集异壁之大成：支持向量机

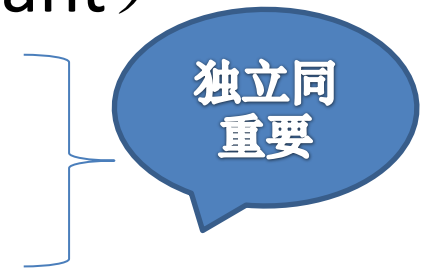
Nothing is more practical than a good theory——Vapnik

胡光能 hugn@nlp.nju.edu.cn

引子：统计学习理论与支持向量机

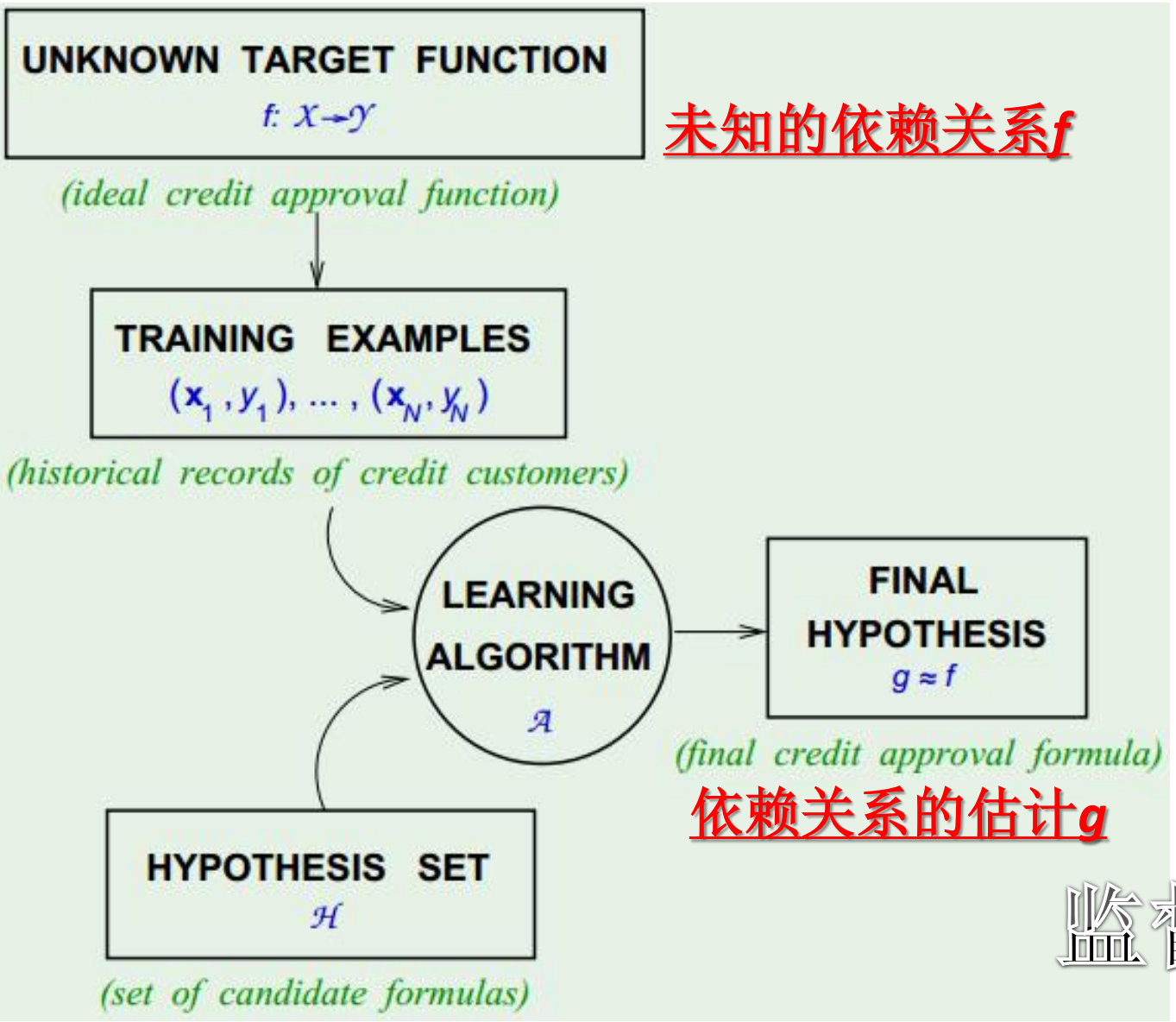
- *Theory & Practical*
- Vapnik等人提出的一种小样本统计学。认为在*观测*的基础上估计**依赖关系**只要知道未知依赖关系所属的*函数集*的一般性质（如VC维）；依赖关系的估计就是基于经验数据的函数估计

- *“Nothing is more practical than a good theory”*
- 科学理论的三要素：（Kant）
  - 问题的表示（问题本质）
  - 问题的解决（基本原理）
  - 证明（模型成为理论）



目录

信用评级



未知的依赖关系**f**

依赖关系的估计**g**

问题：

根据信用评级的历史数据*预测*一个用户是否是信用良好的

条件：

- 数据集
- 假设集
- 学习算法

监督学习！

从数据中学习：哲学

- 归纳推理：经验科学从**单称（特殊）**陈述例如观察和实验结果过渡到**全称（一般）**陈述例如假说或理论的推理叫做归纳推理。
- Popper，1959年《科学发现的逻辑》，书中道：科学与非科学的划界不是证实而是**证伪**；除非有**先验**知识否则纯归纳是不可能的

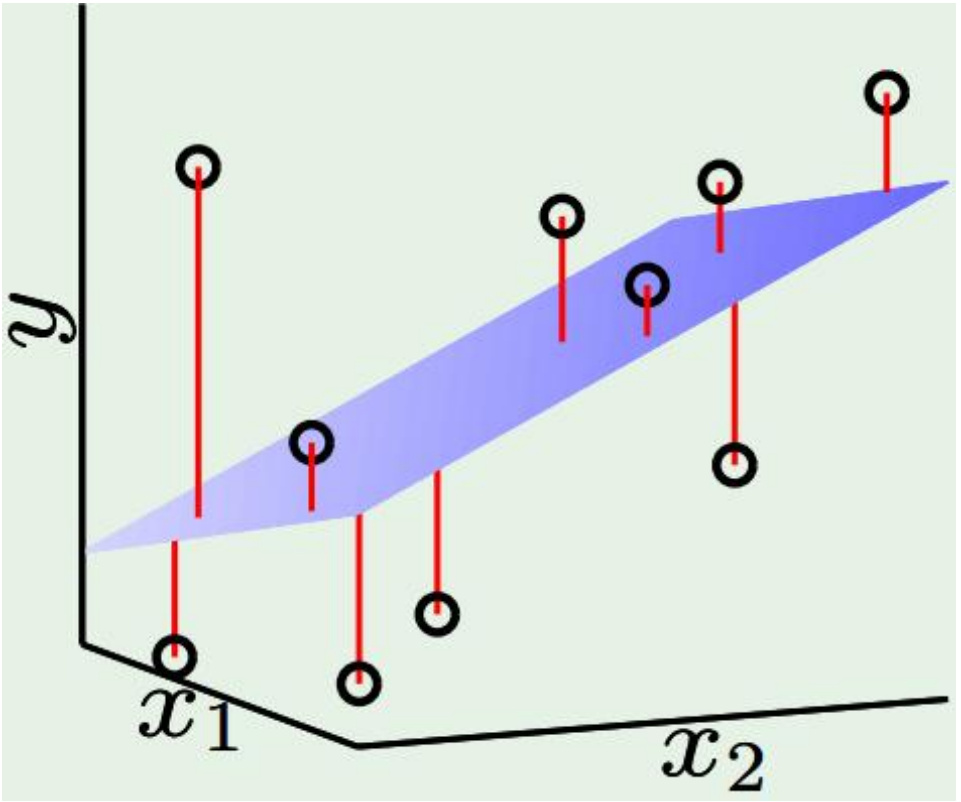
- Vapnik运用可证伪理论证明ERM归纳原则的一致性
- Mitchell运用归纳偏置使归纳推理充分地由演绎推理来论证

科学哲学指导统计学习理论、机器学习

机器学习，Mitchell，2.7节 归纳偏置

统计学——最小二乘回归

- **回归**问题比分类问题古老：Gauss，1809年《天体运动论》为计算谷神星轨道问题提出了*LS*线性插值



历史注记：

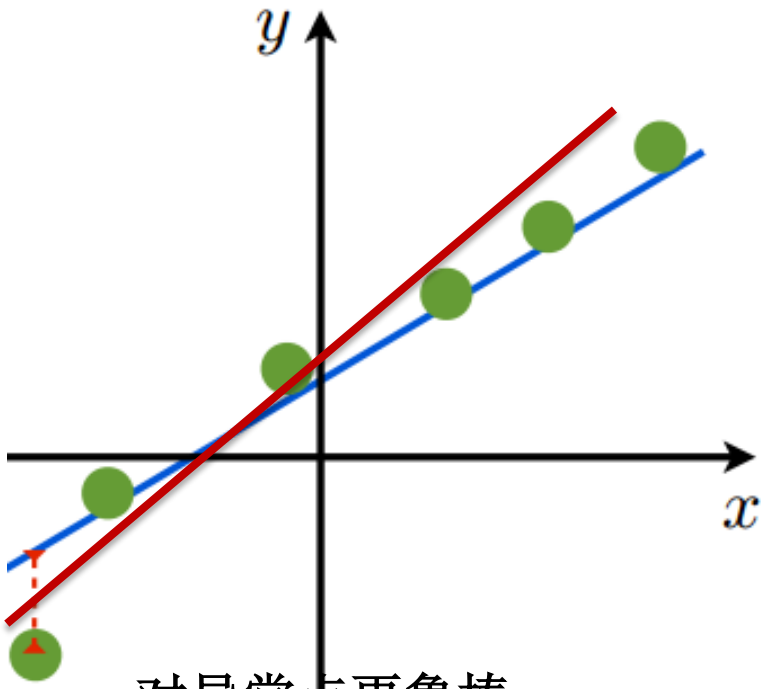
- 勒让德1806年独立发现最小二乘法
- 牛顿，莱布尼茨：微积分

$$\min L(w,b) = \sum_i (y_i - (w \cdot x_i + b))^2$$

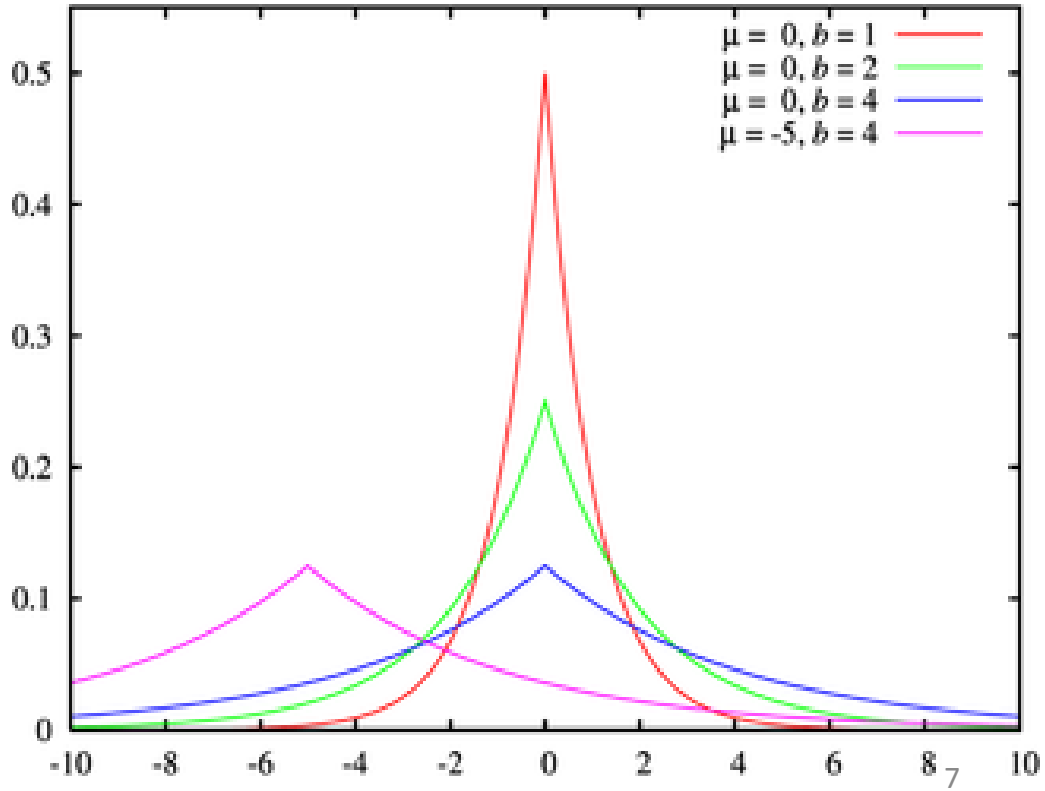
最小绝对偏差回归

- 噪音服从Laplace分布时用*LAD*回归

$$\min L(w,b) = \sum_i | y_i - (w \cdot x_i + b) |$$



对异常点更鲁棒



高斯-马尔科夫定理

- “在所有的线性**无偏**估计中，参数 β 的最小二乘估计有最小方差”
- 然而存在有偏估计能换取到更小的方差
- 偏差-方差分解：

$$MSE(\hat{\theta}) = E(\hat{\theta} - \theta)^2$$

$$= Var(\hat{\theta}) + [E\hat{\theta} - \theta]^2$$

上述论述取自：*统计学习基础：数据挖掘、推理与预测*，Hastie等著，范明等译，电子工业出版社

统计学——分类

- Fisher在30年代，提出了用于**分类**的线性判别分析

$$J(w)=\frac{(\mu_w^+-\mu_w^-)^2}{(\sigma_w^+)^2+(\sigma_w^-)^2}$$

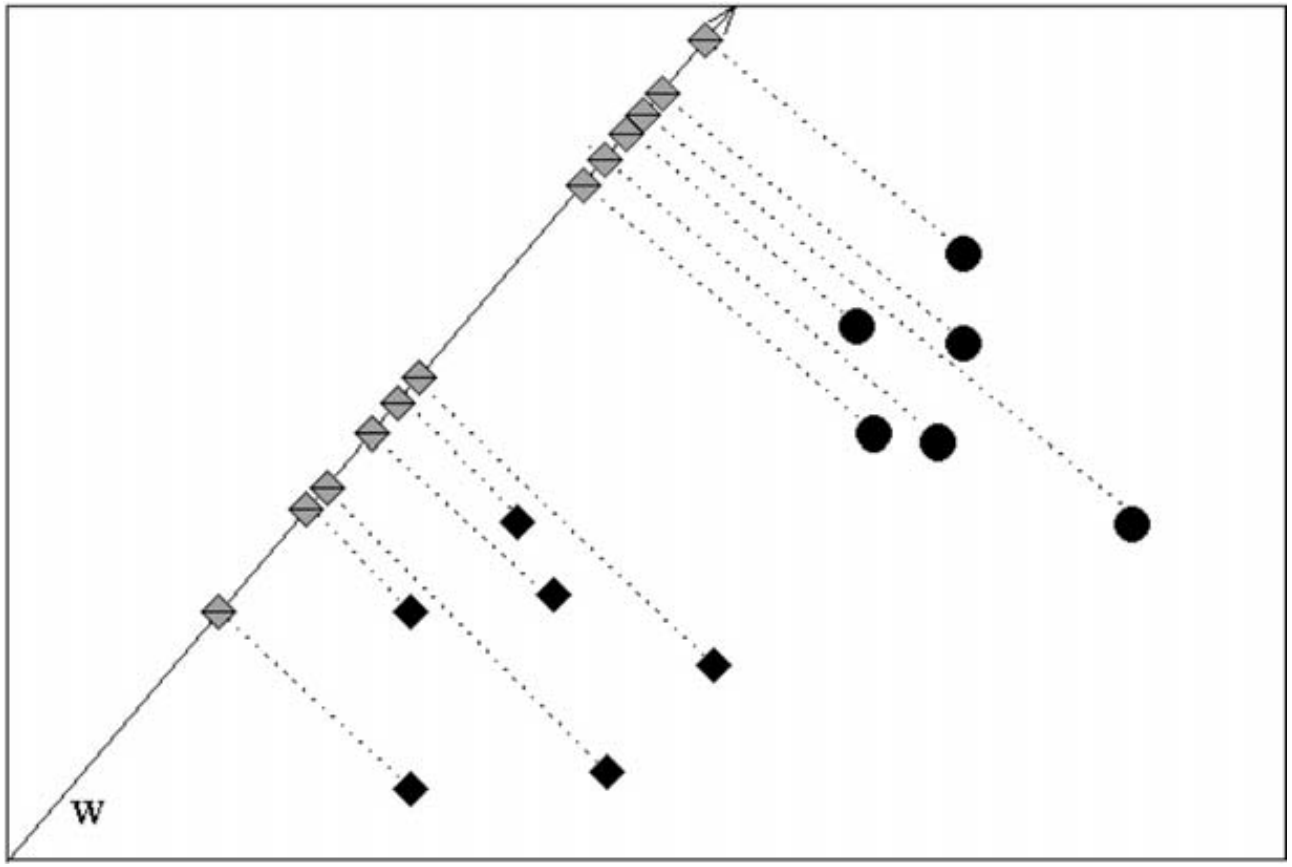
最大化上式：

分子最大化

—类间隔得远

分母最小化

—类内要紧凑

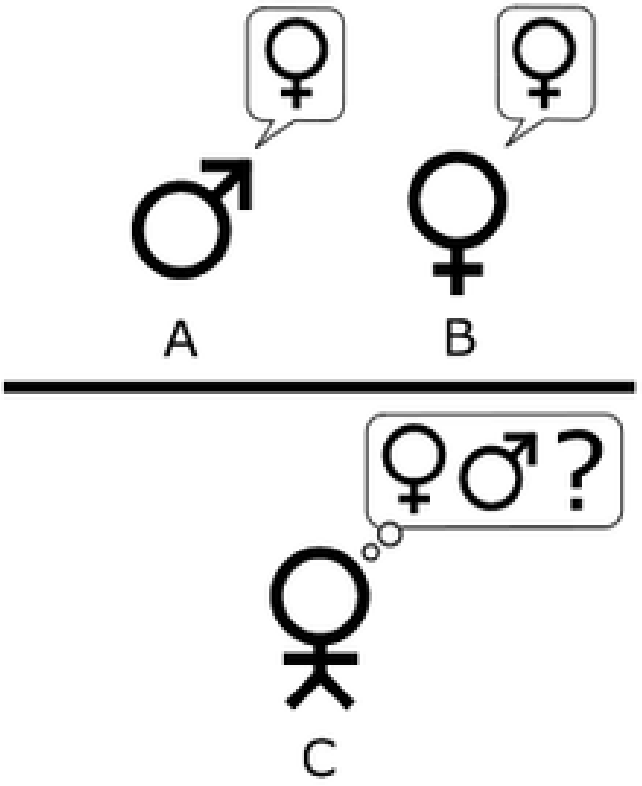


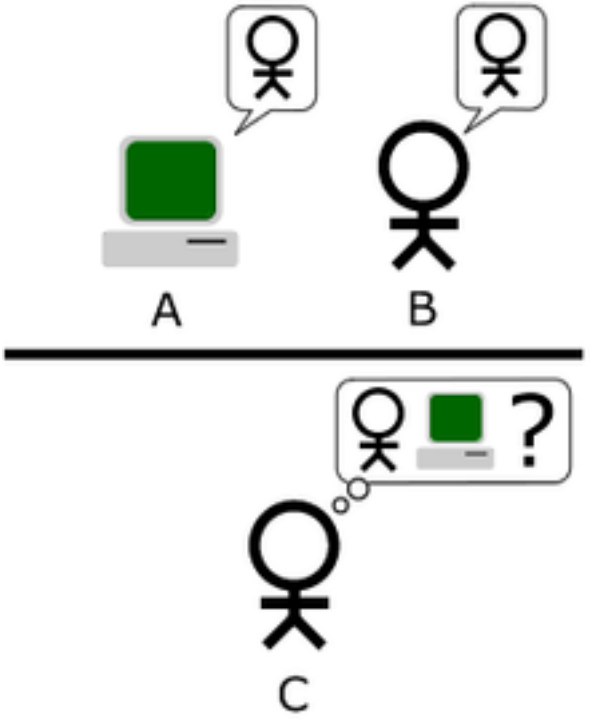
统计学习基本问题

- *“对于一种未知的依赖关系，为了在观测的基础上对它进行估计，我们必须先验地知道些什么”*
- Fisher理论的回答：除了有限个**参数**的取值外，必须知道其它的几乎所有信息(尤其是样本数要足够大)；依赖关系的估计就是基于经验数据的对这些未知参数的估计
- Vapnik理论的回答：只要知道未知依赖关系所属的函数集（**泛函**空间/假设空间）的一般性质（如VC维）；依赖关系的估计就是基于经验数据的函数估计

人工智能——图灵

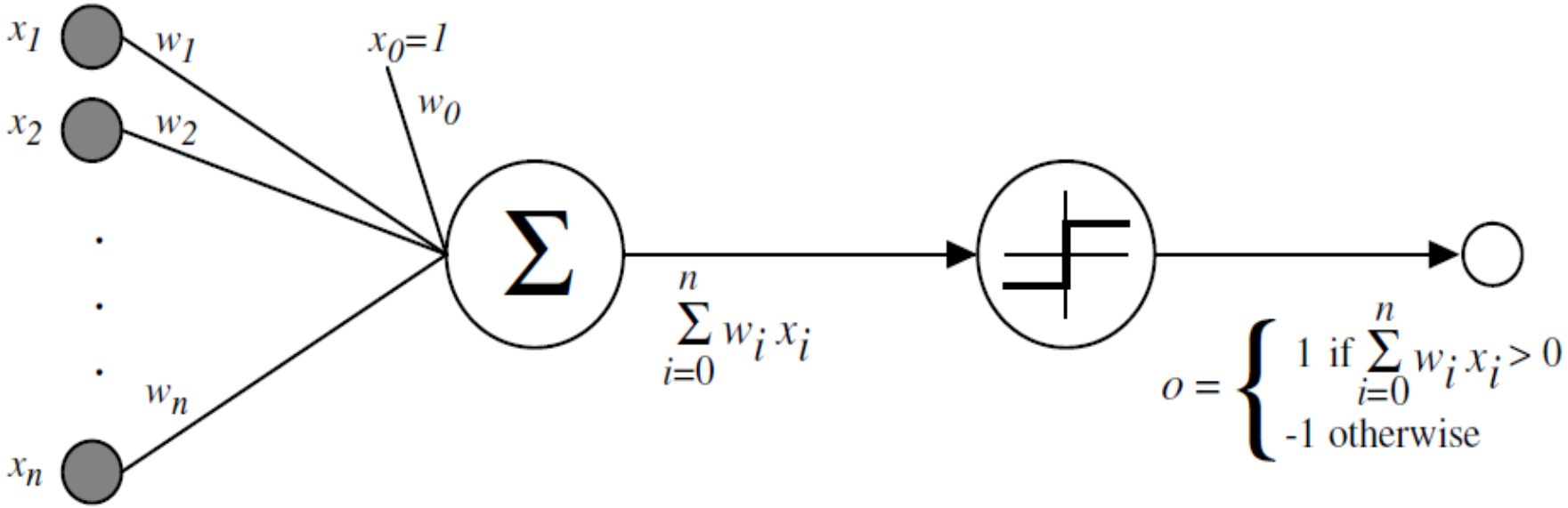
- Turing 《计算机器与智能》1950提出机器能够学习的思想以及判断机器是否智能的“模仿游戏”





Rosenblatt

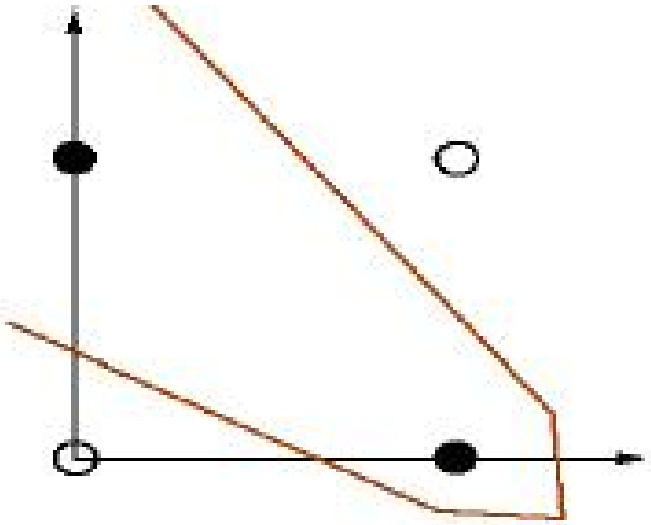
- Rosenblatt 《感知器：用于大脑中信息存储和组织的概率模型》1959，提出二类分类的线性学习器

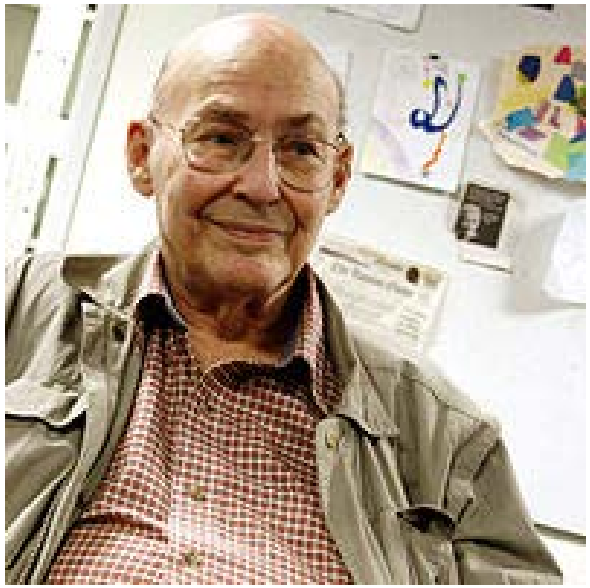


$$o(x_1, \ldots, x_n) = \begin{cases} 1 & \text{if } w_0 + w_1 x_1 + \cdots + w_n x_n > 0 \\ -1 & \text{otherwise.} \end{cases}$$

Minsky

- Minsky等《感知机》1969，分析了感知器的局限：仅有线性学习能力





Marvin Minsky（2008）
Turing Award 1969

图片取自：http://en.wikipedia.org/wiki/Marvin_Minsky

从数据中学习：人工智能

- 把**学习**问题建模成适当假设空间中的**搜索**问题是人工智能的学习方法特点
- 对于学习的称谓
  - 哲学用归纳推理
  - 统计学用估计、逼近和**优化**
  - 人工智能用搜索

- 同一概念的不同称谓我们随处可见

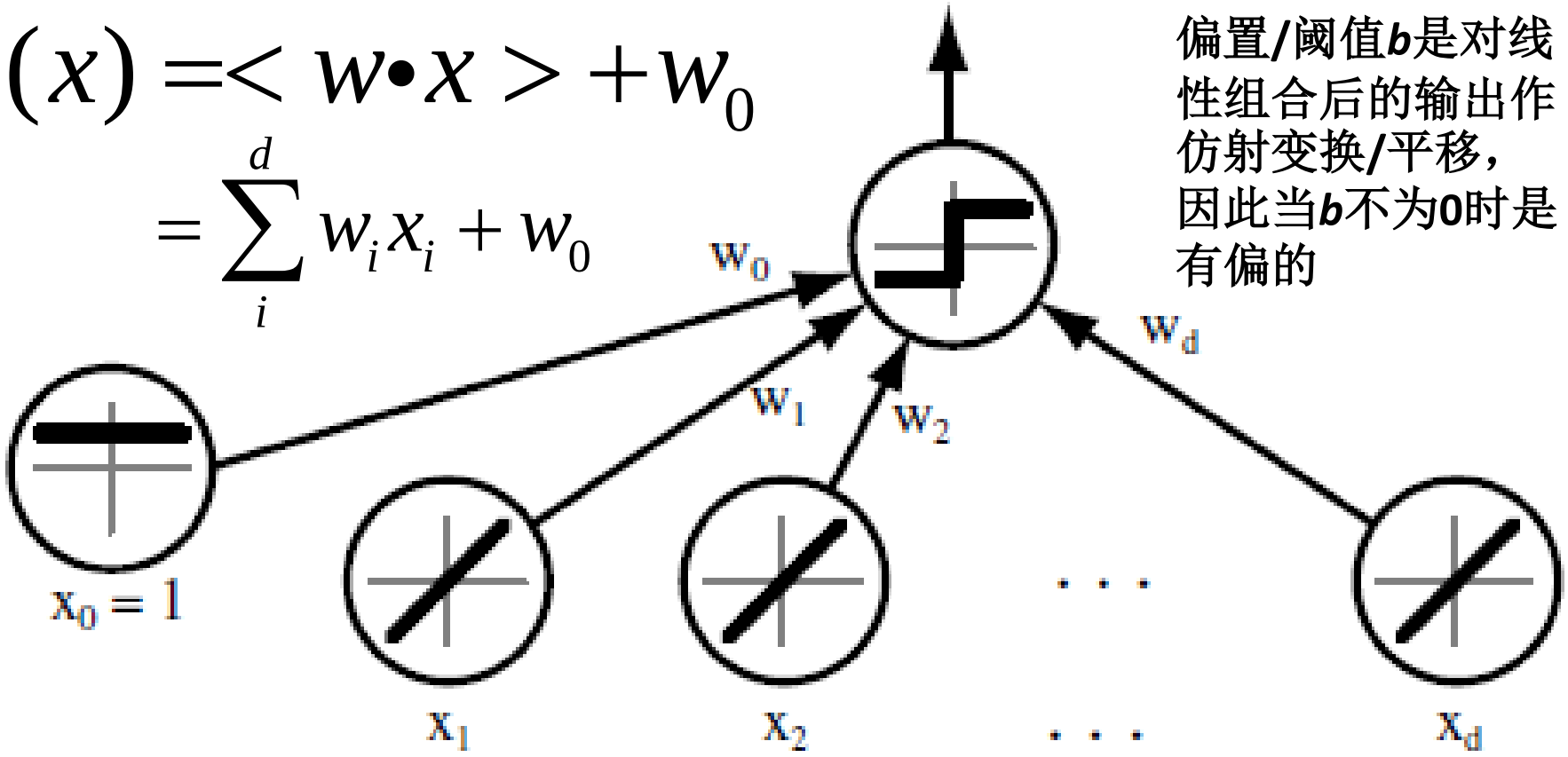
目录

线性学习器

- 线性学习器指假设函数集是线性函数的学习器

$$f(x) = < w \bullet x > + w_0$$

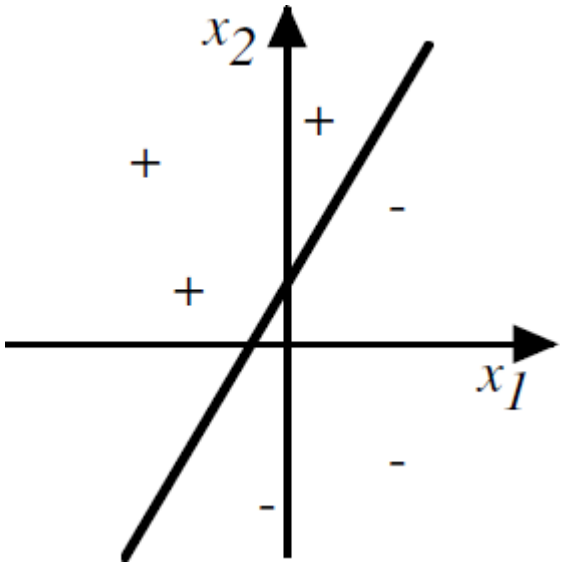
$$= \sum_{i}^{d} w_i x_i + w_0$$

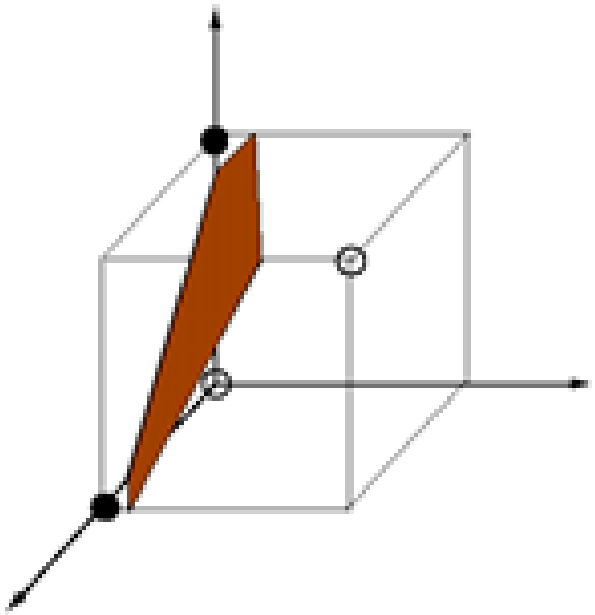


偏置/阈值*b*是对线性组合后的输出作仿射变换/平移，因此当*b*不为0时是有偏的

线性学习器的决策边界

- 决策边界：超平面





感知器的训练：原始形式

for i=1 to *l*

$$\min_{w,b} L(w,b) = -\sum_{x_i \in M} y_i (w \cdot x_i + b)$$

if $y_i(< w_k \bullet x_i > + b_k) \le 0$

$$\begin{cases} w_{k+1} \leftarrow w_k + \eta y_i x_i \\ b_{k+1} \leftarrow b_k + \eta y_i \end{cases} \quad k \leftarrow k+1$$

原始形式
——在线、错误驱动
——随机、梯度下降

$$\nabla_b L(w,b) = -\sum_{x_i \in M} y_i \qquad \nabla_w L(w,b) = -\sum_{x_i \in M} y_i x_i$$

- 收敛的条件：
  - 数据线性可分
  - 学习率充分小

$k \le \left(\frac{R}{\gamma}\right)^2$（Novikoff,1962）

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

收敛的证明可参阅：统计学习方法，李航，p31-33

感知器训练：对偶形式

for i=1 to *l*

if $y_i\left(\sum_{j=1}^{l}\alpha_j y_j < x_j \bullet x_i > +b\right) \le 0$

$$\begin{cases} \alpha_i \leftarrow \alpha_i + 1 \\ b \leftarrow b + \eta y_i \end{cases}$$

对偶形式：将权重向量展成输入和输出的线性组合

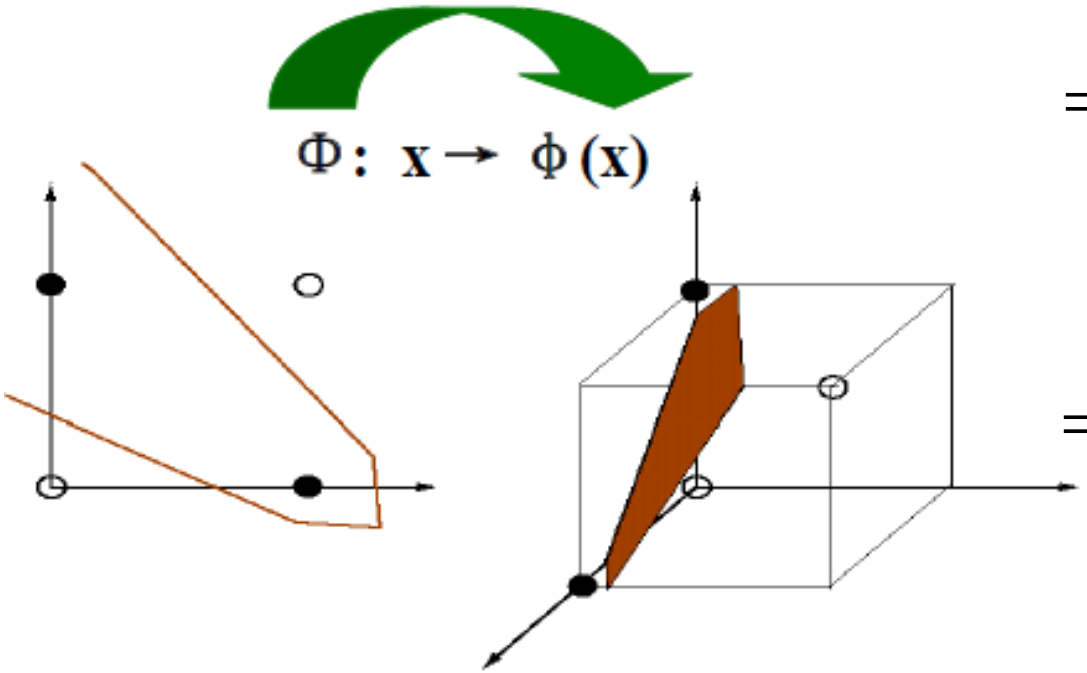
$$w = \sum_{i=1}^{l} \alpha_i y_i x_i$$

- 对偶表示中，训练数据*以内积形式成对出现*在决策函数中，这个特点为**核函数**隐式定义特征空间进而实现非线性变换提供可能
- 对偶表示与**优化理论**自然包容

线性学习器：学习线性不可分

- 如果数据线性不可分但是又只有线性学习器，怎么办？

$$f(x) = \sum_{j=1}^{l} \alpha_j y_j < x_j \bullet x > + b$$



$$= \sum_{i=1}^{l} \alpha_i y_i < \phi(x_i) \bullet \phi(x) > + b$$

$$\phi(x) \bullet \phi(z) = K(x, z)$$

$$= \sum_{i=1}^{l} \alpha_i y_i K(x_i, x) + b$$

目录

- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- **苦其心志：核函数**
- 劳其筋骨：泛化理论
- 饿其体肤：优化理论
- 降大任于是人：支持向量机
- 降妖伏魔：算法实现
- 大显身手：实际应用

核函数：定义

- 核K是一个函数，满足对于输入空间中任意点对x,z有
  $\phi(x)\bullet\phi(z)=K(x,z)$ 其中ϕ 是输入空间到特征空间的映射
- 核函数也叫正定核函数、正定核、内积核和Mercer核

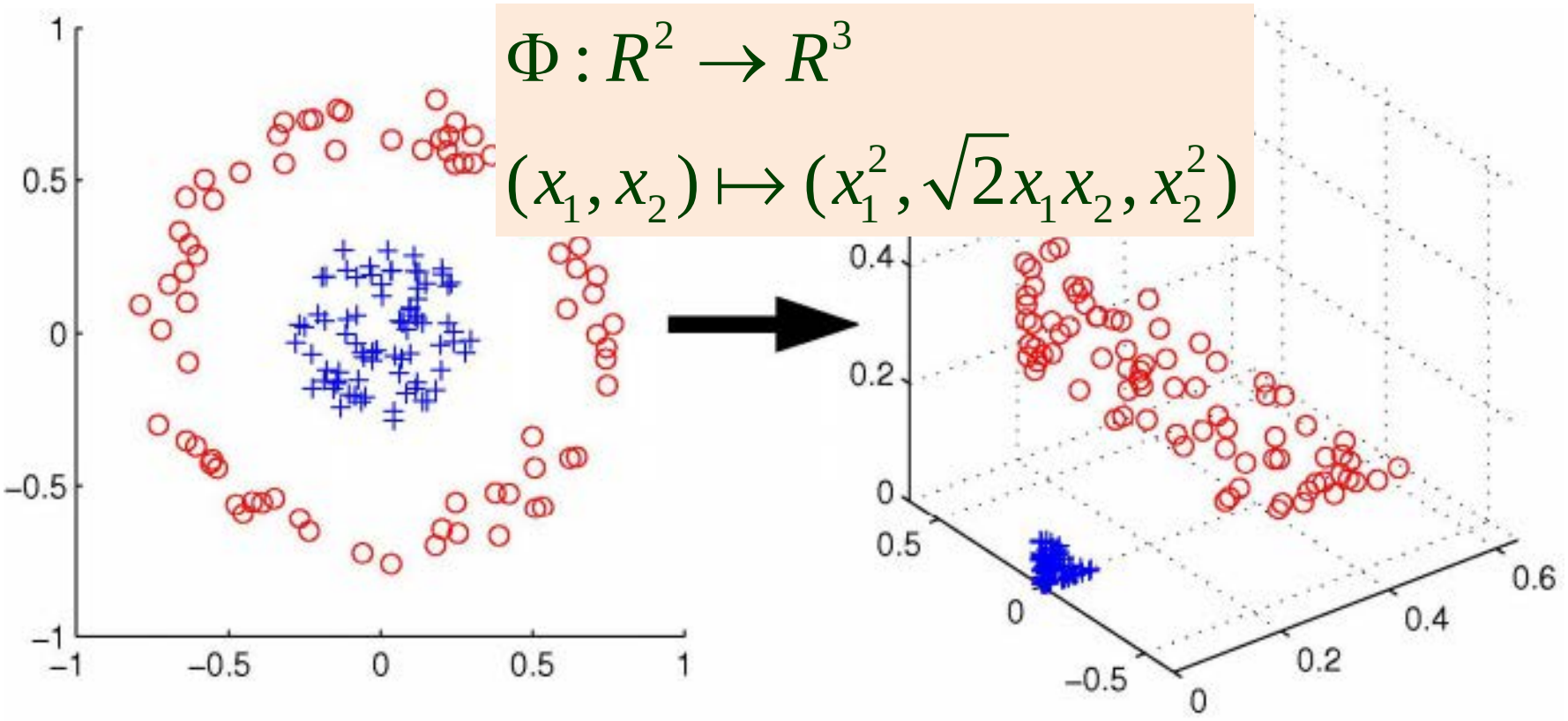
$$f(x)=\sum_{j=1}^{l}\alpha_j y_j <x_j\bullet x>+b$$

$$=\sum_{i=1}^{l}\alpha_i y_i <\phi(x_i)\bullet\phi(x)>+b$$

$$=\sum_{i=1}^{l}\alpha_i y_i K(x_i,x)+b$$

$$\overset{e.g.}{=}\sum_{i=1}^{l}\alpha_i y_i (x_i\bullet x)^2+b$$

- 即使特征空间是无限维的，核技巧也只包括有限项
- 当映射ϕ是非线性变换时也就等价于学习原空间的非线性函数



$$\Phi : R^2 \to R^3$$

$$(x_1, x_2) \mapsto (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

设三维点对r和s对应的二维点对是a和b：

$$r = \Phi(a)$$

$$s = \Phi(b)$$

$$< r, s > = r_1s_1 + r_2s_2 + r_3s_3$$

$$= a_1^2b_1^2 + 2a_1a_2b_1b_2 + a_2^2b_2^2$$

$$=< a, b >^2 = K(a,b) =< \Phi(a), \Phi(b) >$$

多项式核

核函数： Mercer定理

- Mercer定理（Mercer,1908）：

  “一个对称的半正定函数是一个核”

- 一个对称的半正定函数有一个对称的半正定Gram矩阵：

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

$K(\mathbf{x}_1,\mathbf{x}_1)$	$K(\mathbf{x}_1,\mathbf{x}_2)$	$K(\mathbf{x}_1,\mathbf{x}_3)$	…	$K(\mathbf{x}_1,\mathbf{x}_N)$
$K(\mathbf{x}_2,\mathbf{x}_1)$	$K(\mathbf{x}_2,\mathbf{x}_2)$	$K(\mathbf{x}_2,\mathbf{x}_3)$		$K(\mathbf{x}_2,\mathbf{x}_N)$
…	…	…	…	…
$K(\mathbf{x}_N,\mathbf{x}_1)$	$K(\mathbf{x}_N,\mathbf{x}_2)$	$K(\mathbf{x}_N,\mathbf{x}_3)$	…	$K(\mathbf{x}_N,\mathbf{x}_N)$

- 正定核K：$K(x,z)=\Phi(x)\bullet\Phi(z)$
- Gram矩阵：$K=\left[K(x_i,x_j)\right]_{m\times m}$
- 正定核⇔Gram矩阵是半正定的

必要性 $\alpha=(\alpha_1,\ldots,\alpha_m)^T$

$$\alpha^T(K(x_i,x_j))_{m\times m}\alpha$$

$$=\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j K(x_i,x_j)$$

$$=(\sum_{i=1}^{m}\alpha_i\Phi(x_i))\bullet(\sum_{j=1}^{m}\alpha_j\Phi(x_j))$$

$$=\|\sum_{i=1}^{m}\alpha_i\Phi(x_i)\|^2\geq 0$$

充分性 $\Phi: x\to K(\bullet,x)$

$$f(\bullet)=\sum_{i=1}^{m}\alpha_i K(\bullet,x_i)\quad g(\bullet)=\sum_{j=1}^{m}\beta_i K(\bullet,x_j)$$

$$f*g=\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\beta_j K(x_i,x_j)$$

上述*运算就是一个内积算子

$$K(\bullet,x)\bullet f=f(x)$$

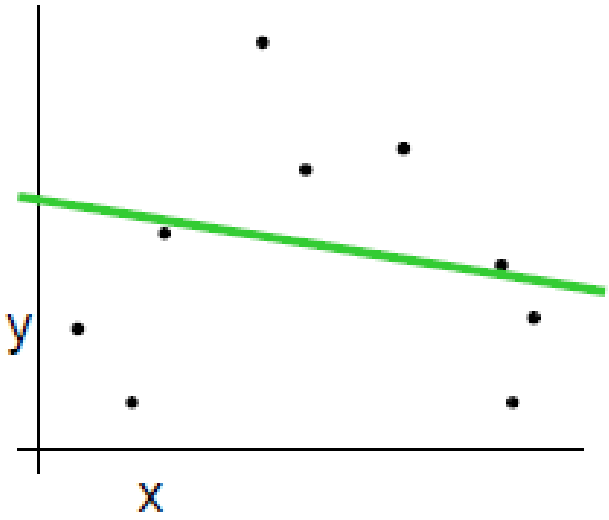
$$K(\bullet,x)\bullet K(\bullet,z)=K(x,z)$$

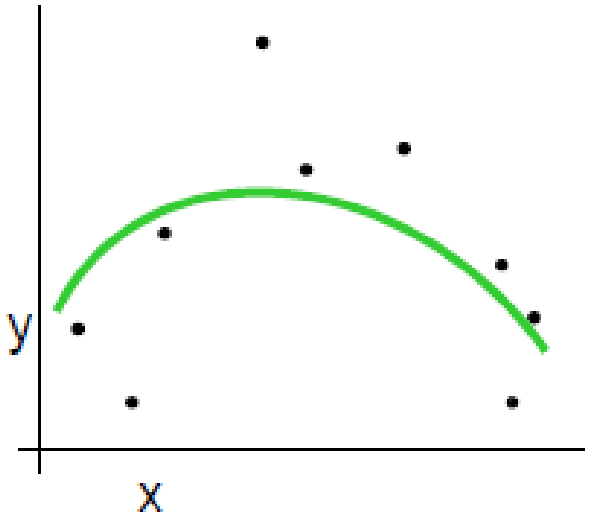
核函数与特征空间

- 核函数隐式定义了一个特征空间：使得在特征空间中的线性学习成为可能——也即在原空间中的非线性学习成为可能
- 核函数隐含了相似性度量
- 多项式核、径向基函数的高斯核、神经网络的Sigmoid函数
- 字符串核函数可用于离散数据集如文本分类等

- 特征空间维数高而样本数少：灵活性的增加容易导致**过拟合**

欠/过拟合





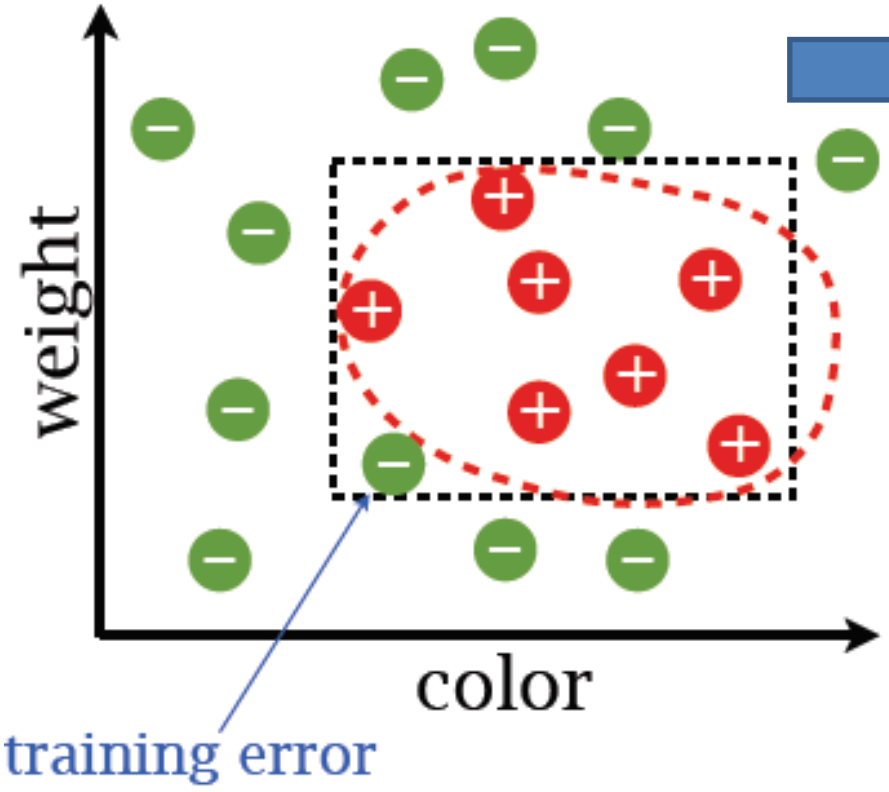
- 模型不够灵活（复杂）时，训练误差和测试误差都会比较大
- 模型太灵活（太复杂）时，训练误差可能为零，但测试误差还是会比较大，即**泛化**能力不够
- 什么样的模型复杂度是“好”的？我们又如何控制复杂度？

目录

- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- 苦其心志：核函数
- **劳其筋骨：泛化理论**
- 饿其体肤：优化理论
- 降大任于是人：支持向量机
- 降妖伏魔：算法实现
- 大显身手：实际应用

泛化理论：举例

- 泛化理论的界定理告诉我们应该控制学习器中的哪些因子才能保证好的泛化能力



界定理

下式以不小于$\mathbf{1}-\boldsymbol{\delta}$的概率成立

$$\varepsilon_g < \varepsilon_t + \sqrt{\frac{1}{m}\left(\ln|H| + \ln\frac{1}{\delta}\right)}$$

界定理告诉我们：

- 训练误差要小
- 样本数要多
- 假设集规模要小

$\ln|H|$？

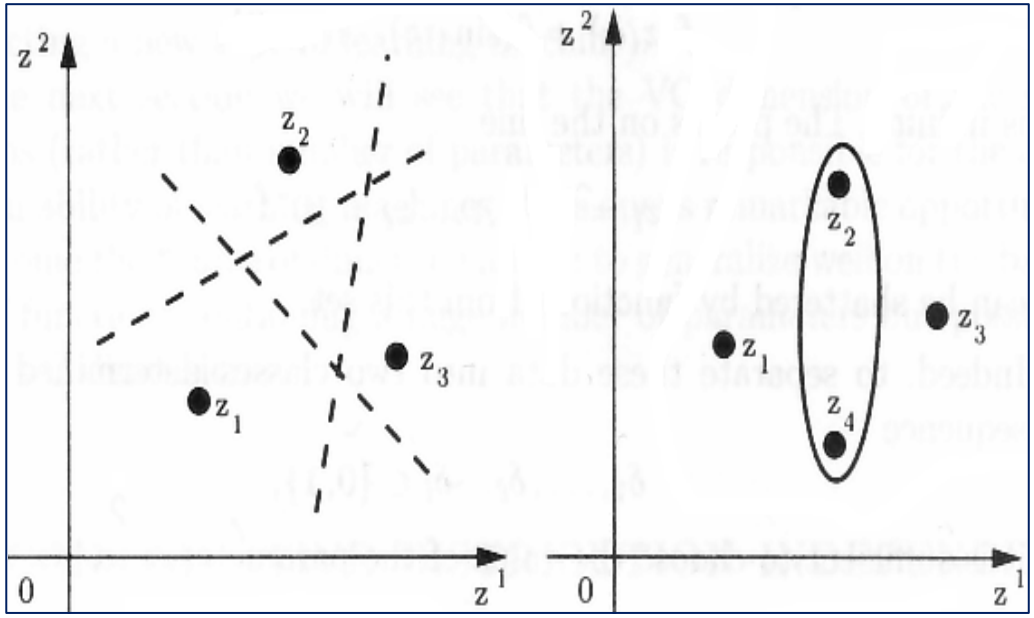
·界太悲观！

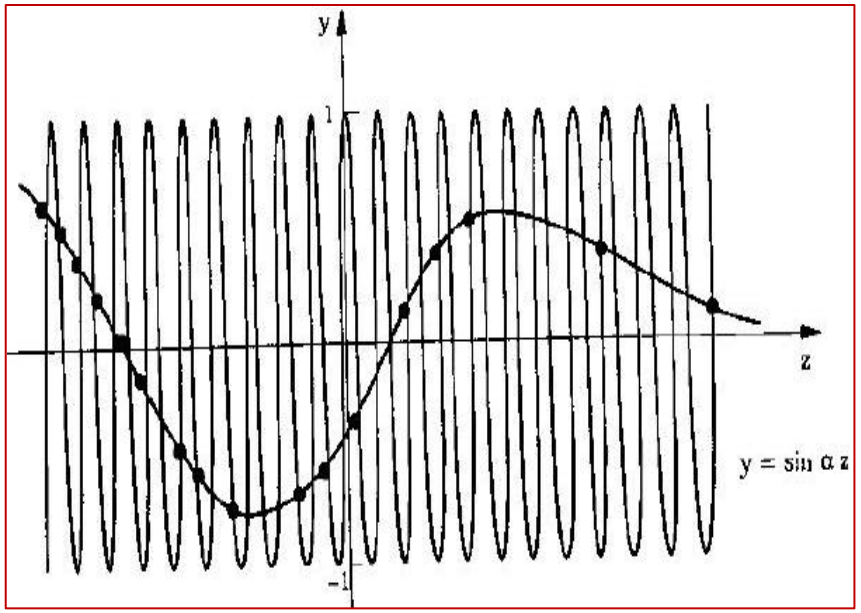
·$|H|$无限大？

此例取自俞杨老师课件

Vapnik-Chervonenkis维

- VC维可看作函数集的“有效”规模，度量其表达能力
  - 线性函数集的VC维





- 高频正弦波的VC维

- VC维：可被函数集打散的最大样本点数

$$\mathrm{VC}(H) \le \log_2 |H|$$ *Many Better Than All*

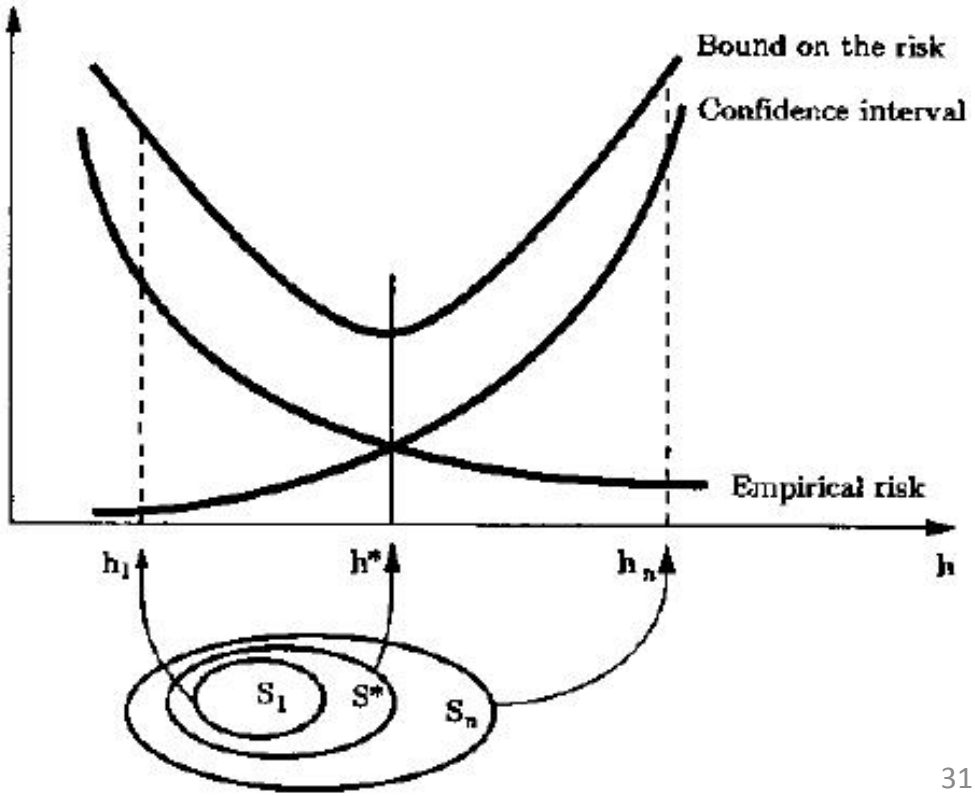
VC维与结构最小化风险

$$R(\alpha) \le R_{emp}(\alpha) + \sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}}$$ Vapnik,1995

结构风险=经验风险+置信范围

界定理告诉我们：

- 不是输入空间的维数影响泛化能力
- 也不是函数集的规模影响泛化能力

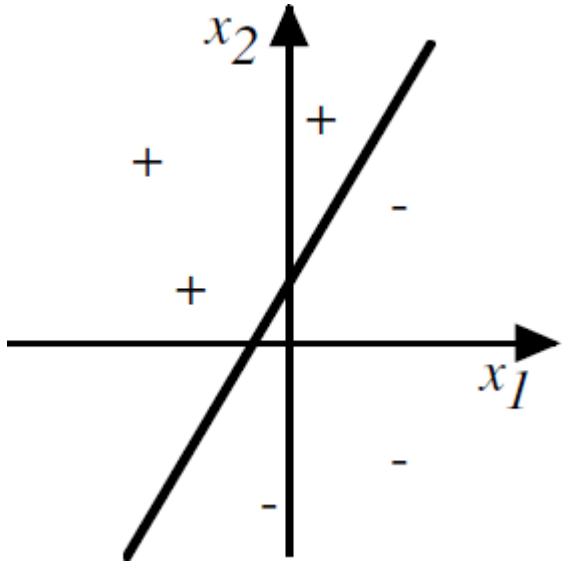


偏差-方差

再看感知器训练：抽象的观点

- 问题描述：确定线性决策边界分隔正类和负类

$$f(x) = sign(w \bullet x + b)$$



- 问题求解：转换为求解某条件或者无条件下的某代价函数极值

$$\min_{w,b} L(w,b) = -\sum_{x_i \in M} y_i (w \cdot x_i + b)$$

■这是**优化理论**的范畴

目录

- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- 苦其心志：核函数
- 劳其筋骨：泛化理论
- **饿其体肤：优化理论**
- 降大任于是人：支持向量机
- 降妖伏魔：算法实现
- 大显身手：实际应用

优化理论的表述及三个里程碑

- 解决如下问题

$$\min_x \quad f(x)$$

$$s.t. \quad g_i(x) \le 0, i = 1, \ldots, m$$

$$h_j(x) = 0,\ j = 1, \ldots,\ p$$

- Fermat1629年：无约束条件下函数的最小值
- Lagrange1788年：等式约束，拉格朗日函数/乘子
- KKT（1939,1951）：不等式约束，对偶互补条件

拉格朗日乘数法的推导

- 求等式约束条件下的极值 $z = f(x,y)\ \ \varphi(x,y) = 0$

1)从约束中得到隐函数： $y = y(x)$

2)带入z中化为无约束问题： $z = f(x, y(x))$

3)求偏导令其为零：

$$\frac{dz}{dx} = f_x + f_y \cdot y'(x) = 0$$

$$y'(x) = -\frac{\varphi_x}{\varphi_y}$$

隐函数求导法则

4)整理得： $f_x - f_y \dfrac{\varphi_x}{\varphi_y} = 0 \Leftrightarrow \dfrac{f_x}{\varphi_x} = \dfrac{f_y}{\varphi_y} = -\lambda$

$$f_x + \lambda\varphi_x = 0 = F_x$$

$$f_y + \lambda\varphi_y = 0 = F_y$$

$$\varphi(x,y) = 0 = F_\lambda$$

$$F(x,y,\lambda) = f(x,y) + \lambda\varphi(x,y)$$

上述推导过程取自：高等数学（第二版）上，李忠等著，北大出版社

Karush-Kuhn-Tucker条件

- 引进广义拉格朗日函数

$$L(x,a,b)=f(x)+\sum_{i=1}^{m}a_i g_i(x)+\sum_{j=1}^{p}b_j h_j(x)$$

- KKT条件由如下三部分构成
  - L 分别对x, a, b求偏导 $\nabla_{x,a,b} L = 0$
  - 原始约束 $g_i(x) \le 0$ $h_j(x) = 0$
  - **对偶互补条件** $a_i g_i(x) = 0, a_i \ge 0$

最优化理论

- 拉格朗日乘子也叫对偶变量，它是将参数向量**展成**输入和输出的线性组合的系数

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

- 指示了输入点的信息量：每个样本点关联一个乘子，难学习的点有较大的乘子值
- 优化理论刻画了解的数学特性：KKT条件
- 训练SVM，凸二次规划就足够了

有关SVM的更多优化理论，请参看：邓乃扬等. 支持向量机. 科学出版社,2009

目录

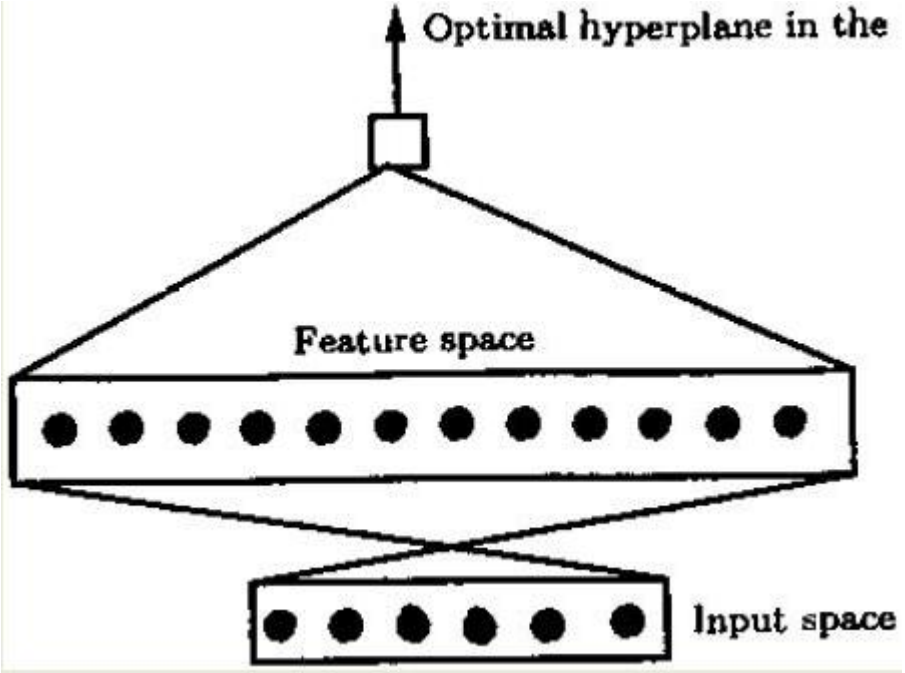
- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- 苦其心志：核函数
- 劳其筋骨：泛化理论
- 饿其体肤：优化理论
- **降大任于是人：支持向量机**
- 降妖伏魔：算法实现
- 大显身手：实际应用

支持向量机

- 定义
- 学习偏置：结构风险最小化
- 间隔超平面
- 三种情况：线性可分、近似可分、不可分
- 支持向量回归与损失函数
- 统计学习方法

定义

- SVM是一个学习器：
  - 在核特征空间中使用线性函数作为假设空间
  - 要学习的偏置受泛化理论的控制
  - 训练算法来自优化理论

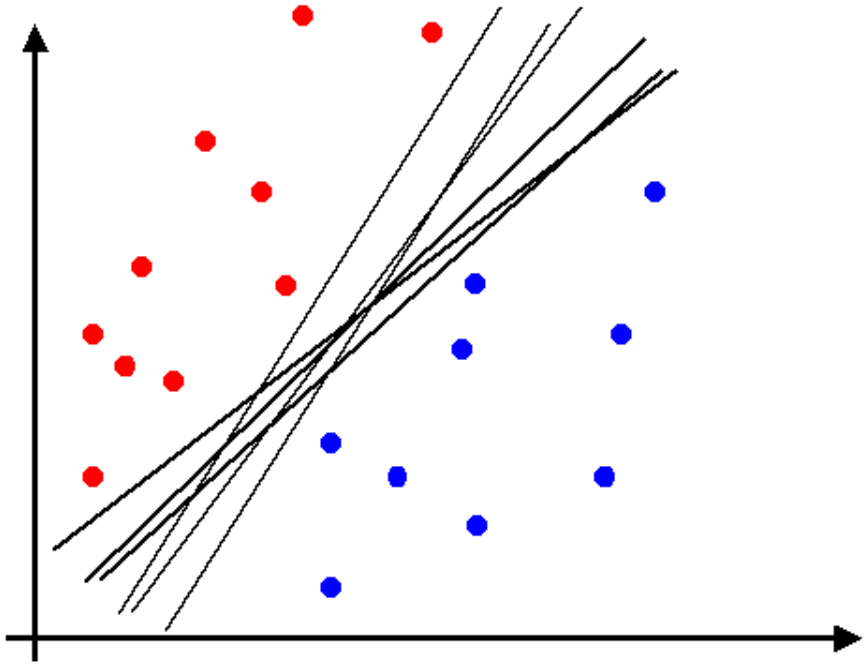


- SVM与SLT
  - SVM becomes hot since the middle of 1990s
  - SLT were obtained in 1960 - 70s： VC维和SRM归纳原则

学习偏置

$$R(\alpha) \le R_{emp}(\alpha) + \sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}}$$ Vapnik1995

- 学习策略一：保持置信范围固定，最小化经验风险（ANN）
- 学习策略二：保持经验风险固定，最小化置信范围（SRM）



•线性可分：硬间隔
•线性近似可分：软间隔
•线性不可分：核函数

■ 大间隔的泛化思想

线性可分：间隔

- 函数间隔 $\gamma* = \min_i y_i(w\bullet x_i + b)$ 归一化
- 几何间隔 $\gamma = \frac{\gamma*}{\|w\|} = \min_i y_i \frac{w\bullet x_i + b}{\|w\|}$

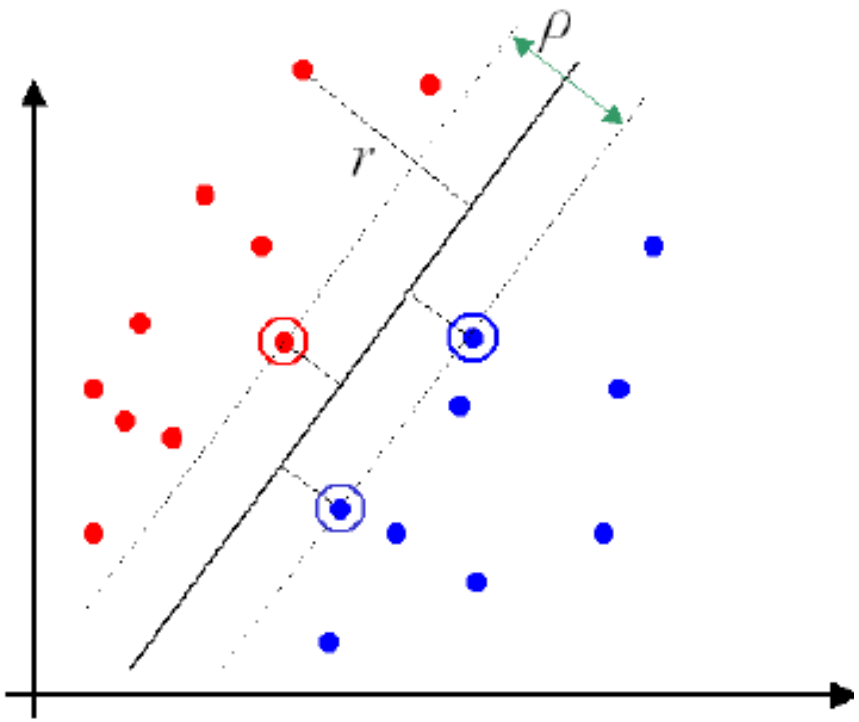
- 带符号的距离：

—正负号表示分类正误

—绝对值表示确信程度

$$r_i = \frac{w\bullet x_i + b}{\|w^*\|}$$

$$\rho = \frac{2}{\|w^*\|}$$



硬间隔

- 最大间隔分类超平面：在分类正确的约束下最小化权重范数

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

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

取$\gamma^*=1$

$$\min_{w,b} \quad \frac{1}{2}||w||^2$$
$$s.t. \quad -y_i(w\cdot x_i + b)+1 \leq 0$$

原始问题

原始问题→对偶问题

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

$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0 \qquad \alpha_i \geq 0$$

对偶问题

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

$$\nabla_w L = w - \sum_{i=1}^{N}\alpha_i y_i x_i = 0 \Rightarrow w = \sum_{i=1}^{N}\alpha_i y_i x_i$$

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

如何求 ***b***呢？

- 原始问题是拉格朗日函数的极小极大问题
- 对偶问题是拉格朗日函数的极大极小问题

KKT条件是原始问题与对偶问题有相同最优值的充要条件

原始问题 vs 对偶问题

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$

$$s.t. \quad -y_i(w \bullet x_i + b) + 1 \le 0$$

- 对偶问题完全根据训练数据表达
  - 通过求解乘子间接求解原参数
- 训练数据点以成对的内积形式出现
  - 为核函数的引入提供了可能
- 乘子有其特有的数学意义和实际意义

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

$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0 \quad \alpha_i \ge 0$$

几个关系

$$\gamma = 1/\| w^* \|_2$$

$$\rho = 2\gamma$$

$$w^* \cdot w^* = \sum_{i,j=1}^{N} y_i y_j \alpha_i^* \alpha_j^* < x_i \cdot x_j >$$

$$y_j(\sum_{i \in SV} y_i \alpha_i^* < x_i \cdot x_j > + b^*) = 1$$

$$= \sum_{j \in SV} \alpha_j^* y_j \sum_{i \in SV} y_i \alpha_i^* < x_i \cdot x_j >$$

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

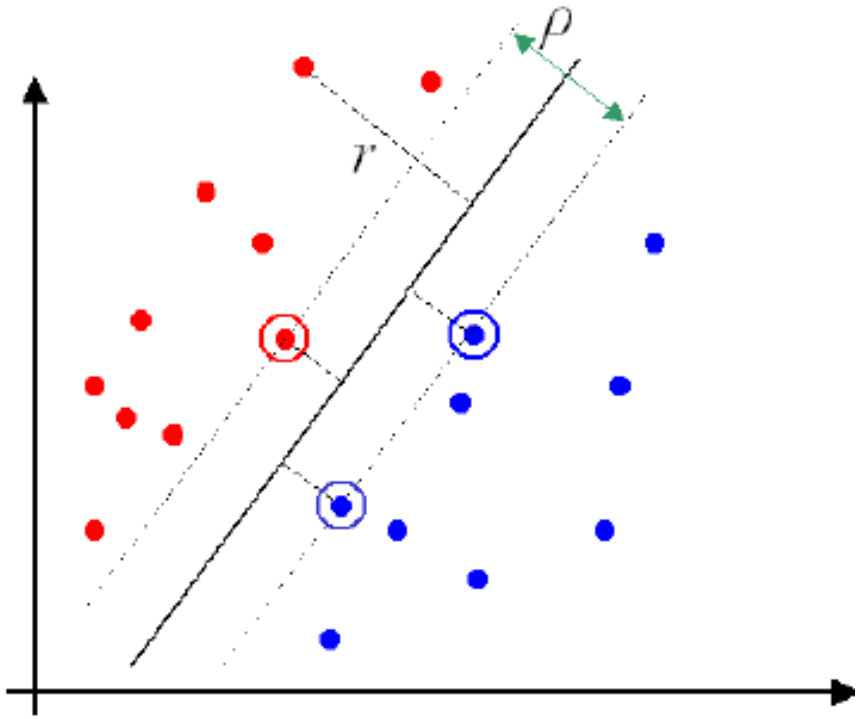
$$= \sum_{j \in SV} \alpha_j^* (1 - y_j b^*)$$

$$= \sum_{i \in SV} \alpha_i^*$$

$$\max \gamma \Rightarrow \min \| w \| \Rightarrow \min \sum_{i \in SV} \alpha_i^*$$

支持向量

- 乘子非零的点叫做支持向量：它们是最靠近决策面的，这样的点也是最难分的，容易犯错误



- KKT的对偶互补条件提供了解的稀疏性的可能
- 支持向量是训练集的子集，意味着解通常是稀疏的
- 不是支持向量的点对决策无影响，它们的轻微扰动也不影响解

- 由支持向量可重构决策超平面，它们是训练集的一个*压缩方案*；而支持向量机构建的分类模型是训练集的一个压缩方案

线性近似可分：松弛和罚项

- 约束引入松弛变量、目标函数加入惩罚项
- 参数C起平衡作用

$$\begin{aligned} \min_{\gamma,w,b} \quad & \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, \quad i = 1, \ldots, m. \end{aligned}$$

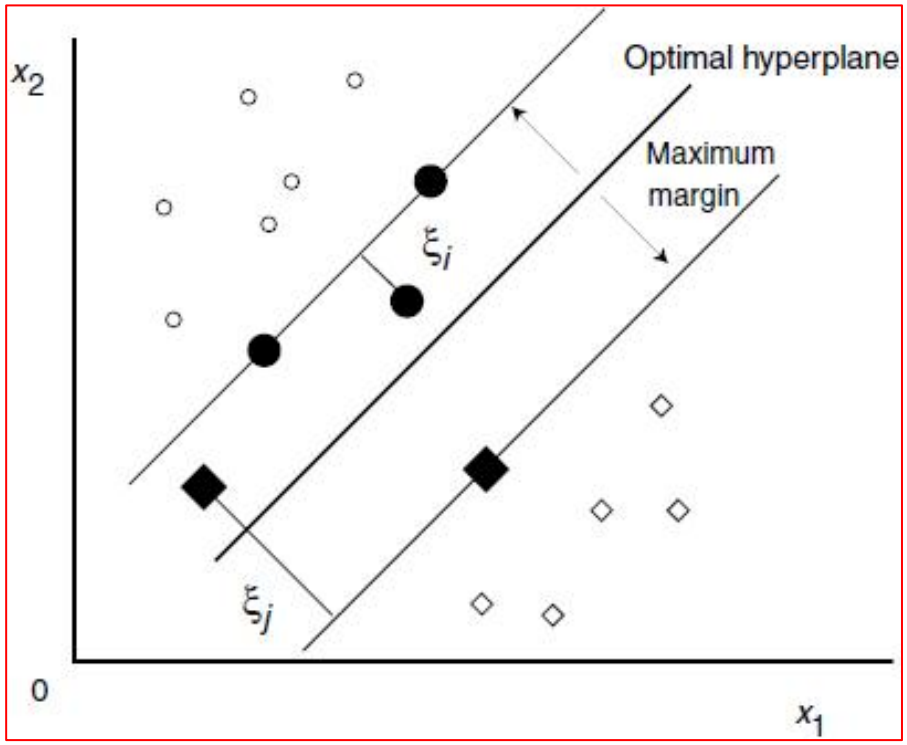
$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, m \\ & \sum_{i=1}^{m}\alpha_i y^{(i)} = 0, \end{aligned}$$

线性等式约束

- 训练点仍以内积形式成对出现
- 松弛变量不出现在对偶形式中

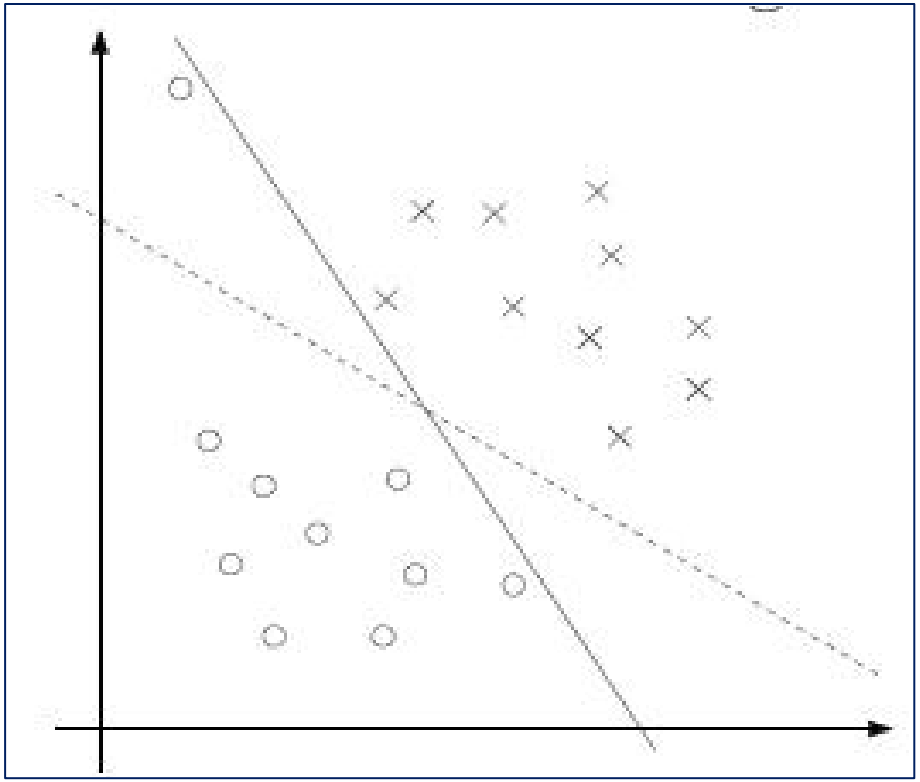
- 罚项技术也叫正则化，解决不适定问题，避免过拟合
- 罚项可以是松弛变量的二范数，此时需用C修正核矩阵
- 一阶或二阶好取决于实际数据，也受噪声模型的影响

软间隔



软间隔的支持向量比较复杂：可位于间隔边界上、可位于超平面上或两者之间；甚至位于超平面误分一侧

有了线性可分（线性不可分）为什么还需要线性近似可分？

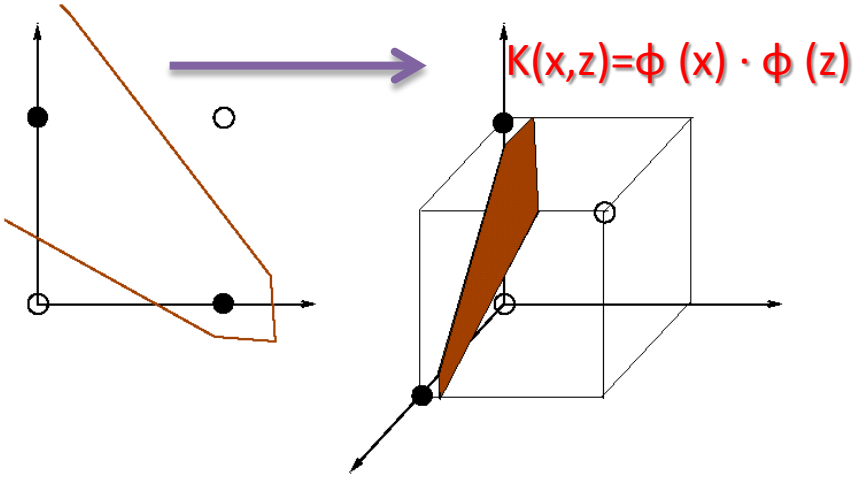


根本原因：小样本

任何数学技巧都不能补救信息的缺失

线性不可分：核函数

- 线性不可分与线性近似可分的求解几乎完全一样，只需要使用核函数隐式实现一个非线性映射的内积
- 代价是维数变高



$$\begin{aligned} &\textit{maximise} && W(\alpha) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j \left(K(\mathbf{x}_i, \mathbf{x}_j) + \frac{1}{C} \delta_{ij} \right), \\ &\textit{subject to} && \sum_{i=1}^{\ell} y_i \alpha_i = 0, \\ & && \alpha_i \geq 0, \; i = 1, \ldots, \ell. \end{aligned}$$

松弛变量的二阶范数

- 支持向量机学习的可能性：概念上和技术/计算上的问题解决

$$EP_{\text{error}} \leq E \min \left(\frac{m}{\ell}, \frac{[R^2 |w|^2]}{\ell}, \frac{n}{\ell} \right)$$

- 最大间隔；压缩方案
- 核技巧：内积的**convolution**

归纳

Input	training set $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$, $\delta > 0$
Process maximise subject to	find $\boldsymbol{\alpha}^*$ as solution of the optimisation problem: $W(\boldsymbol{\alpha}) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y_i y_j \kappa(\mathbf{x}_i, \mathbf{x}_j)$ $\sum_{i=1}^{\ell} y_i \alpha_i = 0$ and $0 \leq \alpha_i$, $i = 1, \ldots, \ell$.
4	$\gamma^* = \left(\sum_{i \in \mathrm{sv}} \alpha_i^*\right)^{-1/2}$
5	choose i such that $0 < \alpha_i^*$
6	$b = y_i - \sum_{j \in \mathrm{sv}} \alpha_j^* y_j \kappa(\mathbf{x}_j, \mathbf{x}_i)$
7	$f(\cdot) = \mathrm{sgn}\left(\sum_{j \in \mathrm{sv}} \alpha_j^* y_j \kappa(\mathbf{x}_j, \cdot) + b\right)$;
8	$\mathbf{w} = \sum_{j \in \mathrm{sv}} y_j \alpha_j^* \phi(\mathbf{x}_j)$
Output	weight vector $\mathbf{w}$, dual solution $\boldsymbol{\alpha}^*$, margin γ^* and function f implementing the decision rule represented by the hyperplane

$0 \leq \alpha_i \leq C$

支持向量机：特点小结

- 核技巧，可以使用线性学习器学习非线性关系

通用性好

- 通过大间隔、VC维和支持向量等与维数无关的量来控制容量

泛化性好

- 解的稀疏性

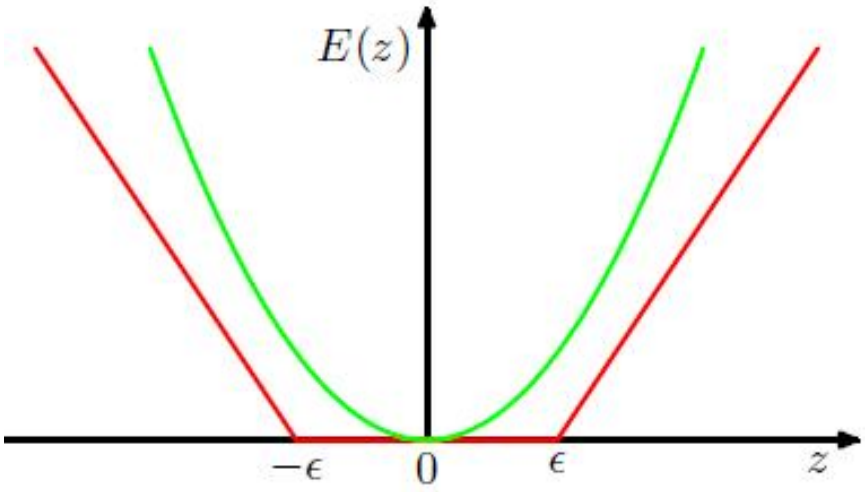
快速算法

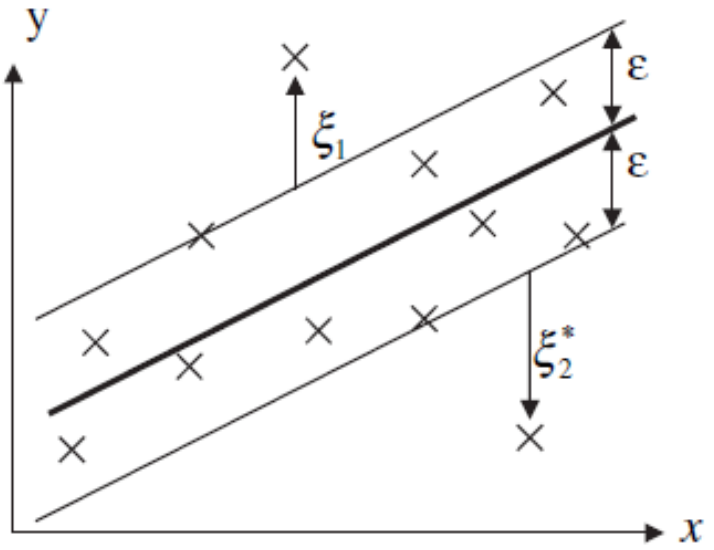
- 凸优化问题、没有局部最小

- 坚实的理论基础
- 快速的算法实现
- 绝对的杀手应用

支持向量机回归

- 支持向量的方法除了用于分类还可用于回归，且保留了最大间隔算法的所有主要特征





$$\underset{\mathbf{w}\in\mathcal{H},\boldsymbol{\xi}^{(*)}\in\mathbb{R}^m,b\in\mathbb{R}}{\text{minimize}} \quad \tau(\mathbf{w},\boldsymbol{\xi}^{(*)}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}(\xi_i+\xi_i^*)$$

$$\text{subject to} \quad f(\mathbf{x}_i) - y_i \le \varepsilon + \xi_i$$
$$y_i - f(\mathbf{x}_i) \le \varepsilon + \xi_i^*$$
$$\xi_i, \xi_i^* \ge 0 \qquad \text{for all } i = 1,\ldots,m$$

统计学习方法：解构的观点

- 方法=模型+策略+算法
  - 模型的*假设空间*界定了学习的范围
  - 策略即在*目标函数*的指导下选择最优模型
- 对SVM：
  - 模型：特征空间上学习线性函数集
  - 策略：特征空间上选择最优分类超平面
  - 算法：？不是有优化理论吗？

■ 优化理论对SVM的贡献不是在算法方面而在于它刻画了解的数学特性。算法要实现的就是求解该优化问题的数值方法/计算方法

目录

- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- 苦其心志：核函数
- 劳其筋骨：泛化理论
- 饿其体肤：优化理论
- 降大任于是人：支持向量机
- **降妖伏魔：算法实现**
- 大显身手：实际应用

SMO

- SVM特定算法的设计利用了其自身独有的特点及其来自优化理论的优美的数学性质
- **S**equential **M**inimal **O**ptimization：
- 从线性等式约束条件入手采用分治策略：每次选择两个对偶变量固定其余变量得到一个针对这两个变量的凸二次规划问题，而这有**解析解**！不断重复该过程直到所有变量均满足KKT条件（或在某精度范围内满足停机条件）从而得到原问题的解

两变量的解析解

- 由线性等式 $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$ 约束有：

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{m} \alpha_i y^{(i)} = Const$$

- 用α_2替换α_1：

$$\alpha_1 = -y^{(1)} \sum_{i=2}^{m} \alpha_i y^{(i)} = (Const - \alpha_2 y^{(2)}) y^{(1)}$$

因此两变量的最优化成为单变量的最优化且变量的**C**约束保证其是有界的，可解析地求解

归并排序？

目录

- 缘起：从数据中学习
- 九层之台起于垒土：线性学习器
- 苦其心志：核函数
- 劳其筋骨：泛化理论
- 饿其体肤：优化理论
- 降大任于是人：支持向量机
- 降妖伏魔：算法实现
- **大显身手：实际应用**

实际应用：手写数字识别

- *统计学习理论，Vapnik 5.7 SV机的实验*

Classifier	Raw error%
Human performance	2.5
Decision tree, C4.5	16.2
Best two-layer neural network	5.9
Five-layer network (LeNet 1)	5.1

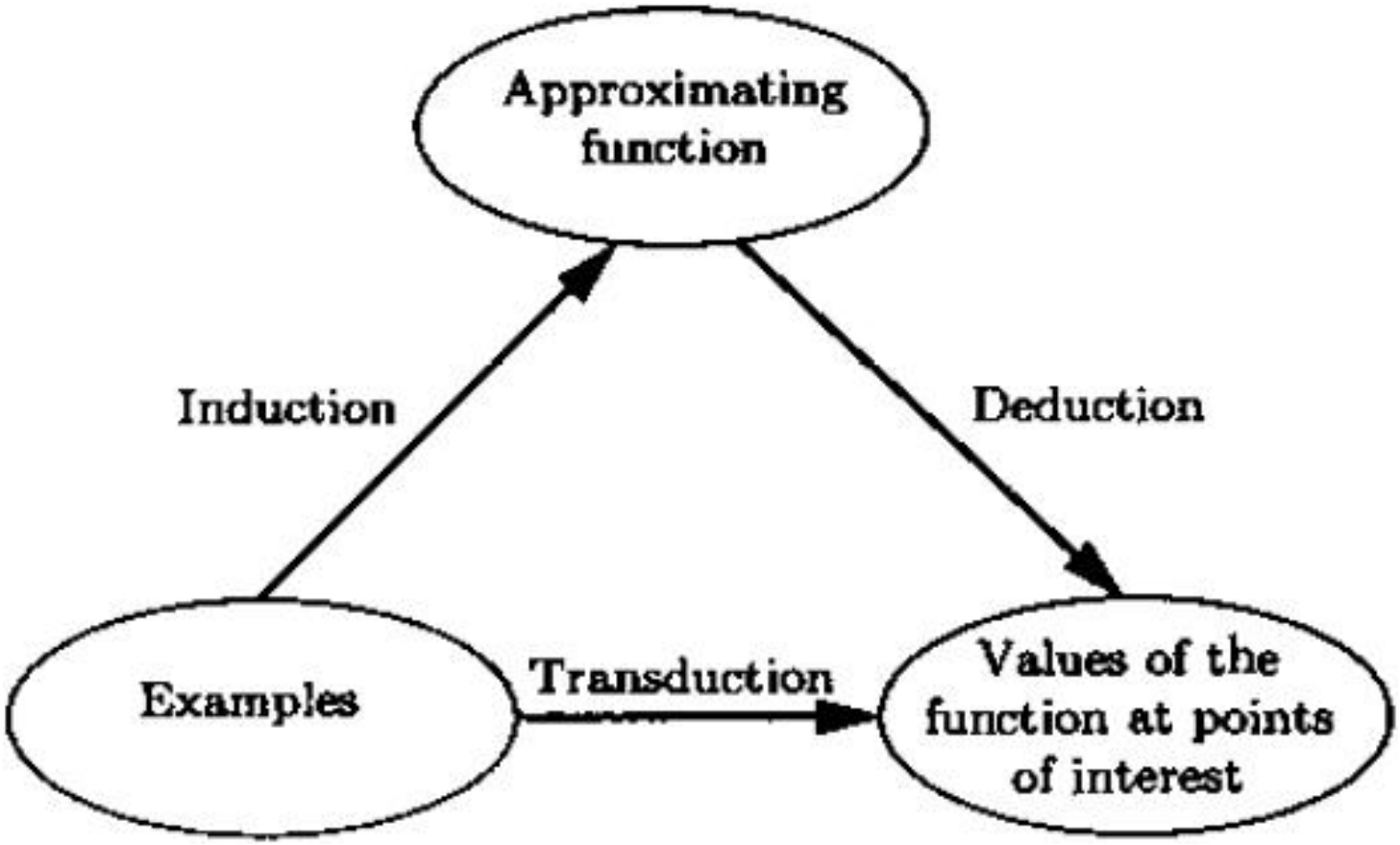
Type of SV classifier	Parameters of classifier	Number of support vectors	Raw error
Polynomials	d=3	274	4.0
RBF classifiers	$\sigma^2 = 0.3$	291	4.1
Neural network	$b = 2,\ c = 1$	254	4.2

degree of polynomial	dimensionality of feature space	support vectors	raw error
1	256	282	8.9
2	≈ 33000	227	4.7
3	$\approx 1 \times 10^6$	274	4.0
4	$\approx 1 \times 10^9$	321	4.2
5	$\approx 1 \times 10^{12}$	374	4.3
6	$\approx 1 \times 10^{14}$	377	4.5
7	$\approx 1 \times 10^{16}$	422	4.5

- 高维学习的可能性（技术上的问题）
- 支持向量（VC维）控制了泛化性能（概念上的问题）

尾声：推理原则

- 演绎还是归纳？还是？？



哲人咖啡厅

- *Nothing is more practical than a good theory——**Vapnik***

参考文档

统计学习理论的本质，Vapnik著，张学工译
支持向量机导论，Cristianini等著，李国正等译
统计学习方法，李航著