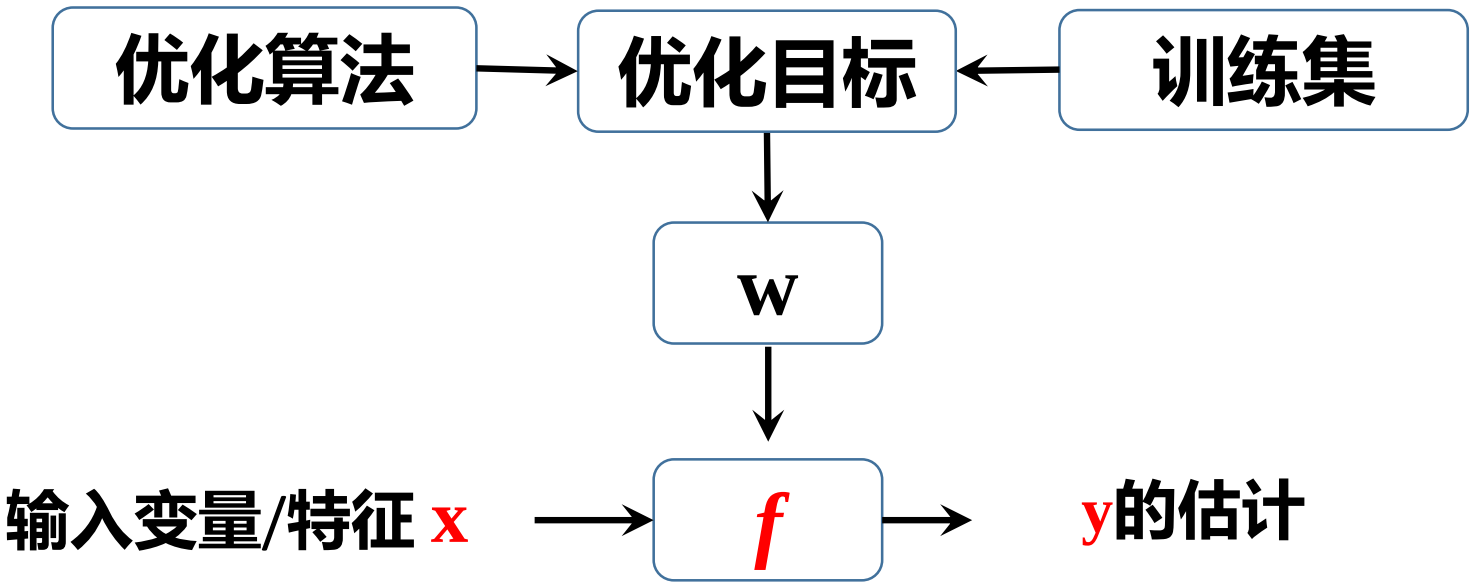
软件开发环境国家重点实验室
State Key Laboratory of Software Development Environment

机器学习

刘祥龙
北京航空航天大学计算机学院
软件开发环境国家重点实验室

第四讲

线性回归



$$f(\mathbf{x}) = w_0 + w_1 x_1 + \ldots + w_m x_m$$

$$= w_0 + \sum_{i=1}^{m} w_i x_i$$

$$= \sum_{i=0}^{m} w_i x_i \qquad x_0 = 1$$

基函数

一般地：

基函数

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x})$$

$\phi(x)$ **称为基函数（Basis function）**

一般情况下 $\phi_0(x) = 1$，**此时**w_0**为截矩或偏置（bias）**

最简单的情况下： $\phi_j(x) = x_j$

多元线性回归 - 最小二乘法

- 线性模型+最小均方误差+优化：最小二乘法

$$\hat{\boldsymbol{w}}^* = \arg\min_{\hat{w}} \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}^{\mathrm{T}}\right)\left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)$$

令 $E_{\hat{\boldsymbol{w}}} = \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)^{\mathrm{T}}\left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)$ ，对$\hat{\boldsymbol{w}}$ 求导得到

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

令上式为零可得$\hat{\boldsymbol{w}}$ 最优解的闭式解

多元线性回归 - 最小二乘法

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 是满秩矩阵或正定矩阵，则

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

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

$$f\left(\hat{\boldsymbol{x}}_i\right) = \hat{\boldsymbol{x}}_i^{\mathrm{T}}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

- $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 不是满秩矩阵：引入正则化

概率视角：最大似然估计

- 假定观测数据由确定的函数加高斯噪声组成：

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \qquad \text{where} \qquad p(\epsilon|\beta) = \mathcal{N}(\epsilon|0, \beta^{-1})$$

即：$$p(t|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}).$$

似然函数 $$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}).$$

对数似然

$$\begin{aligned} \ln p(\mathbf{t}|\mathbf{w}, \beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}) \\ &= \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi) - \beta E_D(\mathbf{w}) \end{aligned}$$

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2$$

概率视角：最大似然估计

- **求梯度并求零点的值：**

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t}|\mathbf{w}, \beta) = \beta \sum_{n=1}^{N} \left\{ t_n - \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}_n) \right\} \boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} = \mathbf{0}.$$

解得w：

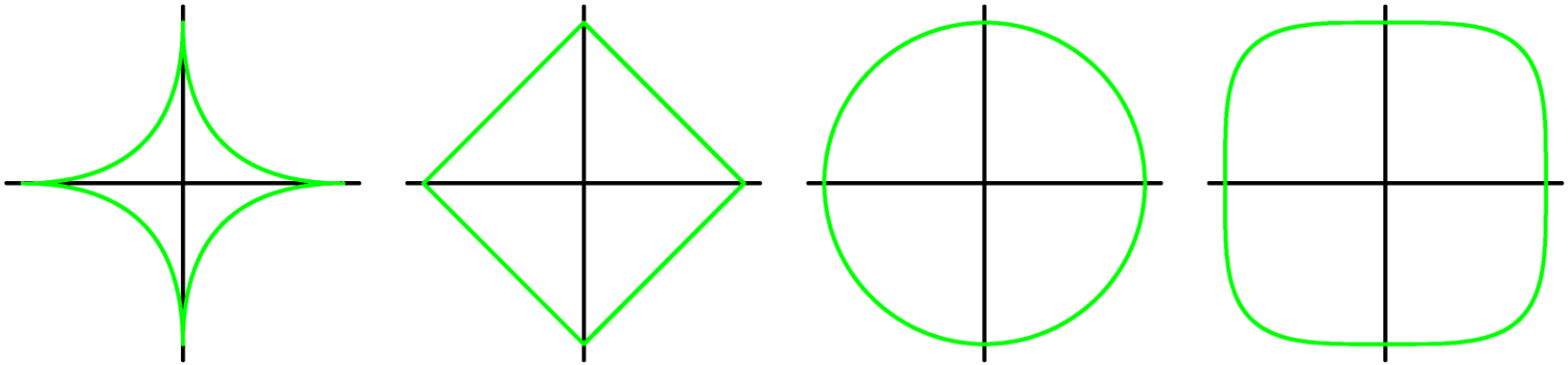
$$\mathbf{w}_{\mathrm{ML}} = \left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{t}$$

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}.$$

如何防止过拟合

- 加入正则项

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$$



Lasso　　Quadratic

如何防止过拟合

- 加入正则项

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

二次正则项

求得 w： $\mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}.$

软件开发环境国家重点实验室
State Key Laboratory of Software Development Environment

分类问题

离散值预测：二分类

- 线性回归：$z = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$
- 二分类：$y \in \{0, 1\}$

- 如何建立分类与线性回归的联系？
  - 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
  - 缺点：不连续

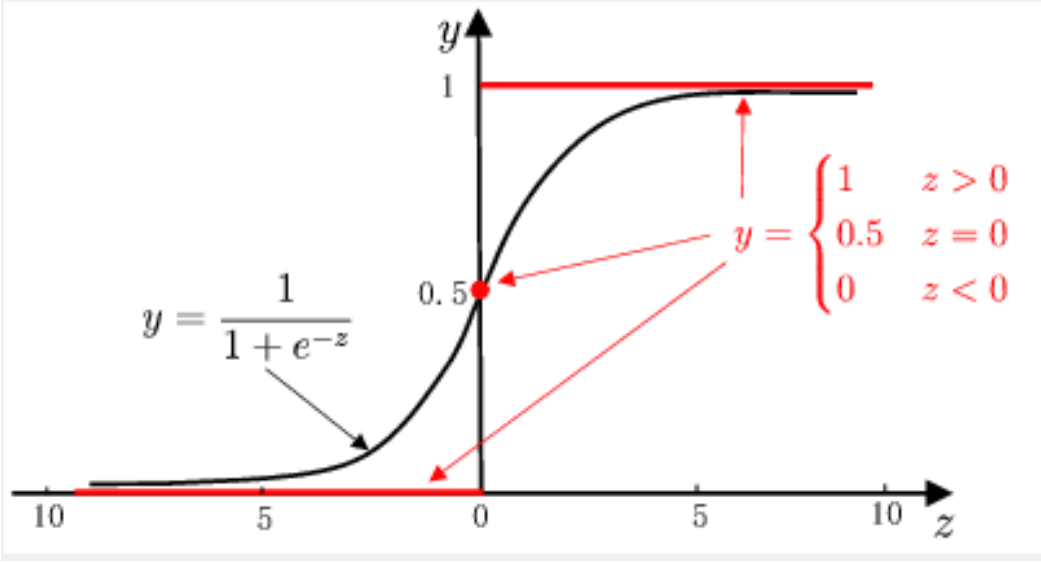
二分类

- 替代函数——逻辑函数（logistic function）
  - 单调可微、任意阶可导

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

S函数

皮埃尔·弗朗索瓦·韦吕勒在1844或1845年在研究它与人口增长的关系时命名的



逻辑回归（ logistic regression)

- 运用逻辑函数

$$y = \frac{1}{1 + e^{-z}} \quad \text{变为} \quad y = \frac{1}{1 + e^{-(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b)}}$$

- 逻辑回归的优点
  - 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

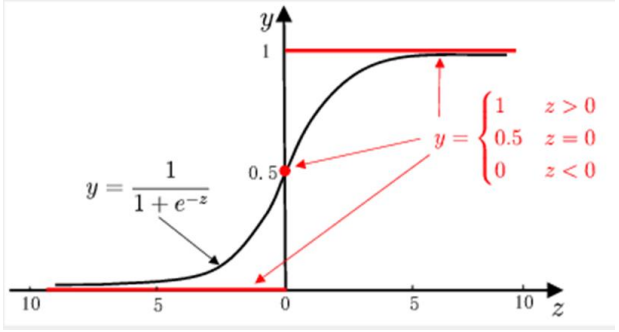
逻辑回归 – 极大似然法

- 分类概率

$$\ln \frac{p\left(y=1 \mid \boldsymbol{x}\right)}{p\left(y=0 \mid \boldsymbol{x}\right)} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$$

$$p\left(y=1 \mid \boldsymbol{x}\right) = \frac{e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$$

$$p\left(y=0 \mid \boldsymbol{x}\right) = \frac{1}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$$



逻辑回归 – 极大似然法

- 极大似然法（maximum likelihood）
  - 给定数据集

$$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m}$$

  - 最大化样本属于其真实标记的概率
    - 最大化似然函数

$$\ell(\boldsymbol{w}, b) = \prod_{i=1}^{m} p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}_i, b)$$

    - 最大化对数似然函数

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

逻辑回归 – 极大似然法

- 转化为最小化负对数似然函数求解
  - 令 $\boldsymbol{\beta}=(\boldsymbol{w};b)$, $\hat{\boldsymbol{x}}=(\boldsymbol{x};1)$，则 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b$可简写为 $\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}$
  - 再令 $$p_1\left(\hat{\boldsymbol{x}}_i;\boldsymbol{\beta}\right)=p\left(y=1\mid\hat{\boldsymbol{x}};\boldsymbol{\beta}\right)$$

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

则似然项可重写为

$$p\left(y_i\mid\boldsymbol{x}_i;\boldsymbol{w}_i,b\right)=y_ip_1\left(\hat{\boldsymbol{x}}_i;\boldsymbol{\beta}\right)+\left(1-y_i\right)p_0\left(\hat{\boldsymbol{x}}_i;\boldsymbol{\beta}\right)$$

  - 目标函数等价于

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

逻辑回归 - 极大似然法

- 优化求解 $$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta})$$

- **牛顿法**第t+1轮迭代解的更新公式

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

其中关于 β 的一阶、二阶导数分别为

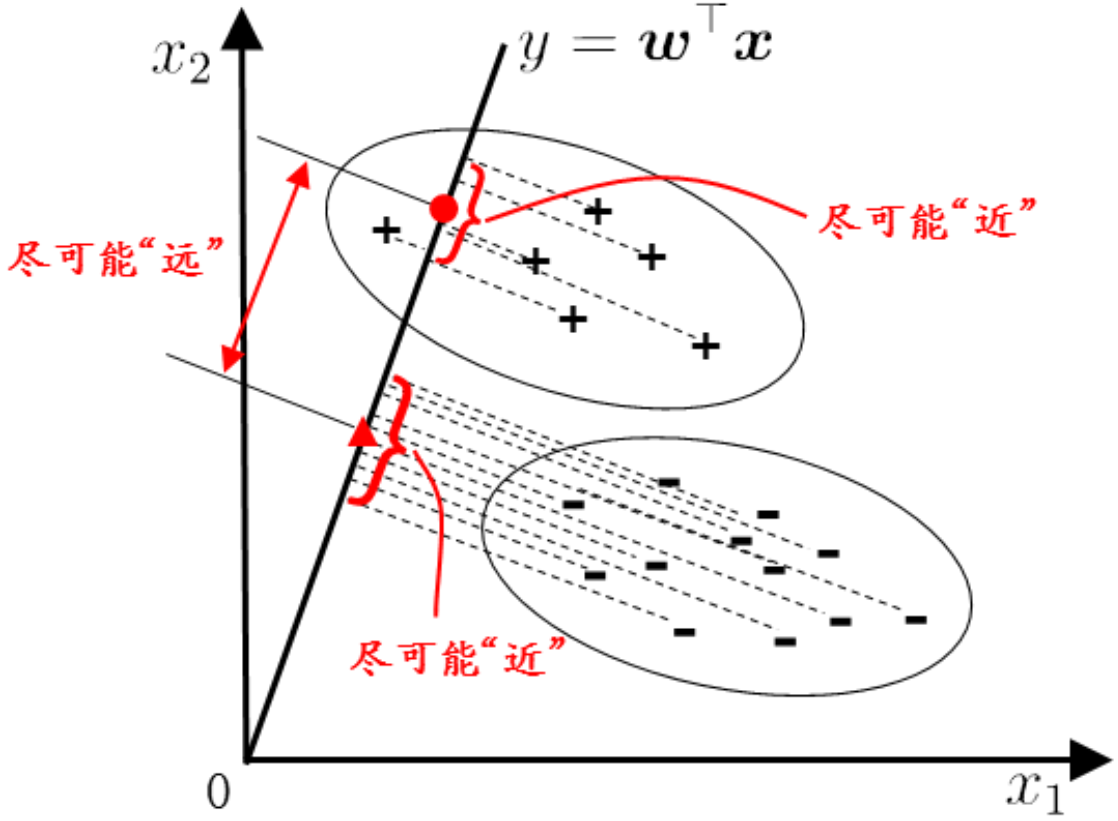
$$\frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -\sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \left(y_i - p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)\right)$$

$$\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathrm{T}}} = \sum_{i=1}^{m} \hat{\boldsymbol{x}}_i \hat{\boldsymbol{x}}_i^{\mathrm{T}} p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)\left(1 - p_1\left(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}\right)\right)$$

高阶可导连续凸函数，梯度下降法/牛顿法 [Boyd and Vandenberghe, 2004]

其他二分类任务 - 线性判别分析

- 线性判别分析（Linear Discriminant Analysis）
  [Fisher, 1936]



LDA也可被视为一种监督降维技术

线性判别分析

- LDA的思想
  - 欲使同类样例的投影点尽可能接近：可以让同类样例投影点的协方差尽可能小
  - 欲使异类样例的投影点尽可能远离：可以让类中心之间的距离尽可能大

- 变量说明
  - 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：$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0$ 和 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1$
  - 两类样本的协方差：$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w}$ 和 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}$

线性判别分析

- 最大化目标

$$
\begin{aligned}
J &= \frac{\left\| \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\mu}_1 \right\|_2^2}{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{\Sigma}_0 \boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}} \boldsymbol{\Sigma}_1 \boldsymbol{w}} \\
&= \frac{\boldsymbol{w}^{\mathrm{T}} \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right)^{\mathrm{T}} \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \left(\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1 \right) \boldsymbol{w}}
\end{aligned}
$$

- 类内散度矩阵

$$
\begin{aligned}
\mathbf{S}_w &= \boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1 \\
&= \sum_{\boldsymbol{x} \in X_0} \left(\boldsymbol{x} - \boldsymbol{\mu}_0 \right) \left(\boldsymbol{x} - \boldsymbol{\mu}_0 \right)^{\mathrm{T}} + \sum_{\boldsymbol{x} \in X_1} \left(\boldsymbol{x} - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{x} - \boldsymbol{\mu}_1 \right)^{\mathrm{T}}
\end{aligned}
$$

- 类间散度矩阵 $\mathbf{S}_b = \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right) \left(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1 \right)^{\mathrm{T}}$

线性判别分析

- 广义瑞利商（generalized Rayleigh quotient）

$$J = \frac{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}}$$

- 令 $\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} = 1$，最大化广义瑞利商等价形式为

$$\min_{\boldsymbol{w}} \ -\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}$$

$$\text{s.t. } \boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} = 1$$

- 运用拉格朗日乘子法 $\mathbf{S}_b\boldsymbol{w} = \lambda\mathbf{S}_w\boldsymbol{w}$ $\mathbf{S}_w^{-1}\mathbf{S}_b w = \lambda\, w$

线性判别分析

- 同向向量 $S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$ $S_B w = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w = (\mu_1 - \mu_2) * \lambda_w$

$$\mathbf{S}_b \boldsymbol{w} = \lambda (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

- 结果

$$\mathbf{S}_w^{-1} \mathbf{S}_b w = \lambda w \longrightarrow \boldsymbol{w} = \mathbf{S}_w^{-1} (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

- 求解
  - 奇异值分解 $\mathbf{S}_w = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T$

LDA推广 - 多分类任务

- 优化目标

$$\max_{\mathbf{W}} \frac{\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}\right)}{\operatorname{tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_w\mathbf{W}\right)}$$

其中 $\mathbf{W} \in \mathbb{R}^{d\times(N-1)}$

$$\mathbf{S}_b = \sum_{i=1}^{N} m_i \left(\boldsymbol{\mu}_i - \boldsymbol{\mu}\right)\left(\boldsymbol{\mu}_i - \boldsymbol{\mu}\right)^T$$

$$\mathbf{S}_w = \sum_{i=1}^{N} \mathbf{S}_{w_i}$$

$$\mathbf{S}_{w_i} = \sum_{\boldsymbol{x}\in X_i} \left(\boldsymbol{x} - \boldsymbol{\mu}_i\right)\left(\boldsymbol{x} - \boldsymbol{\mu}_i\right)^T$$

$$\mathbf{S}_b\mathbf{W} = \lambda\mathbf{S}_w\mathbf{W}$$

$\mathbf{W}$的闭式解则是 $\mathbf{S}_w^{-1}\mathbf{S}_b$ 的N-1个最大广义特征值所对应的特征向量组成的矩阵

- 多分类LDA将样本投影到N-1维空间，N-1通常远小于数据原有的属性数，因此LDA也被视为一种监督降维技术

多分类学习

- 多分类学习方法
  - 二分类学习方法推广到多类
  - 利用二分类学习器解决多分类问题（常用）
    - 对问题进行拆分，为拆出的每个二分类任务训练一个分类器
    - 对于每个分类器的预测结果进行集成以获得最终的多分类结果

- 拆分策略
  - 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

- 拆分阶段
  - N个类别两两配对
    - N(N-1)/2 个二类任务
  - 各个二类任务学习分类器
    - N(N-1)/2 个二类分类器

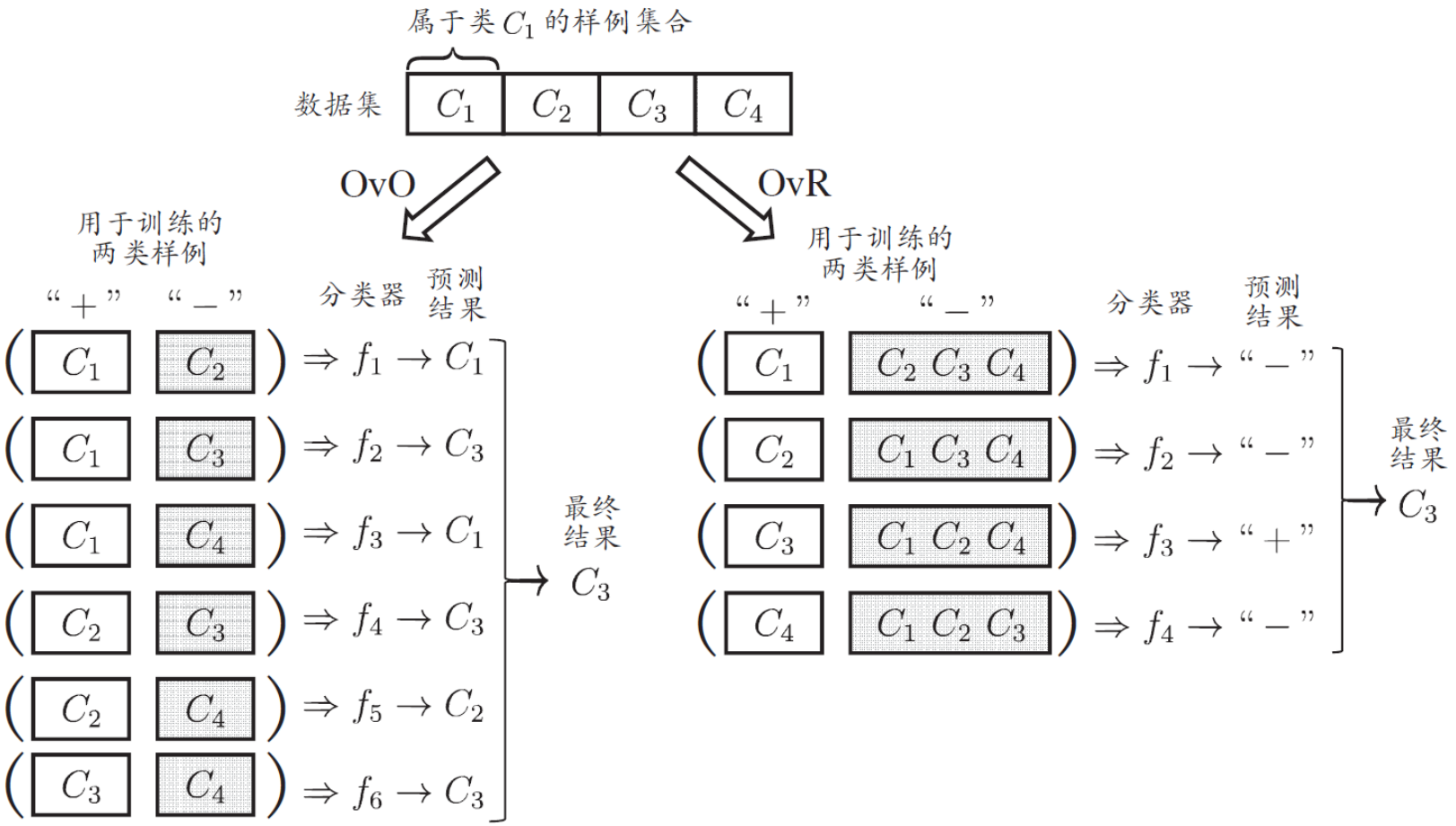
- 测试阶段
  - 新样本提交给所有分类器预测
    - N(N-1)/2 个分类结果
  - 投票产生最终分类结果
    - 被预测最多的类别为最终类别

多分类学习 - 一对其余

- 任务拆分
  - 某一类作为正例，其他反例
    - N 个二类任务
  - 各个二类任务学习分类器
    - N 个二类分类器

- 测试阶段
  - 新样本提交给所有分类器预测
    - N 个分类结果
  - 比较各分类器预测置信度
    - 置信度最大类别作为最终类别

多分类学习 - 两种策略比较



多分类学习- 两种策略比较

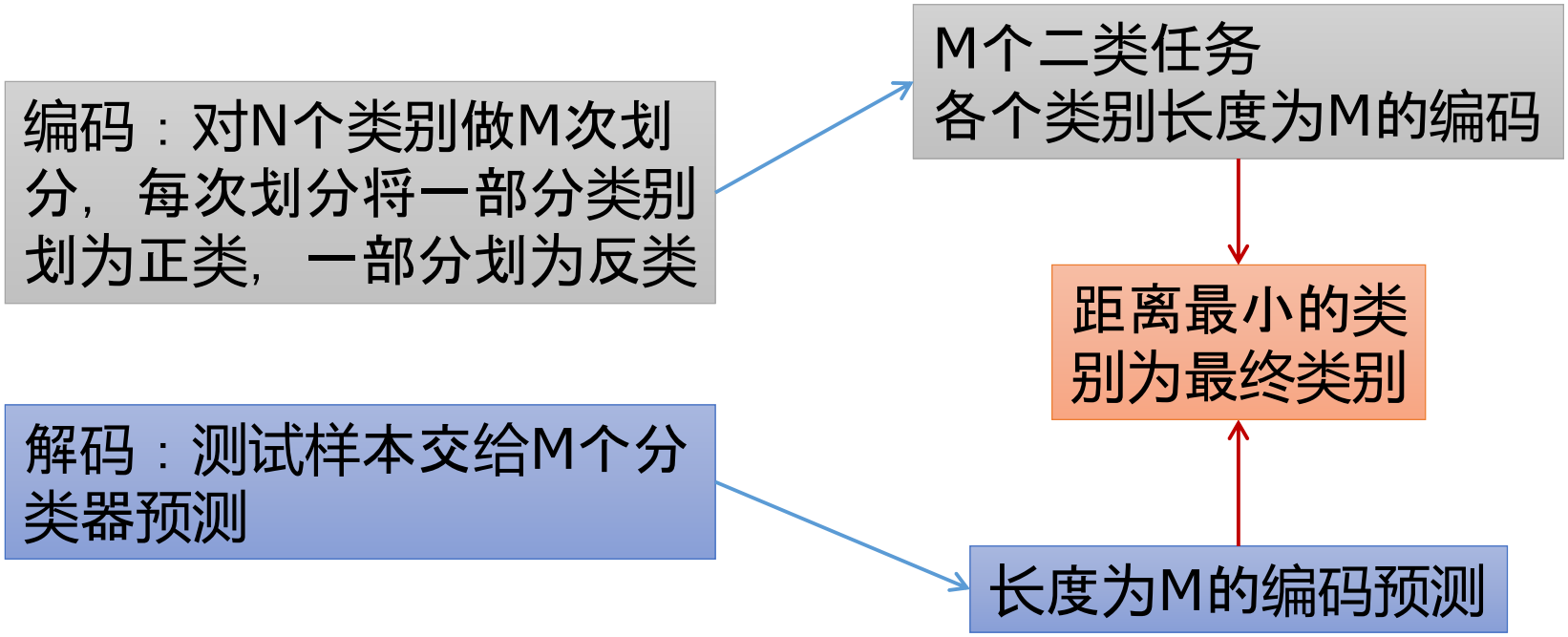
- 一对一
- 训练N(N-1)/2个分类器，存储开销和测试时间大
- 训练只用两个类的样例，训练时间短

- 一对其余
- 训练N个分类器，存储开销和测试时间小
- 训练用到全部训练样例，训练时间长

预测性能取决于具体数据分布，多数情况下两者差不多

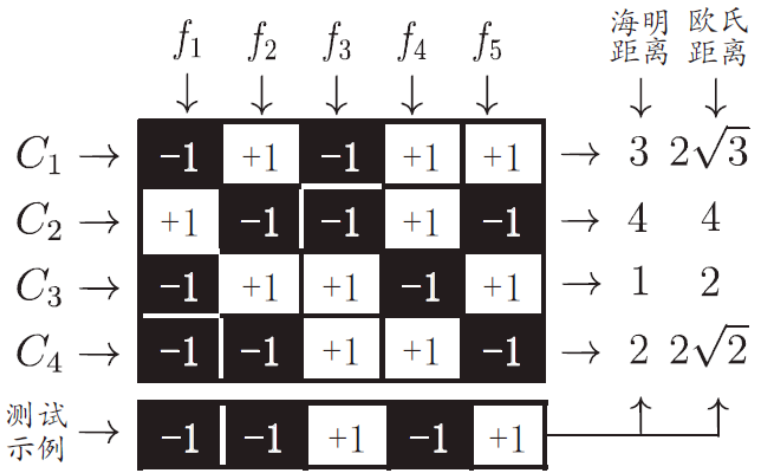
多分类学习 - 多对多

- 多对多（Many vs Many, MvM）
  - 若干类作为正类，若干类作为反类
- 纠错输出码（Error Correcting Output Code, ECOC）



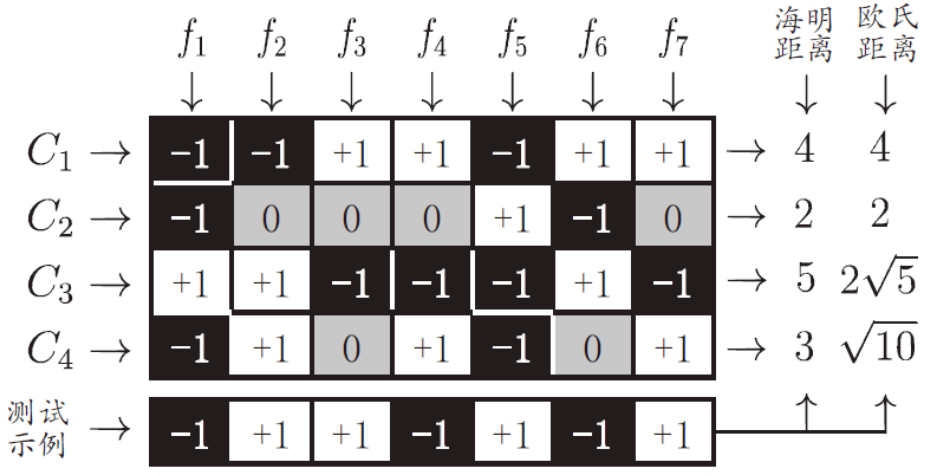
多分类学习 - 多对多

- 纠错输出码(Error Correcting Output Code, ECOC)



(a) 二元 ECOC 码

[Dietterich and Bakiri, 1995]



(b) 三元 ECOC 码

[Allwein et al. 2000]

- ECOC编码对分类器错误有一定容忍和修正能力，编码越长、纠错能力越强
- 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

软件开发环境国家重点实验室
State Key Laboratory of Software Development Environment

支持向量机

支持向量机

- C. Cortes和V. Vapnik (1995年提出)
  - 支持向量机是基于统计学习理论(Statistical Learning Theory, SLT)发展起来的一种新的机器学习的方法。
  - 统计学习理论主要创立者是Vladimir N. Vapnik。

Google

like us on facebook

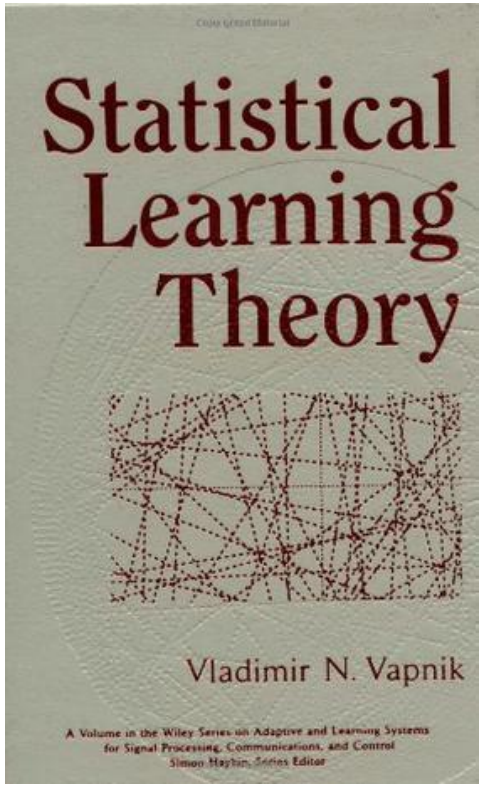
AT&T

Vladimir N. Vapnik

- 1936年 出生于苏联
- 1958年 乌兹别克国立大学 硕士
- 1964年 莫斯科控制科学学院 博士
- 1964-1990年 莫斯科控制科学学院
- 1991-2001年 美国AT&T贝尔实验室
  发明支持向量机理论
- 2002-2014年 NEC实验室(美国)
  从事机器学习研究
- 1995年和2003年 伦敦大学皇家霍洛威学院和美国哥伦比亚大学计算机专业的教授
- 2006年 美国国家工程院院士
- 2014年至今 美国Facebook公司 从事人工智能研究
- 2017年 IEEE John von Neumann Medal

V. Vapnik对于统计机器学习的贡献

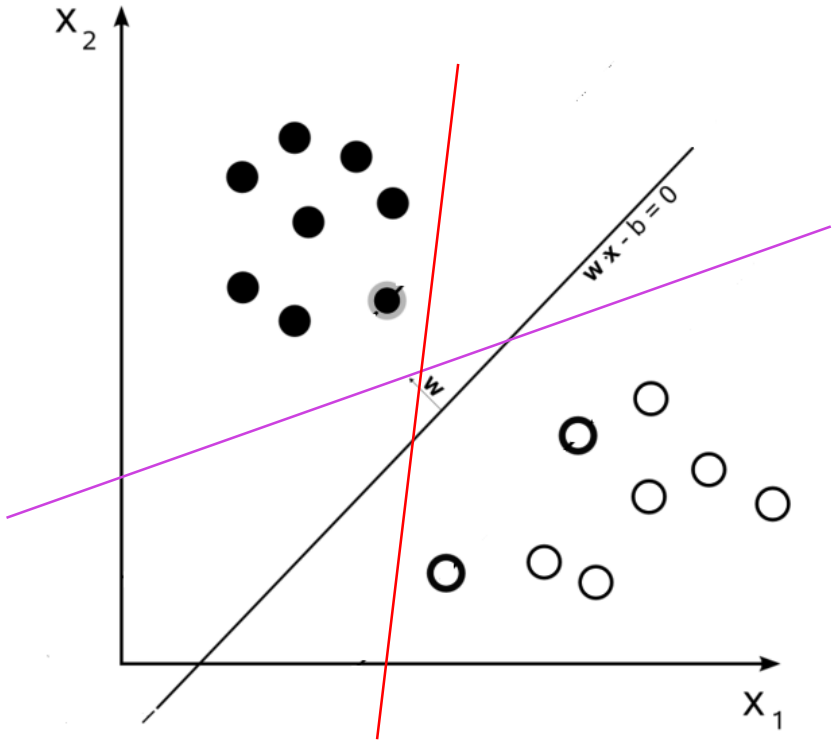
- 1968年，Vapnik和Chervonenkis提出了VC熵和VC维的概念，这些是统计学习理论的核心概念。同时，他们发现了泛函空间的大数定理，得到了关于收敛速度的非渐进界的主要结论。
- 1974年，Vapnik和Chervonenkis提出了结构风险最小化归纳原则。
- 1989年，Vapnik和Chervonenkis发现了经验风险最小化归纳原则和最大似然方法一致性的充分必要条件，完成了对经验风险最小化归纳推理的分析。
- 90年代中期，有限样本情况下的机器学习理论研究逐渐成熟起来，形成了较完善的理论体系—统计学习理论。



线性可分支持向量机：函数

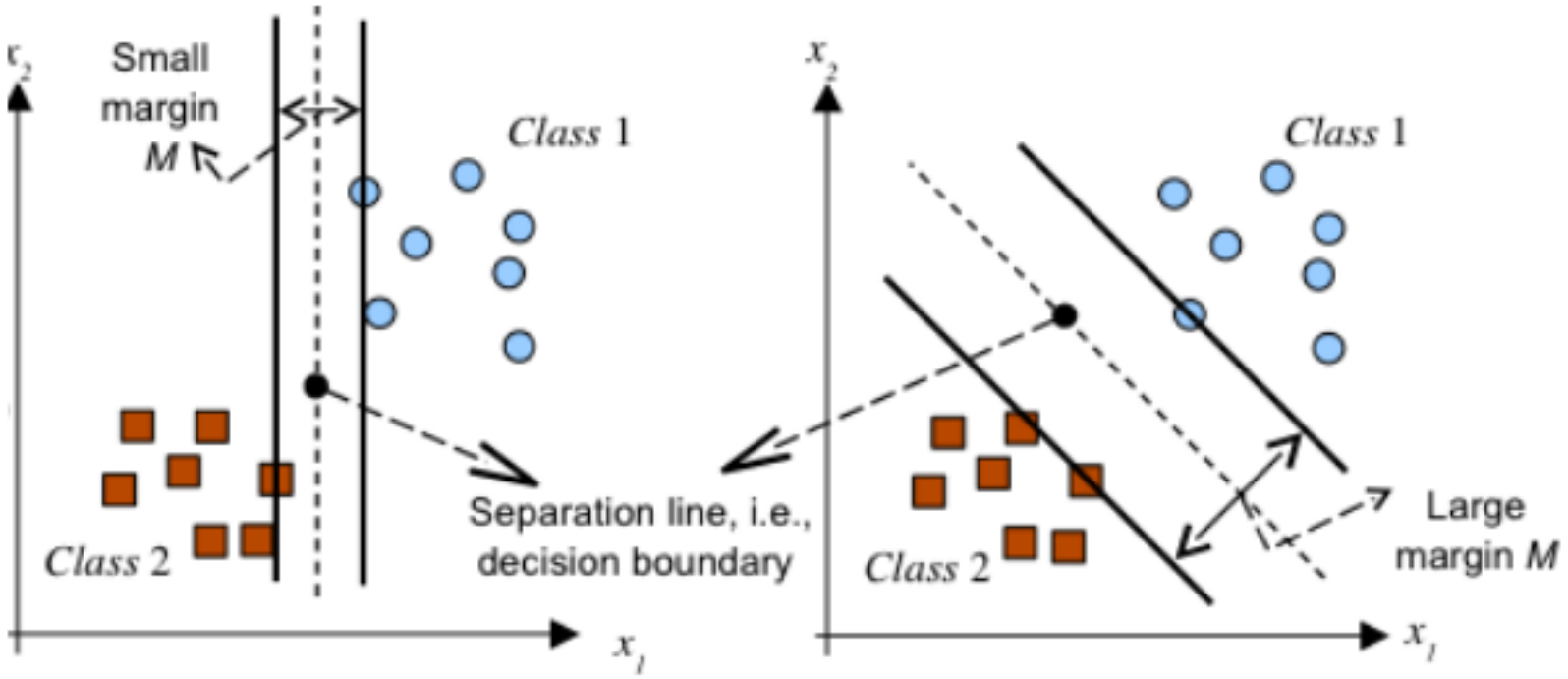
- 训练数据集
  - $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\},\ x_i \in R^n, y_i \in \{+1, -1\}$

- 二分类目标
  - 函数：分离超平面$w^T x_i + b = 0$
  - 二分类函数$y_i = \text{sign}(w^T x_i + b)$
  - 若线性可分
    - 感知机：存在无数分离超平面
    - 线性可分支持向量机：唯一解



支持向量机：优化目标

- 线性可分的最优分类面
  - 最优分类面就是要求分类线不但能将两类正确分开(训练错误率为0)，且使分类间隔最大。SVM考虑寻找一个满足分类要求的超平面，并且使训练集中的点距离分类面尽可能的远，也就是寻找一个分类面使它两侧的空白区域(Margin)最大。



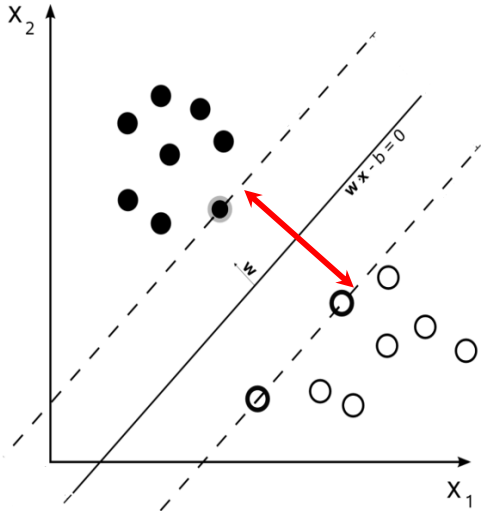
线性可分支持向量机

- 函数间隔
  - $|w^T x + b|$表示点相对分离超平面的距离，反映分类确信度
  - $y(w^T x + b)$同时反映预测与实际分类是否一致及确信度
  - 数据集T上函数间隔

$$\min_{i=1,2,\ldots,N} y_i(w^T x_i + b)$$

- 几何间隔
  - $(\mathrm{w}, \mathrm{b}) \rightarrow (2\mathrm{w}, 2\mathrm{b})$时超平面不变，函数间隔发生变化
  - 函数间隔→几何间隔，可约束$\|w\| = 1$

$$\gamma = \min_{i=1,2,\ldots,N} \frac{y_i(w^T x_i + b)}{\|w\|}$$



线性可分支持向量机

- 间隔最大化
  - 正确划分，几何间隔最大
  - 直观解释：充分大的确信度对数据进行分类
    - *最难分的点以足够大的确信度分开*
    - *对未知新数据有很好的泛化能力*
- 最大间隔分类超平面

$$\max_{w,b} \gamma \quad s.t. \quad \frac{y_i(w^T x_i + b)}{\|w\|} \geq \gamma$$

$\hat{\gamma} = \|w\|\gamma$

$$\max_{w,b} \frac{\hat{\gamma}}{\|w\|} \quad s.t. \quad y_i(w^T x_i + b) \geq \hat{\gamma}$$

$\hat{\gamma} = 1$

$$\max_{w,b} \frac{1}{\|w\|} \quad s.t. \quad y_i(w^T x_i + b) \geq 1$$

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad s.t. \quad y_i(w^T x_i + b) \geq 1$$

线性可分支持向量机：优化求解

- 可解
  - 凸优化: f, g凸函数， h仿射函数

$$\min_{w} f(w)$$
$$s.t. \quad g_i(w) \leq 0$$
$$h_i(w) = 0$$

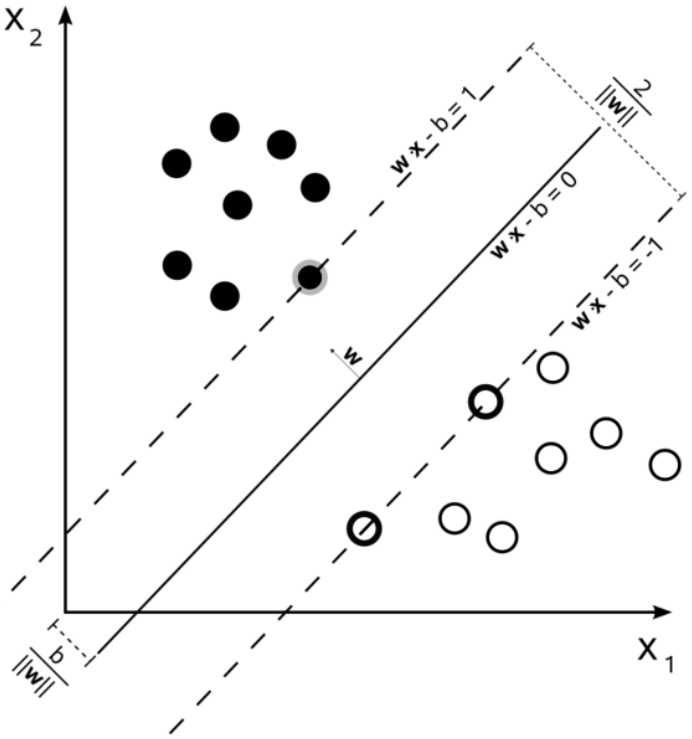
$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t. \quad y_i(w^T x_i + b) \geq 1$$

- 解存在唯一
  - 存在性：目标函数有解(w^*, b^*)，且非 $(w^*, b^*) \neq (0, b)$
  - 唯一性：假设存在两个解 → $w^* = 0$或者两个解相同 → w^*唯一

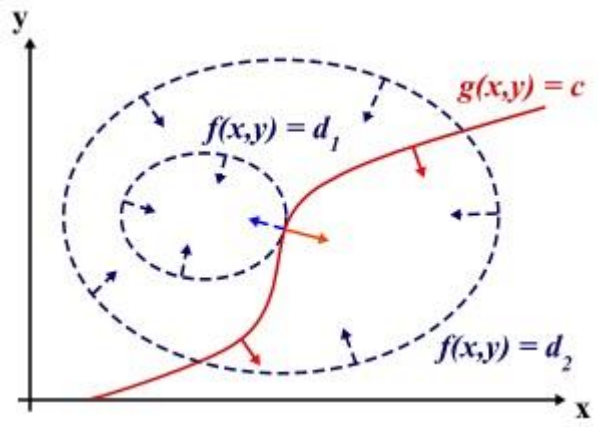
线性可分支持向量机

- 支持向量
  - 使 $y_i(w^T x_i + b) = 1$成立的x_i
  - 支持向量所在平面为$\mathrm{w}^{\mathrm{T}}\mathrm{x} + b = \pm 1$
  - 两平面距离称为间隔

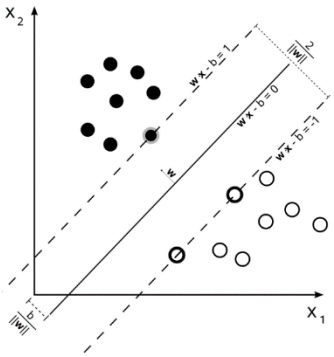
- 特点
  - 只有支持向量决定分离平面
  - 支持向量较少，具有稀疏性



线性可分支持向量机



等式约束的优化问题



$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t.\ \ y_i(w^T x_i + b) \geq 1$$

不等式约束的优化问题?

$$min \quad y = f(x_1, x_2)$$
$$s.t. \quad g(x_1, x_2) = 0$$

$$y_\lambda = f(x_1, x_2) + \lambda(g(x_1, x_2))$$

$$\min f_0(x)$$
$$s.t.\ f_i(x) \leq 0, \quad i = 1, \ldots, m$$
$$h_i(x) = 0, \quad i = 1, \ldots, p$$

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

对偶问题

$$\begin{aligned} &\min\ f_0(x) \\ &s.t.\ f_i(x) \le 0, \quad i = 1, \dots, m \\ &\quad\ h_i(x) = 0, \quad i = 1, \dots, p \end{aligned}$$

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

定义 Lagrange 对偶函数为 Lagrange 函数关于 x 取最小值，即对于 $\lambda \in \mathbf{R}^m$ ， $\nu \in \mathbf{R}^p$ ，有

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right) 。$$

Lagrange 对偶函数是一族关于 (λ, ν) 的仿射函数的逐点下确界，即使原问题不是凸的，对偶函数也是凹函数。

对偶问题

可以证明，Lagrange 对偶函数构成了原问题(1)最优值 p^* 的下界，即对任意 $\lambda \succcurlyeq = 0$ 和 ν 有

$$\inf_{x\in\mathcal{D}}\left(f_0(x)+\sum_{i=1}^{m}\lambda_i f_i(x)+\sum_{i=1}^{p}\nu_i h_i(x)\right) \qquad g(\lambda,\nu)\le p^*$$

$$\begin{aligned}&\min\ f_0(x)\\ &s.t.\ f_i(x)\le 0,\quad i=1,\dots,m\\ &\qquad h_i(x)=0,\quad i=1,\dots,p\end{aligned}$$

求解从 Lagrange 对偶函数求得原函数最优值的最好下界，构成 Lagrange 对偶问题

$$\max g(\lambda,\nu)$$
$$s.t.\quad \lambda \succcurlyeq = 0$$

Lagrange 对偶函数是凹函数，所以 Lagrange 对偶问题是凸优化问题。

对偶问题

弱对偶性

我们用 d^* 表示 Lagrange 对偶问题的最优解，则有

$$d^* \leq p^*$$

即使原问题不是凸优化问题，这个不等式也成立。这个性质称作弱对偶性。

强对偶性

如果 $d^* = p^*$，则称强对偶性成立。如果原问题是凸优化问题，则通常（但不总是）强对偶性成立。

KKT条件

$$\inf_{x\in\mathcal{D}}\left(f_0(x)+\sum_{i=1}^{m}\lambda_i f_i(x)+\sum_{i=1}^{p}\nu_i h_i(x)\right) = p^*$$

$$\begin{aligned}&\min\ f_0(x)\\ &s.t.\ f_i(x)\le 0,\quad i=1,\dots,m\\ &\quad\ h_i(x)=0,\quad i=1,\dots,p\end{aligned}$$

当强对偶性成立时，x^* 和 (λ^*,ν^*) 分别是原问题和对偶问题的最优解，则其必须满足 KKT 条件：

$$f_i(x^*)\le 0,\quad i=1,\cdots,m$$

$$h_i(x^*)=0,\quad i=1,\cdots,p$$

$$\lambda_i^*\ge 0,\quad i=1,\cdots,m$$

$$\lambda_i^* f_i(x^*)=0,\quad i=1,\cdots,m$$

$$\nabla f_0(x^*)+\sum_{i=1}^{m}\lambda_i^*\nabla f_i(x^*)+\sum_{i=1}^{p}\nu_i^*\nabla h_i(x^*)=0$$

线性可分支持向量机

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t.\ \ y_i(w^T x_i + b) \geq 1$$

- 对偶算法
  - 拉格朗日函数：拉格朗日乘子$\alpha = (\alpha_1, \ldots, \alpha_N)^T$

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \alpha_i y_i (w^T x_i + b) + \sum_{i=1}^{N} \alpha_i$$

  - 对偶问题：$\min\limits_{w,b} \max\limits_{\alpha} L(w, b, \alpha)$ -> $\max\limits_{\alpha} \min\limits_{w,b} L(w, b, \alpha)$

线性可分支持向量机

- 对偶算法
  - 对偶问题

$$\max_{\alpha} \min_{w,b} L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \alpha_i y_i (w^T x_i + b) + \sum_{i=1}^{N} \alpha_i$$

(1) 求 $\min_{w,b} L(w,b,\alpha)$：对w, b分别求导

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i,$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

$$L(w,b,\alpha) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j) + \sum_{i=1}^{N} \alpha_i$$

线性可分支持向量机

- 对偶算法
  - 对偶问题：$\max\limits_{\alpha} \min\limits_{w,b} L(w,b,\alpha)$

(2) 求 $\max\limits_{\alpha} \min\limits_{w,b} L(w,b,\alpha)$

$$\max_{\alpha} \ -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i^T x_j) + \sum_{i=1}^{N}\alpha_i$$

$$s.t.\ \sum_{i=1}^{N}\alpha_i y_i = 0,\ \alpha_i \geq 0, i = 1,2,\dots,N$$

  - 最优解：由KKT条件$w^* = \sum_{i=1}^{N}\alpha_i^* y_i x_i$和$\alpha_i^*(y_i(w^T x_i + b) - 1) = 0$，存在$\alpha_i^* > 0$

$$w^* = \sum_{i=1}^{N}\alpha_i^* y_i x_i, \quad b^* = y_j - \sum_{i=1}^{N}\alpha_i^* y_i (x_i^* x_j)$$

超平面法向量w^*是支持向量的线性组合

  - 决策函数：$f(x) = \text{sign}\left(\sum_{i=1}^{N}\alpha_i^* y_i (x^T x_i) + b^*\right),\ \alpha_i^* > 0$

线性可分支持向量机

- 支持向量
  - KKT互补条件$\alpha_i^*(y_i(w^T x_i + b) - 1) = 0$
  - 决策函数$f(x) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i^* y_i (x^T x_i) + b^*\right),\ \alpha_i^* > 0$
  - 对应$\alpha_i^* > 0$的样例：$y_i(w^T x_i + b) = 1$，落在间隔边界

- 示例

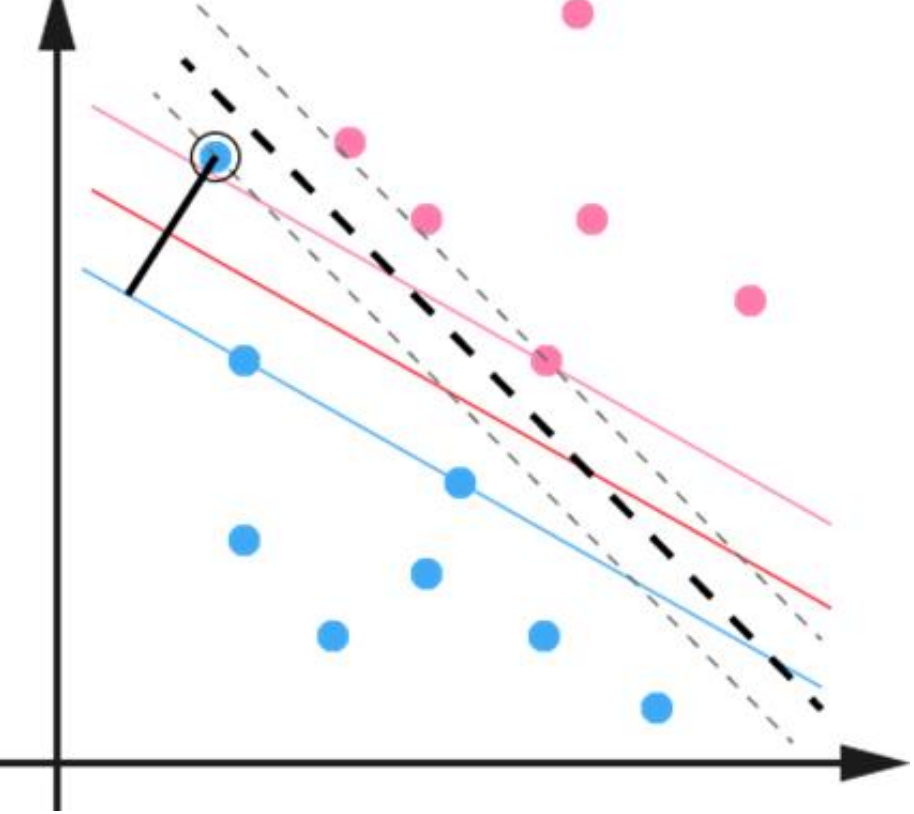
  http://www.csie.ntu.edu.tw/~cjlin/libsvm/

  http://vision.stanford.edu/teaching/cs231n-demos/linear-classify/

支持向量机

- 处理噪声和离群点
  - 求解最优分类面的时间代价大还可能导致泛化性能差。因此，对于分布有交集的数据需要有一定范围内的“错分”，又有较大分界区域的广义最优分类面。

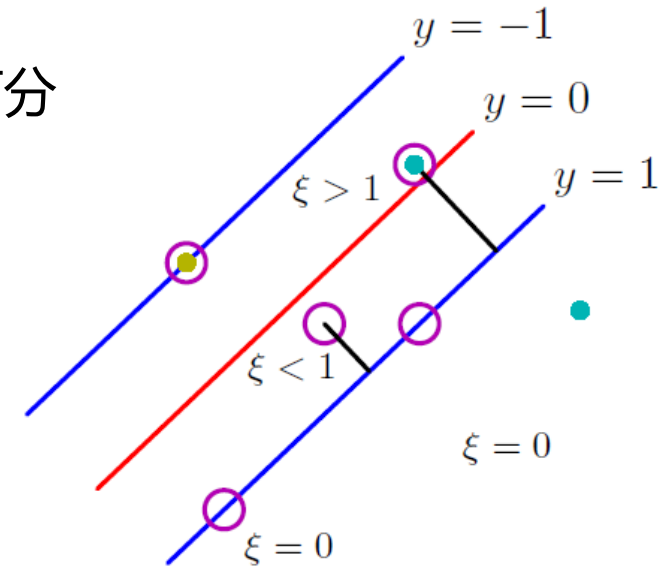
准确性



泛化性

线性支持向量机

- 线性不可分训练数据集
  - 通常存在异常点，去除后可线性可分
  - 硬间隔无法满足



- 目标函数

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i$$

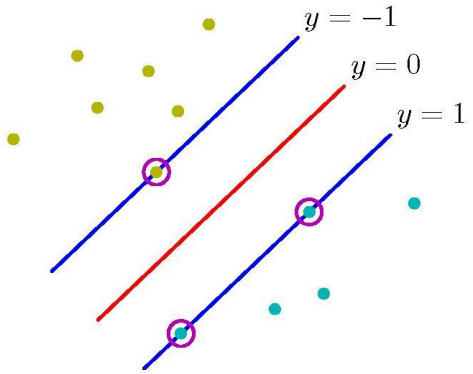
$$s.t.\ \ y_i(w^T x_i + b) \geq 1 - \xi_i$$

支持向量机

- 处理噪声和离群点
  - 这种处理方式也被视为是从硬间隔(Hard Margin)向软间隔(Soft Margin)的转变。

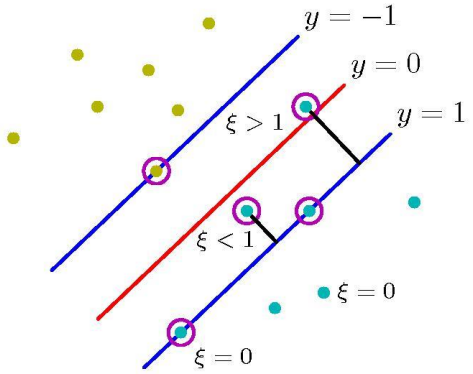
硬间隔

$$\min_{w,b} \frac{1}{\|w\|^2}$$
$$s.t.\ t_n(w^T x_n + b) \geq 1,\ n = 1, ..., N$$



软间隔

$$\min_{w,b,\xi} \frac{1}{\|w\|^2} + C\sum_{n=1}^{N} \xi_n$$
$$s.t.\ t_n(w^T x_n + b) \geq 1 - \xi_n,\ n = 1, ..., N$$
$$\xi_n \geq 0$$



线性支持向量机

- 对偶算法
  - 拉格朗日函数求$L(w,b,\xi,\alpha,\mu)$
  - 求解对偶
    - 求$\min\limits_{w,b,\xi} L(w,b,\xi,\alpha,\mu)$
    - 求$\max\limits_{\alpha} \ -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j(x_i^T x_j)+\sum_{i=1}^{N}\alpha_i$，满足$C-\alpha_i-\mu_i=0$
  - 最优解：由KKT条件$w^*=\sum_{i=1}^{N}\alpha_i^* y_i x_i$，$\mu_i^*\xi_i^*=0$，及$\alpha_i^*(y_i(w^T x_i+$

超平面法向量w^*是支持向量的线性组合

线性支持向量机

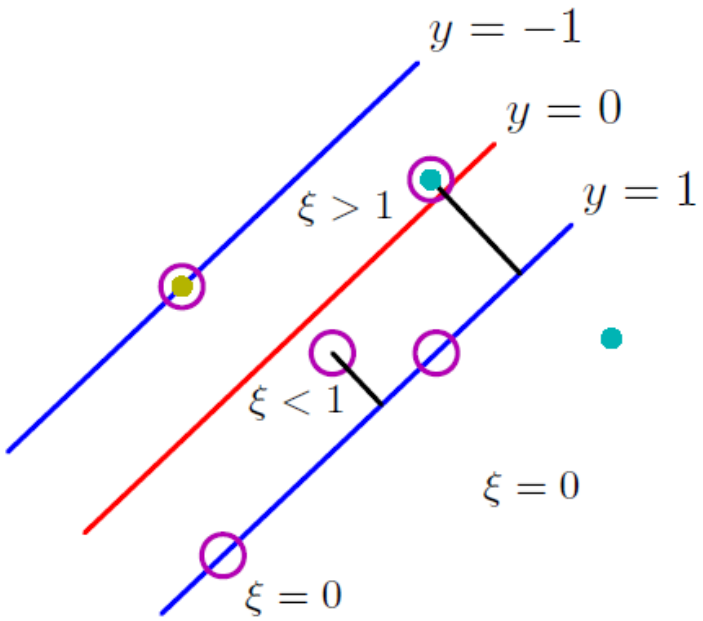
- 支持向量

$$\mu_i^*\xi_i^* = 0, \quad \alpha_i^*(y_i(w^T x_i + b) - 1 + \xi_i) = 0, \quad C - \alpha_i - \mu_i = 0$$

  - $0 < \alpha_i^* < C$: $y_i(w^T x_i + b) - 1 + \xi_i = 0$
  - $\alpha_i^* < C \rightarrow \xi_i = 0$，$x_i$落在间隔边界
  - $\alpha_i^* = C \rightarrow \xi_i < 1, \quad \xi_i = 1, \quad \xi_i > 1$

- 示例

$$C - \alpha_i - \mu_i = 0$$
$$\mu_i^*\xi_i^* = 0$$



序列最小最优化算法

- 凸二次规划对偶问题
  - 样本容量大时，直接求解效率较低
  - 序列最小最优化（SMO）算法求解

$$\max_{\alpha} \ -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i,x_j)+\sum_{i=1}^{N}\alpha_i$$

$$s.t.\ \sum_{i=1}^{N}\alpha_i y_i = 0,\ \alpha_i \geq 0, i = 1,2,\dots,N$$

- 启发式算法
  - 所有变量满足KKT条件，选择最不满足条件的两个变量，固定其他变量，迭代求解。
  - 两变量构成二次规划问题

序列最小优化算法

- J. C. Platt(1999年提出)
  - 支持向量机的学习问题可以形式化为求解具有全局最优解的凸二次规划问题。许多方法可以用于求解这一问题，但当训练样本容量很大时，这些算法往往效率较低，以致无法使用。
  - 序列最小优化算法(Sequential Minimal Optimization, SMO)是一种启发式算法。基本思想是：如果所有变量都满足此优化问题的KKT条件，那么这个问题的解就得到了。
  - SMO算法的特点是不断地将原二次规划问题分解为只有两个变量的二次规划问题，并对子问题进行解析求解，直到所有变量都满足KKT条件为止。因为子问题解析解存在，所以每次计算子问题都很快，虽然子问题次数很多，但是总体上还是高效的。

序列最小最优化算法

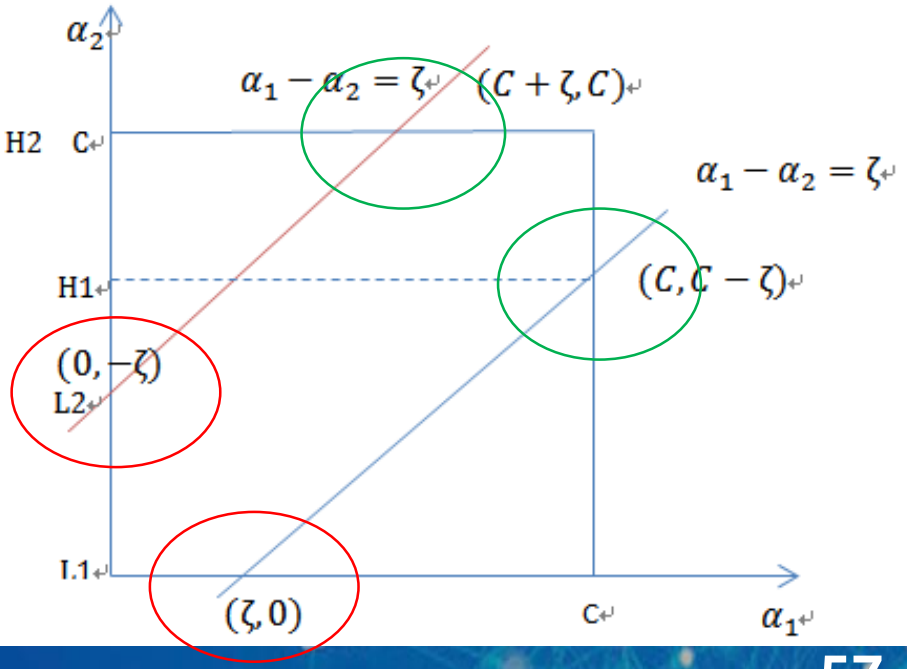
- 两变量二次规划

$$\min_{\alpha_1,\alpha_2} W(\alpha_1,\alpha_2)$$

$$s.t.\ \alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3} y_i \alpha_i,\ C \geq \alpha_i \geq 0, i = 1,2$$

  - 迭代求解：$(\alpha_1^{old}, \alpha_2^{old})$ -> $(\alpha_1^{new}, \alpha_2^{new})$
  - $\frac{\partial W}{\partial \alpha_2} = 0 \Rightarrow \alpha_2^{new} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta}$

- 剪辑α_2^{new}，变量约束范围
  - y_1, y_2异号：最小值，最大值
  - y_1, y_2同号：最大值，最小值



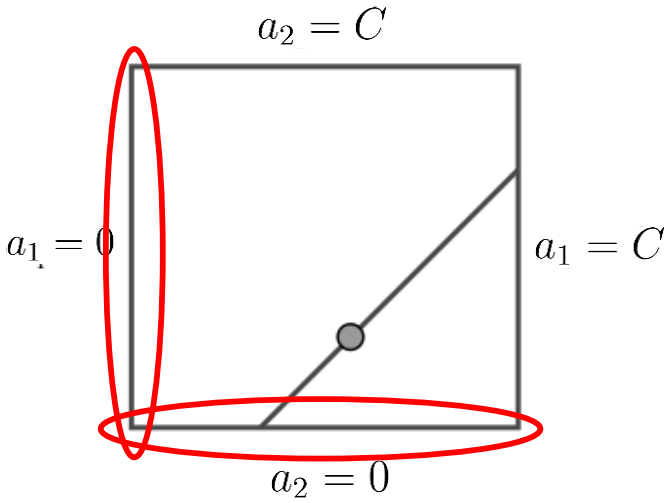
序列最小优化算法

- 两个变量二次规划的解析方法
  - 两个变量(a1, a2)的约束可用二维空间中的图形表示

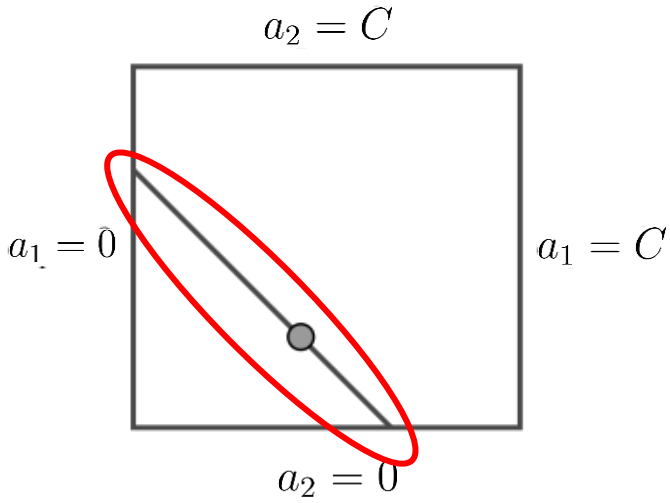
$$a_1 t_1 + a_2 t_2 = -\sum_{n=3}^{N} t_n a_n = \zeta$$

$$0 \le a_n \le C, i = 1, 2$$

目标函数在一条平行于对角线的线段上的最优值
两个变量的最优化问题转化为单变量最优化问题



$$t_1 \neq t_2 \Rightarrow a_1 - a_2 = k$$



$$t_1 = t_2 \Rightarrow a_1 + a_2 = k$$

序列最小最优化算法

- 变量选取
  - 第一个变量选择（外层循环）：违反KKT条件最严重的样本点
    - $0 < \alpha_i^* < C$：$y_i(w^T x_i + b)=1$
    - $\alpha_i^* = 0$：$y_i(w^T x_i + b) \geq 1$
    - $\alpha_i^* = C$：$y_i(w^T x_i + b) \leq 1$
  - 第二个变量选择（内层循环）：使得α_2变化足够大
    - $\alpha_2^{new} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta}$，$E_i = [\sum_{i=1}^{N} \alpha_i^* y_i \mathrm{K}(\mathrm{x}, x_i) + b^*] - \mathrm{y_i}$
    - $E_1 - E_2$足够大
  - 更新b和E_i
  - 重复上述步骤，直到KKT条件满足

序列最小优化算法

- SMO算法解凸二次规划问题

$$\min_{a} \ \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m t_n t_m k(x_n, x_m) - \sum_{n=1}^{N} a_n$$

$$s.t.\ 0 \le a_n \le C, n = 1, 2, ..., N$$

$$\sum_{n=1}^{N} a_n t_n = 0$$

  - 子问题有两个变量，一个是违反KKT条件最严重的，另一个有约束条件自动确定。两个变量中只有一个是自由变量。假设a1，a2为两个变量， a3, a4,..., aN固定，那么：

$$a_1 = -t_1 \sum_{n=2}^{N} a_n t_n = 0$$

  - 即a2确定，a1也随之确定。
  - SMO算法包括：求解两个变量二次规划的解析方法和选择变量的启发式方法。

序列最小优化算法

- 两个变量二次规划的解析方法
  - 不失一般性，假设选择的两个变量是a1和a2，其他变量ai (i=3, 4,..., N)是固定的，原问题的子问题可以写成：

$$\min_{a_1,a_2} W(a_1,a_2) = \frac{1}{2}K_{11}a_1^2 + \frac{1}{2}K_{22}a_2^2 + t_1t_2K_{12}a_1a_2$$

$$-(a_1+a_2) + t_1a_1\sum_{n=3}^{N} t_na_nK_{n1} + t_2a_2\sum_{n=3}^{N} t_na_nK_{n2} + const$$

$$s.t.\ a_1t_1 + a_2t_2 = -\sum_{n=3}^{N} t_na_n = \zeta$$

$$0 \le a_n \le C, i = 1,2$$

  - 其中，$K_{mn} = K(x_m, x_n), m, n = 1, 2, ...N$ ，ζ是常数。

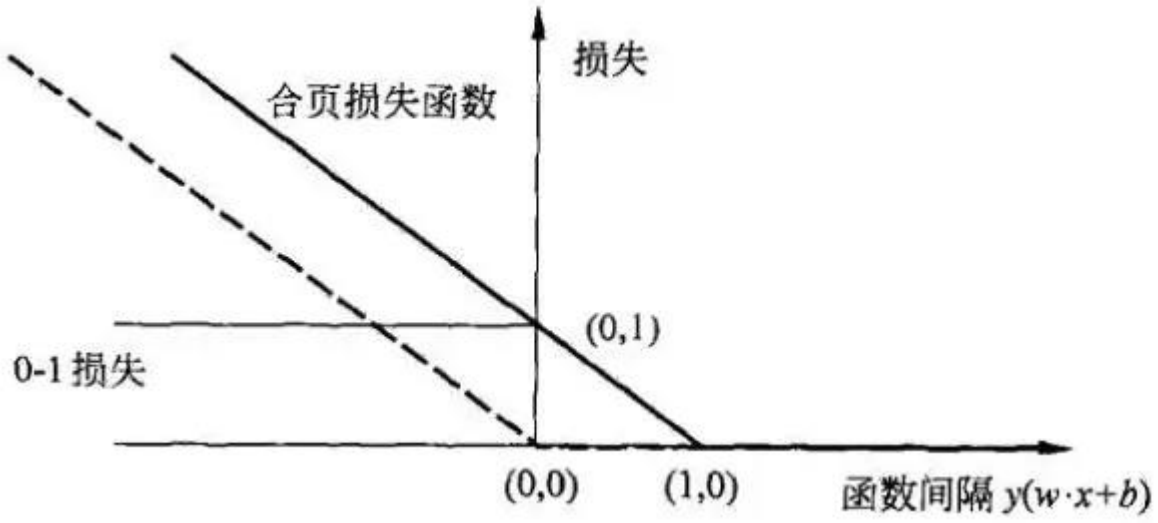
合页损失函数

- 另外一种解释

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad s.t. \quad y_i(w^T x_i + b) \geq 1$$

$$\Rightarrow \quad \min_{w,b} \sum_{i=1}^{N} [1 - y_i(w^T x_i + b)]_+ + \lambda\|w\|^2$$

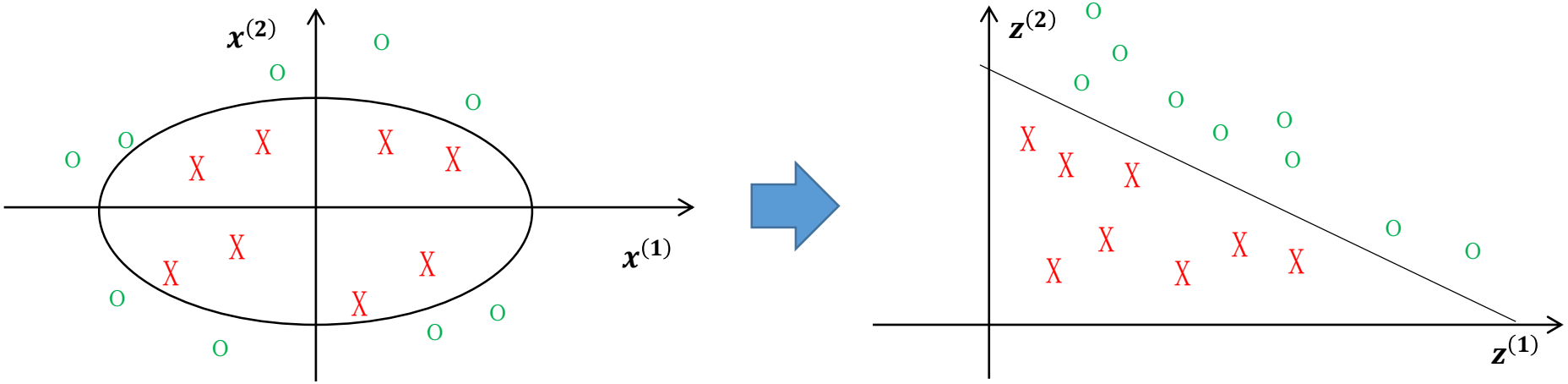
$$[z]_+ = \begin{cases} z, z > 0 \\ 0, z \leq 0 \end{cases}$$



核方法

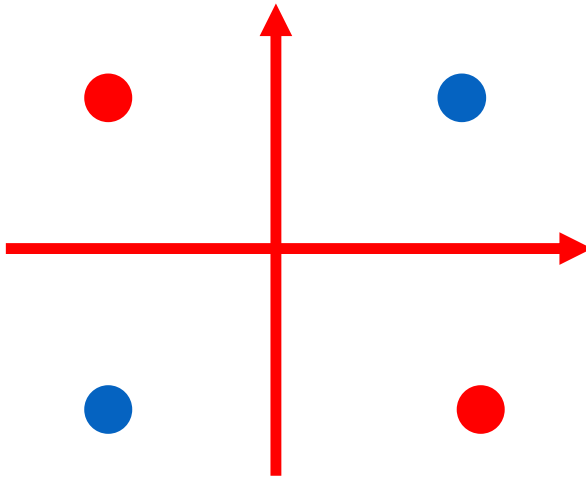
- 核技巧
  - 线性不可分，可采取非线性映射
  - 例子：线性不可分$(x^{(1)}, x^{(2)})$分类平面$w_1(x^{(1)})^2 + w_2(x^{(2)})^2 + b = 0$
  - 非线性映射$z^{(1)} = (x^{(1)})^2$，$z^{(2)} = (x^{(2)})^2$，线性可分

$$w_1 z^{(1)} + w_2 z^{(2)} + b = 0$$



支持向量机

- XOR问题
  - 二维样本集 $x = (x_1, x_2)$
  - 第一类(0, 0)和 (1, 1)，第二类(1, 0)和 (0, 1)



将二维数据映射到三维

- 映射函数

$$\phi(x) = (x_1, x_2, x_1 x_2)$$

$$(0,0) \to (0,0,0)$$
$$(1,1) \mapsto (1,1,1)$$
$$(0,1) \mapsto (0,1,0)$$
$$(1,0) \to (1,0,0)$$

线性不可分　线性可分

核方法

- 定义：
  - 映射$\phi(x)$：输入空间 → 特征空间（希尔伯特空间）
  - 核函数$\mathrm{K}(\mathrm{x},\mathrm{z}) = \phi(x)^T\phi(z)$
- 思想
  - 显式定义核函数，而不显式定义映射函数
  - 直接计算$\mathrm{K}(\mathrm{x},\mathrm{z})$容易，$\phi(x)$通常无穷维，难以定义
- 支持向量机
  - 决策函数$f(x) = sign(\sum_{i=1}^{N} \alpha_i^* y_i (x^T x_i) + b^*)$
  - 内积$x^T x_i$由核函数代替$\mathrm{K}(\mathrm{x}, x_i)$，实际上进行了特征映射
  - 决策函数$f(x) = sign(\sum_{i=1}^{N} \alpha_i^* y_i \mathrm{K}(\mathrm{x}, x_i) + b^*)$

核方法

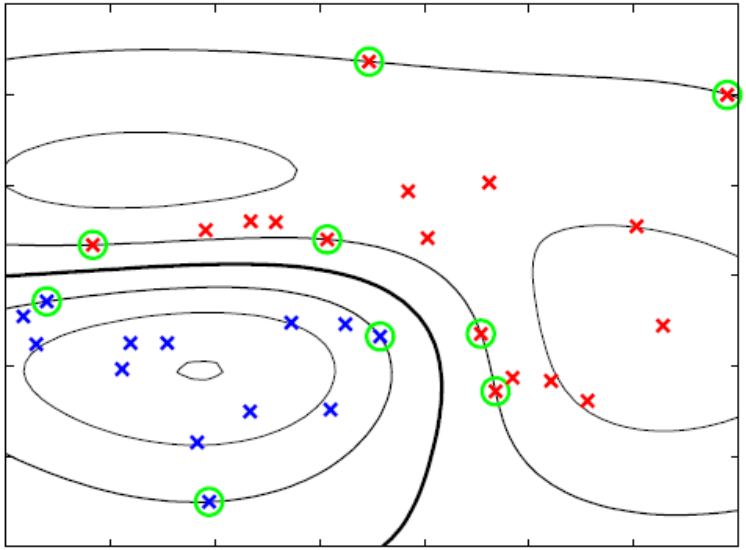
- 正定核
  - 核函数$\mathrm{K}(\mathrm{x}, z)$应该满足什么性质？→ 构造或判断
  - Gram矩阵：一组向量的内积构成矩阵的元素
  - 正定：对任意的向量v，$\mathrm{v}^{\mathrm{T}}\mathrm{Kv} \geq 0$
  - 核函数：$\mathrm{K}(\mathrm{x}, z)$正定核函数，⇔ $\mathrm{K}(\mathrm{x}, z)$对应的Gram矩阵半正定

$$\mathrm{K} = \left[K\left(x_i, x_j\right)\right]_{m \times m}$$

    - 必要性：$\mathrm{K} = \left[K\left(x_i, x_j\right)\right]_{m \times m} = \left[\phi(x_i)^T \phi\left(x_j\right)\right]_{m \times m}$ => $\mathrm{v}^{\mathrm{T}}\mathrm{Kv} = [\sum_i v_i \phi(x_i)]^2 \geq 0$
    - 充分性：Gram矩阵可定义$\phi: \mathrm{x} \rightarrow \mathrm{K}(., \mathrm{x})$ => $\mathrm{K}(\mathrm{x}, z) = \phi(x)^T \phi(z)$
  - 正定核函数判定：
    - 对称函数
    - 对任意的$\mathrm{K} = \left[K\left(x_i, x_j\right)\right]_{m \times m}$半正定

核方法

- 常用核函数
  - 多项式核函数$\mathrm{K}(\mathrm{x}, z) = (x^T z + 1)^P$
  - 高斯核函数$\mathrm{K}(\mathrm{x}, z) = exp\left(-\frac{||x-z||^2}{2\sigma^2}\right)$
  - 字符串核函数



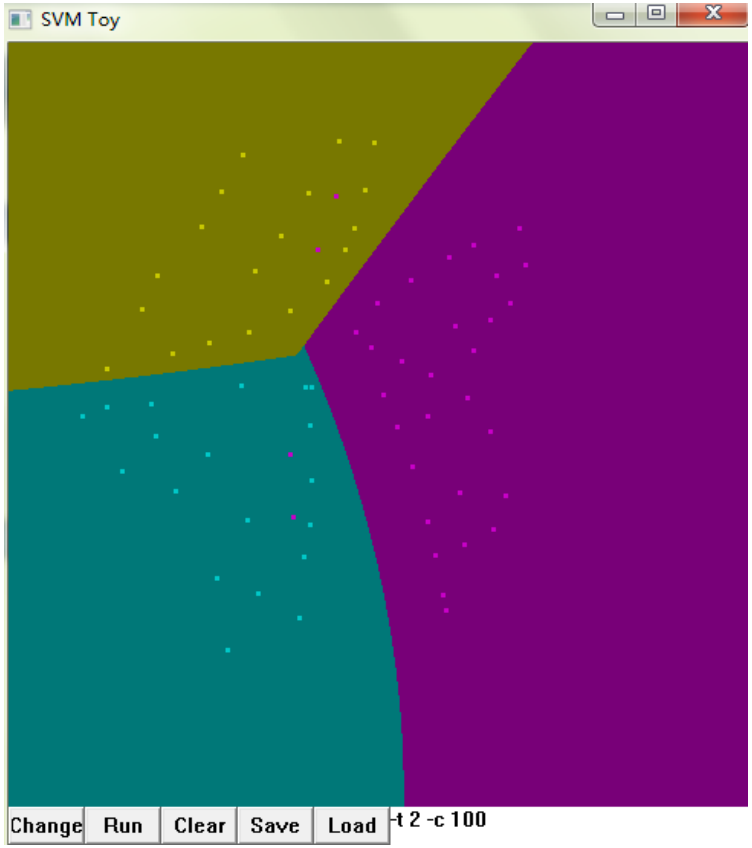
- 非线性支持向量机
  - $f(x) = \mathrm{sign}(\sum_{i=1}^{N} \alpha_i^* y_i \mathrm{K}(\mathrm{x}, x_i) + b^*)$
  - 算法

$$\max_{\alpha} \ -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i, x_j) + \sum_{i=1}^{N}\alpha_i$$

$$s.t. \ \sum_{i=1}^{N}\alpha_i y_i = 0, \ \alpha_i \geq 0, i = 1,2,\dots,N$$

支持向量机

- 支持向量机工具
  - LibSVM: http://www.csie.ntu.edu.tw/~cjlin/libsvm/



VC维

- SLT中衡量函数集性能的指标
- 为了研究经验风险最小化函数集的学习一致收敛速度和推广性，统计学力理论定义了一些指标来衡量函数集的性能，其中最重要的就是VC维(Vapnik-Chervonenkis Dimension)

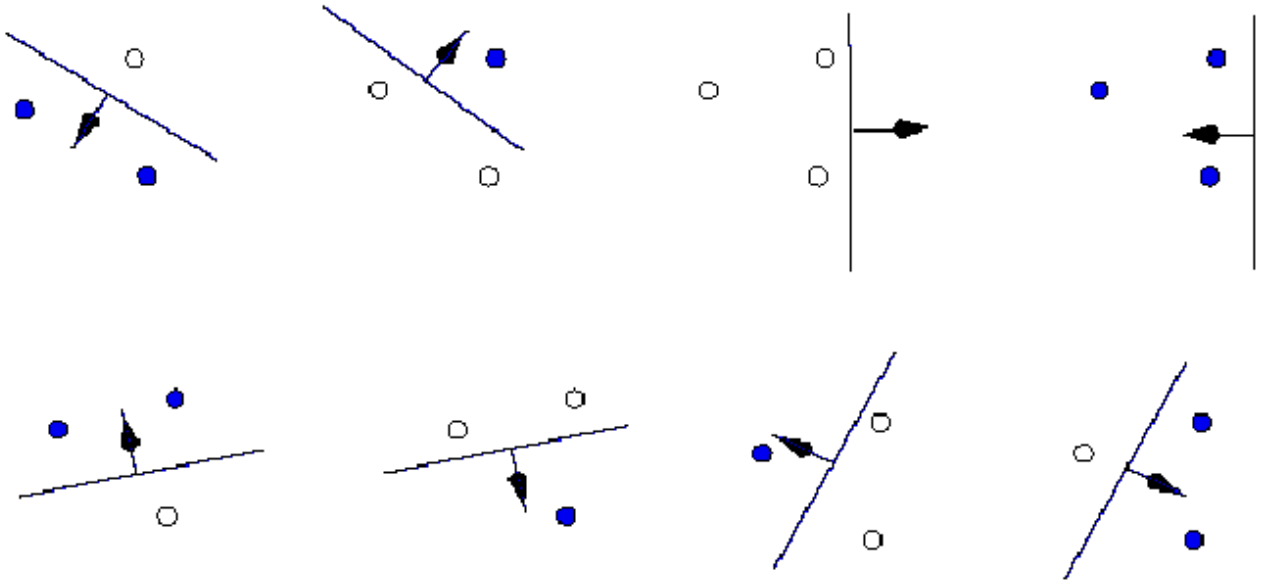
VC维定义：

对于一个指示函数(即只有0和1两种取值的函数)集，如果存在h个样本能够被函数集里的函数按照所有可能的2^h种形式分开，则称函数集能够把h个样本打散，函数集的VC维就是能够打散的最大样本数目。

VC维

- SLT中衡量函数集性能的指标
  - 对于2维空间的线性分类面函数
  - 其能够以任意方式打散(分类)的最大样本数为3。

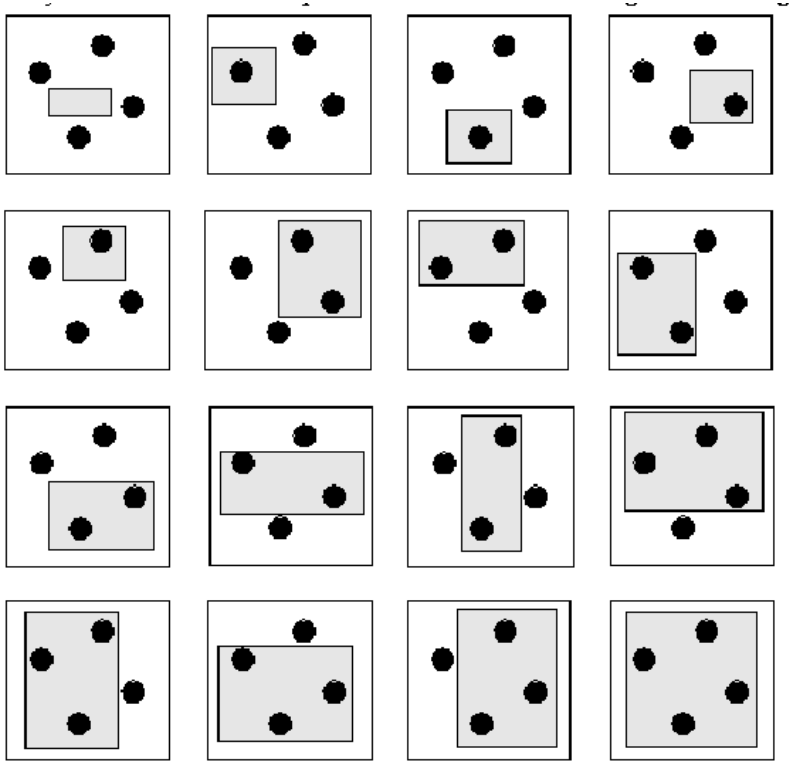
$$y = w_0 + w_1 x_1 + w_2 x_2$$



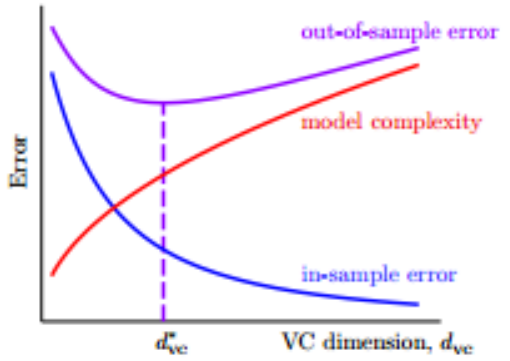
VC维

- SLT中衡量函数集性能的指标
  - 一般来说,对于n维空间的线性分类面函数
  - 的VC维为n+1。

$$y = w_0 + w_1 x_1 + \ldots + w_n x_n$$



VC维



- $d_{vc} \uparrow$: $E_{in} \downarrow$ but $\Omega \uparrow$
- $d_{vc} \downarrow$: $\Omega \downarrow$ but $E_{in} \uparrow$
- best d_{vc}^* in the middle

powerful $\mathcal{H}$ not always good!

$$\mathbb{P}_{\mathcal{D}}\left[|E_{in}(g) - E_{out}(g)| > \epsilon\right] \quad \leq \quad 4(2N)^{d_{vc}} \exp\left(-\tfrac{1}{8}\epsilon^2 N\right)$$

theory: $N \approx 10,000 d_{vc}$; practice: $N \approx 10 d_{vc}$

given specs $\epsilon = 0.1$, $\delta = 0.1$, $d_{vc} = 3$, want $4(2N)^{d_{vc}} \exp\left(-\tfrac{1}{8}\epsilon^2 N\right) \leq \delta$

N	bound
100	2.82×10^7
1,000	9.17×10^9
10,000	1.19×10^8
100,000	1.65×10^{-38}
29,300	9.99×10^{-2}

sample complexity:
need $N \approx 10,000 d_{vc}$ in theory

VC维

- SLT中衡量函数集性能的指标
  - 一般而言，VC维反映了函数集的学习能力，VC维越大则学习机器越复杂，学习内容量就越大。
  - 目前没有通用的关于任意函数集VC维计算的理论，只对一些特殊的函数集知道其函数维。
  - 在n维实数空间中线性分类器和线性实函数的VC维是n+1
  - 而f(x, a)=sin(ax)的VC维则为无穷大
  - 如何用理论和试验的方法计算其VC维是当前SLT中一个待研究的问题。

软件开发环境国家重点实验室
State Key Laboratory of Software Development Environment

本节课结束