模式识别

1. 绪论

1. 1. 模式识别

模：法也；式，法也。

模式：一种规律

Pattern：事物的模板或原型，或者表征事物特点的特征或性状的组合。

模式识别中的模式：

- 存在于时间、空间中可观察的事物，具有时间空间分布的信息。
- 对客体特征的描述。

样本：

- 一个具体的研究对象

模式类：

- 具有某些共同特性的模式的集合

模式识别：

- 确定一个样本的类别属性（模式类）的过程
- 把某一样本归属于多个类型中的某个类型

模式识别——用计算机实现人对各种事物或现象的分析、描述、判断、识别。

1. 2. 模式识别系统

1. 2. 1. 典型构成

1. 2. 1. 1. 监督模式识别

回归：

- 反映样本数据集中样本的属性值的特性，通过函数表达样本映射的关系来发现属性值之间的依赖关系。

分类：

- 将样本数据集中的样本映射到某个给定的类别中。

一般步骤：

1. 分析问题
2. 原始特征获取
3. 特征提取与选择
4. 分类器设计
5. 分类决策

1. 2. 1. 2. 非监督模式识别

聚类：

- 将样本数据集中的样本分为几个类别，属于同一类别的样本相似性比较大。

一般步骤：

1. 分析问题
2. 原始特征获取
3. 特征提取与选择
4. 聚类分析
5. 结果解释

1. 2. 1. 3. 比较

有已知样本：监督模式识别

无已知样本：非监督模式识别

2. 贝叶斯决策理论

2. 1. 基本符号与定义

决策：根据观测到的x，利用先验和类条件概率决定x属于哪一类。决策是从样本空间到决策空间的一个映射。

Bayes决策

两种常用的准则：

- 最小错误概率准则
- 最小风险准则

2. 1. 1. 基于最小错误

贝叶斯概率：

$$p(\omega_i|x)=\frac{p(x|\omega_i)p(\omega_i)}{p(x)}$$

决策规则

$$p(\omega_i|x)=\max_{j=1,2}p(\omega_j|x),x\in\omega_i$$

等价形式：

$$p(x|\omega_i)p(\omega_i)=\max_{j=1,2}p(x|\omega_j)p(\omega_i)$$

$$若l(x)=\frac{p(x|\omega_1)}{p(x|\omega_2)}<\frac{p(\omega_2)}{p(\omega_1)},则x\in\frac{\omega_1}{\omega_2}$$

错误率：

$$p(e)=\int_{-\infty}^{+\infty}p(e,x)\mathrm{d}x=\int_{-\infty}^{+\infty}p(e|x)p(x)\mathrm{d}x$$

条件错误率$p(e|x)$的计算：

- 以两类问题为例，当获得观测值x之后，有两种决策可能：决定$x \in \omega_1$或$x \in \omega_2$。则条件错误率为：

$$p(e|x) = \begin{cases} p(\omega_2|x) = 1 - p(\omega_1|x) & \text{if } x \in \omega_1 \\ p(\omega_1|x) = 1 - p(\omega_2|x) & \text{if } x \in \omega_2 \end{cases}$$

贝叶斯最小错误率决策：

- 选择后验概率$p(\omega_1|x)$，$p(\omega_2|x)$中大的ω作为决策，使得在观测值x下的条件错误率最小：

$$D(x) = \arg\max_i p(\omega_i|x)$$

条件错误率：

$$p(e|x) = 1 - \max_i p(\omega_i|x)$$

错误率：

$$p(e) = \mathbb{E}(p(e|x))$$

同时可以证明，此决策保证了每个观测值下的条件错误率最小。Bayes决策是一致最优决策。

2. 1. 2. 基于最小风险

引入风险函数（损失函数）$\lambda(x)$

风险损失：采用决策a_i时的风险

$$R(a_i|x) = \mathbb{E}(\lambda(a_i, \omega_j)) = \sum_{j=1}^{c} \lambda(a_i, \omega_j) p(\omega_j|x)$$

x是随机变量，采用x不同的观测值，产生的条件风险不同。决策a可以看成x的函数。定义期望风险：

$$R = \int R(a(x)|x) p(x) \mathrm{d}x$$

条件风险对应的是x，期望风险对应的是$a(x)$。

在考虑错判带来的损失时，我们希望损失最小，如果在采取每个决策都使其条件风险最小，则对所有的x做出决策时，其期望风险也必然最小。这样的决策就是最小风险贝叶斯准则。

即：

如果有：

$$R(a_k|x) = \min_{i=1,2,\ldots,a} R(a_i|x)$$

则有：$a = a_k$

计算步骤：

1. 计算后验概率
2. 根据决策风险表，计算采取a_i的条件风险
3. 对a个条件风险值进行比较，找出使条件风险最小的决策a。

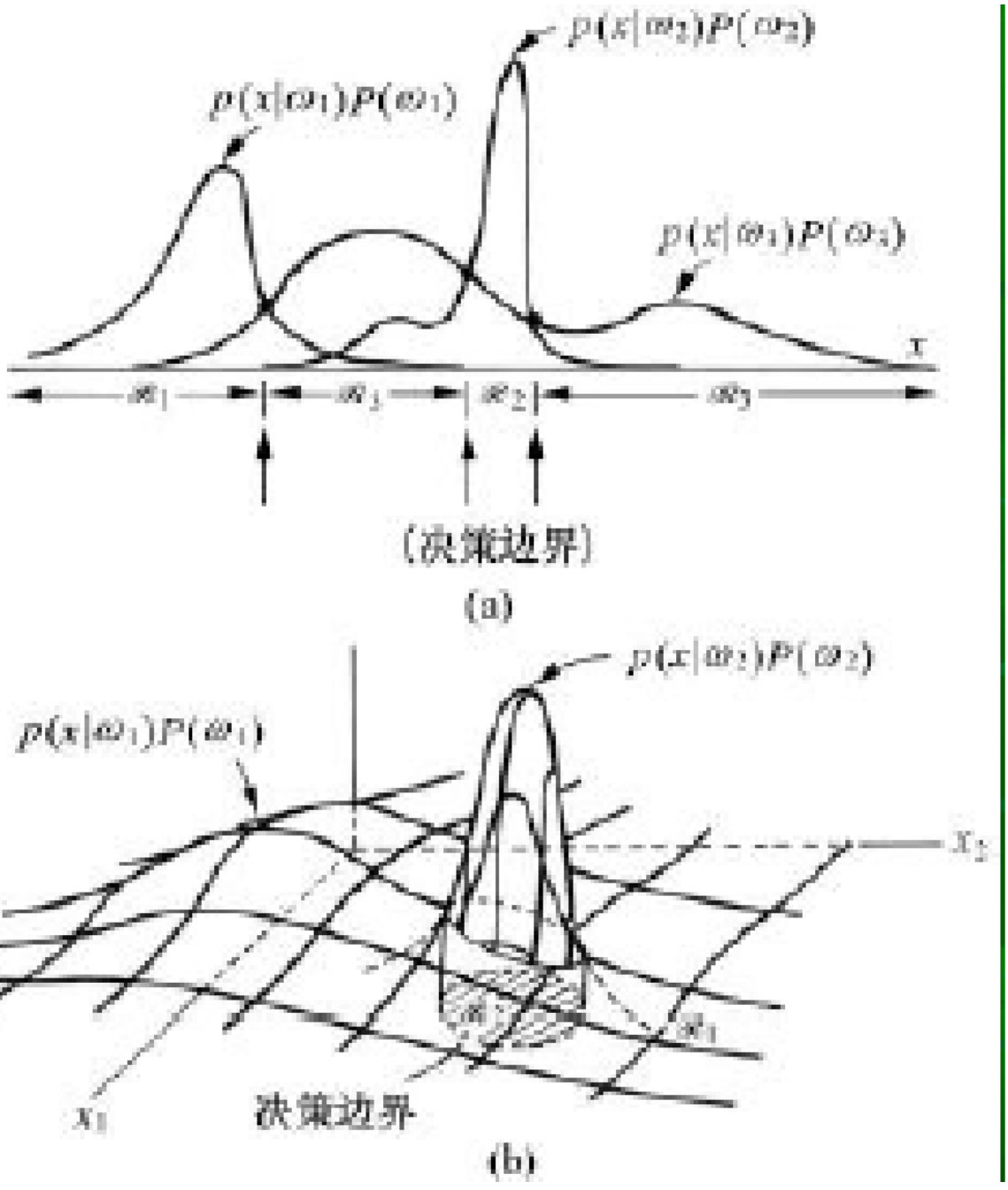
2. 2. 多类情况贝叶斯决策规则

2. 2. 1. 判别函数

$g_i(x)$，一般选取$g_i(x) = \max_j g_j(x)$，则将x归于ω_i类。

2. 2. 2. 决策面方程

$$g_i(x) = g_j(x)$$



2. 3. 正态分布统计决策

最小错误率贝叶斯判别函数：

$$g_i(x) = \ln p(x|\omega_i) + \ln p(\omega_i)$$

带入正态分布：

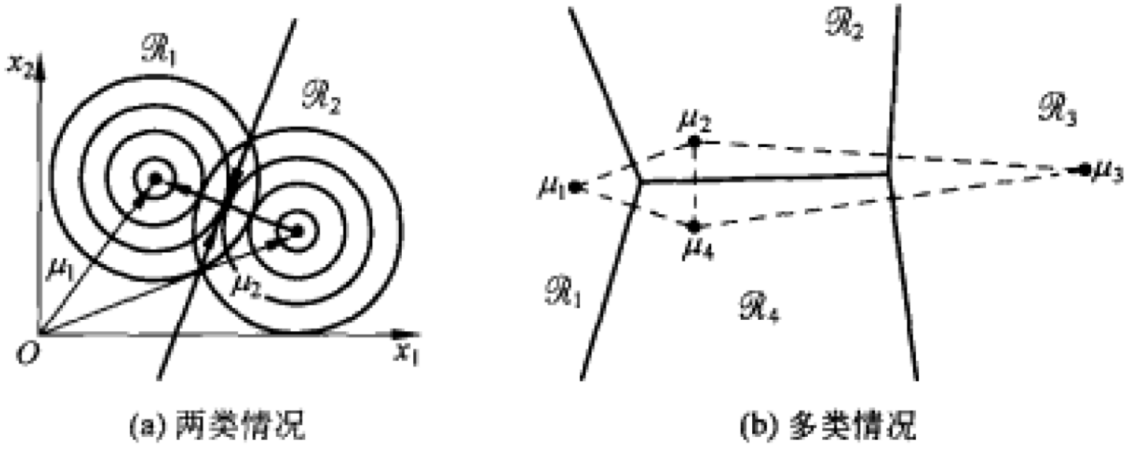
$$g_i(x) = -0.5(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i) - \frac{d}{2}\ln 2\pi - 0.5\ln|\Sigma_i| + \ln p(\omega_i)$$

决策面方程：

$$-0.5[(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)-(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)]-0.5\ln\frac{|\Sigma_i|}{|\Sigma_j|}+\ln\frac{p(\omega_i)}{p(\omega_j)}=0$$

几个特殊情况：

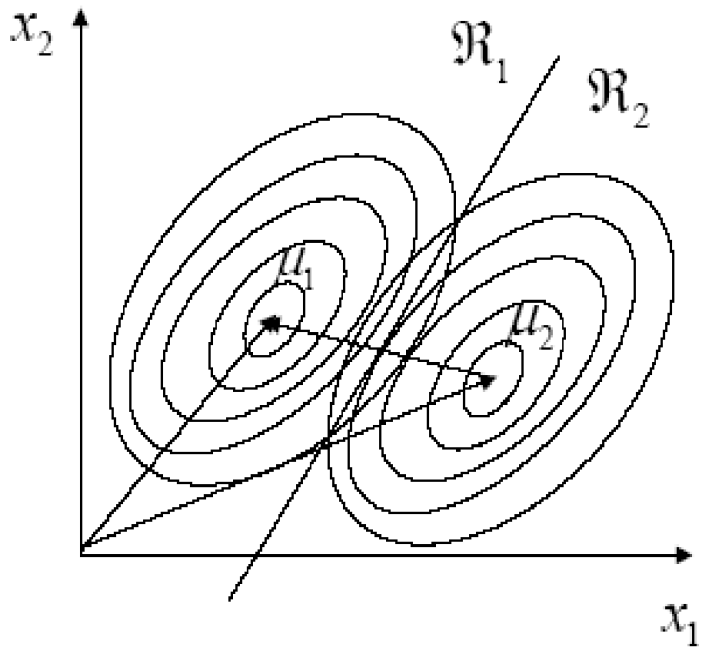
1. $\Sigma_i=\sigma^2 I$，则判别函数为：$-\frac{1}{2\sigma^2}(x-\mu_i)^T(x-\mu_i)+\ln p(\omega_i)$。若先验概率相等，直接等价于最小距离分类器。



(a) 两类情况　　(b) 多类情况

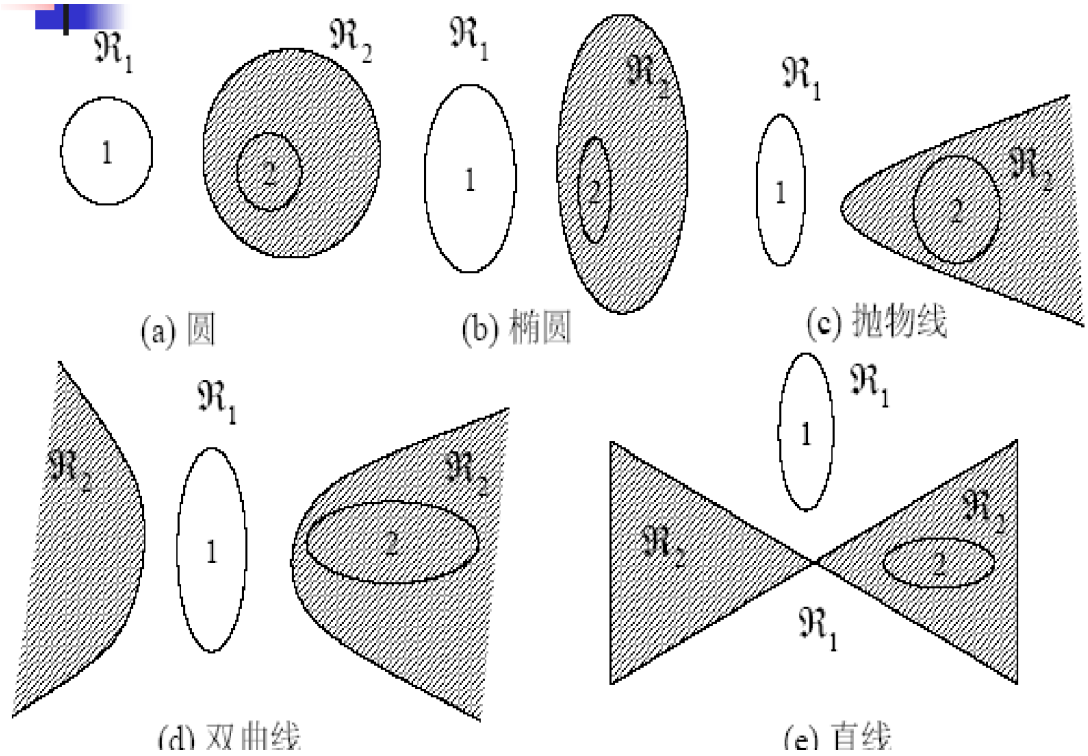
正态分布且 $P(\omega_i)=P(\omega_j)$，$\Sigma_i=\sigma^2 I$ 时的决策面

2. 每一个协方差矩阵都相等，则判别函数为：
$-\frac{1}{2\sigma^2}(x-\mu_i)^T(x-\mu_i)-\frac{d}{2}\ln 2\pi-0.5\ln\sigma^{2d}+\ln p(\omega_i)$式中$d$为维度。可简化为：
$-\frac{1}{2\sigma^2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)+\ln p(\omega_i)$



正态分布且$P(\omega_1)=P(\omega_2),\Sigma_1=\Sigma_2$ 时的决策面

3. 若协方差矩阵各不相等：



3. 概率密度函数的估计

3. 1. 基于样本的两步贝叶斯决策

- 利用样本集估计$P(\omega_i)$和$P(x|\omega_i)$
- 得到$\hat{P}(\omega_i)$和$\hat{P}(x|\omega_i)$
- 将估计量带入贝叶斯决策规则
- 得到决策结果

首先通过训练样本估计概率密度函数

- 先验概率估计-训练样本分布情况/根据领域知识认定
- 但类条件概率密度估计难

统计决策进行类别判定

- 训练样本有限，难以涵盖所有情况
- 但当训练样本多，就可以趋近于理论贝叶斯决策

3. 2. 概率密度估计方法

由训练样本集估计总体概率密度的方法可分为：

- 监督参数估计
- 非监督参数估计
- 非参数估计

3. 3. 参数估计的基本概念

统计量、参数空间

点估计：构造一个统计量作为某参数的估计

估计量、估计值、区间估计

3. 4. 概率密度估计的评估

无偏性：$\mathbb{E}\theta = \hat{\theta}$

渐进无偏：N趋于无穷有无偏性

有效性：一种估计比另一种方差小

一致估计

$$\lim_{n\to\infty} p(|\hat{\theta_n} - \theta| > \epsilon) = 0$$

则$\hat{\theta}$是θ的一致估计

3. 5. 最大似然估计

3. 5. 1. 基本假设

- θ确定但未知
- 按类别把样本集分开，$\mathfrak{R}_j$类中每个样本都是对立从概率密度$p(x|\omega_i)$的总体中独立抽取出来的。每即个样本i.i.d
- $p(x|\omega_j)$为已知分布，参数向量未知，且每个不同类别参数在函数上独立

3. 5. 2. 似然函数定义

在一类中独立抽取样本集来估计未知参数。

假设某类样本集中有N个样本

$$\mathfrak{R} = \{x_1, x_2, \dots, x_N\}$$

因样本独立抽取，样本出现在样本集中的概率

$$l(\theta) = p(\mathfrak{R}|\theta) = p(x_1, x_2, \dots, x_N|\theta) = p(x_1|\theta)p(x_2|\theta)\dots p(x_N|\theta)$$

也可以取对数。

3. 5. 3. 求解

由于每个训练样本独立，可对概率乘积取对数，再求导。

$$\forall i, \frac{\partial}{\partial\theta_i}\sum_{k=1}^{N}\log p(x_k|\theta^i) = 0$$

注意：方程有可能有多解，但只有一个解最大。

3. 5. 4. 多维正态分布的最大似然估计

3. 5. 4. 1. Σ已知，μ未知，求μ

$$\mu = \hat{\mu} = \frac{1}{N}\sum_{k=1}^{N}x_k$$

3. 5. 4. 2. Σ, μ均未知

$$\hat{\theta}_1 = \hat{\mu}_1 = \frac{1}{N}\sum_{k=1}^{N}x_k$$

$$\hat{\theta}_2 = \hat{\sigma}_1^2 = \frac{1}{N}\sum_{k=1}^{N}(x_k - \hat{\mu})^2$$

3. 6. 贝叶斯估计和贝叶斯学习

贝叶斯估计实质：贝叶斯决策来决策参数的取值。

与最大似然估计的区别：

- 最大似然估计把待估计的参数当作未知但固定的量
- 贝叶斯估计把待估计的参数也看为随机变量

贝叶斯学习：把贝叶斯估计的原理用于直接从数据对概率密度函数进行迭代估计。

可以回顾最小风险贝叶斯决策

3. 6. 1. 贝叶斯估计量

$R(\hat{\theta}|x)$为给定x条件下估计量$\hat{\theta}$的期望损失，称为条件风险

定义：如果θ的估计量$\hat{\theta}$使得条件风险最小，则为贝叶斯估计量。

3. 6. 2. 损失函数

损失函数有多种，最常见为平方误差：

$$\lambda(\hat{\theta}, \theta) = (\hat{\theta} - \theta)^2$$

3. 6. 3. 贝叶斯估计

定理：如果损失函数为二次函数，则贝叶斯估计量为给定x时θ的条件期望：

$$\hat{\theta} = \mathbb{E}[\theta|x] = \int_{\Theta} \theta p(\theta|x)\mathrm{d}\theta$$

3. 6. 4. 估计基本步骤

1. 确定θ的先验分布$p(\theta)$，待估参数为随机变量
2. 用第i类样本$X^i = (X_1, X_2, \ldots, X_N)$求出样本的联合概率密度分布$p(X^i|\theta)$，是一个$\theta$的函数
3. 利用贝叶斯公式，求θ的后验概率

$$p(\theta|X^i) = \frac{p(X^i|\theta)p(\theta)}{\int p(X^i|\theta)p(\theta)\mathrm{d}\theta}$$

4. 求贝叶斯估计

$$\hat{\theta} = \int_{\Theta} \theta p(\theta|X^i)\mathrm{d}\theta$$

这两种参数估计方法，最终目的是估计总体分布

$$p(x|\mathfrak{R})\ \mathfrak{R} \to X^i$$

其实在第三步后可以直接通过联合密度求类条件概率密度

$$p(x|X^i)\int p(x,\theta|X^i)\mathrm{d}\theta = \int p(x|\theta)p(\theta|X^i)\mathrm{d}\theta$$

3. 6. 5. 正态分布的均值估计

一维正态分布，已知σ^2，估计μ。

假设概率密度服从正态分布，则$p(X|\mu) \sim N(\mu, \sigma^2), p(\mu) \sim N(\mu_0, \sigma^2)$

用第i类样本$X^i = (X_1, X_2, \dots, X_N)$，求出后验概率：

$$p(\mu|X^i) = \frac{p(X^i|\mu)p(\mu)}{\int p(X^i|\mu)p(\mu)\mathrm{d}\mu}$$

因为N个样本独立抽取，且$\int p(X^i|\mu)p(\mu)\mathrm{d}\mu$仅仅与$x$有关，则上式可改写为：

$$\begin{aligned}
p(\mu|X^i) &= a\prod_{k=1}^{N} p(x_k|\mu)p(\mu) \\
&= a\prod_{k=1}^{N} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}(\frac{x_k-\mu}{\sigma}\right)^2 \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}(\frac{\mu-\mu_0}{\sigma}\right)^2)) \\
&= a^*\exp\left(-\frac{1}{2}(\sum_{k=1}^{N}(\frac{x_k-\mu}{\sigma}\right)^2 + (\frac{\mu-\mu_0}{\sigma})^2)) \\
&= a^{**}
\end{aligned}$$

3. 6. 6. 非监督参数估计

在未知样本类别的条件下的参数估计称为**非监督参数估计**

几个基本假设

- 样本来自类数为c的各类中，但不知道每个样本究竟来自于哪一类
- 每类的先验概率$p(\omega_j)$已知
- 类条件概率密度的形式$p(x|\omega_j, \theta_j)$已知
- 未知的只是c个参数向量$\theta_1, \theta_2, \dots, \theta_c$的值

似然函数：

$$l(\theta) = p(\Re|\theta)$$

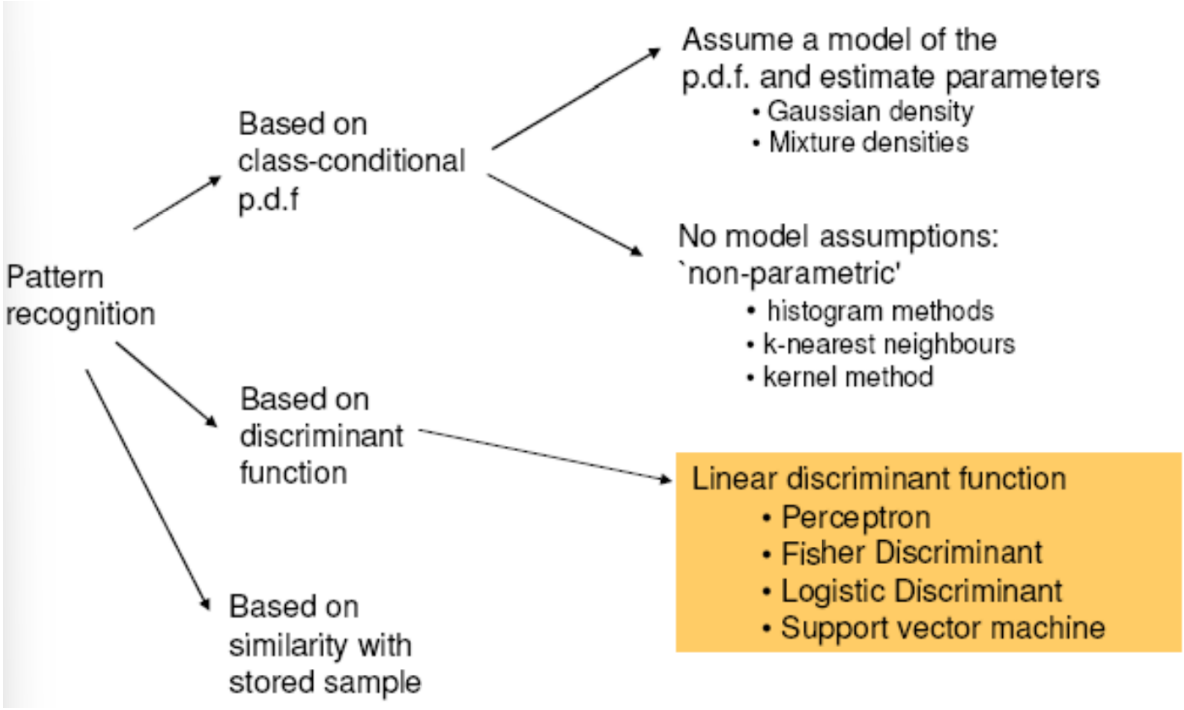
4. 线性判别函数

4. 1. 引言

基于样本的Bayes分类器：通过估计类条件概率密度，设计相应判别函数

如果可以估计概率密度函数，则可以使用贝叶斯决策来最优的实现分类

从样本数据中直接估计参数->根据对样本/问题的理解直接设定判别函数形式，直接求解！



4. 1. 1. 参数方法

概率密度函数已知，训练样本去估计参数

4. 1. 2. 线性判别方法

与非参数方法类似，与参数没有直接关系。次优方法。

4. 1. 3. 基于样本的直接确定判别函数的方法

1. 针对不同情况，使用不同准则函数，设计出满足这些不同准则要求的分类器。
2. 这些准则的“最优”并不一定与错误率最小相一致：次优分类器。

4. 2. 线性判别函数的基本概念

d维空间中线性判别函数的一般形式：

$$g(x) = w^T x + w_0$$

其中：x是样本向量-样本在d维特征空间中的描述，w是权向量，w_0是一个常数。

令$g(x) = g(x_1) - g(x_2)$，$g(x) = 0$定义了一个决策面，分开了两类。

判别函数$g(x)$也可以看作是特征空间中某个点x到超平面距离的一种代数度量。可定义：

$$x = x_p + r\frac{w}{||w||}$$

其中x_p是x到H的投影向量，r是x到H的垂直距离。

则

$$\begin{aligned} g(x) &= w^T(x_p + r\frac{w}{||w||}) + w_0 \\ &= w^T x_p + w_0 + r\frac{w^T w}{||w||} \\ &= r||w|| \end{aligned}$$

特殊情况：x为原点，则$g(x) = w_0$。

4. 2. 1. 设计分类器的主要步骤

1. 有一组具有类别标志的样本集
2. 根据实际情况确定一个准则函数 J,满足：J是样本集和w, wo, a的函数；J的值能反映分类器的性能，它的极值解对应于 “最好”的决策
3. 利用最优化方法求出准则函数的极值解和w, wo, a，进而得到$g(x)$

4. 3. Fisher线性判别分析

4. 3. 1. 基本思想

希望投影后的一维数据满足：

1. 两类之间尽可能远
2. 每一类自身尽可能紧凑

即：用投影后数据的统计性质（均值和离散度）的函数作为判别优劣的标准

4. 3. 2. 定义符号

m_1, m_2两类数据均值向量

S_1, S_2两类数据离散度矩阵

μ_1, μ_2两类数据投影后一维数据均值

σ_1, σ_2两类数据投影后一维数据离散度

$$m_i = \frac{1}{N}\sum^{i} x$$

$$S_i = \sum^{i}((x - m_i)(x - m_i)^T)$$

则有：

$$\begin{aligned}\mu_i &= w^T m_i \\ \sigma_i^2 &= \sum (w^T x - \mu_i)^2 \\ &= w^T \sum (x - m_i)(x - m_i)^T w \\ &= w^T S_i w\end{aligned}$$

4. 3. 3. Fisher准则函数

$$J_F(w) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

$$w_{opt} = \arg\max J_F(w)$$

称$S_b = (m_1 - m_2)(m_1 - m_2)^T$类间离散度矩阵

称$S_t = S_1 + S_2$类内总离散度矩阵

则：

$$J_F(w) = \frac{w^T S_b w}{w^T S_t w}$$

4. 3. 4. Fisher准则合理性

$J_F(w)$只与投影方向有关，与$||w||$无关，若w是最优解，则λw也是最优解。

4. 3. 5. Fisher最佳投影方向求解

需要$S_t = S_1 + S_2$正定，否则存在w使得$w^T S_t w = 0$，导致$J_F(w)$无极大值。

又因为有界性，则$J_F(w) \leq \frac{\max \lambda(S_b)}{\min \lambda(S_t)}$

$\lambda(S)$表示S的特征根。

又因为S_t正定，则存在最优的w，使得$w^T S_t w = 1$

本来是无约束优化$\max \frac{w^T S_b w}{w^T S_t w}$，等价为带约束最优化：

$$\max w^t S_b w$$
$$s.t.\ w^T S_t w = 1$$
$$L(w, \lambda) = w^T S_b w - \lambda(w^T S_t w - 1)$$
$$\frac{\partial L(w, \lambda)}{\partial w} = S_b w - \lambda S_t w = 0$$

由定义：

$$(m_1 - m_2)(m_1 - m_2)^T w_{opt} = \lambda S_t w_{opt}$$

记$c = (m_1 - m_2)^T w_{opt}$，则：

$$w_{opt} = \frac{c}{\lambda} S_t^{-1}(m_1 - m_2)$$

而我们只关心投影方向，所以：

$$w_{opt} = S_t^{-1}(m_1 - m_2) = (S_1 + S_2)^{-1}(m_1 - m_2)$$

映射就此完成，现在确定分类阈值w_0，即两个样本投影后的加权平均值（权为样本个数）

4. 4. 感知准则函数

为讨论方便，将x增加一维：$y = [1, x_1, x_2, \ldots, x_d]^T$，增广的权向量为：
$a = [w_0, w_1, w_2, \ldots, w_d]^T$，线性判别函数为：$g(y) = a^T y$

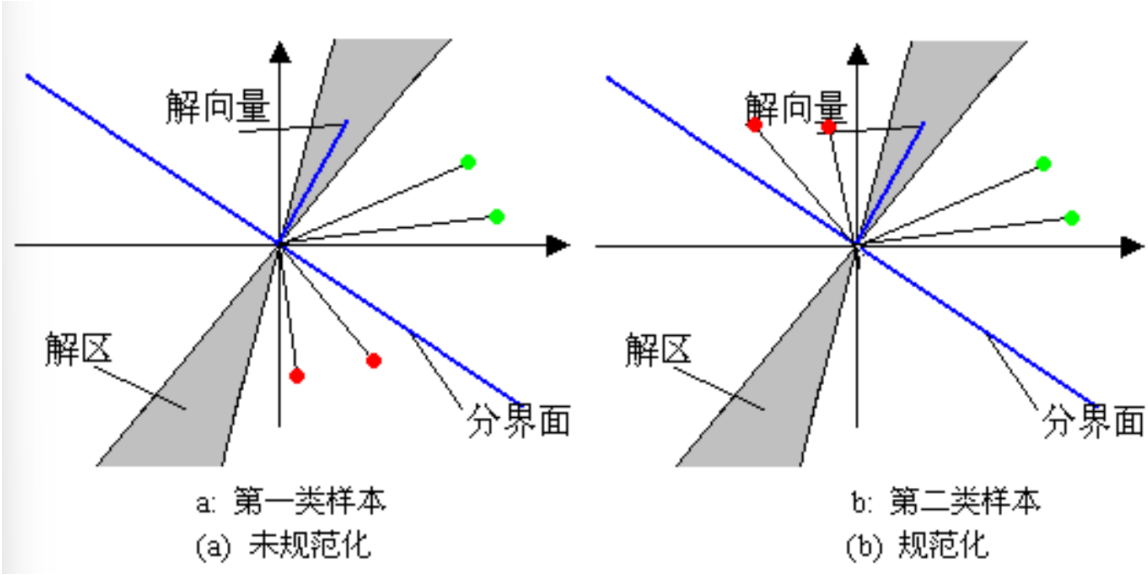
4. 4. 1. 基本概念

4. 4. 1. 1. 线性可分性

训练样本集中的两类样本在特征空间可以用一个线性分界面正确无误的分开。

4. 4. 1. 2. 规范化样本向量

将第二类样本取其反向向量。



a: 第一类样本
(a) 未规范化

b: 第二类样本
(b) 规范化

4. 4. 2. 感知准则函数

$$J_P(a) = \sum_{y \in Y_M} (-a^T y)$$

式中Y_M是被a错误分类的实例集合，由于全部反向，导致所有错误分类的实例有$a^T y < 0$，因此$J_P(a)$非负。

定义梯度：

$$\nabla_a J_P(a) = \sum_{y \in Y_M} (-y)$$

梯度衰减更新法则为：

$$a_{k+1} = a_k + \eta_k \sum_{y \in Y_M(k)} y$$

也被称为感知机批次更新法则。

- 1 begin initialize $\mathbf{a}$, $k = 0$
- 2 do $k \leftarrow (k+1)$
- 3 if $\mathbf{y}^{(k)}$ is misclassified by $\boldsymbol{a}$
  then $\mathbf{a} \leftarrow \mathbf{a} + \mathbf{y}^{(k)}$ $\Delta \mathbf{a} = -\eta \nabla J_p(\mathbf{a}) = -\eta \left(\sum_{\mathbf{y} \in Y} (-\mathbf{y}) \right)$
- 4 until all patterns properly classified
- 5 return
- 6 end

感知准则函数修正法：

1. 单样本修正法：样本集视为不断重复出现的序列，逐个样本检查，修正权向量
2. 批量样本修正法：样本成批或全部检查后，修正权向量。

4. 4. 3. 小结

感知准则函数方法思路：

1. 找初始向量a_1，用训练样本集中每个样本计算
2. 若发现某个y有$a^Ty<0$，则只要$a_{k+1}=a_k+\eta_k y$，则必有$a_{k+1}^Ty=a_k^Ty+\eta_k y^Ty$，有趋势使得$a_{k+1}^Ty>0$

当然，修正后的a也有可能使某些y出现$a_{k+1}^Ty<0$，但只要训练样本集线性可分，无论初值为什么，有限次迭代都能收敛。

4. 5. 最小平方误差准则函数

规范化增广样本向量y_i，增广权向量a，正确分类要求：$a^Ty_i>0$

本质就是求一组N个线性不等式的解。

样本集增广矩阵Y及一组N个线性不等式可由矩阵表示

引入余量$b=[b_1,b_2,\dots,b_N]^T$，b_i为任意给定正常数，$a^Ty_i=b_i>0$，则可表示为：

$$Ya=b$$

定义误差向量$e=Ya-b$，定义平方误差准则函数：

$$J_S(a)=||e||^2=||Ya-b||^2=\sum_{i=1}^N(a^Ty_i-b_i)^2$$

求解

$$a^*=\arg\min_a J_S(a)$$

$$\nabla J_S(a)\sum_{i=1}^N 2(a^Ty_i-b_i)y_i=2Y^T(Ya-b)$$

则：

$$\nabla J_S(a^*)=0\iff Y^TYa^*=Y^Tb$$
$$a^*=(Y^TY)^{-1}Y^Tb=Y^+b$$

4. 5. 1. 与其他方法的关系

与Fisher：当$b=[N/N_1,N/N_1,\dots,N/N_1(N_1个),N/N_2,N/N_2,\dots,N/N_2(N_2个)]^T$

MSE等价于Fisher

与Bayes：当$N\to\infty,b=u_N$时，则它以最小均方误差逼近Bayes判别函数
$g(x)=P(\omega_1|x)-P(\omega_2|x)$

4. 5. 2. MSE方法的迭代解

伪逆解计算量大，实际使用的梯度下降法

$$\nabla J_S(a)\sum_{i=1}^N 2(a^Ty_i-b_i)y_i=2Y^T(Ya-b)=\begin{cases}a_1\text{随机初始化}\\a_{k+1}=a_k-\eta_kY^T(Ya-b)\end{cases}$$

满足一定条件时梯度下降结束。

也可以使用单样本修正调整权向量：Widrow-Hoff/最小均方根/LMS算法

$$a_{k+1} = a_k + \eta_k(b_i - a_K^T y_i)y_i$$

其中y_i是使得$a_K^T y_i \neq b_i$的样本。

4. 6. 多类问题

三种方法：

4. 6. 1. c-1 二类问题

分离符合与不符合ω_1的点

4. 6. 2. c(c-1)/2 二类问题

分离每一对

4. 6. 3. 定义c个线性离散函数

将x归于w_j类如果$g_i(x) > g_j(x), \forall j \neq i$

$$g_i(x) = w_i^T x + w_{i0}$$

5. 非线性判别函数

5. 1. 引言

线性判别函数：简单实用，但线性不可分时错误率大。

1. 使用新的特征
2. 非线性变换
3. 非线性分类器

5. 2. 分段线性判别函数

- 决策面由若干超平面段组成，计算相对比较简单
- 能够逼近各种形状的超平面，适应性强
- 多类情况下的线性判别函数分类
- 树状分类

如果两类可划分为线性可分的若干子类，则可以设计多个线性分类器，实现分段线性分类器。

5. 2. 1. 基于距离的分段线性判别函数

分段线性距离分类器：将各类别划分为相对密集的子类，每个子类以它们的均值作为代表点，然后按最小距离分类。

判别函数：ω_i有l_i个子类，将ω_i的决策域R_i分成l_i个子域$R_i^1, R_i^2, \ldots, R_i^{l_i}$，每个区域用均值$m_i^k$代表：

$$g_i(x) = \min_{k=1,\ldots,l_i} ||x - m_i^k||$$

判别规则：

$$j = \arg \min_{i=1,\ldots,c} g_i(x)$$

如果$g_j(x) = \arg\min_{i=1,\ldots,c} g_i(x)$，则$x$属于$\omega_j$。

5. 2. 2. 一般的分段线性判别函数

将每个大类分成若干子类，针对每个子类定义一个线性判别函数。

一般形式：$g_i^k(x)$表示第i类第k段线性判别函数，l_i是第i类所具有的判别函数个数，$w_i^k.w_{i_0}^k$ 拘别是第k段的权向量和阈值权：

$$g_i^k(x) = {w_i^{(k)}}^T x + w_{i_0}^k$$

第i类判别函数：

$$g_i(x) - \max_{k=1,2,\ldots,l_i} g_i^k(x)$$

判别规则：如果$g_j(x) = \arg\min_{i=1,\ldots,c} g_i(x)$，则$x$属于$\omega_j$。

决策面：$g_i^n(x) = g_j^m(x)$

问题：如何确定子类数目？如何求得各子类的线性判别函数?

5. 3. 分段线性判别函数的设计

5. 3. 1. 已知子类划分

直接使用多类线性分类方法

如何知道子类划分？**先验、聚类分析**

5. 3. 2. 只知道子类数目，不知道子类划分

使用错误修正法（此法介绍使用增广的判别函数形式表示$g_i^k(y) = {a_i^{(k)}}^T y$)

假设ω_i，ω_i类中有l_i个子类，每一类均存在一定数量训练样本：

1. 初始化：任意给定各类各子类权值${a_i^{(k)}}^T(0)$，通常使用小随机数
2. 在时刻t：当前权值为${a_i^{(k)}}^T(t)$，考虑某个训练样本$y_v \in \omega_j$，找出ω_j类的子类中最大的判别函数 ${a_i^{(m)}}^T(t)y_v = \max_k {a_j^{(k)}}^T(t)y_v$
3. 考察当前权值对y_v的分类情况，若${a_j^{(m)}}^T(t)y_v > {a_i^{(k)}}^T(t)y_v, \forall i \neq j$，则${a_i^{(k)}}^T(t)$不变。若 $\exists i \neq j, k = n, s.t. {a_j^{(m)}}^T(t)y_v \leq {a_i^{(n)}}^T(t)y_v$，则$y_v$被错分，对其中最大者记为$(i^{'}, n^{'})$修正：

$$a_j^{(m)}(t+1) = a_j^{(m)}(t) + \rho_t y_t$$
$$a_{i^{'}}^{(n^{'})}(t+1) = a_{i^{'}}^{(n^{'})}(t) - \rho_t y_t$$

重复迭代，直至收敛。式中ρ为自取步长

5. 3. 3. 未知子类数目

使用树状分段线性分类器。

5. 3. 4. 用凹函数的并来表示分段线性函数

设l_i为线性判别函数，则：

1. $l_1, l_2, \ldots, l_r$都是分段线性判别函数
2. 若A, B都是分段线性判别函数，则：$A \wedge B, A \vee B$也是分段线性判别函数。
3. 对任何分段线性函数都可以表示为析取范式或合取范式。

析取范式中最小项$(L_{11} \wedge L_{22} \wedge \cdots \wedge L_{1m})$称为**凹函数。**

对于多峰二类问题：设第一类有q个峰，则有q个凹函数，即：

$$P = P_1 \vee P_2 \vee \cdots \vee P_q$$

每个凹函数P_i由m个线性判别函数来构成：

$$P_i = L_{i1} \wedge L_{i2} \wedge \cdots \wedge L_{im}$$

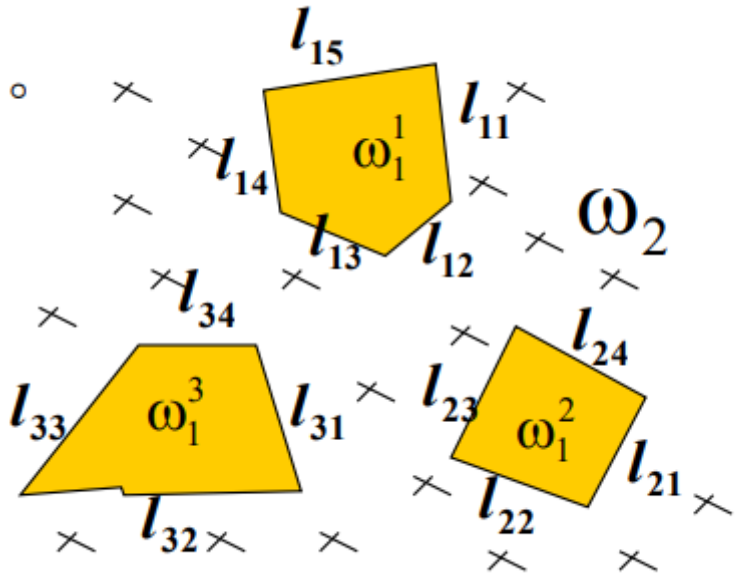
假设对于每个子类的线性判别函数L_{ij}都设计成：

$$L_{ij} = w_{ij}x \begin{cases} > 0, x \in \omega_1, i = 1, 2, \ldots, q\text{子类} \\ < 0, x \in \omega_2, j = 1, 2, \ldots, m\text{每个子类的判别函数数} \end{cases}$$

最终判别规则：

$$P > 0, \text{则} x \in \omega_1$$
$$P \leq 0, \text{则} x \in \omega_2$$

例：



ω_1分三个峰，$q = 3$。

判别函数：

$$m_1 = 5, m_2 = 4, m_3 = 4$$

所以共有十三个分段判别函数

$$\begin{aligned} P &= (L_{11} \wedge L_{12} \wedge L_{13} \wedge L_{14} \wedge L_{15}) \vee (L_{21} \wedge L_{22} \wedge L_{23} \wedge L_{24}) \vee (L_{31} \wedge L_{32} \wedge L_{33} \wedge L_{34}) \\ &= \max(\min(l_{11}, l_{12}, l_{13}, l_{14}, l_{15}), \min(l_{21}, l_{22}, l_{23}, l_{24}), \min(l_{31}, l_{32}, l_{33}, l_{34})) \end{aligned}$$

若$P > 0$则$x \in \omega_1$，否则$x \in \omega_2$

5. 3. 5. 用交遇区的样本设计分段线性函数

思想：寻找两类中最靠近的样本子集，用他们设计分类器。

步骤：

1. 用聚类分析等方法把每类样本分为若干子类
2. 考察子类间的距离$d(v_i^m, v_j^n)$
3. 寻找紧互对原型对$d(v_i^m, v_j^n) = \min_l(d(v_i^l, v_j^n)) = \min_l(d(v_i^m, v_j^l))$

4. 用紧互对原型对设计分类面
5. 决策规则（可能有错分）：设最后得到m个超平面$H_i:\alpha_i^Ty=0$，记$z_i(x)=if\ \alpha_i^Ty>0:1,\ else\ 0$，得$z(x)=[z_1(x),z_2(x),\dots,z_m(x)]$，对$z(x)$的每一种可能取值$z_j$，统计其在$\chi_1,\chi_2$两类样本中出现的次数$N_1(z_j),N_2(z_j)$，定义$\Omega(z_j)$：若$N_1(z_j),N_2(z_j)$很小，则$\Omega(z_j)=\delta$，否则若$L=\frac{N_1(z_j)}{N_1(z_j)+N_2(z_j)}>1$，则$\Omega(z_j)=1$，否则若$L<2$，$\Omega(z_j)=0$。
6. 最终决策规则：对输入x，若$\Omega(z_j)=1$则$x\in\omega_1$；若$\Omega(z_j)=0$，则$x\in\omega_2$；若$\Omega(z_j)=\delta$则拒绝。

5. 4. 二次判别函数

二次判别函数一般表示为：

$$\begin{aligned}g(x)&=X^T\bar{W}X+W^TX+W_0\\&=\sum_{i=1}^{n}w_{ii}x_i^2+2\sum_{j=1}^{n-1}\sum_{i=j+1}^{n}w_{ji}x_jx_i+\sum_{j=1}^{n}w_jx_j+W_0\end{aligned}$$

其中：$\bar{W}$是$n\times n$维权向量，W为n维权向量，系数共有$l=\frac{1}{2}n(n+3)+1$（很大）

二次决策面为超二次曲面。

5. 4. 1. 已知样本ω_1集中，而ω_2分散

定义ω_1判别函数：

$$g(x)=k^2-(x-\bar{\mu})^T\Sigma^{-1}(x-\bar{\mu})$$

k决定超平面大小，μ为ω_1均值，Σ为ω_1协方差。

5. 4. 2. 已知样本ω_1,ω_2都集中

可以定义两个判别函数：

$$g_i(x)=k_i^2-(x-\bar{\mu}_i)^T\Sigma_i^{-1}(x-\bar{\mu}_i)$$

μ_i为ω_i均值，Σ_i为ω_i协方差。

判别面方程：

$$g(x)=g_1(x)-g_2(x)=0$$

5. 5. 神经网络综述

5. 5. 1. 简介

神经网是一种模拟动物神经网络行为特征，进行分布式并行信息处理的算法。

这种网络依靠系统的复杂程度，通过调整内部大量节点之间相互连接的关系，从而达到处理信息的目的。

基础：

1. 构成：大量简单的基本元件——神经元
2. 工作原理：模拟生物神经处理信息
3. 功能：进行信息的并行处理和非线性转换

特点：

1. 比较轻松地实现非线性映射

2. 具有大规模的计算能力

5. 5. 2. 发展

1. MP模型
2. Hebb学习规则
3. 感知机模型
4. 感知器，局限性
5. 能量函数
6. 反向传播，解决多层前向神经网络学习问题

5. 5. 3. 优劣势

优势：

1. 很强的自适应学习能力
2. 实现特征空间较复杂的划分
3. 能用高速并行处理系统实现

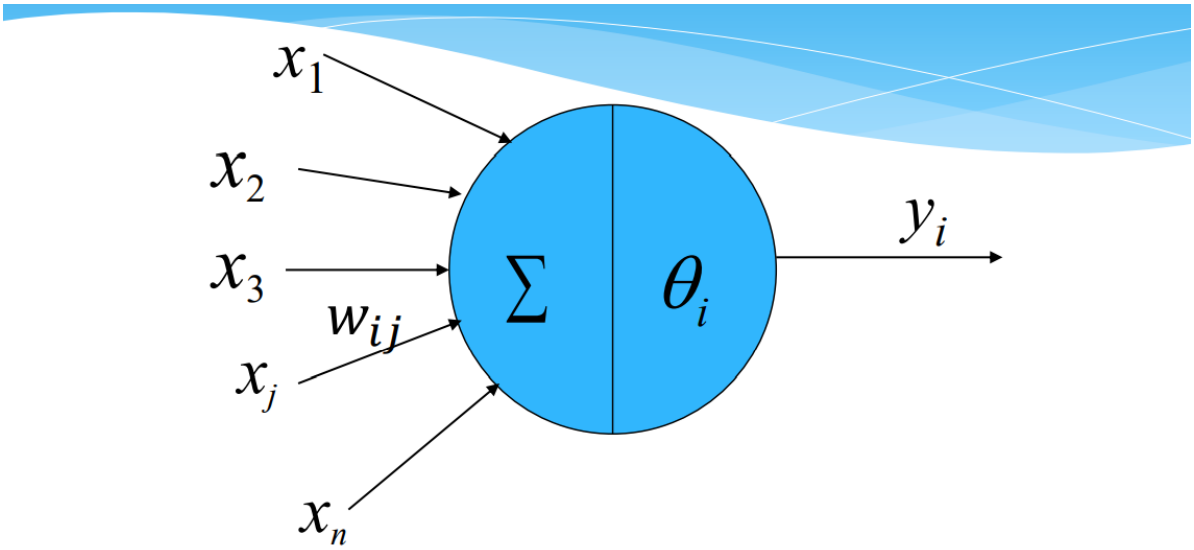
劣势：

1. 需要更多训练数据
2. 在普通计算机模拟运行速度较慢
3. 无法解释

5. 5. 4. 本质

利用计算机语言模拟人脑做决定

5. 5. 5. 神经元结构模型



* x_j为输入信号，θ_i为阈值，w_{ij}表示与神经元x_j连接的权值
* y_i表示输出值
* 判断$x_j w_{ij}$是否大于阈值θ_i

5. 5. 6. 几种代表性的网络模型

1. 单层前向神经网络——线性网络
2. 阶跃网络
3. 多层前向神经网络（反推学习规则，即BP神经网络）
4. Elman、Hopfield、双向联想记忆网络、自组织竞争网络

5. 5. 7. 神经网络能干什么

函数逼近、数据聚类、模式分类、优化计算等。

5. 6. BP神经网络

BP 神经网络是一种按**误差逆传播算法(反向传播)**训练的多层前馈网络，是应用广泛的神经网络模型之一。

BP网络能学习和存贮大量的输入-输出模式映射关系，而无需事前揭示描述这种映射关系的数学方程。

学习规则是使用最速下降法，通过反向传播来不断调整网络的权值和阈值，使网络的误差平方和最小。

BP神经网络模型拓扑结构包括输入层(input layer)隐层(hidden layer)和输出层(output layer)

5. 6. 1. 特点

多层前馈神经网络，信号向前传播，误差向后传播。

5. 6. 2. 工作原理

基本BP算法包括两个方面：信号的前向传播和误差的反向传播。即计算实际输出时按从输入到输出的方向进行，而权值和阈值的修正从输出到输入的方向进行。利用输出后的误差来估计输出层的直接前一层的误差，再 用这个误差估计更前一层的误差，如此反传下去，就获得了所有其他各层的误差估计。

5. 6. 3. 工作流程

1. 网络初始化：根据训练数据确定网络输入神经元、隐含神经元、输出神经元数目，初始化各层神经元间的连接权值，初始化隐含层、输出层的阈值，给定学习率和神经元传递函数。
2. 隐含层输出计算：根据输入向量、输入层和隐含层连接权值、阈值，计算隐含层输出
3. 输出层输出计算：根据隐含层输出、连接权值、阈值，计算BP神经网络预测输出
4. 误差计算：根据预测输出和期望输出计算网络预测误差
5. 权值更新：根据网络误差更新网络权值
6. 阈值更新：根据网络误差更新
7. 判断是否结束

5. 6. 4. 学习过程

正向传播：

输入样本→输入层→隐含层→输出层

判断是否转入反向传播阶段：若输出层的实际输出与期望的输出不符。

误差反传：误差以某种形式在各层表示——修正各层单元的权值。

5. 6. 5. 缺陷与改进

缺陷：

1. 学习效率低，收敛速度慢
2. 易陷入局部极小状态

改进：

隐含层层数：过少则误差大，过多则会过拟合，学习时间增长。

隐含层神经元数量一般使用经验判断，m为隐含层神经元数，n为输入层神经元数。I为输出层神经元数，α为$1-10$间的常数，则：$m=\sqrt{n+l}+a, m=\log_2 n, m=\sqrt{nl}$ 等。

5. 6. 6. 其它常用传统神经网络模型

5. 6. 6. 1. 径向基函数网络

特点：**只有一个隐层**，隐层单元采用径向基函数作为输出函数，输入层到隐层的权值固定为1，输出节点为线性求和单元，隐层到输出节点间权值可调。

径向基函数

某种沿径向对称的标量函数。通常定义为空间中任一点x到某一中心x_c之间欧氏距离的单调函数，可记作$k(||x-x_c||)$。

最常用的是高斯核函数：$k(||x-x_c||)=\exp\left\{-\frac{||x-x_c||^2}{2\delta^2}\right\}$

作用

把网络看作对未知函数的逼近器，输入层到隐层的基函数是非线性的，而输出是线性的，可看作先对原始非线性可分特征空间变换到另一空间，通过这一变换使得在新空间线性可分，再用线性单元解决问题。

可调参数

隐层基函数中心、方差，输出单元权值。

5. 6. 6. 2. Hopfield 网络

Hopfield网络是一种反馈网络。反馈网络具有一般非线性系统的许多性质，如稳定性问题、各种类型的吸引子以及混沌现象等。它比前馈网络的内容复杂。

Hopfield还满足：

1. 权值对称：权矩阵为对称阵
2. 无自反馈：权矩阵对角线元素为0

5. 6. 6. 3. 自适应共振理论神经网络ART

自适应共振理论神经网络既能模拟人脑的可塑性，可以学习知识，又能模拟人脑的稳定性，学习新的知识但不破坏原有知识，即这种网络**不仅能记忆新的知识，而且还保留已记忆的内容。**

5. 6. 6. 4. 自组织特征映射神经网络SOM

人脑的记忆不是神经元与记忆模式的一一对应，而是一群神经元对应一个模式。

6. 支持向量机

回顾：

- 线性判别函数
- Fisher线性判别函数
- 感知准则函数
- 最小平方误差准则函数
- 多类问题

6. 1. 线性支持向量机

SVM从线性可分情况下的最优分类面发展而来。

最优分类面就是要求分类线不仅能将两类正确分开，并且**分类间隔最大**。

SVM考虑找到一个超平面，并且使训练集中的点距离分类面尽可能远，也就是使其两侧的空白区域（margin）尽可能大。

样本集：$\{x_n.t_n\}, n=1,2,\ldots,N, x_n \in \mathbb{R}^d; t_n \in \{-1,1\}$

分类器：$y(x)=w^Tx+b$

其中：

$$t_n=\begin{cases}1, y(x_n)>0 \text{ if } x_i \in w_1\\ -1, y(x_n)<0 \text{ if } x_i \in w_2\end{cases}$$

即：

$$t_n y(x_n)>0$$

样本集任意一点x_n到分类面的距离：

$$\frac{t_n y(x_n)}{||w||}=\frac{t_n(w^Tx_n+b)}{||w||}$$

优化w, b使得margin最大。

求解很复杂，可以令超平面最近的点有：$t_n(w^Tx_n+b)=1$，则所有点都有：$t_n(w^Tx_n+b)\geq 1$

这时只需要最大化$||w||^{-1}$，等价于：

$$\arg\min_{w,b}\frac{1}{2}||w||^2$$
$$s.t. \quad t_i[(w^Tx_i)+b]\geq 1$$

转化为二次规划问题。

拉格朗日乘子法求解即可。

6. 2. 非线性支持向量机

对于非线性可分的数据样本，可通过适当的函数变换，将其在高维空间中转化为线性可分。

例如异或问题：

二维样本集：$x=(x_1.x_2)$

第一类$(0,0),(1,1)$，第二类$(1,0),(0,1)$

映射函数$\phi(x)=(x_1,x_2,x_1x_2)$

则：$(0,0)\to(0,0,0),(1,1)\to(1,1,1)$,$(1,0)\to(1,0,0),(0,1)\to(0,1,0)$

线性可分了。即将$y(x)=w^Tx+b$转化为$y(x)=w^T\phi(x)+b$。

6. 2. 1. 核函数

决策时：原本为

$$y(x)=\sum_{n=1}^{N}a_nt_nx^Tx_n+b$$

转化为

$$y(x)=\sum_{n=1}^{N}a_nt_nK(x,x_n)+b$$

其中：

$$K(x,x_n)=\phi(x)^T\phi(x_n)$$

核函数在特征空间中直接计算数据映射后的内积，简化了计算过程。

基于泛函理论，我们只需要知道核函数，没必要知道非线性变换的表达形式。

核函数需要满足**Mercer定理**：

$\mathbb{R}^n\times\mathbb{R}^n\to\mathbb{R}$上的映射$k$是一个有效核函数，当且仅当对于训练样本其相应的核函数矩阵是对称半正定的。即对于任何平方可积函数$g(x)$，有$\iint k(x,y)g(x)g(y)\mathrm{d}x\mathrm{d}y\geq 0$。

根据问题和数据不同，选择不同的核函数，如：

- 线性核：$k(x_1,x_2)=x_1^Tx_2$
- 多项式核：$k(x_1,x_2)=(<x_1,x_2>+R)^d$
- 高斯核：$k(x_1,x_2)=\exp\left\{-\frac{||x_1-x_2||^2}{2\sigma^2}\right\}$
- sigmoid核：$k(x_1,x_2)=\tanh\left(\beta_0x_1^Tx_2+\beta_1\right)$

6. 2. 2. 非线性SVM基本思想

1. 通过非线性变换将输入空间变换到高维空间。
2. 在新空间中求最优分类面。

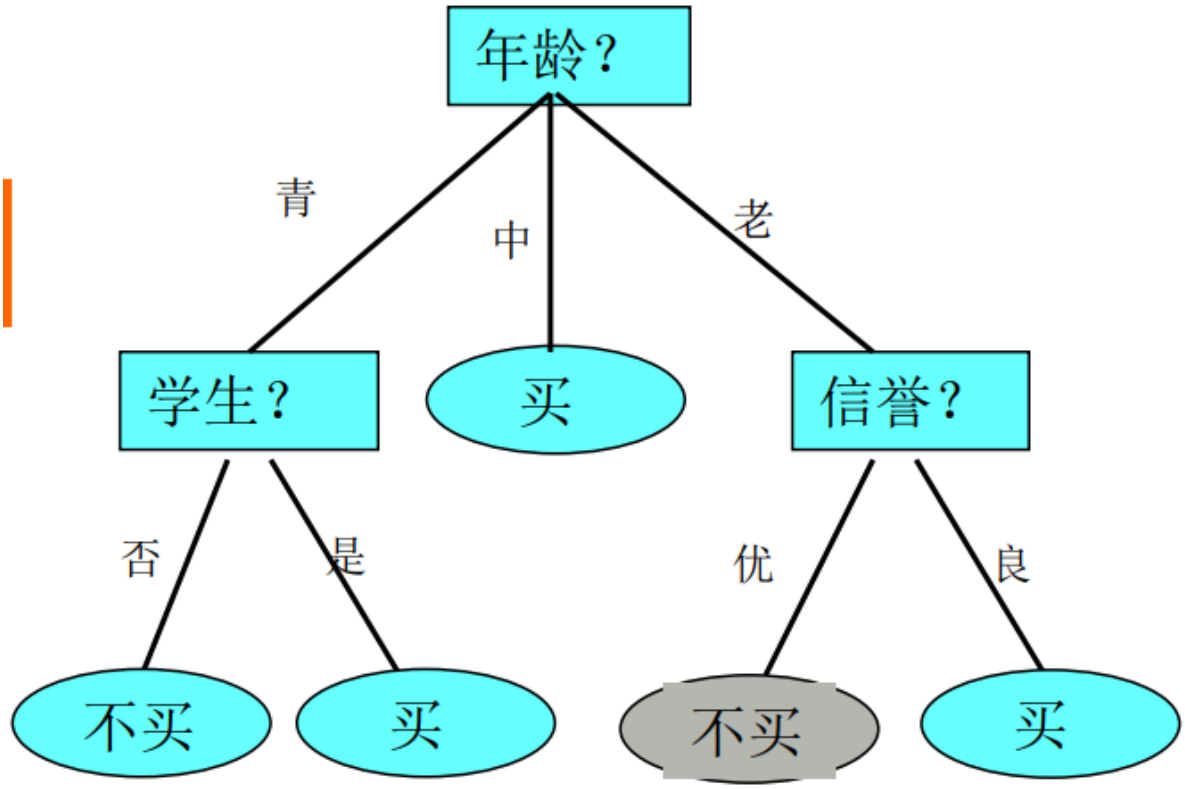
非线性变换通过定义适当的内积核函数实现。而选择不同的核函数，可以看作是选择不同的相似性度量。线性支持向量机就是采用欧氏空间中的内积作为相似性度量。

6. 3. 其他非线性分类方法

6. 3. 1. 决策树

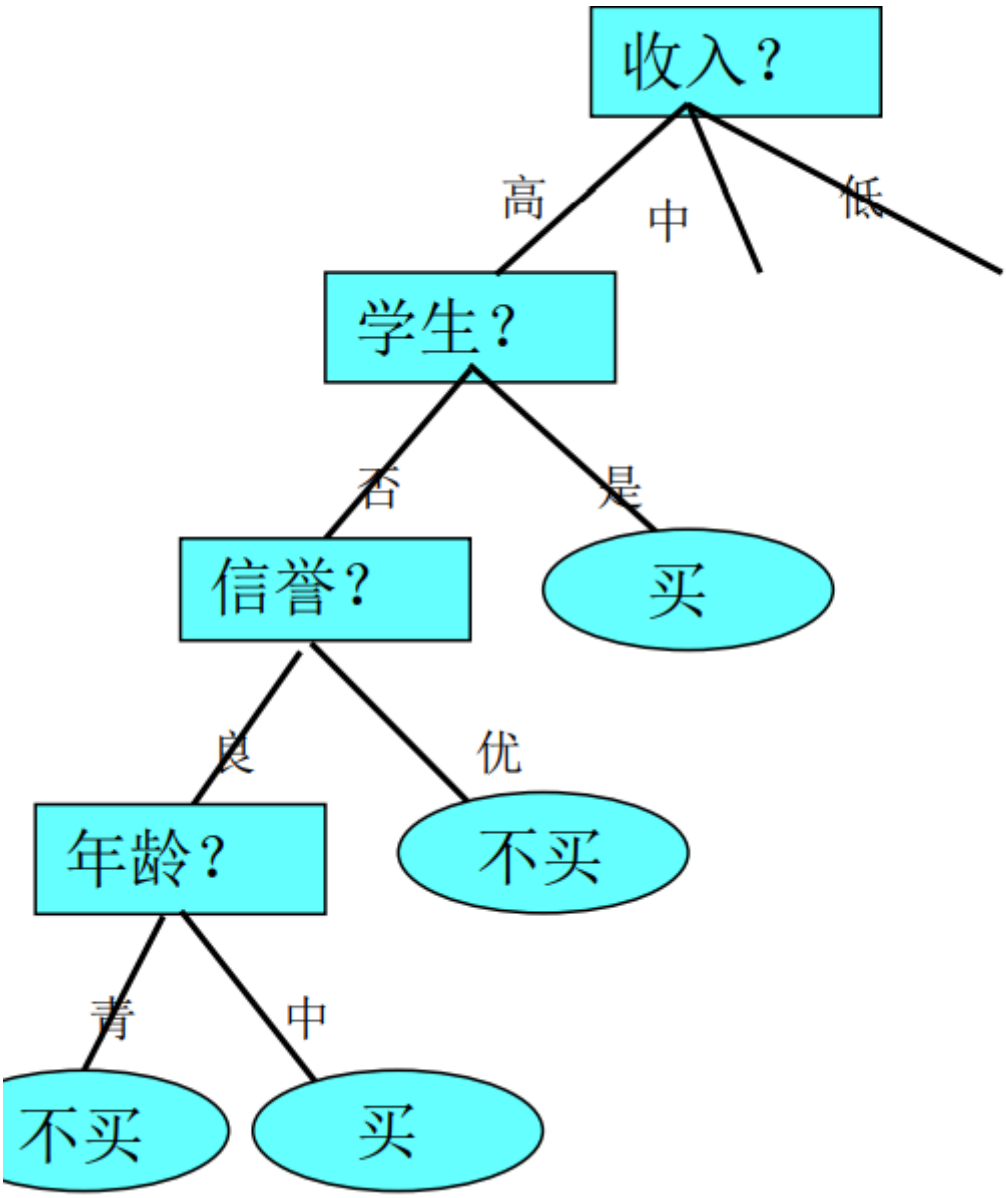
内部节点上选用一个属性进行分割，分支表示输出，叶子节点表示类。

如：



决策树算法对数据处理的过程中，将数据按树状结构分成若干分支形成决策树，从根到树叶的每条路径创建一个规则。

但是，也有很烂的决策树，如：



怎样让决策树深度没那么深？需要对决策树生成步骤进行改变。

6. 3. 2. 决策树的生成

1. 在条件属性集中选择最有分类标识能力的属性作为决策树当前节点。
2. 根据当前决策属性取值不同，将训练样本数据集划分为若干子集。
3. 针对上一步得到每一个子集，重复上述过程，直到子集中所有元组都属于同一类，不能再进一步划分为止。

6. 3. 3. 决策树分类算法

ID3算法。

基本思想：按一定准则选择一个条件属性作为根节点，根据其属性取值将整个例子空间划分为几个子空间，然后递归使用这一准则继续划分，直至所有底层子空间只含有一类例子。

一些数学定义：

6. 3. 3. 1. 熵

度量样例的纯度。

定义：设S是n个数据样本的集合，将样本划分为c个不同的类，每个类含样本数n_i，则S划分为c个类的熵为：

$$\text{Entropy}(S) = -\sum_{i=1}^{c} \frac{n_i}{n} \log_2 \frac{n_i}{n}$$

6. 3. 3. 2. 信息增益

衡量属性区分训练样例的能力：一个属性的信息增益就是由于**使用这个属性**分割样例而**导致的熵的降低**。

属性A相对样例集合S的信息增益定义：

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S)$$

则ID3算法每次选择信息增益值最大的属性作为最佳属性，进行分类。

6. 3. 4. 其它决策树建立方法

CART算法、C4.5算法等。

6. 3. 5. 决策树的数据准备

- 数据清理：删除、减少噪声，补填空缺值。
- 相关性分析： 对于问题无关的属性或者不能归纳的属性，直接删除。
- 数据变换：数据标准化，数据归纳，并控制属性可能值不超过七种。

6. 3. 6. 剪枝

先剪枝：

- 数据划分法
- 阈值法
- 信息增益的统计显著性分析

后剪枝：

- 减少分类错误修剪法
- 最小代价与复杂性折中
- 最小描述长度准则

6. 3. 7. 随机森林

决策树方法尤其受数据集影响大，容易过学习。

统计学策略：自举（bootstrap）通过对现有样本重采样形成多个样本集，模拟数据中的随机性。

随机森林步骤：

1. 对样本数据进行自举重采样
2. 用每个重采样样本集作为训练样本构造一个决策树
3. 得到所需数据的决策后，对这些树的输出进行决策，得票最多的类作为随机森林的决策。