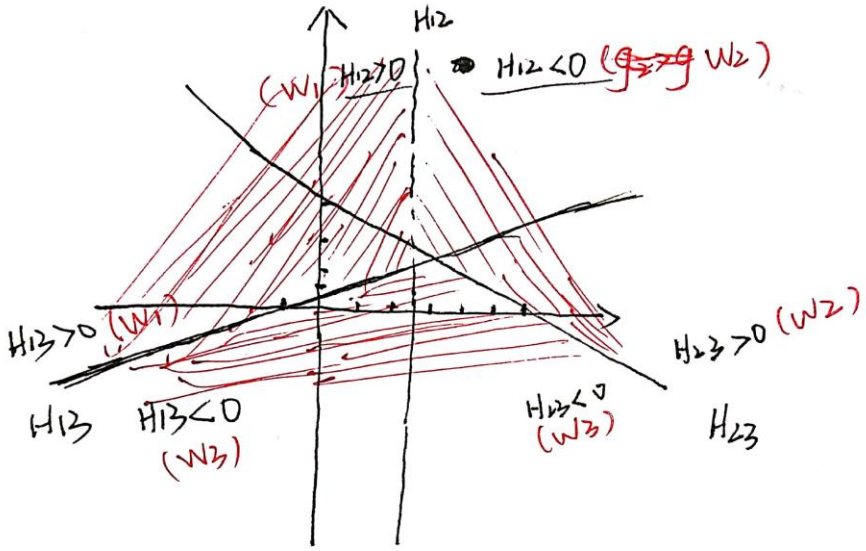
1)



$$H_{12}=g_1(x)-g_2(x)=-2x_1+5$$

$$H_{23}=g_2(x)-g_3(x)=x_1+2x_2-6$$

$$H_{13}=g_1(x)-g_3(x)=-x_1+2x_2-1$$



2)

- **感知准则函数－基本思想**
  - 考虑如下准则函数：

$$J_p(\mathbf{a}) = \sum_{\mathbf{y}\in Y}(-\mathbf{a}^T\mathbf{y}),\ \text{其中，}Y\text{为错分样本集合}$$

  - 当 $\mathbf{y}$ 被错分时， $\mathbf{a}^T\mathbf{y} \le 0$，则 $-\mathbf{a}^T\mathbf{y} \ge 0$。因此 $J_p(\mathbf{a})$ 总是大于等于0。在可分情形下，当且仅当 Y 为空集时 $J_p(\mathbf{a})$ 将等于零，这时将不存在错分样本。
  - 因此，目标是最小化 $J_p(\mathbf{a})$: $\min_{\mathbf{a}} J_p(\mathbf{a})$

可变增量批处理修正方法（另一种表述形式）

Batch Variable-Increment Perceptron

```
1   begin initialize: a, η0, k=0
2     do k ← k+1 (mod n)
3       Yk={}, j = 0
4       do j ← j + 1
5         if yj is misclassified, then append yj to Yk
6       until j = n
7       a = a + ηk Σ_{y∈Y(k)} y      //发现所有错分，然后再修正
8     until Yk={}                    //直到所有样本均正确分类
9     return a
10  end
```

5.

1)

1. 原理：假设各类出现的先验概率均相等，每个样本点以概率为1属于1个类，计算数据点到类中心的欧氏距离的平方，寻找与样本点最近的类中心点，将其分给最近的类。

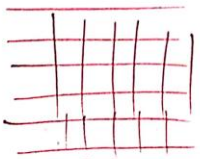
2)

5.(2) $x_1=(-6,1)^T$, $x_2=(-6,-1)^T$, $x_3=(-4,0)^T$, $x_4=(4,0)^T$, $x_5=(5,1)^T$
$x_6=(6,-1)^T$

① 把每个样本看作一类，计算点对之间的距离

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	0					
x_2	2	0				
x_3	$\sqrt{5}$	$\sqrt{5}$	0			
x_4	$\sqrt{101}$	$\sqrt{101}$	$\sqrt{64}$	0		
x_5	$\sqrt{104}$	$\sqrt{104}$	$\sqrt{85}$	2	0	
x_6	$\sqrt{149}$	$\sqrt{149}$	$\sqrt{101}$	$\sqrt{5}$	$\sqrt{5}$	0

② $G_1=\{x_4, x_5\}$.



图略.

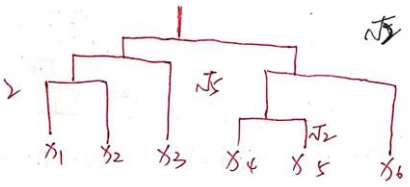
③ 根据最小距离原则，$G_1=\{x_4, x_5\}$，$G_2=\{x_1, x_2\}$

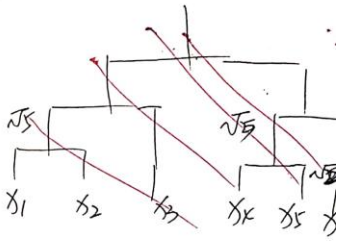
	G_1	x_3	G_2	x_6
G_1	0			
x_3	$\sqrt{5}$	0		
G_2	$\sqrt{101}$	$\sqrt{64}$	0	
x_6	$\sqrt{149}$	$\sqrt{101}$	$\sqrt{5}$	0

④ $G_3=G_1\cup\{x_3\}=\{x_4, x_5, x_3\}$. $G_4=G_2\cup\{x_6\}=\{x_1, x_2, x_6\}$

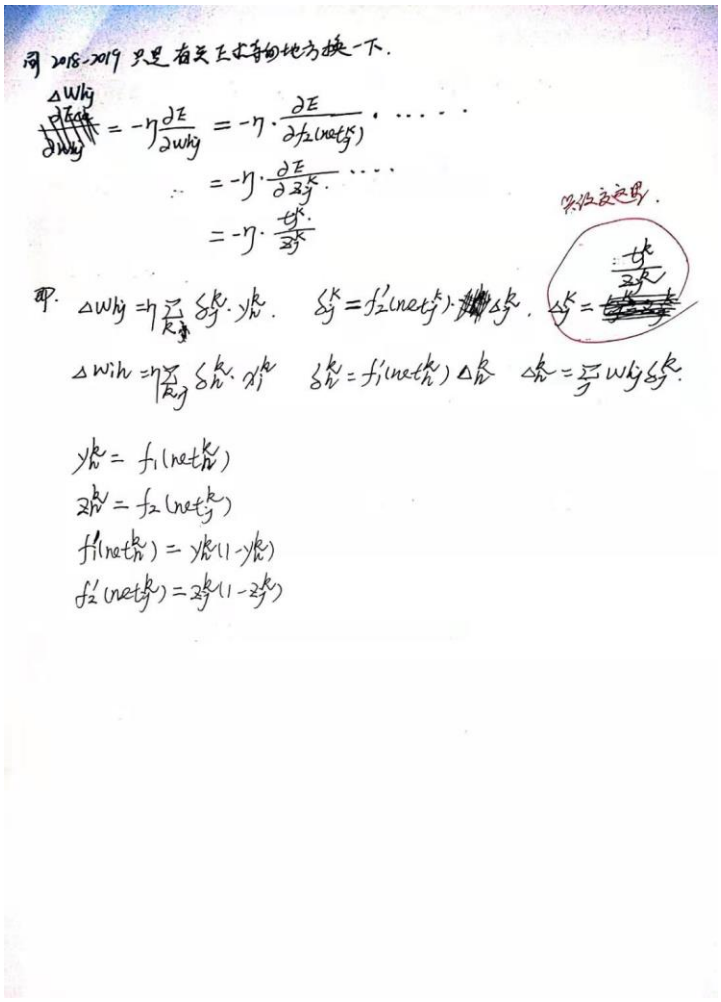
	G_3	G_4
G_3	0	
G_4	$\sqrt{101}$	0

决策树.





6.



7.

1) 简述 PCA 的主要思想和求解过程

2) 比较 PCA、CCA、LDA、ICA 的区别与适用场景。

3) 解释 LDA 所基于的数据分布假设，并阐述其不足之处

解：

1) PCA 的主要思想：PCA 就是将高维的数据通过线性变换投影到低维空间上去，得到一组无关的基。

求解过程：

3. 有 N 个样本$x_1,\cdots,x_N$，每个样本维数 D，希望将样本维数降低到 K，请给出 PCA 算法的计算过程

Step1：计算数据点的均值

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

Step2：计算样本的协方差矩阵

$$S = \frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})(x_n - \bar{x})^{\top}$$

Step3：对 D x D 的 S 矩阵进行特征值分解

Step4：对特征值进行从大到小排序，并选取最大的 K 个，记为 u1，u2……uk

Step5：U=【u1，u2……uk】就是 D x K 的投影矩阵

Step6：利用 $Z = U^T X$ 实现 PCA 降维

2)

PCA：主成分分析。PCA 属于无监督（训练样本无标签）的降维方法，是一种正交投影，侧重选择样本点投影方差最大的方向，减少特征相关性。

适用场景：

a) 原始数据特征多而且特征冗余。

b) 需要对样本进行可视化的时候，三维以上的特征无法进行可视化

CCA：典型相关分析。它选择的投影标准是降维到 1 维后，两组数据的相关系数最大。

适用场景：侧重于两组数据有相关关系的时候。

LDA：线性判别分析，可以降维可以分类，是从更利于分类的角度的，有监督（训练样本有标签）。希望数据投影后类内方差最小，类间方差最大。

适用场景：侧重于分类。

ICA：独立成分分析。ICA 信号需要是非高斯的，寻找的是最能使数据的相互独立的方向。

应用场景：盲信号分离。

3)

假设：

每一个类是单模态高斯分布 → 多模态 LDA

每一个类的协方差矩阵都相同 → 异方差 LDA

不足：

类分离问题。

降维维数不能超过 C-1

补充知识：

LDA 用于降维，和 PCA 有很多相同，也有很多不同的地方，因此值得好好的比较一下两者的降维异同点。

首先我们看看相同点：

1）两者均可以对数据进行降维。

2）两者在降维时均使用了矩阵特征分解的思想。

3）两者都假设数据符合高斯分布。

我们接着看看不同点：

1）LDA 是有监督的降维方法，而 PCA 是无监督的降维方法

2）LDA 降维最多降到类别数 k-1 的维数，而 PCA 没有这个限制。

3）LDA 除了可以用于降维，还可以用于分类。

4）LDA 选择分类性能最好的投影方向，而 PCA 选择样本点投影具有最大方差的方向。

LDA 算法的主要优点有：

1）在降维过程中可以使用类别的先验知识经验，而像 PCA 这样的无监督学习则无法使用类别先验知识。

2）LDA 在样本分类信息依赖均值而不是方差的时候，比 PCA 之类的算法较优。

LDA 算法的主要缺点有：

1）LDA 不适合对非高斯分布样本进行降维，PCA 也有这个问题。

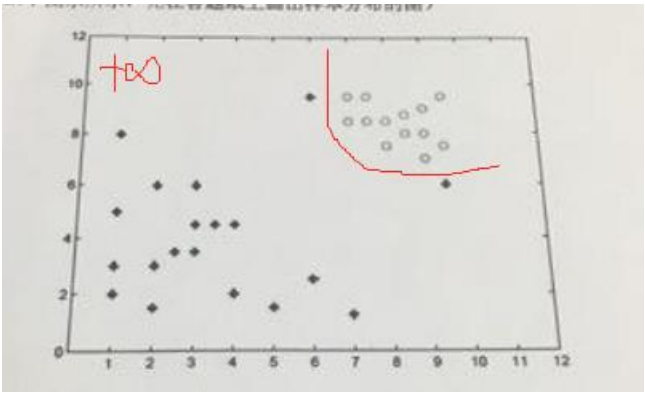
2）LDA 降维最多降到类别数 k-1 的维数，如果我们降维的维度大于 k-1，则不能使用 LDA。当然目前有一些 LDA 的进化版算法可以绕过这个问题。

3）LDA 在样本分类信息依赖方差而不是均值的时候，降维效果不好。

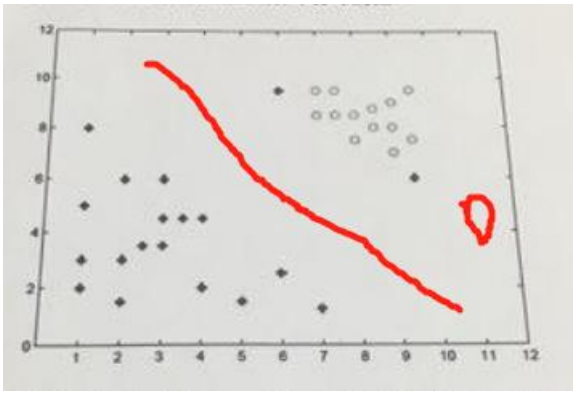
4）LDA 可能过度拟合数据。

9.SVM

1) C 取无穷大分类边界



2) C 取无穷小时的边界



3) 惩罚因子 C 越大，则 SVM 会更倾向把所有数据分对，往往出现较小的 margin，最终导致过拟合现象，泛化性能不好。

C 过于小，则惩罚力度不够，SVM 会更倾向实现最大化的 margin，而对样本分对分错不关心，不利于分类。

综上来看，在测试集中，C 取无穷小的时候效果会相对更好一些，泛化性能好。

4) Soft-margin SVM 的原问题和对偶问题，并阐述核方法的基本思想是如何将线性模型转换非线性模型的

原问题：

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ \text{s.t.} \quad & y_i(w \cdot x_i + b) \geqslant 1 - \xi_i, \quad i = 1, 2, \cdots, N \\ & \xi_i \geqslant 0, \quad i = 1, 2, \cdots, N \end{aligned}$$

对偶问题：

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i \\ \text{s.t.} \quad & \sum_{i=1}^{N}\alpha_i y_i = 0 \\ & 0 \leqslant \alpha_i \leqslant C, \quad i = 1, 2, \cdots, N \end{aligned}$$

将在原始低维空间线性不可分的分类问题通过非线性变换成高维空间线性可分的分类问题，在高维空间学习线性支持向量机。在线性支持向量机学习的对偶问题中，把低维到高维的非线性变换的内积形式用核函数表示。