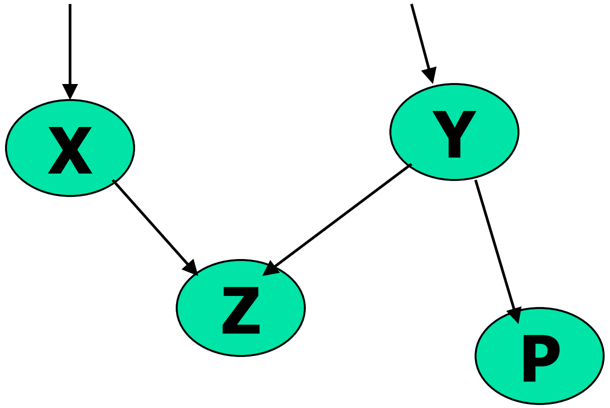
Chapter 8. 分类: Advanced Methods

Chapter 8. 分类: Advanced Methods

- 贝叶斯信念网络
- 后向传播分类 Classification by Backpropagation
- 支持向量机 Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- 其他分类方法
- Additional Topics Regarding Classification
- Summary

贝叶斯信念网络

- **Bayesian belief networks (又称为 Bayesian networks, probabilistic networks): 允许变量子集间定义类条件独立**
- **(*有向无环*) 因果关系的图模型**
  - 表示变量间的<u>依赖</u>关系
  - 给出了一个联合概率分布



❑ Nodes: 随机变量

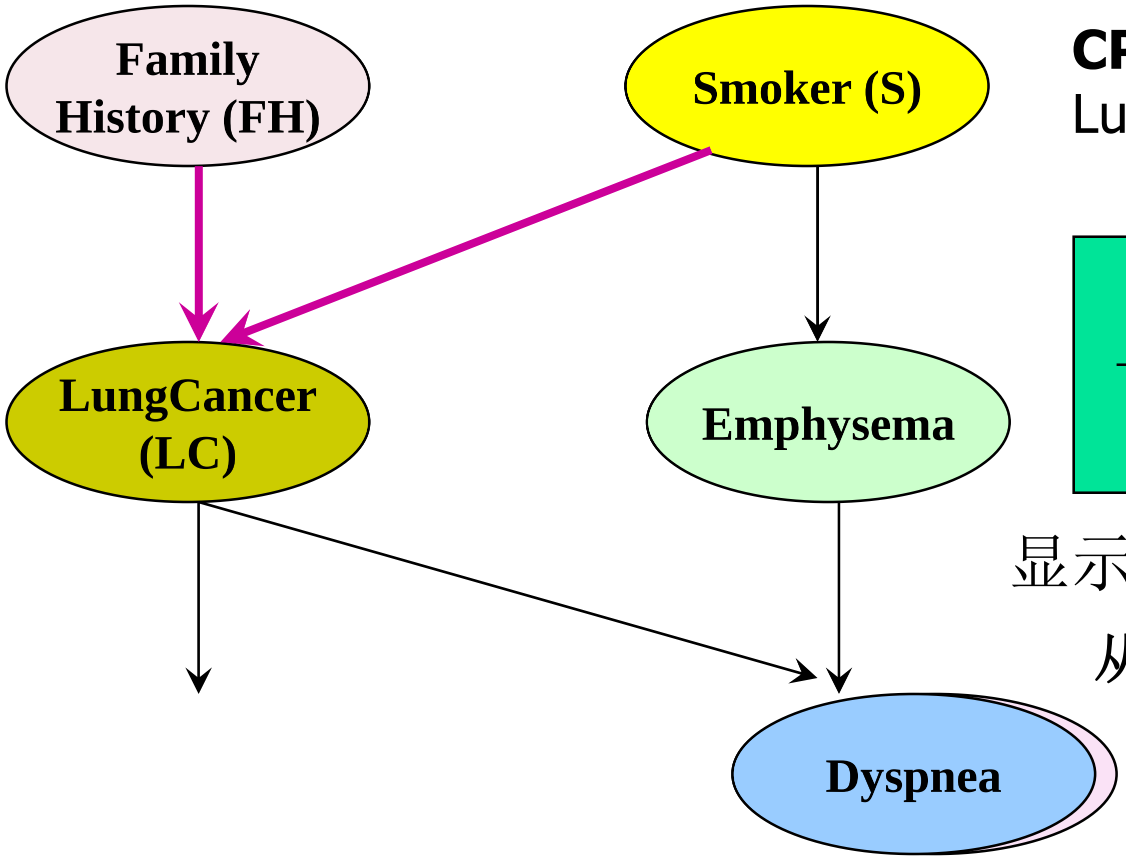
❑ Links: 依赖关系

❑ X,Y 是Z的双亲, Y is the parent of P

❑Z 和 P间没有依赖关系

❑ 没有环

贝叶斯信念网络: An Example



CPT: **Conditional Probability Table** for variable LungCancer:

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

显示父母的每个可能组合的条件概率

从CPT推倒 X的特定值得概率

$$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(Y_i))$$

训练贝叶斯网路:几种方案

- **Scenario 1:给定网络结构和所有变量观察：只计算CPT**
- **Scenario 2: 网络结构已知, 某些变量隐藏: 梯度下降法(贪心爬山), i.e., 沿着准则函数的最速下降方向搜索解**
  - **权重初始化为随机值**
  - **每次迭代中,似乎是对目前的最佳解决方案前进，没有回溯**
  - **每次迭代中权重被更新，并且收敛到局部最优解**
- **Scenario 3: 网络结构未知, 所有变量可知: 搜索模型空间构造网络拓扑**
- **Scenario 4: 未知结构, 隐藏变量: 目前没有好的算法**
- **D. Heckerman. A Tutorial on Learning with Bayesian Networks. In *Learning in Graphical Models*, M. Jordan, ed.. MIT Press, 1999.**

Chapter 8. 分类: Advanced Methods

- Bayesian Belief Networks
- Classification by Backpropagation
- Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- Other Classification Methods
- Additional Topics Regarding Classification
- Summary

用反向传播分类

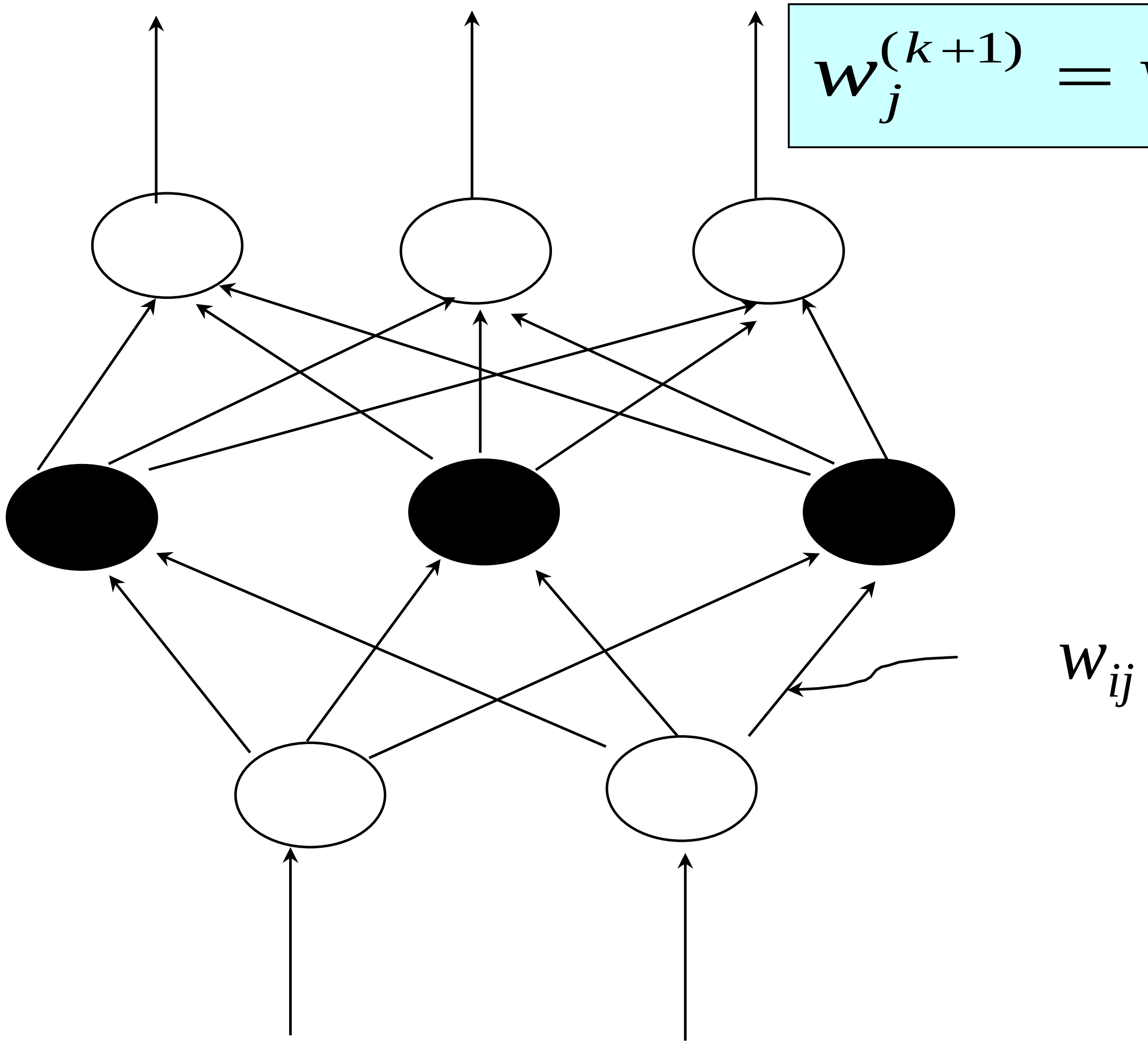
- 反向传播: 一种神经网络学习算法
- 最早是由心理学家和神经学家开创的，开发和测试神经元计算模拟
- 神经网络: 一组连接的输入/输出单元，其中每个连接都与一个权重关联
- 通过调整权重来学习, 能够输入元组的正确类别标号
- 又被称为连接者学习**connectionist learning**

神经网络作为分类器

- 弱点
  - 学习时间很长
  - 需要很多参数（常靠经验确定）, 如网络的结构
  - 可解释性差: 很难解释权重和网络中“隐藏单元”的含义
- 优势
  - 对噪音数据的高承受能力
  - 分类未经训练的模式的能力
  - 非常适合处理连续值的输入/输出
  - 成功地应用于现实数据, e.g., 手写字符识别
  - 算法是固有并行的
  - 已经发展了一些从训练好的神经网路提取规则的技术

多层前馈神经网络

$$w_j^{(k+1)} = w_j^{(k)} + \lambda(y_i - \hat{y}_i^{(k)})x_{ij}$$



Output vector

输出层

隐藏层

输入层

Input vector: *X*

多层前馈神经网络

- 网络的输入对应于每个训练元组的测量属性
  - 输入同时传给称作输入层的单元
- 加权后同时传递给隐藏层
- 隐藏层的数目是任意的, 通常只有一个
- 最后一个隐藏层的输出权重后作为输入传递给称为输出层，此处给出网络的预测
- 前馈feed-forward: 权重都不反馈到输入单元或前一层的输出单元
- 从统计学观点, 网络进行一个非线性回归；给定足够的隐藏单元和训练数据, 可以逼近任何函数

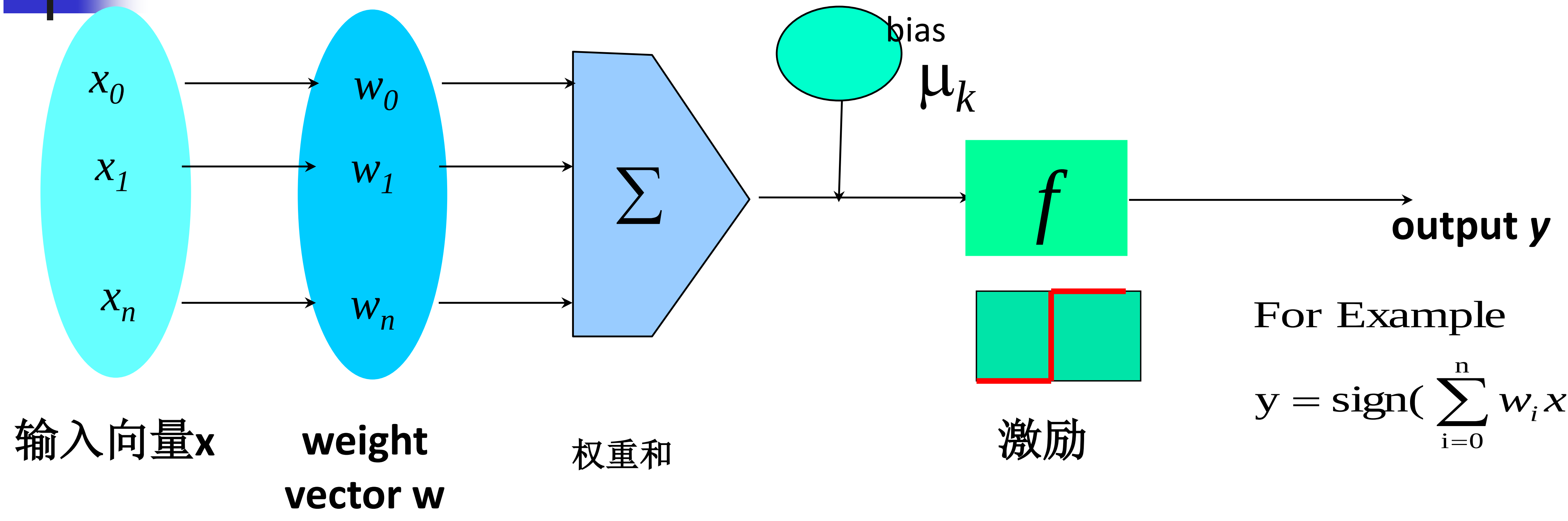
定义网络拓扑

- 确定网络拓扑: 给定输入层的单元数, 隐藏层数(if > 1), 每个隐藏层的单元数, 输出层的单元数
- 规格化训练元组的输入值 [0.0—1.0]
  - 对于离散值，可重新编码，每个可能的值一个输入单元并初始化0
- 输出, 如果涉及超过两个类别则一个输出单元对应一个类别
- 一旦一个训练好的网络其准确率达不到要求时，用不同的网络拓扑和初始值重新训练网络

反向传播Backpropagation

- 迭代地处理训练数据 & 比较网络的预测值和实际的目标值
- 对每个训练元组, 修改权重最小化目标的预测值和实际值之间的**mean squared error**
- 这种修改后向进行: 从输出层开始, 通过每个隐藏层直到第一个隐藏层
- 步骤
  - 初始化权重为一个小的随机数, 以及偏倚 **biases**
  - 向前传播输入 (应用激励函数)
  - 向后传播误差 (更新权重和偏倚)
  - 停止条件 (当误差非常小, etc.)

神经元: 一个隐藏/输出层单元

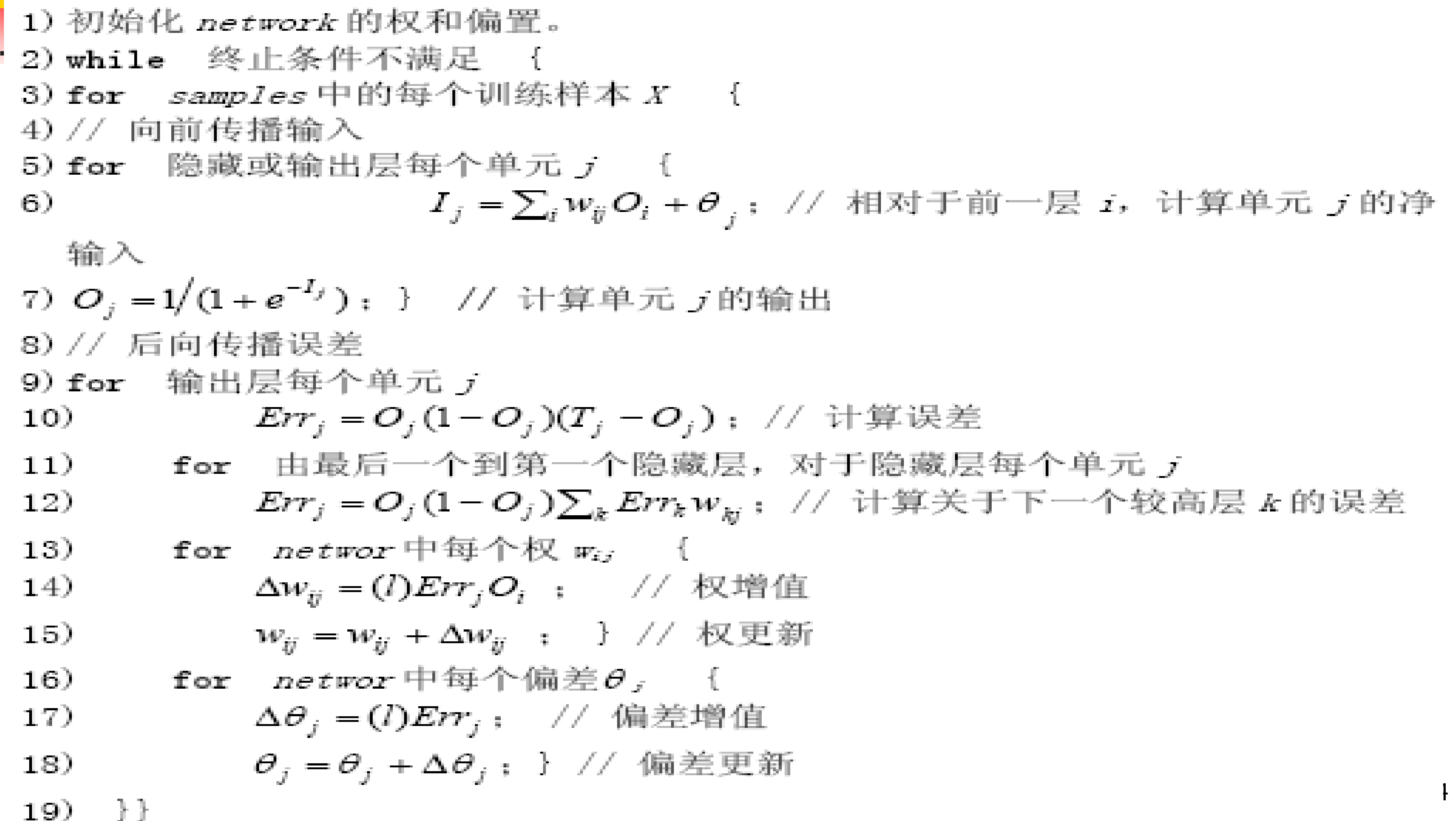


For Example

$$y = \text{sign}(\sum_{i=0}^{n} w_i x_i - \mu_k)$$

- 一个***n***-维输入向量x被映射到变量y，通过非线性函数映射
- 单元的输入是前一层的输出. 被乘上权重后求和且加上此单元的偏倚. 然后应用一个非线性激励函数.

后向传播算法



```
初始化 network 的权和偏置。
while 终止条件不满足 {
for samples 中的每个训练样本 X {
// 向前传播输入
for 隐藏或输出层每个单元 j {
                $I_j = \sum_i w_{ij} O_i + \theta_j$；// 相对于前一层 i，计算单元 j 的净输入
$O_j = 1/(1+e^{-I_j})$；}  // 计算单元 j 的输出
// 后向传播误差
for 输出层每个单元 j
        $Err_j = O_j(1-O_j)(T_j-O_j)$；// 计算误差
    for 由最后一个到第一个隐藏层，对于隐藏层每个单元 j
        $Err_j = O_j(1-O_j)\sum_k Err_k w_{kj}$；// 计算关于下一个较高层 k 的误差
    for networ 中每个权 $w_{ij}$ {
        $\Delta w_{ij} = (l) Err_j O_i$ ；  // 权增值
        $w_{ij} = w_{ij} + \Delta w_{ij}$ ；  } // 权更新
    for networ 中每个偏差 $\theta_j$ {
        $\Delta\theta_j = (l) Err_j$；  // 偏差增值
        $\theta_j = \theta_j + \Delta\theta_j$；} // 偏差更新
}}
```

效率和可解释性

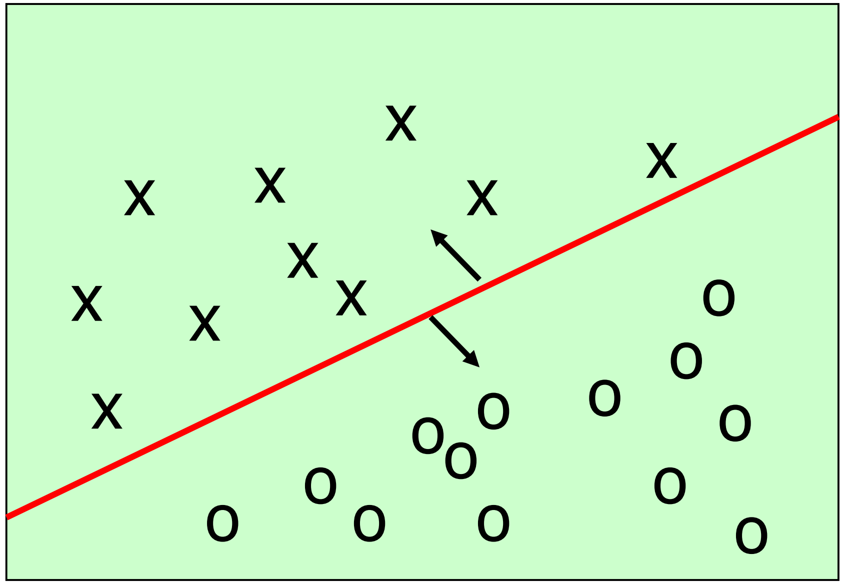
- 向后传播的效率: 每次迭代 O(|D| * *w*), |D|为元组数，*w* 个权重, 最坏的情况下迭代的次数可能是元组数的指数
- 为了更容易理解: 通过网络修剪提取规则
  - 简化网络结构，去除对训练的网络有最弱影响的权重连接
  - 对连接, 单元, or 活跃值聚类
  - 输入和活跃值集合用来推导描述输入和隐藏层间关系的规则
- Sensitivity analysis: 评估一个给定的输入变量对网络输出的影响。从中获得的知识可以表示为规则。
  - **IF X 减少5% THEN Y增加…**

Chapter 8. 分类: Advanced Methods

- 贝叶斯信念网络
- 后向传播分类 Classification by Backpropagation
- 支持向量机 Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- 其他分类方法
- Additional Topics Regarding Classification
- Summary

分类:一个数学映射

- Classification:预测分类的类标签
  - E.g., 个人主页分类
    - $x_i = (x_1, x_2, x_3, \ldots)$, $y_i = +1$ or -1
    - x_1 : # of word “homepage”
    - x_2 : # of word “welcome”
- $x \in X = \Re^n$, $y \in Y = \{+1, -1\}$,
  - 推导一个函数 $f: X \rightarrow Y$
- 线性分类
  - 二元分类问题
  - 红线上面的点属于 class ‘x’
  - 下面的点属于 class ‘o’
  - Examples: SVM, Perceptron, Probabilistic Classifiers



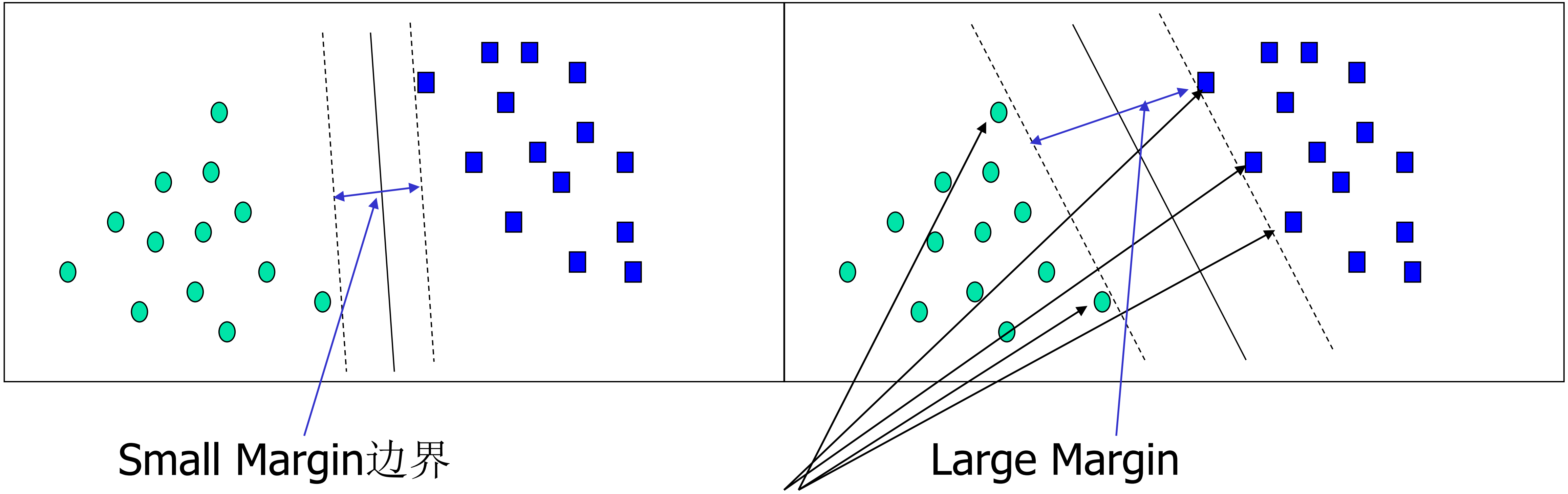
SVM—Support Vector Machines

- 一个相对新的分类方法，适用于linear and nonlinear data
- 使用一个非线性映射把原始训练数据变换到高维空间中
- 在新的维上, 搜索线性优化分离超平面hyperplane (i.e., “决策边界”)
- 用一个合适的足够高维的映射, 两类数据总是可以被超平面分开
- SVM 使用support vectors (“基本” 选练元组) 和边缘margins (由支持向量定义) 发现超平面

SVM—历史和应用

- **Vapnik and colleagues (1992)—基础工作来自于Vapnik & Chervonenkis' statistical learning theory in 1960s**
- **Features: 训练慢但是准确度高，由于能够建模非线性决策边界 (margin maximization)**
- **Used for: 分类和数值预测**
- **应用:**
  - 手写数字识别, **object recognition, speaker identification,** 基准时间序列预测检验

支持向量机的一般哲学

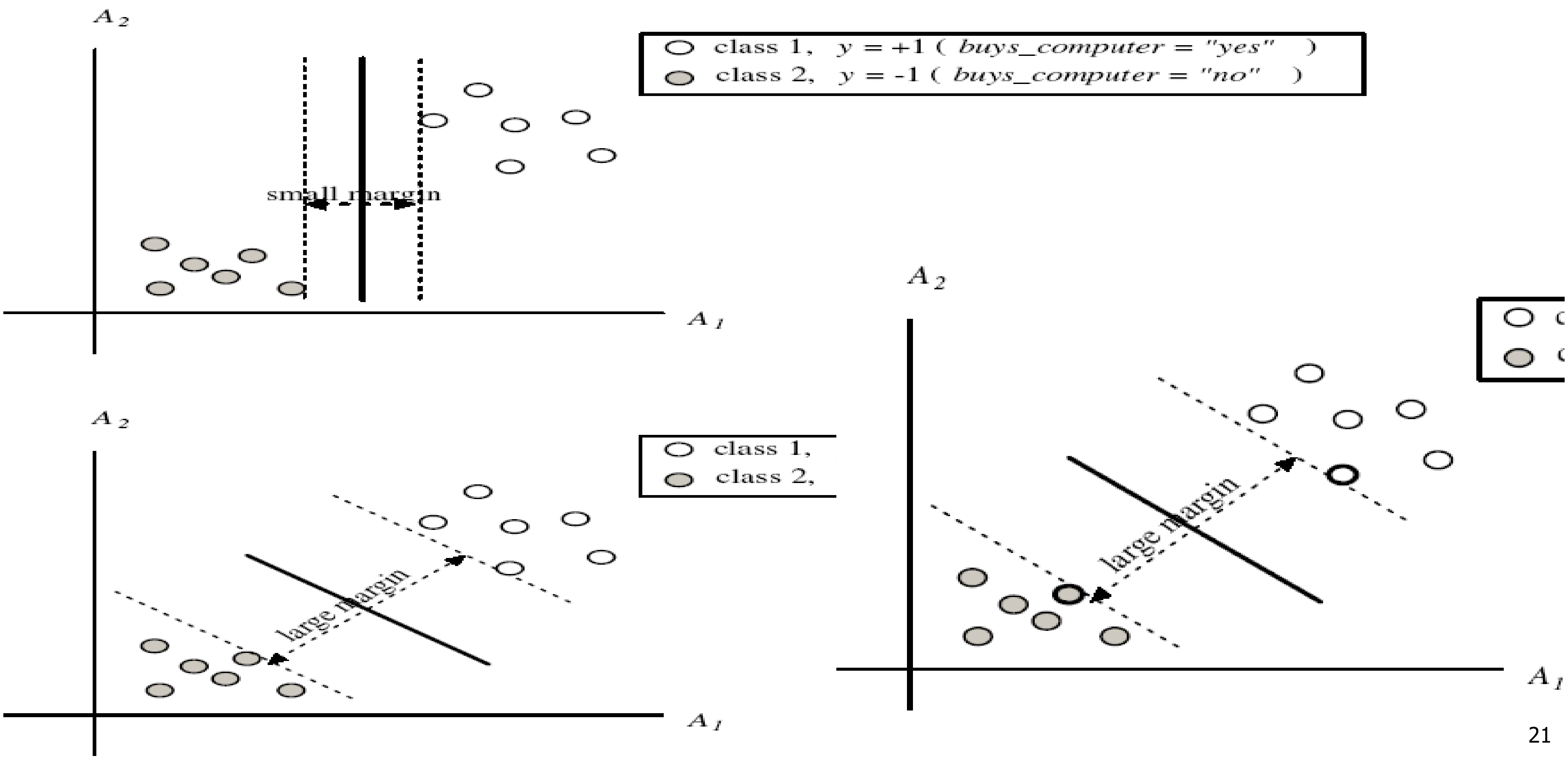


Small Margin边界

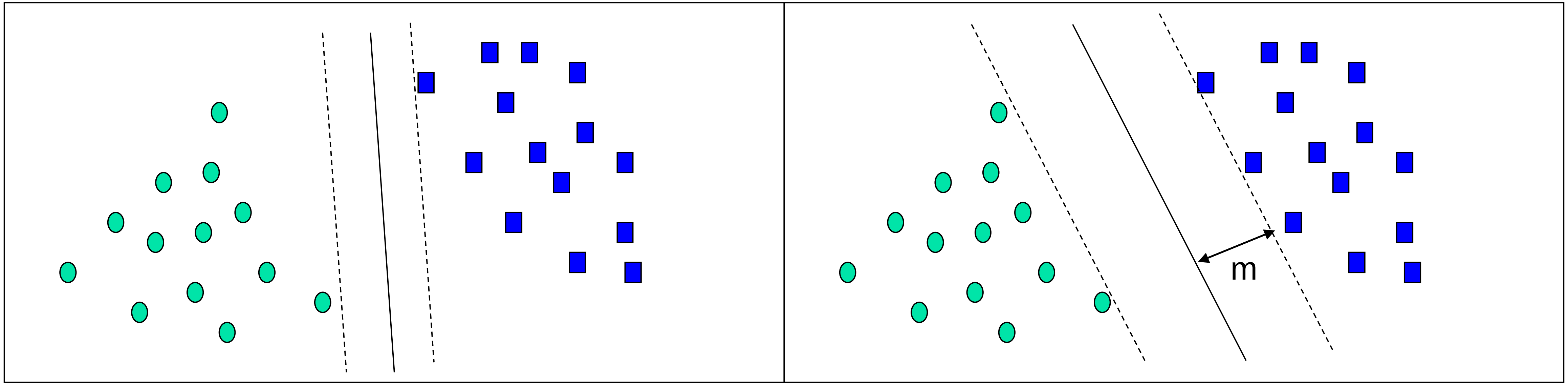
Large Margin

Support Vectors

SVM—Margins and Support Vectors


A 2
A 1
○ class 1, y = +1 (buys_computer = "yes")
● class 2, y = -1 (buys_computer = "no")
small margin
A 2
A 1
○ class 1,
● class 2,
large margin
A 2
A 1
large margin

SVM—当数据线性可分时



D 为($\mathbf{X}_1$, y_1), …, ($\mathbf{X}_{|D|}$, $y_{|D|}$), 其中 $\mathbf{X}_i$ 带标签yi的训练元组

有无数条直线(hyperplanes) 可以分离两个类，但我们需要发现最好的一个(对未知数据有最小化的分类误差)

SVM searches for the hyperplane with the largest margin, i.e., **maximum marginal hyperplane** (MMH)

SVM—线性可分

- 一个分离超平面可以写成

  $$\mathbf{W} \bullet \mathbf{X} + b = 0$$

  $\mathbf{W}=\{w_1, w_2, \ldots, w_n\}$ 权重向量和标量b (bias)

- 对于2-D，可以写成

  $$w_0 + w_1\, x_1 + w_2\, x_2 = 0$$

- 超平面定义了边缘的边界:

  H_1: $w_0 + w_1\, x_1 + w_2\, x_2 \geq 1$ for $y_i = +1$, and

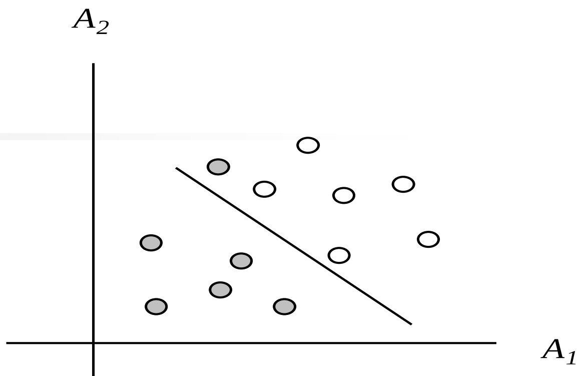
  H_2: $w_0 + w_1\, x_1 + w_2\, x_2 \leq -1$ for $y_i = -1$

- 任何一个位于超平面H_1 or H_2 (i.e., the sides defining the margin) 的样本为 **support vectors**
- **最大边缘是 2/||w||→max**
- 是一个 **constrained (convex) quadratic optimization** problem: 二次目标函数和线性约束 → *Quadratic Programming (QP)* → Lagrangian multipliers

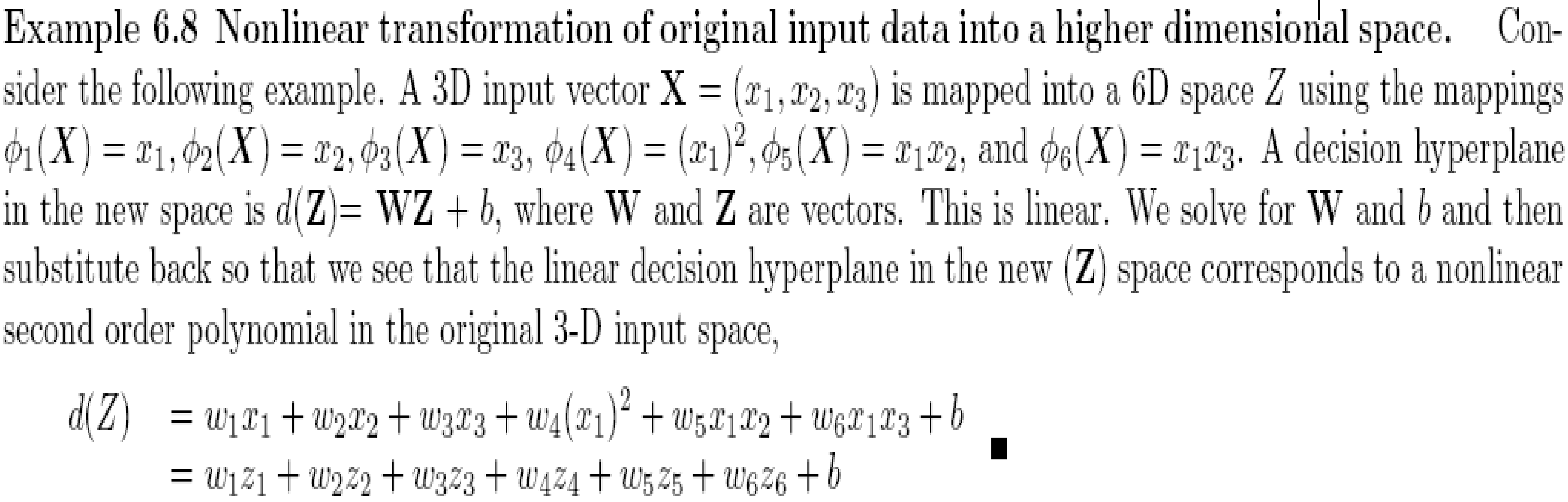
Why Is SVM Effective on High Dimensional Data?

- 训练后的分类器的**complexity**由支持向量数而不是数据维度刻画
- 支持向量**support vectors**是基本的/临界的训练元组—离决策边界最近 (MMH)
- 如果其他的样本删掉后重新训练仍然会发现相同的分离超平面
- 支持向量的数目可用于计算（svm分类器）期望误差率的上界 (upper), 其独立于数据维度
- 一个只有少量支持向量的svm有很好的推广性能, 即使数据的维度很高时

SVM—线性不可分



- 把原始输入数据变换到一个更高维的空间



Example 6.8 Nonlinear transformation of original input data into a higher dimensional space. Consider the following example. A 3D input vector $\mathbf{X} = (x_1, x_2, x_3)$ is mapped into a 6D space Z using the mappings $\phi_1(X) = x_1, \phi_2(X) = x_2, \phi_3(X) = x_3$, $\phi_4(X) = (x_1)^2, \phi_5(X) = x_1x_2$, and $\phi_6(X) = x_1x_3$. A decision hyperplane in the new space is $d(\mathbf{Z})= \mathbf{WZ} + b$, where $\mathbf{W}$ and $\mathbf{Z}$ are vectors. This is linear. We solve for $\mathbf{W}$ and b and then substitute back so that we see that the linear decision hyperplane in the new $(\mathbf{Z})$ space corresponds to a nonlinear second order polynomial in the original 3-D input space,

$$
\begin{aligned}
d(Z) &= w_1x_1 + w_2x_2 + w_3x_3 + w_4(x_1)^2 + w_5x_1x_2 + w_6x_1x_3 + b \\
&= w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4 + w_5z_5 + w_6z_6 + b
\end{aligned}
$$

- Search for a linear separating hyperplane in the new space

SVM: 不同的核函数

- 计算变换后数据的点积, 数学上等价于应用一个核函数K($\mathbf{X}_i$, $\mathbf{X}_j$) 于原始数据, i.e., K($\mathbf{X}_i$, $\mathbf{X}_j$) = Φ($\mathbf{X}_i$) Φ($\mathbf{X}_j$)
- Typical Kernel Functions

$$\begin{aligned} \textbf{Polynomial kernel of degree } h : &\quad K(X_i, X_j) = (X_i \cdot X_j + 1)^h \\ \textbf{Gaussian radial basis function kernel} : &\quad K(X_i, X_j) = e^{-\|X_i - X_j\|^2/2\sigma^2} \\ \textbf{Sigmoid kernel} : &\quad K(X_i, X_j) = \tanh(\kappa X_i \cdot X_j - \delta) \end{aligned}$$

- SVM 也可用于多类数据 (> 2)和回归分析(需要附加参数)

SVM vs. Neural Network

- SVM
  - **Deterministic algorithm**
  - **Nice generalization properties**
  - **Hard to learn – 使用 quadratic programming techniques批量学习**
  - **Using kernels can learn very complex functions**

- Neural Network
  - **Nondeterministic algorithm**
  - **Generalizes well but doesn’t have strong mathematical foundation**
  - **Can easily be learned in incremental fashion**
  - **To learn complex functions—use multilayer perceptron (nontrivial)**

SVM Related Links

- SVM Website: http://www.kernel-machines.org/
- Representative implementations
  - **LIBSVM: an efficient implementation of SVM, multi-class classifications, nu-SVM, one-class SVM, including also various interfaces with java, python, etc.**
  - **SVM-light: simpler but performance is not better than LIBSVM, support only binary classification and only in C**
  - **SVM-torch: another recent implementation also written in C**

Chapter 8. 惰性学习

- Bayesian Belief Networks
- Classification by Backpropagation
- Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- Other Classification Methods
- Additional Topics Regarding Classification
- Summary

Lazy vs. Eager Learning

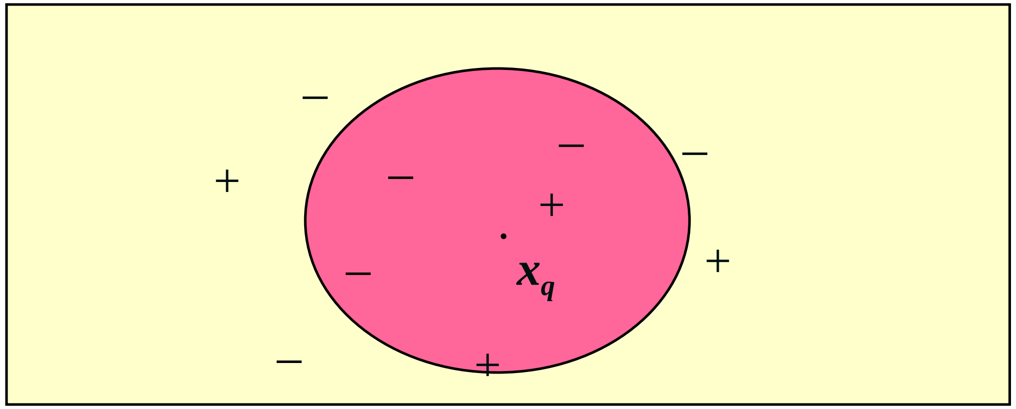
- Lazy vs. eager learning
  - **Lazy learning (e.g.,** 基于实例的学习**):** 仅存储数据 **(**或稍加处理**)** 直到碰到检验元组才开始处理
  - **Eager learning (**前面介绍的方法**):** 给定训练数据**,**在遇到待处理的新数据前构造分类模型
- Lazy: 训练用时很少，预测时用时多
- 准确性
  - 惰性学习方法可以有效地利用更丰富的假设空间，使用多个局部线性函数来对目标函数形成一个隐式的全局逼近
  - **Eager:** 必须限于一个假设，它覆盖了整个实例空间

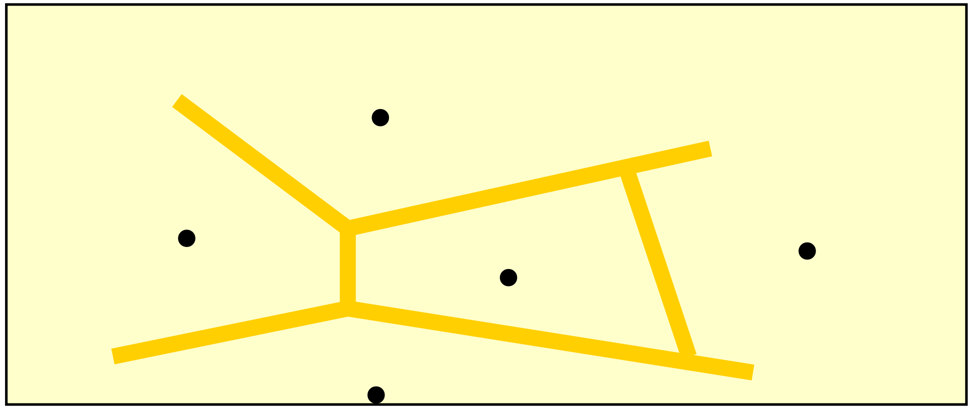
Lazy Learner:基于实例的方法

- Instance-based learning:
  - **Store training examples and delay the processing (“lazy evaluation”) until a new instance must be classified**
- 典型的方法
  - ***k*-nearest neighbor approach**
    - 实例表示为欧氏空间中的点.
  - **Locally weighted regression**
    - **Constructs local approximation**
  - 基于案例的推理**Case-based reasoning**
    - 使用符号表示和知识为基础的推理

k-最近邻算法

- 所有的样本对应于n-D 空间的点
- 通过Euclidean distance, dist(X_1, X_2) 定义最近邻居
- 目标函数可以是discrete- or real- 值
- 对于离散值, *k*-NN 返回与目标元组最近的k个训练样本的多数类
- Vonoroi diagram: the decision surface induced by 1-NN for a typical set of training examples





k-NN Algorithm的讨论

- ***k*-NN：元组的未知实值的预测时**
  - 返回与未知元组***k***个最近邻居的平均值（对应属性）

- **Distance-weighted nearest neighbor algorithm**
  - 根据与目标元组的距离权重组合**k**个近邻的贡献
    - **Give greater weight to closer neighbors**

$$w \equiv \frac{1}{d(x_q, x_i)^2}$$

- **Robust to noisy data by averaging *k*-nearest neighbors**
- **Curse of dimensionality: 邻居间的距离会被无关联的属性影响**
  - 坐标轴伸缩或去除次要的属性

基于案例的推理 (CBR)

- CBR: 使用一个问题解的数据库来求解新问题
- 存储符号描述(tuples or cases)—不是Euclidean space的点
- 应用: 顾客-服务台 (产品有关的诊断), 合法裁决
- Methodology
  - 实例表示为复杂的符号描述(e.g., function graphs)
  - 搜索相似的案例, 组合多个返回的例子
  - Tight coupling between case retrieval, knowledge-based reasoning, and problem solving
- Challenges
  - Find a good similarity metric
  - Indexing based on syntactic similarity measure, and when failure, backtracking, and adapting to additional cases

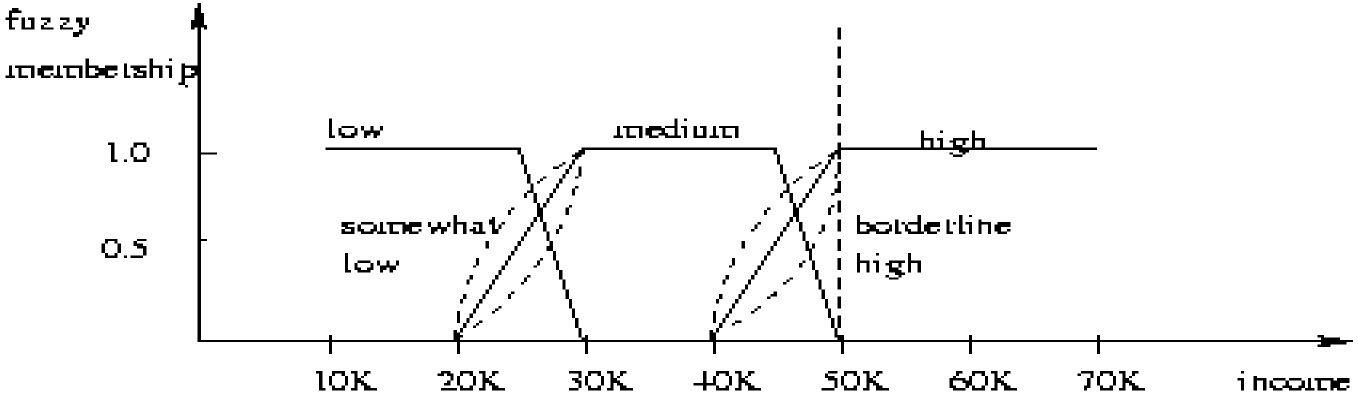
Chapter 8. 分类: 其他方法

- **Bayesian Belief Networks**
- **Classification by Backpropagation**
- **Support Vector Machines**
- **Classification by Using Frequent Patterns**
- **Lazy Learners (or Learning from Your Neighbors)**
- **Other Classification Methods**
- **Additional Topics Regarding Classification**
- **Summary**

遗传算法 (GA)

- Genetic Algorithm: 模仿生物进化
- 使用随机产生的规则组成一个最初的population
  - 每个规则有一系列位表示
  - E.g., if A_1 and ¬A_2 then C_2 can be encoded as 100
  - 如果一个属性有 k > 2 个值, 使用k位
- 基于适者生存原理, 最适合的规则及其后代组成新的种群
- 规则的拟合度用它在训练样本的准确率来评估
- 通过交叉和突变来产生后代
- 此过程持续下去，直到种群P进化到其中的每个规则满足给定的拟合度阈值
- 算法慢，但易于并行

Fuzzy Set Approaches



- Fuzzy logic 使用[0.0, 1.0]真值来表示类的成员的隶属度
- 属性值被转化成模糊值. Ex.:
  - 对于每个离散类别收入{low, medium, high}，*x* 被分配一个模糊的隶属值, e.g. $49K 属于 “medium income” 0.15，属于“high income” 的隶属值是0.96
  - 模糊隶属值的和不一定等于1.
- 每个可用的规则为类的隶属贡献一票
- 通常, 对每个预测分类的真值求和, 并组合这些值

Chapter 8. 分类: Advanced Methods

- Bayesian Belief Networks
- Classification by Backpropagation
- Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- Other Classification Methods
- Additional Topics Regarding Classification
- Summary

多类分类

- 分类时设计多个类别 (i.e., > 2 Classes)
- Method 1. One-vs.-all (OVA): 每次学习一个分类器
  - 给定**m**个类, 训练**m**个分类其，每个类别一个
  - 分类器 **j:** 把类别 **j**的元组定义为 ***positive*** **&** 其他的为 ***negative***
  - 为分类样本 **X,** 所有分类器投票来集成
- Method 2. All-vs.-all (AVA): 为每一对类别学习一个分类器
  - **Given m classes, construct m(m-1)/2 binary classifiers**
  - 使用两个类别的元组训练一个分类器
  - 为分类元组**X,** 每个分类器投票. **X is assigned to the class with maximal vote**
- Comparison
  - **All-vs.-all tends to be superior to one-vs.-all**
  - **Problem: Binary classifier is sensitive to errors, and errors affect vote count**

多类分类的Error-Correcting Codes

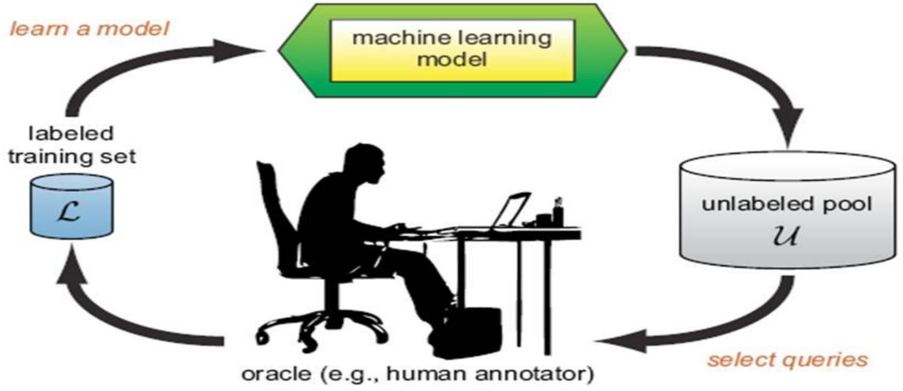
Class	Error-Corr. Codeword						
C_1	1	1	1	1	1	1	1
C_2	0	0	0	0	1	1	1
C_3	0	0	1	1	0	0	1
C_4	0	1	0	1	0	1	0

- 最初目的是在数据传输的通讯任务中通过探索数据冗余来修正误差。例:
  - **A 7-bit codeword associated with classes 1-4**
  - 给定未知元组**X**, 7个分类器的结果为: 0001010
  - Hamming distance: # 两个码字间不同位数的和
  - H(**X**, C_1) = 5, 检查 [1111111] & [0001010]间不同位数和
  - H(**X**, C_2) = 3, H(**X**, C_3) = 3, H(**X**, C_4) = 1, thus C_4 as the label for **X**
- Error-correcting codes can correct up to (h-1)/h 1-bit error, where h is the minimum Hamming distance between any two codewords
- If we use 1-bit per class, it is equiv. to one-vs.-all approach, the code are insufficient to self-correct
- When selecting error-correcting codes, there should be good row-wise and col.-wise separation between the codewords

半监督分类

- Semi-supervised: 使用有标签和无标签数据构造分类器
- Self-training:
  - **Build a classifier using the labeled data**
  - **Use it to label the unlabeled data, and those with the most confident label prediction are added to the set of labeled data**
  - 重复以上过程
  - **Adv:** 容易理解**; disadv:** 可能增大误差
- Co-training: Use two or more classifiers to teach each other
  - 每个学习者使用元组的相互独立的特征集合来训练一个好的分类器**F1**
  - 然后 **f_1 and f_2** 用来预测未知元组 **X** 的类别标签
  - **Teach each other: The tuple having the most confident prediction from f_1 is added to the set of labeled data for f_2, & vice versa**
- Other methods, e.g., joint probability distribution of features and labels

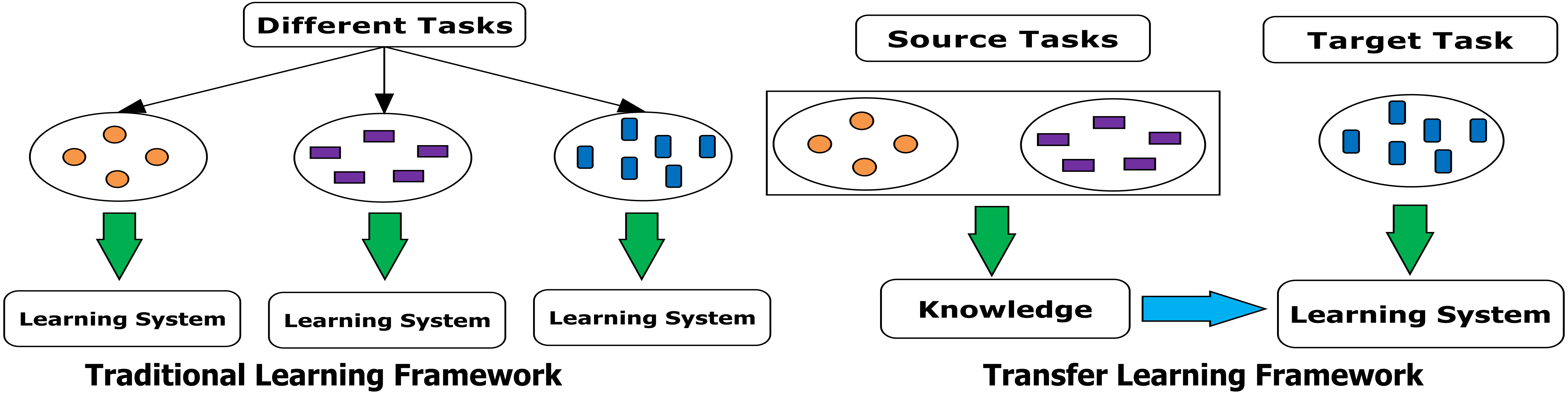
主动学习



- 获取类标签是昂贵
- Active learner: query human (oracle) for labels
- Pool-based approach: Uses a pool of unlabeled data
  - **L: D中有标签的样本子集, U: D的一个未标记数据集**
  - 使用一个查询函数小心地从**U**选择**1**或多个元组，并咨询标签**an oracle (a human annotator)**
  - **The newly labeled samples are added to L, and learn a model**
  - **Goal: Achieve high accuracy using as few labeled data as possible**
- Evaluated using *learning curves*: Accuracy as a function of the number of instances queried (# of tuples to be queried should be small)
- Research issue: How to choose the data tuples to be queried?
  - **Uncertainty sampling: choose the least certain ones**
  - **Reduce *version space*, the subset of hypotheses consistent w. the training data**
  - **Reduce expected entropy over U: Find the greatest reduction in the total number of incorrect predictions**

迁移学习：概念框架

- Transfer learning: Extract knowledge from one or more source tasks and apply the knowledge to a target task
- Traditional learning:每一个任务建立分类器
- Transfer learning: Build new classifier by applying existing knowledge learned from source tasks



迁移学习: Methods and Applications

- 应用:数据过时或分布的变化时, e.g., Web document classification, e-mail spam filtering
- *Instance-based transfer learning*: Reweight some of the data from source tasks and use it to learn the target task
- TrAdaBoost (Transfer AdaBoost)
  - 假定源和目标数据用相同的属性和类别描述, but rather diff. distributions
  - Require only labeling a small amount of target data
  - 训练中使用源数据: When a source tuple is misclassified, reduce the weight of such tupels so that they will have less effect on the subsequent classifier
- Research issues
  - Negative transfer: When it performs worse than no transfer at all
  - Heterogeneous transfer learning: Transfer knowledge from different feature space or multiple source domains
  - Large-scale transfer learning

Chapter 8. 分类:频繁模式

- Bayesian Belief Networks
- Classification by Backpropagation
- Support Vector Machines
- Classification by Using Frequent Patterns
- Lazy Learners (or Learning from Your Neighbors)
- Other Classification Methods
- Additional Topics Regarding Classification
- Summary

关联分类

- 关联分类: 主要步骤
  - 挖掘关于频繁模式(属性-值对的联结) 和类标签间的强关联
  - 产生以下形似的关联规则

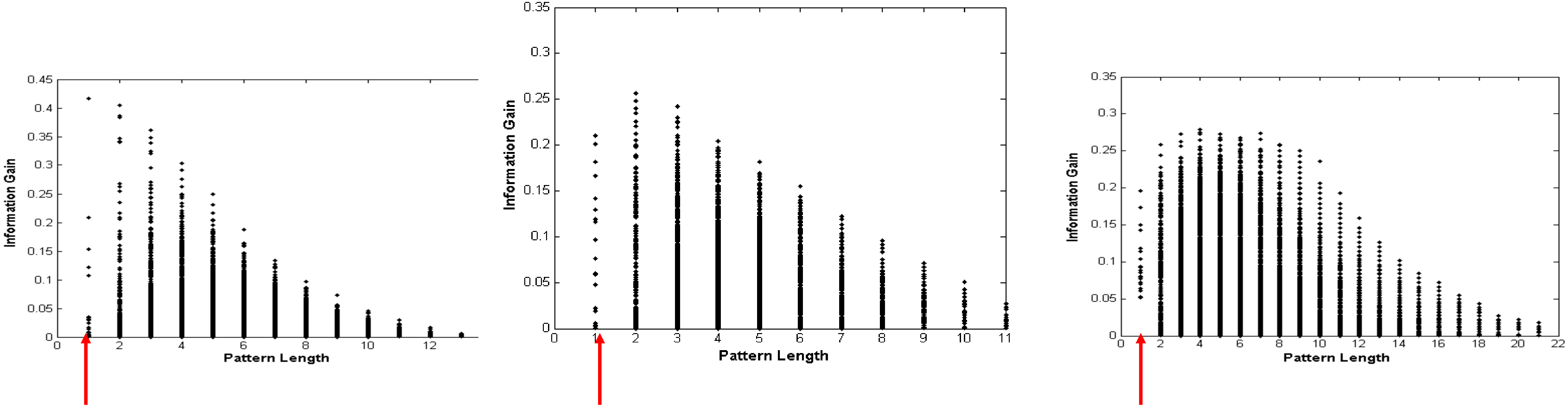
    $P_1 \wedge p_2 \ldots \wedge p_l \rightarrow$ "A_{class} = C" (conf, sup)
  - 组织规则，形成基于规则的分类器
- 为什么有效?
  - 可以发现（在多个属性间）高置信度的关联，可以克服决策树规约引入的约束, 决策树一次考虑一个属性
  - 研究发现，关联分类通常比某些传统的分类方法更精确, 例如C4.5

典型的关联分类方法

- CBA (Classification Based on Associations: Liu, Hsu & Ma, KDD’98)
  - 挖掘可能关联规则：**Cond-set (属性-值 的集合) → class label**
  - 建立分类器**:** 基于置信度和支持度的下降序组织规则
- CMAR (Classification based on Multiple Association Rules: Li, Han, Pei, ICDM’01)
  - 分类**:** 多个规则的统计分析
- CPAR (Classification based on Predictive Association Rules: Yin & Han, SDM’03)
  - 产生预测性规则 **(FOIL-like analysis)** 允许覆盖的元组以降低权重形式保留下来构造新规则
  - （根据期望准确率）使用最好的**k** 个规则预测
  - 更有效（产生规则少）**,** 精确性类似**CMAR**

频繁模式 vs. 单个特征

某些频繁模式的判别能力高于单个特征.



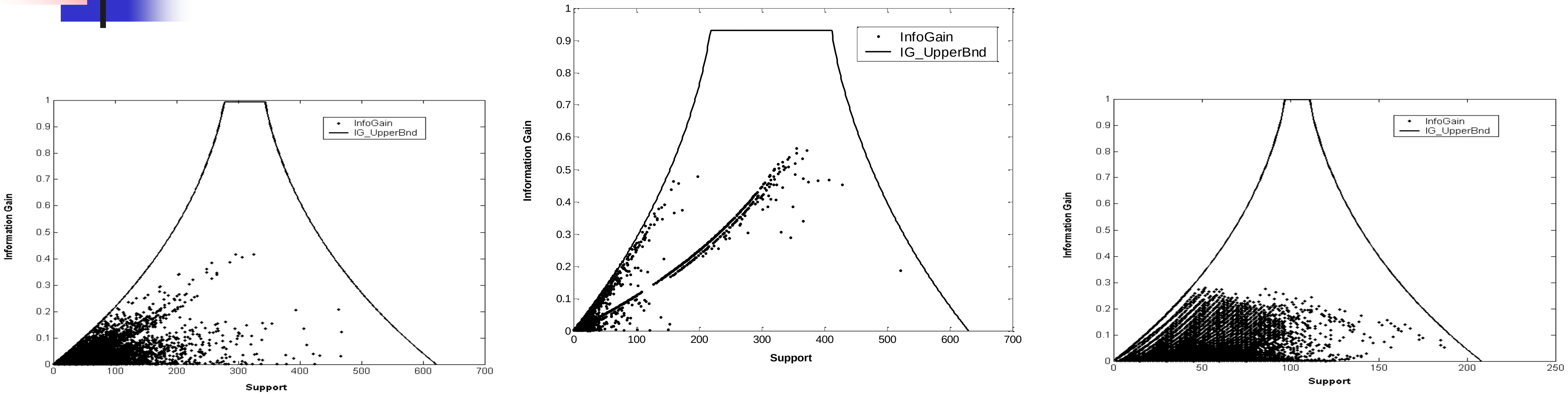
(a) Austral

(b) Cleve

(c) Sonar

Fig. 1. Information Gain vs. Pattern Length

经验结果



(a) Austral

(b) Breast

(c) Sonar

Fig. 2. Information Gain vs. Pattern Frequency

特征选择Feature Selection

- 给定频繁模式集合, 存在non-discriminative和redundant 的模式, 他们会引起过度拟合
- 我们希望选出discriminative patterns，并且去除冗余
- 借用Maximal Marginal Relevance (MMR)的概念
  - A document has high marginal relevance if it is both relevant to the query and contains minimal marginal similarity to previously selected documents

实验结果

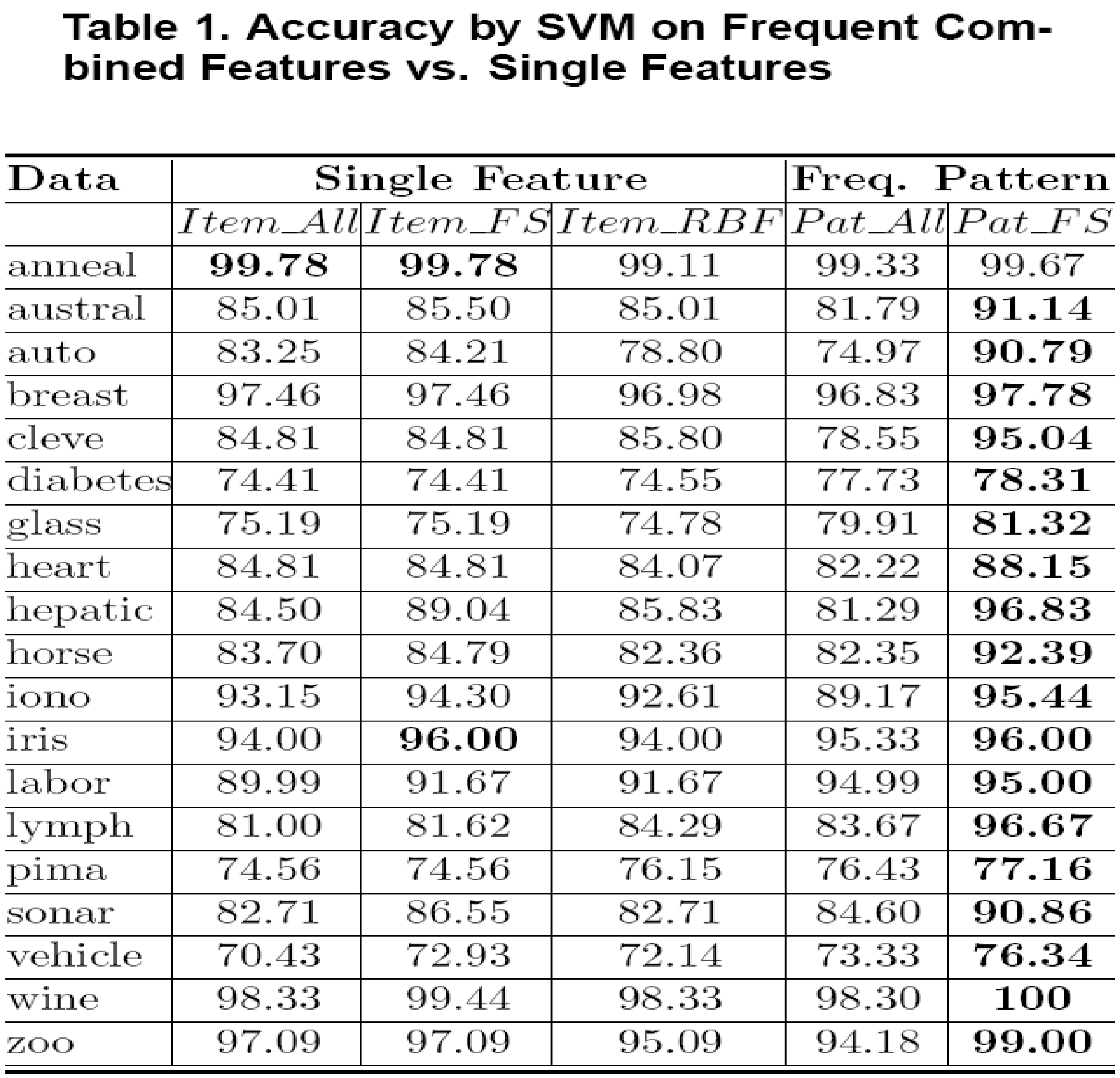


Table 1. Accuracy by SVM on Frequent Combined Features vs. Single Features

Data	Single Feature			Freq. Pattern	
	Item_All	*Item_FS*	*Item_RBF*	*Pat_All*	*Pat_FS*
anneal	**99.78**	**99.78**	99.11	99.33	99.67
austral	85.01	85.50	85.01	81.79	**91.14**
auto	83.25	84.21	78.80	74.97	**90.79**
breast	97.46	97.46	96.98	96.83	**97.78**
cleve	84.81	84.81	85.80	78.55	**95.04**
diabetes	74.41	74.41	74.55	77.73	**78.31**
glass	75.19	75.19	74.78	79.91	**81.32**
heart	84.81	84.81	84.07	82.22	**88.15**
hepatic	84.50	89.04	85.83	81.29	**96.83**
horse	83.70	84.79	82.36	82.35	**92.39**
iono	93.15	94.30	92.61	89.17	**95.44**
iris	94.00	**96.00**	94.00	95.33	**96.00**
labor	89.99	91.67	91.67	94.99	**95.00**
lymph	81.00	81.62	84.29	83.67	**96.67**
pima	74.56	74.56	76.15	76.43	**77.16**
sonar	82.71	86.55	82.71	84.60	**90.86**
vehicle	70.43	72.93	72.14	73.33	**76.34**
wine	98.33	99.44	98.33	98.30	**100**
zoo	97.09	97.09	95.09	94.18	**99.00**

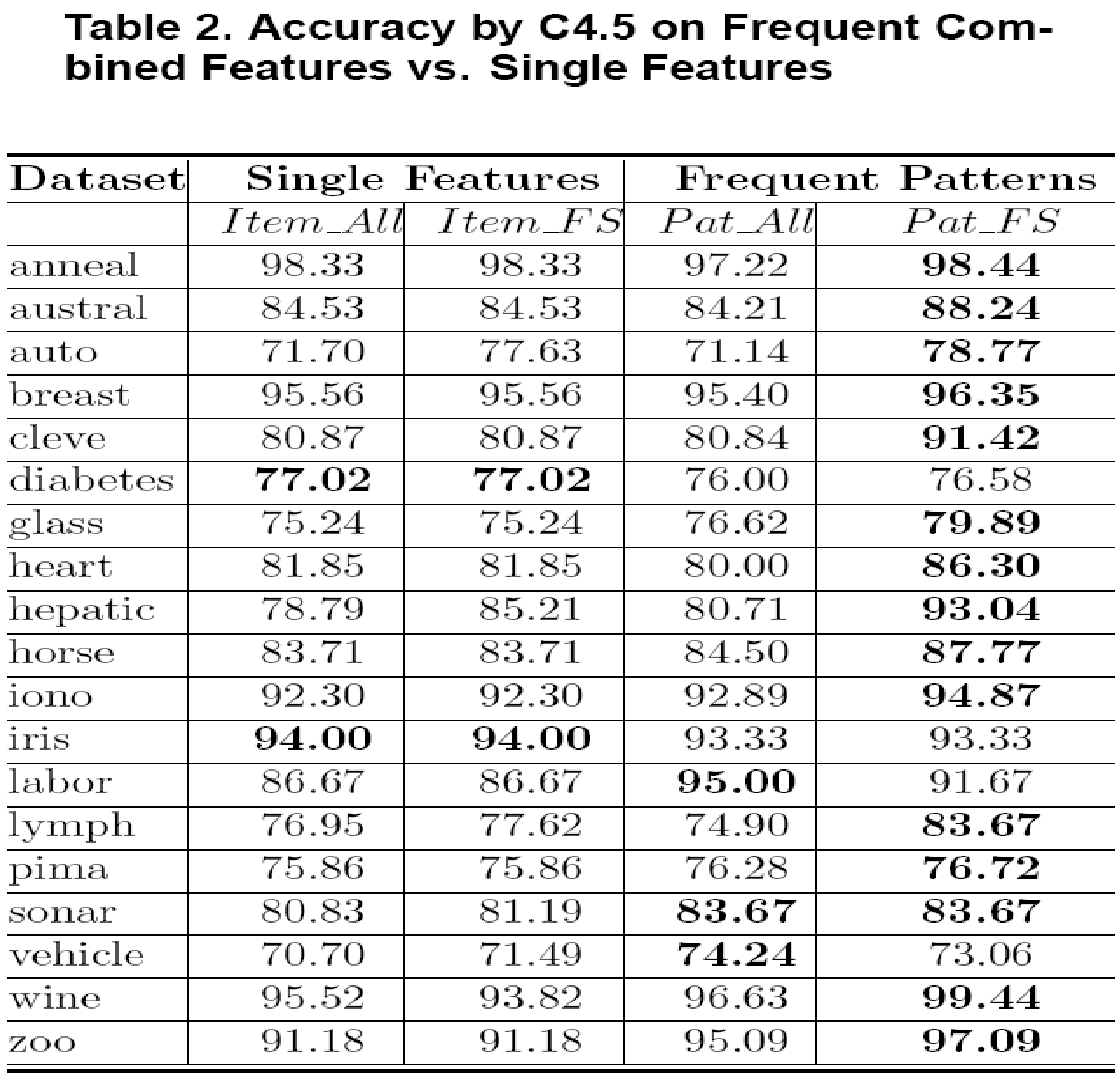
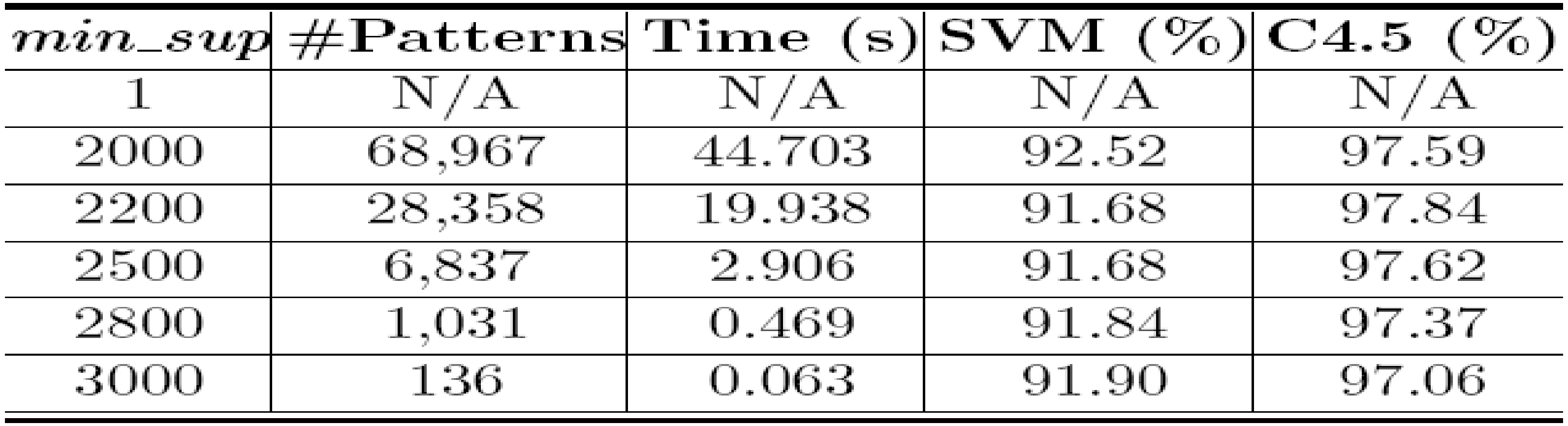


Table 2. Accuracy by C4.5 on Frequent Combined Features vs. Single Features

Dataset	Single Features		Frequent Patterns	
	Item_All	*Item_FS*	*Pat_All*	*Pat_FS*
anneal	98.33	98.33	97.22	**98.44**
austral	84.53	84.53	84.21	**88.24**
auto	71.70	77.63	71.14	**78.77**
breast	95.56	95.56	95.40	**96.35**
cleve	80.87	80.87	80.84	**91.42**
diabetes	**77.02**	**77.02**	76.00	76.58
glass	75.24	75.24	76.62	**79.89**
heart	81.85	81.85	80.00	**86.30**
hepatic	78.79	85.21	80.71	**93.04**
horse	83.71	83.71	84.50	**87.77**
iono	92.30	92.30	92.89	**94.87**
iris	**94.00**	**94.00**	93.33	93.33
labor	86.67	86.67	**95.00**	91.67
lymph	76.95	77.62	74.90	**83.67**
pima	75.86	75.86	76.28	**76.72**
sonar	80.83	81.19	**83.67**	**83.67**
vehicle	70.70	71.49	**74.24**	73.06
wine	95.52	93.82	96.63	**99.44**
zoo	91.18	91.18	95.09	**97.09**

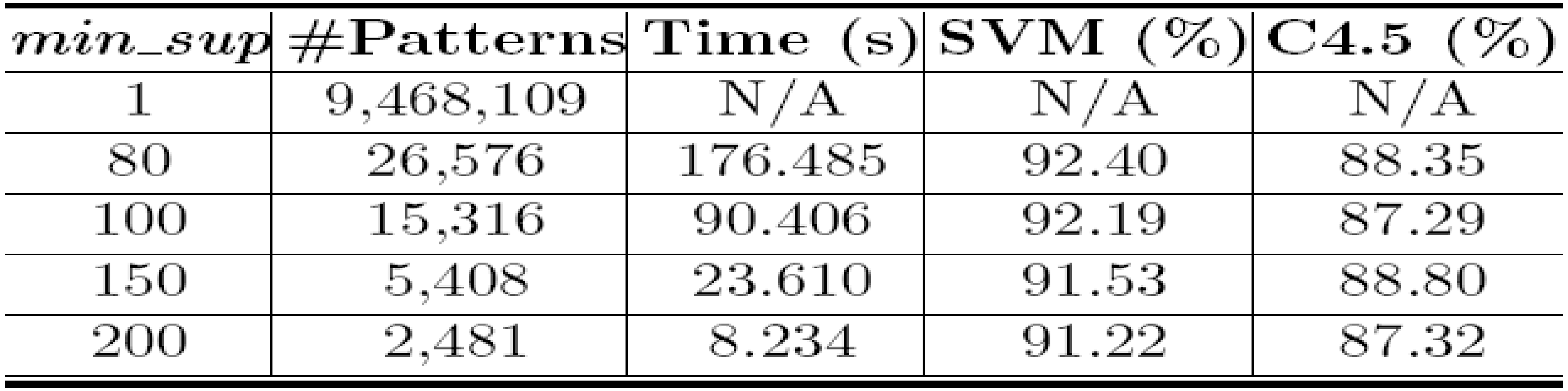
Scalability Tests

Table 3. Accuracy & Time on Chess Data



min_sup	#Patterns	Time (s)	SVM (%)	C4.5 (%)
1	N/A	N/A	N/A	N/A
2000	68,967	44.703	92.52	97.59
2200	28,358	19.938	91.68	97.84
2500	6,837	2.906	91.68	97.62
2800	1,031	0.469	91.84	97.37
3000	136	0.063	91.90	97.06

Table 4. Accuracy & Time on Waveform Data

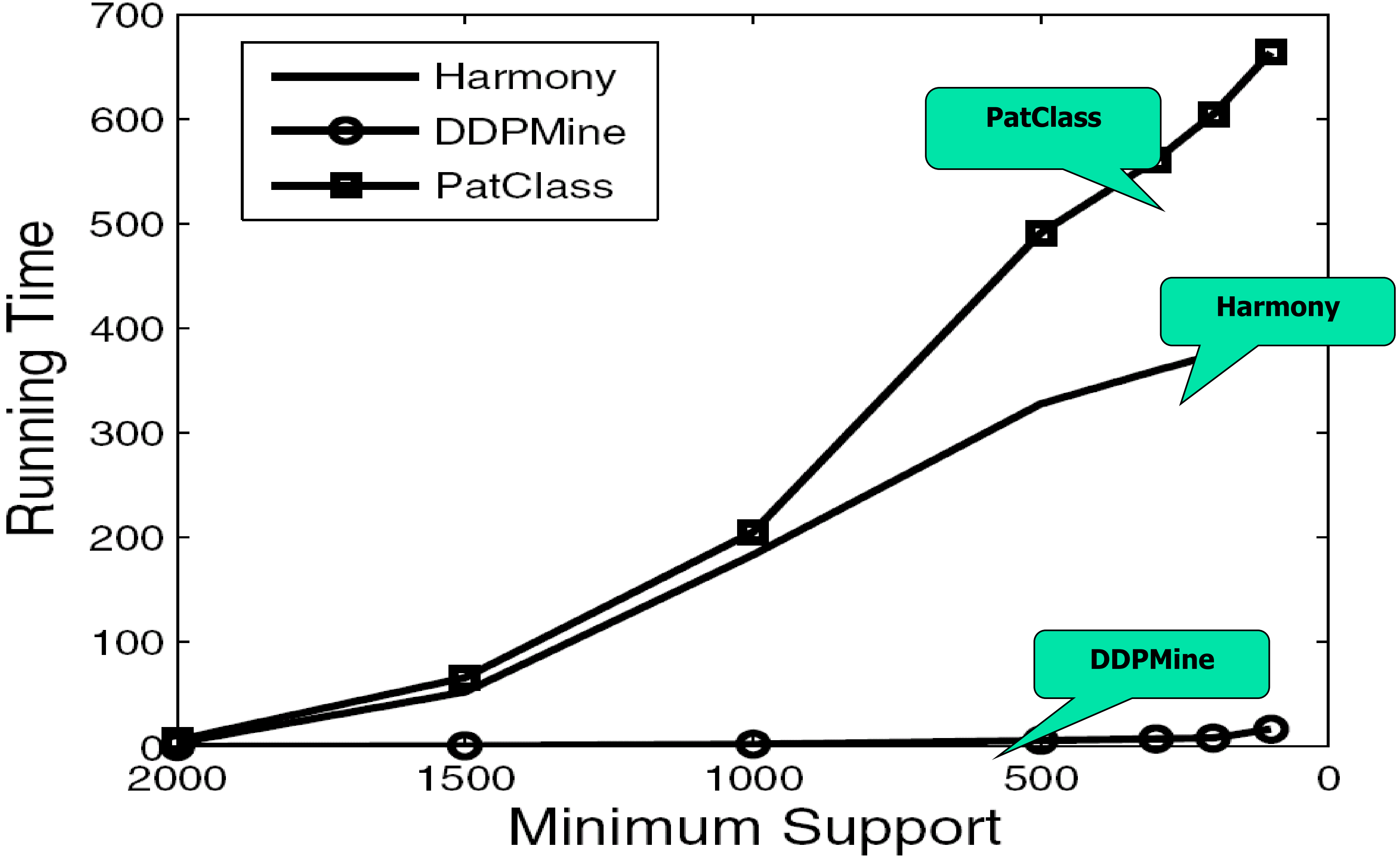


min_sup	#Patterns	Time (s)	SVM (%)	C4.5 (%)
1	9,468,109	N/A	N/A	N/A
80	26,576	176.485	92.40	88.35
100	15,316	90.406	92.19	87.29
150	5,408	23.610	91.53	88.80
200	2,481	8.234	91.22	87.32

基于频繁模式的分类

- H. Cheng, X. Yan, J. Han, and C.-W. Hsu, "Discriminative Frequent Pattern Analysis for Effective Classification", ICDE'07
- Accuracy issue问题
  - Increase the discriminative power
  - Increase the expressive power of the feature space
- Scalability issue问题
  - It is computationally infeasible to generate all feature combinations and filter them with an information gain threshold
  - Efficient method (DDPMine: FPtree pruning): H. Cheng, X. Yan, J. Han, and P. S. Yu, "Direct Discriminative Pattern Mining for Effective Classification", ICDE'08

DDPMine Efficiency: Runtime



PatClass: ICDE'07 Pattern Classification Alg.

Summary

- Effective and advanced classification methods
  - **Bayesian belief network (probabilistic networks)**
  - **Backpropagation (Neural networks)**
  - **Support Vector Machine (SVM)**
  - **Pattern-based classification**
  - **Other classification methods: lazy learners (KNN, case-based reasoning), genetic algorithms, rough set and fuzzy set approaches**
- Additional Topics on Classification
  - **Multiclass classification**
  - **Semi-supervised classification**
  - **Active learning**
  - **Transfer learning**

References (1)

- C. M. Bishop, Neural Networks for Pattern Recognition. Oxford University Press, 1995
- C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2): 121-168, 1998
- H. Cheng, X. Yan, J. Han, and C.-W. Hsu, Discriminative Frequent pattern Analysis for Effective Classification, ICDE'07
- H. Cheng, X. Yan, J. Han, and P. S. Yu, Direct Discriminative Pattern Mining for Effective Classification, ICDE'08
- N. Cristianini and J. Shawe-Taylor, Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University Press, 2000
- A. J. Dobson. An Introduction to Generalized Linear Models. Chapman & Hall, 1990
- G. Dong and J. Li. Efficient mining of emerging patterns: Discovering trends and differences. KDD'99

References (2)

- R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification, 2ed. John Wiley, 2001
- T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, 2001
- S. Haykin, Neural Networks and Learning Machines, Prentice Hall, 2008
- D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. Machine Learning, 1995.
- V. Kecman, Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic, MIT Press, 2001
- W. Li, J. Han, and J. Pei, CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules, ICDM'01
- T.-S. Lim, W.-Y. Loh, and Y.-S. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. Machine Learning, 2000

References (3)

- B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining, p. 80-86, KDD’98.
- T. M. Mitchell. Machine Learning. McGraw Hill, 1997.
- D.E. Rumelhart, and J.L. McClelland, editors, Parallel Distributed Processing, MIT Press, 1986.
- P. Tan, M. Steinbach, and V. Kumar. Introduction to Data Mining. Addison Wesley, 2005.
- S. M. Weiss and N. Indurkhya. Predictive Data Mining. Morgan Kaufmann, 1997.
- I. H. Witten and E. Frank. Data Mining: Practical Machine Learning Tools and Techniques, 2ed. Morgan Kaufmann, 2005.
- X. Yin and J. Han. CPAR: Classification based on predictive association rules. SDM'03
- H. Yu, J. Yang, and J. Han. Classifying large data sets using SVM with hierarchical clusters. KDD'03.

SVM—Introductory Literature

- “Statistical Learning Theory” by Vapnik: extremely hard to understand, containing many errors too.
- C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining*, 2(2), 1998.
  - **Better than the Vapnik’s book, but still written too hard for introduction, and the examples are so not-intuitive**
- The book “An Introduction to Support Vector Machines” by N. Cristianini and J. Shawe-Taylor
  - **Also written hard for introduction, but the explanation about the mercer’s theorem is better than above literatures**
- The neural network book by Haykins
  - **Contains one nice chapter of SVM introduction**

Notes about SVM—Introductory Literature

- "Statistical Learning Theory" by Vapnik: difficult to understand, containing many errors.
- C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining*, 2(2), 1998.
  - **Easier than Vapnik's book, but still not introductory level; the examples are not so intuitive**
- The book An Introduction to Support Vector Machines by Cristianini and Shawe-Taylor
  - **Not introductory level, but the explanation about Mercer's Theorem is better than above literatures**
- Neural Networks and Learning Machines by Haykin
  - **Contains a nice chapter on SVM introduction**