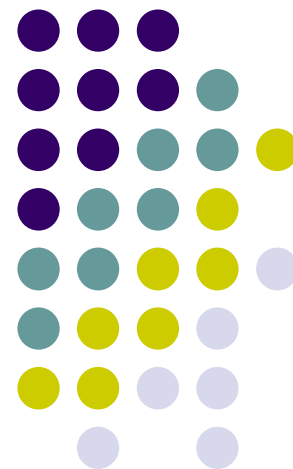


# 机器学习



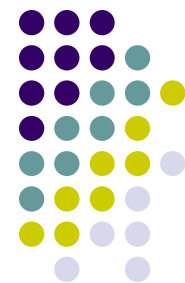


# 什么是机器学习？

- 人工智能大师Herb Simon这样定义学习：

学习：系统在不断重复的工作中对本身能力的增强或改进，使得系统在下一次执行相同任务或类似任务（指的是具有相同分布的任务）时，比现在做的更好或效率更高。

机器学习：通过经验提高系统自身的性能的过程（系统自我改进）。



# 机器学习的重要性

- 机器学习是人工智能的主要核心研究领域之一,也是现代智能系统的关键环节和瓶颈。
- 很难想象:一个没有学习功能的系统是能被称为是具有智能的系统。

# 实例1: 网络安全问题

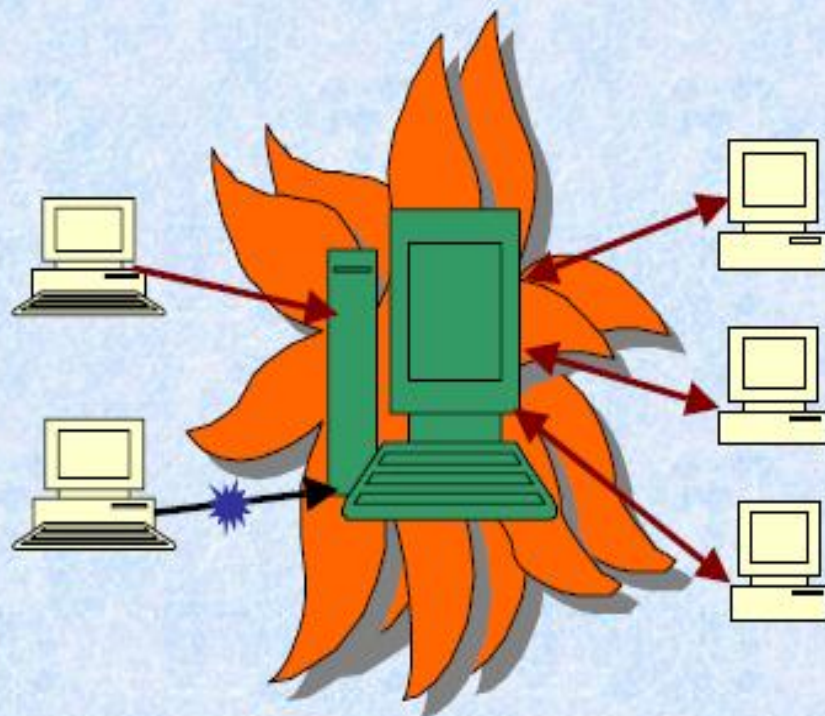
## 入侵检测系统IDS:

是否是入侵? 是何种入侵?

## 如何检测?

- 历史数据: 以往的正常访问模式及其表现、以往的入侵模式及其表现.....

- 对当前访问模式分类



这是一个典型的机器学习问题

常用技术:

神经网络, 决策树, 支持向量机, 贝叶斯分类器, k近邻, 聚类, 序列分析, 免疫网络等 .....



# 实例2: 天气预报

## 如何预测？

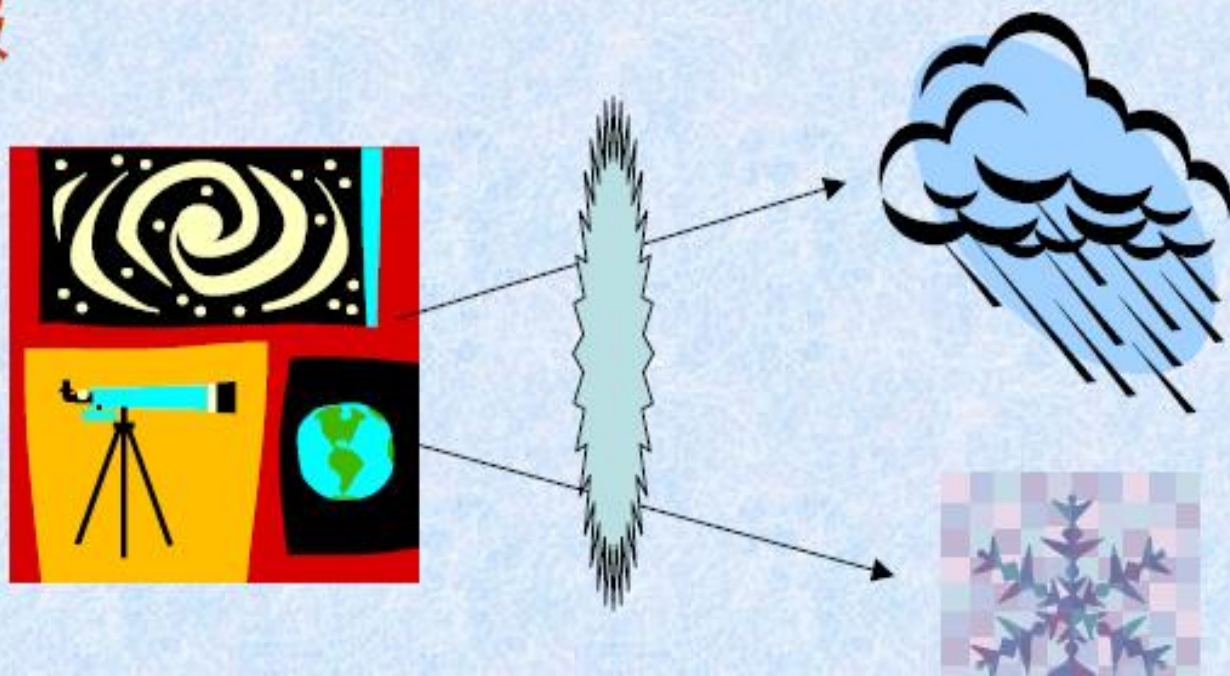
- 气象历史数据：以往的各种气象数据及其变化特点.....
- 对未来一周天气的预报

这是一个典型的机器学习问题

常用技术：

统计多参数模型，神经网络，  
贝叶斯分类器，k近邻，聚类

.....





## 实例3: 搜索引擎



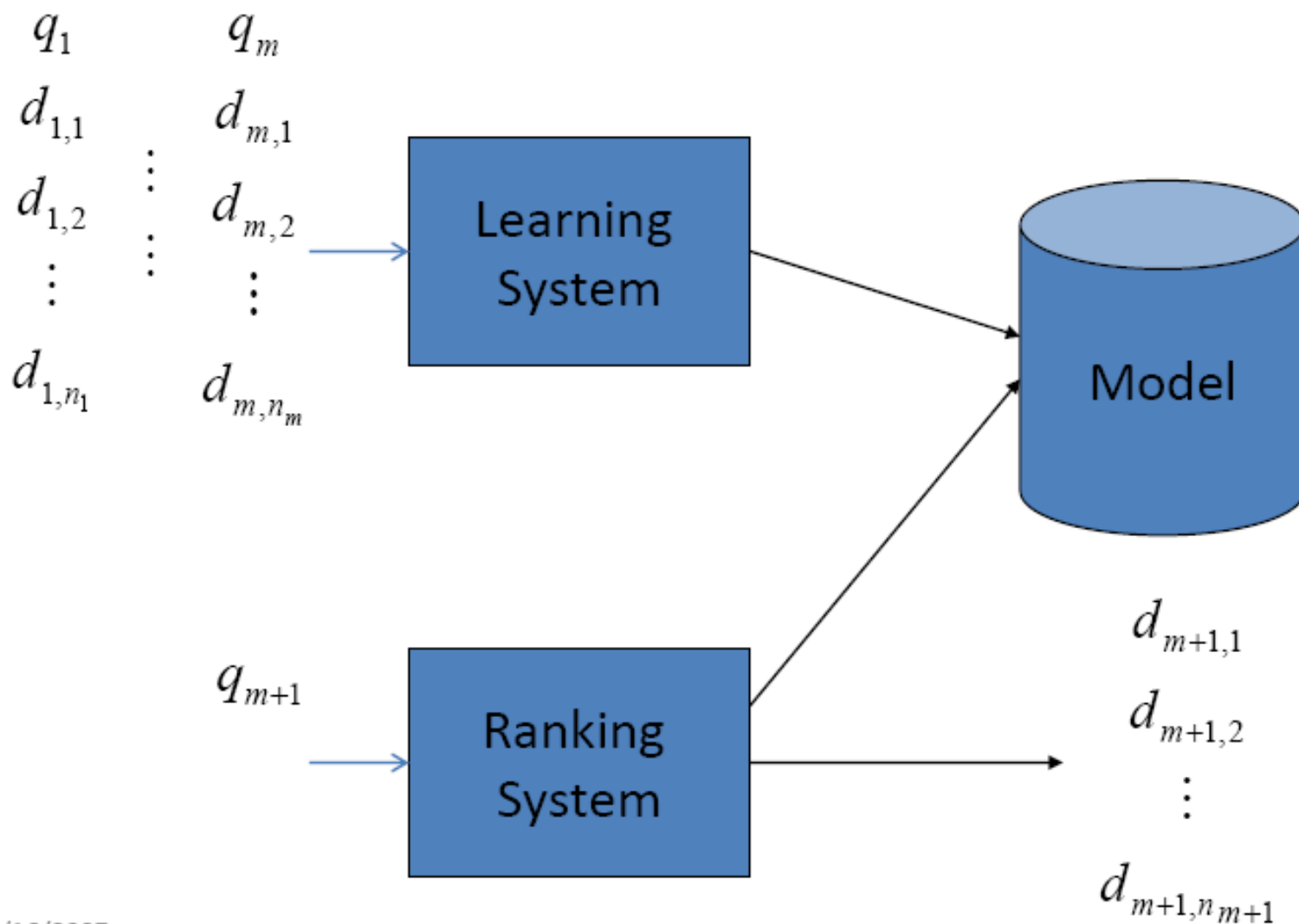
Google的成功，使得Internet搜索引擎成为一个新兴的产业出现有众多专营搜索引擎的公司（例如百度等专门针对中文搜索），而且Microsoft等巨头也开始投入巨资进行搜索引擎的研发

Google掘到的第一桶金，来源于其创始人Larry Page和Sergey Brin提出的PageRank算法

机器学习技术正在支撑着各类搜索引擎

（尤其是贝叶斯学习技术）

# 信息检索 (Information Retrieval)

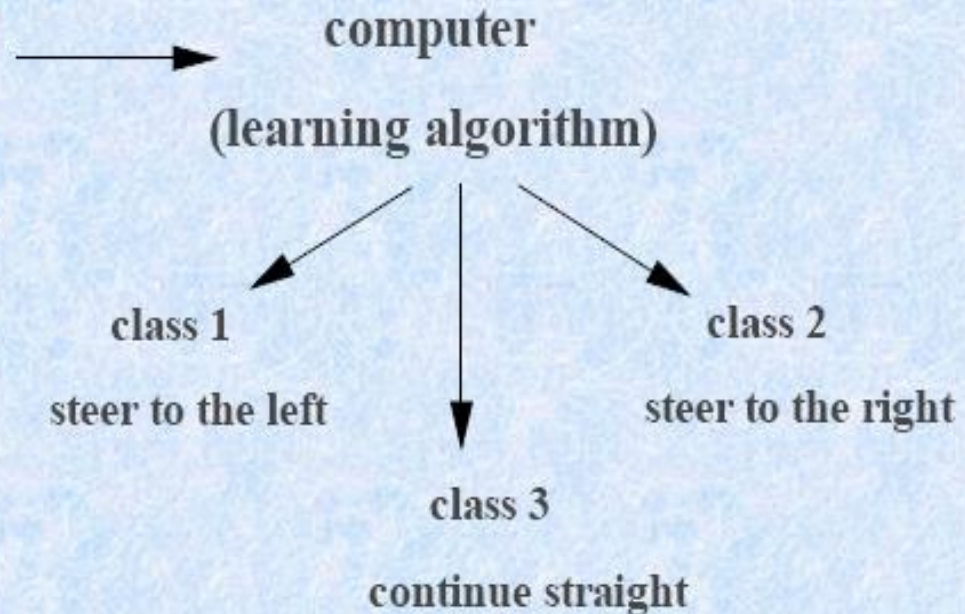




# Application 4

## Automatic car drive (ALVINN 1989)

Train computer-controlled vehicle to steer correctly when driving on a variety of road types.





## Application 4

### *Automatic Car Drive*

*Class of Tasks:* Learning to drive on highways from vision stereos.

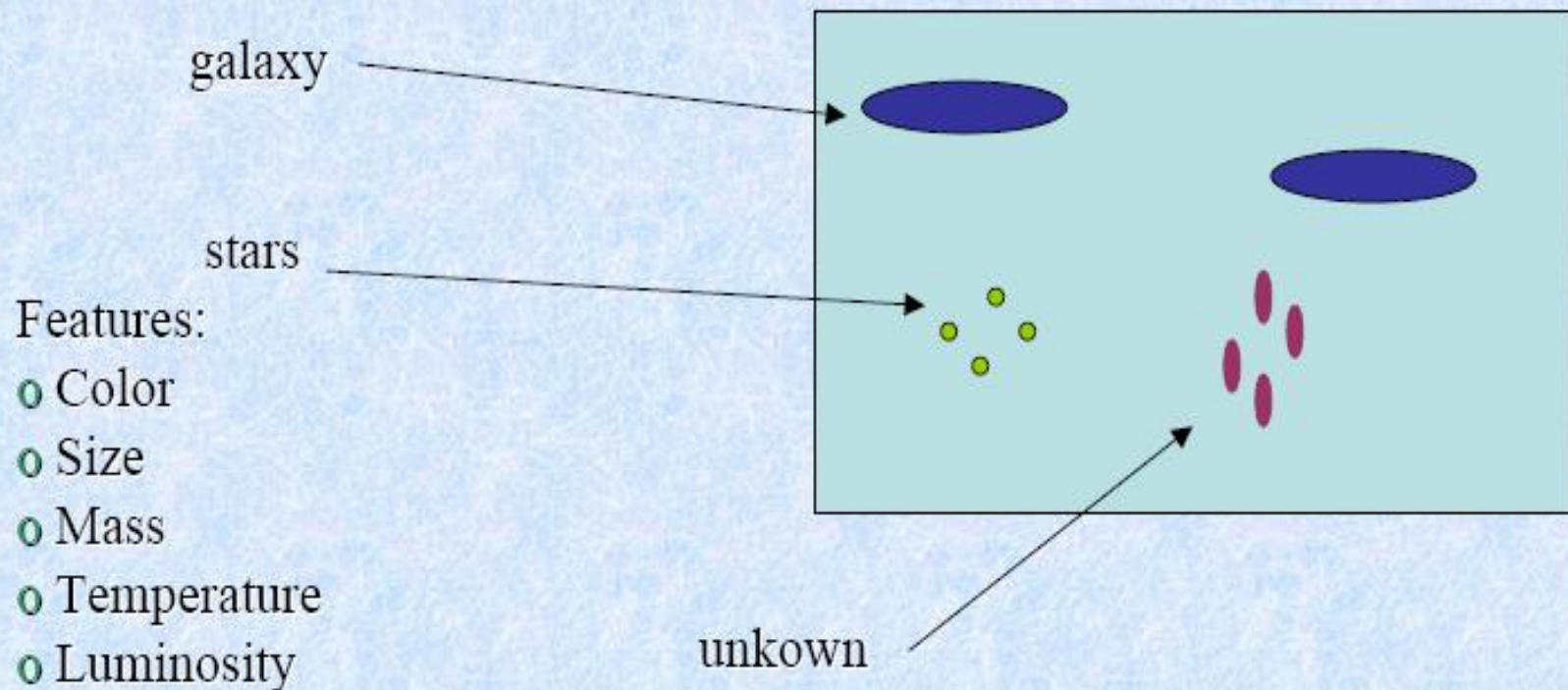
*Knowledge:* Images and steering commands recorded while observing a human driver.

*Performance Module:* Accuracy in classification



# Application 5

Learning to classify astronomical structures.





## Application 5

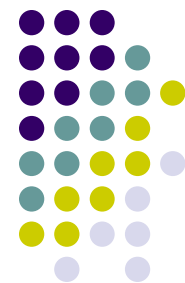
### *Classifying Astronomical Objects*

*Class of Tasks:* Learning to classify new objects.

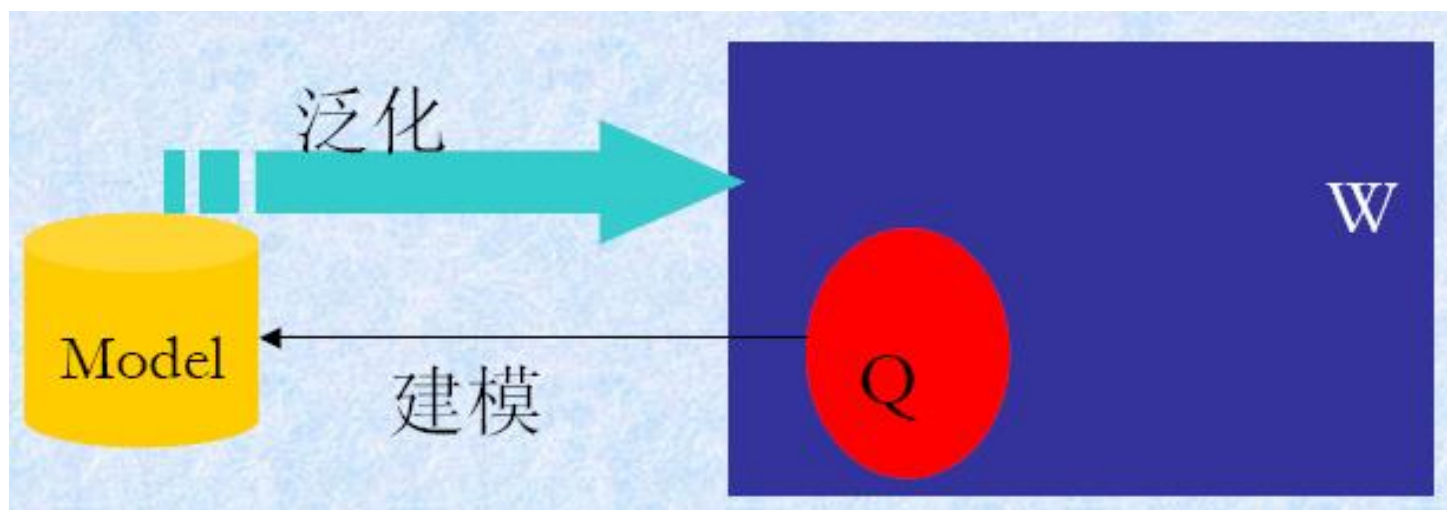
*Knowledge:* database of images with correct classification.

*Performance Module:* Accuracy in classification

# 机器学习的任务



- 令 $W$ 是这个给定世界的有限或无限所有对象的集合，由于观察能力的限制，我们只能获得这个世界的一个有限的子集 $Q \subset W$ ，称为样本集。
- 机器学习就是根据这个有限样本集 $Q$ ，推算这个世界的模型，使得其对这个世界为真。





# 机器学习的三要素



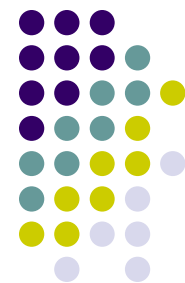
- 一致性假设：机器学习的条件。
- 样本空间划分：决定模型对样本集合的有效性。
- 泛化能力：决定模型对世界的有效性。



# 要素1：一致性假设

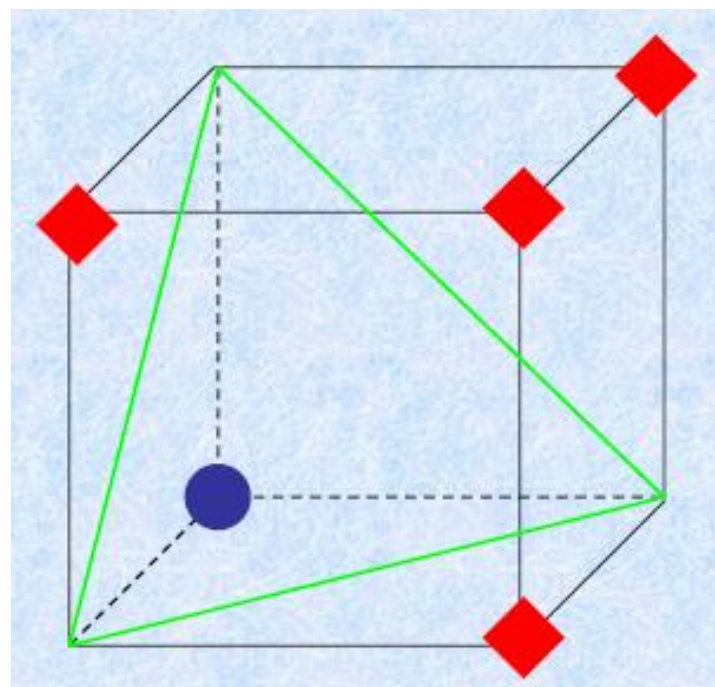
- 假设世界 $W$ 与样本集 $Q$ 具有某种相同的性质。
- 原则上说，存在各种各样的一致性假设。
- 在统计意义下，一般假设：
  - $W$ 与 $Q$ 具有同分布。或，
  - 给定世界 $W$ 的所有对象独立同分布。





## 要素2：对样本空间的划分

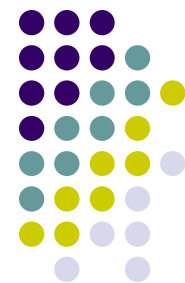
- 样本集合模型：  
将样本集放到一个 $n$ 维空间，寻找一个超平面(等价关系)，使得问题决定的不同对象被划分在不相交的区域。





## 要素3： 泛化能力

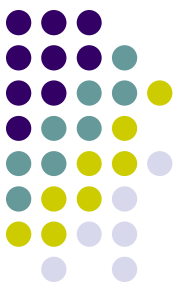
- 泛化能力：学习的目的是学到隐含在数据对背后的规律,对具有同一规律的学习集以外的数据,该神经网络仍具有正确的响应能力,称为泛化能力。
- 通过机器学习方法，从给定有限样本集合计算一个模型，泛化能力是这个模型对世界为真程度的指标。



# 关于三要素

- 不同时期，研究的侧重点不同
- 划分:早期研究主要集中在该要素上
- 泛化能力(在多项式划分):80年代以来的近期研究
- 一致性假设:未来必须考虑 ([Transfer learning](#))





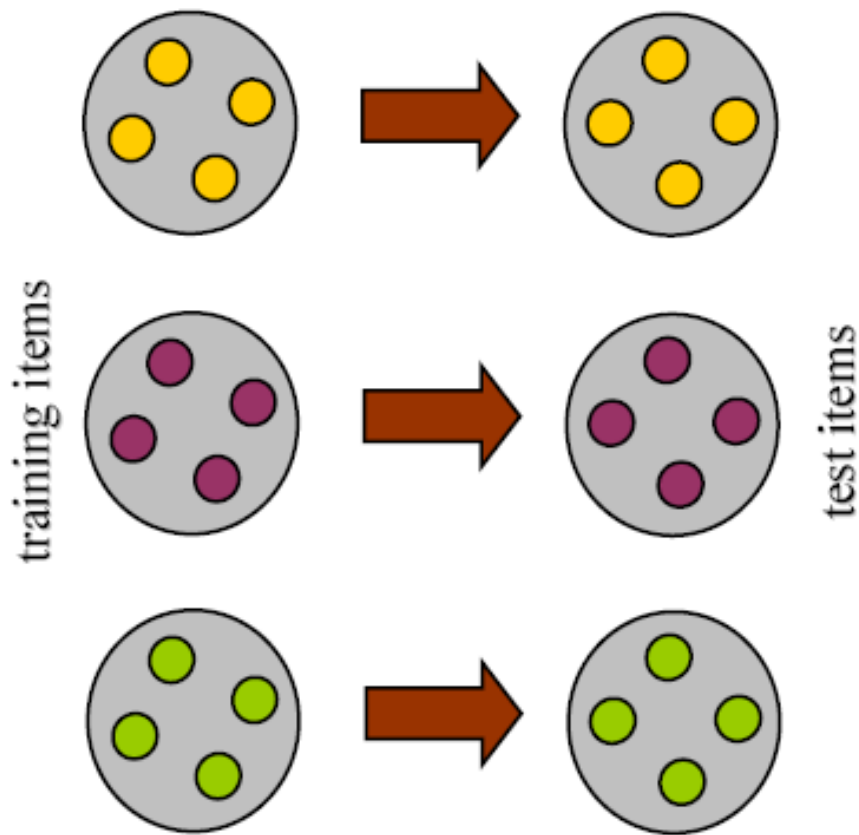
# Transfer learning

- **Transfer learning** 这一概念是由 **DARPA**（美国国防高级研究计划局）在2005年正式提出来的一项研究计划。
- **Transfer Learning** 是指系统能够将在先前任务中学到的知识或技能应用于一个新的任务或新的领域。

# 传统机器学习&转移学习

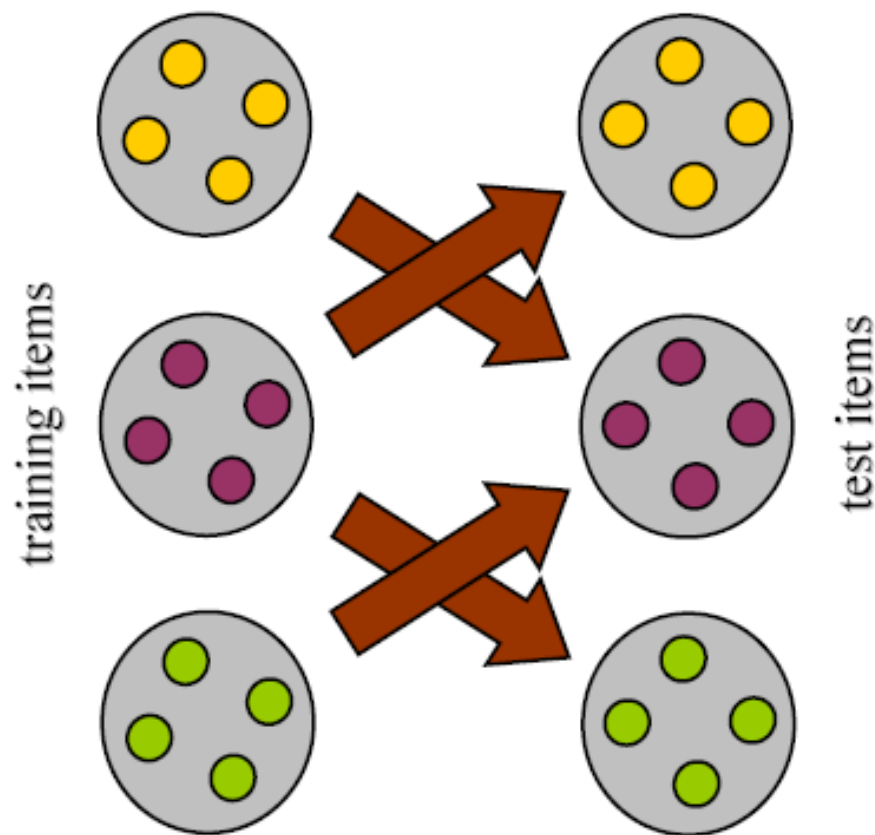


general learning in  
multiple domains

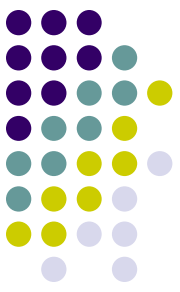


Humans can learn in many domains.

transfer of learning  
across domains



Humans can also transfer from one  
domain to other domains.

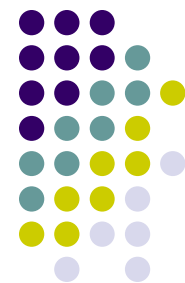


# Transfer Learning = “举一反三”

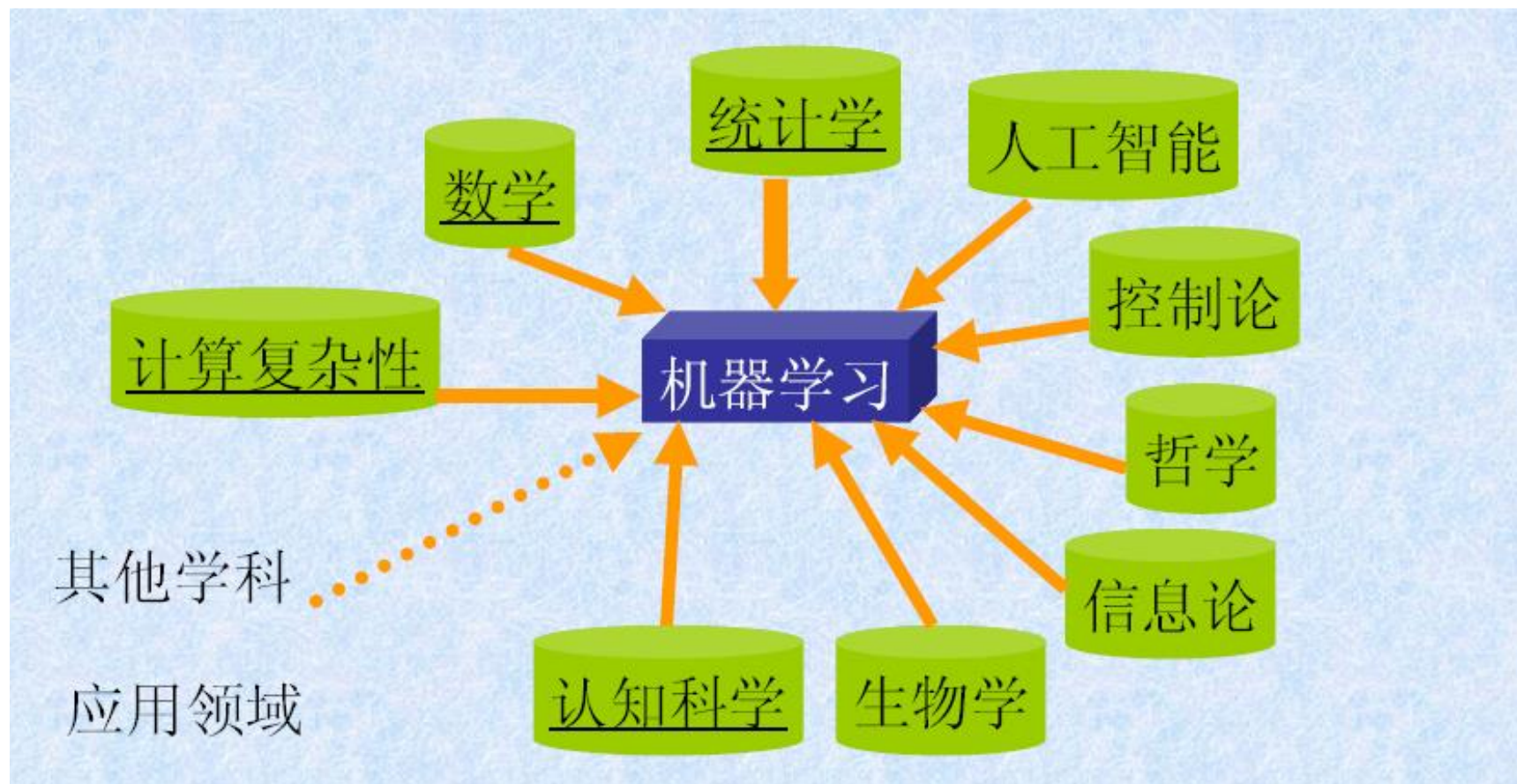
- 我们人类也具有这样的能力，比如我们学会了国际象棋，就可以将下棋的方法应用于跳棋，或者说学起跳棋来会更容易一些；学会了C++，可以把它的一些思想用在学习Java中；再比如某人原来是学物理的，后来学习计算机时，总习惯把物理中的某些思想和概念用于计算机科学中。用我们通俗的话总结，就是
- 传统机器学习 = “种瓜得瓜，种豆得豆”
- 迁移学习 = “举一反三”







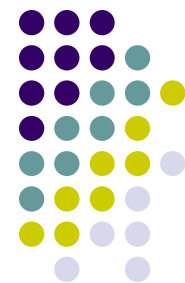
# 机器学习是多学科的交叉





# 机器学习学科

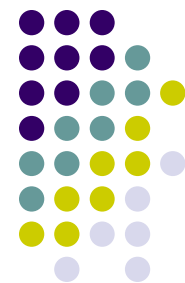
- 1983年，R.S. Michalski等人撰写《机器学习：通往人工智能的途径》一书
- 1986年，Machine Learning杂志创刊
- 1997年以Tom Mitchell的经典教科书《**machine learning**》中都没有贯穿始终的基础体系，只是个不同方法和技术的罗列
- 机器学习还非常年轻、很不成熟



# 机器学习的分类

- 传统上，大致可分为4类：
- 归纳学习
- 解释学习
- 遗传学习 (GA)
- 连接学习(神经网络)





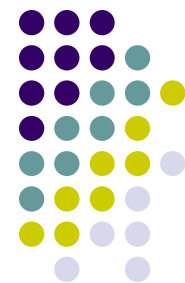
# 归纳学习

- 是从某一概念的分类例子集出发归纳出一般的概念描述。
- 这是目前研究得最多的学习方法，其学习目的是为了获得新的概念、构造新的规则或发现新的理论。
- 这种方法要求大量的训练例，而且归纳性能受到描述语言、概念类型、信噪比、实例空间分布、归纳模式等的影响。



# 解释学习

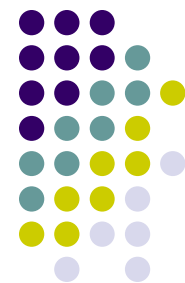
- （分析学习）是从完善的领域理论出发演绎出有助于更有效地利用领域理论的规则。其学习目的是提高系统性能，而不是修改领域理论。
- 它与归纳学习相反，只需要少量的训练例，但要求有完善的领域理论，而且学习效果也与例子表示形式、学习方法（正例学习或反例学习）、概括程度等有关。



# 机器学习面临的挑战

- 随着应用的不断深入,出现了很多被传统机器学习研究忽视、但非常重要的问题 (下面将以医疗和金融为代表来举几个例子)
- 机器学习正与众多学科领域产生了交叉,交叉领域越多,问题也越多,也正是大有可为处.





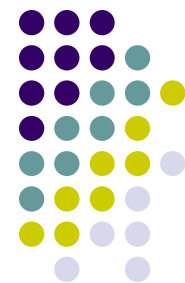
# 例子1：代价敏感问题

- 医疗：以癌症诊断为例，“将病人误诊为健康人的代价”与“将健康人误诊为病人的代价”是不同的。
- 金融：以信用卡盗用检测为例，“将盗用误认为正常使用的代价”与“将正常使用误认为盗用的代价”是不同的。
- 传统的ML技术基本上只考虑同一代价
- 如何处理代价敏感性？
- 在教科书中找不到现成的答案。



## 例子2：不平衡数据问题

- 医疗：以癌症诊断为例，“健康人”样本远远多于“病人”样本。
- 金融：以信用卡盗用检测为例，“正常使用”样本远远多于“被盗用”样本。
- 传统的ML技术基本上只考虑平衡数据
- 如何处理数据不平衡性？
- 在教科书中找不到现成的答案



## 例子3：可理解性问题

- 医疗：以乳腺癌诊断为例，需要向病人解释“为什么做出这样的诊断”
- 金融：以信用卡盗用检测为例，需要向保安部门解释“为什么这是正在被盗用的卡”
- 传统的ML技术基本上只考虑泛化不考虑理解
- 如何处理可理解性？
- 在教科书中找不到现成的答案





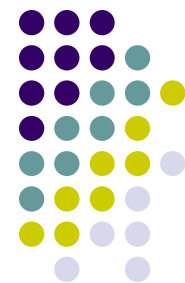
# 机器学习的最新进展

- 算法驱动(建模与数据分析)
- 应用驱动



# 算法驱动

- 海量非线性数据( $10^8-10^9$ )
- 算法的泛化能力考虑
- 学习结果数据的解释
- 代价加权的处理方法
- 不同数据类型的学习方法

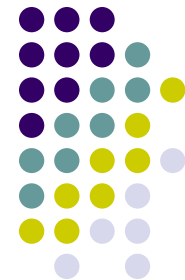


# 应用驱动

- 自然语言分析、网络与电信数据分析、图像数据分析、金融与经济数据分析、零售业数据分析、情报分析。
  - Web信息的有效获取(新一代搜索引擎)。由此导致各种学习任务：数据流学习、多实例学习(部分放弃独立同分布条件)、Ranking学习、蛋白质功能分析,DNA数据分析, …….
- 它们需要使用各种不同方法，解决实际问题。



# 应用驱动机器学习



- 流形机器学习
- 半监督机器学习
- 多实例机器学习
- Ranking机器学习
- 数据流机器学习
- 图模型机器学习
- .....



# 流形机器学习—高维数据的低维表示

- 流形 (manifold) 就是一般的几何对象的总称。比如人，有中国人、美国人等等；流形就包括各种维数的曲线曲面等。和一般的降维分析一样，流形学习把一组在高维空间中的数据在低维空间中重新表示。
- 比如在基于内容的图像检索中，当特征向量的维数非常高时，建立图像特征库时的存储高维特征的空间复杂度和度量图像之间相似性的运算复杂度都将非常的高。
- 线性方法
  - – PCA (Principal Component Analysis)
  - – ICA (Independent Component Analysis)
- 非线性方法
  - – LLE (Local linear Embedding)(Roweis, Science,2000)
  - – Isomap (Tenenbaum, Science, 2000)



- 比较常用的降维算法比如PCA，是针对线性分布的高维数据进行降维的算法，并且有局部最优的问题。而LLE（Local Linear Embedding）算法则针对于非线性数据。

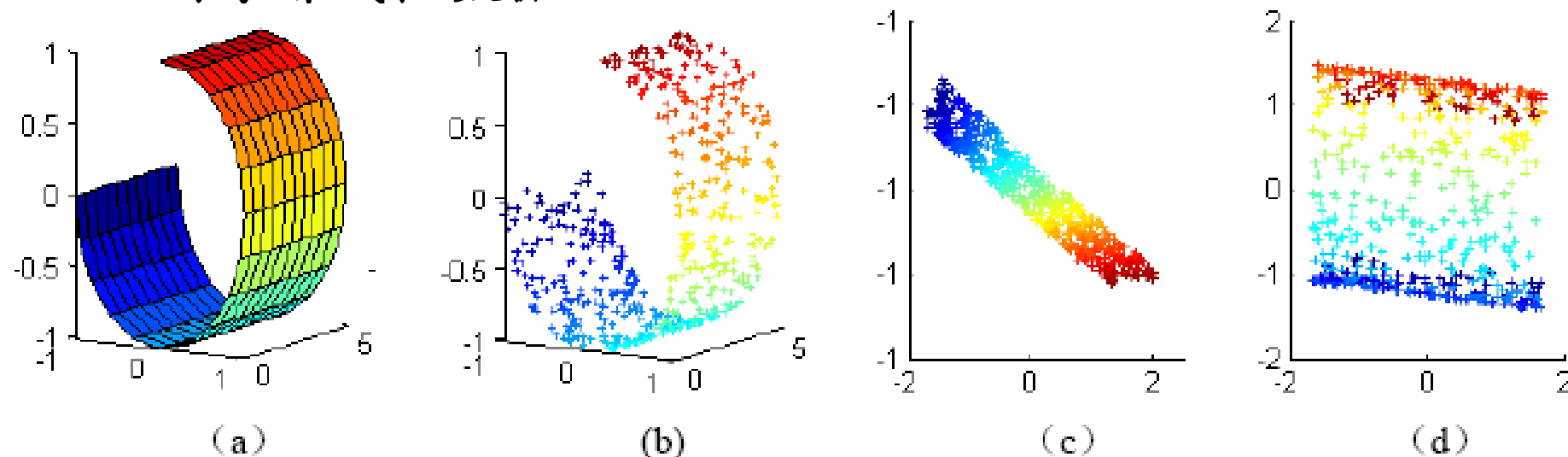
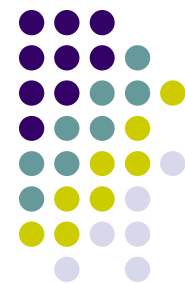


图 1. (b) 是从一个曲面 (a) 上采样得到的一组 3 维数据；  
(c) 是经 LLE 降维后的二维数据分布；(d) 是经 PCA 降维后的数据分布

在这个例子里，用LLE 进行降维成功的体现了数据内在的局部分布结构，而用PCA 映射则会将高维空间里的远点映射到低维空间后变成了近邻点。



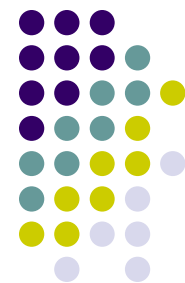
# 半监督机器学习

- 半监督的学习:有少量训练样本,学习机以从训练样本获得的知识为基础,结合测试样本的分布情况逐步修正已有知识,并判断测试样本的类别。



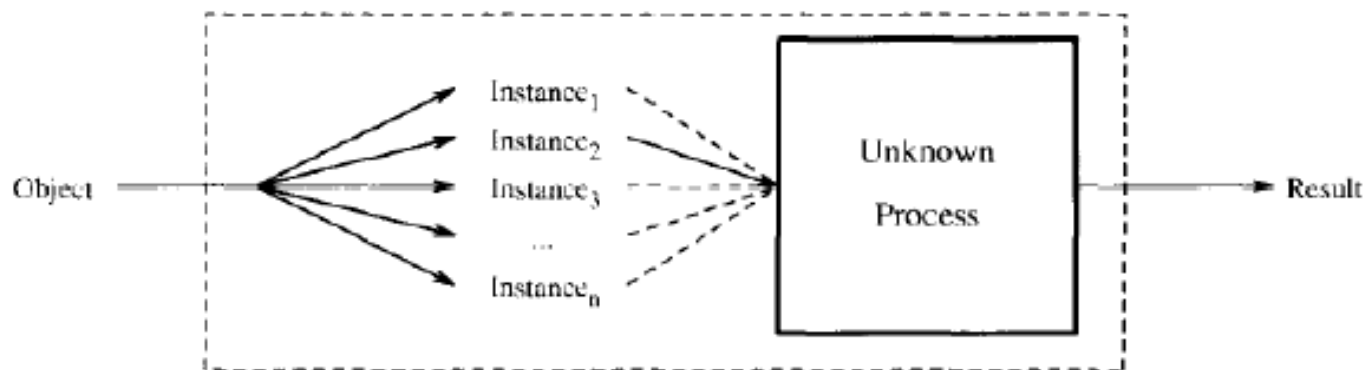
Transfer Learning

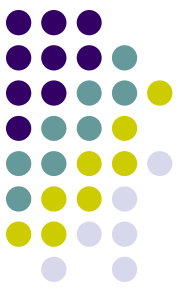




# 多示例机器学习

- 传统的机器学习中，一个对象有一个描述，而在一些实际问题中，一个对象可能同时有多个描述，到底哪个描述是决定对象性质(例如类别)的，却并不知道。解决这种“对象：描述：类别”之间1:N:1关系的学习就是多示例学习

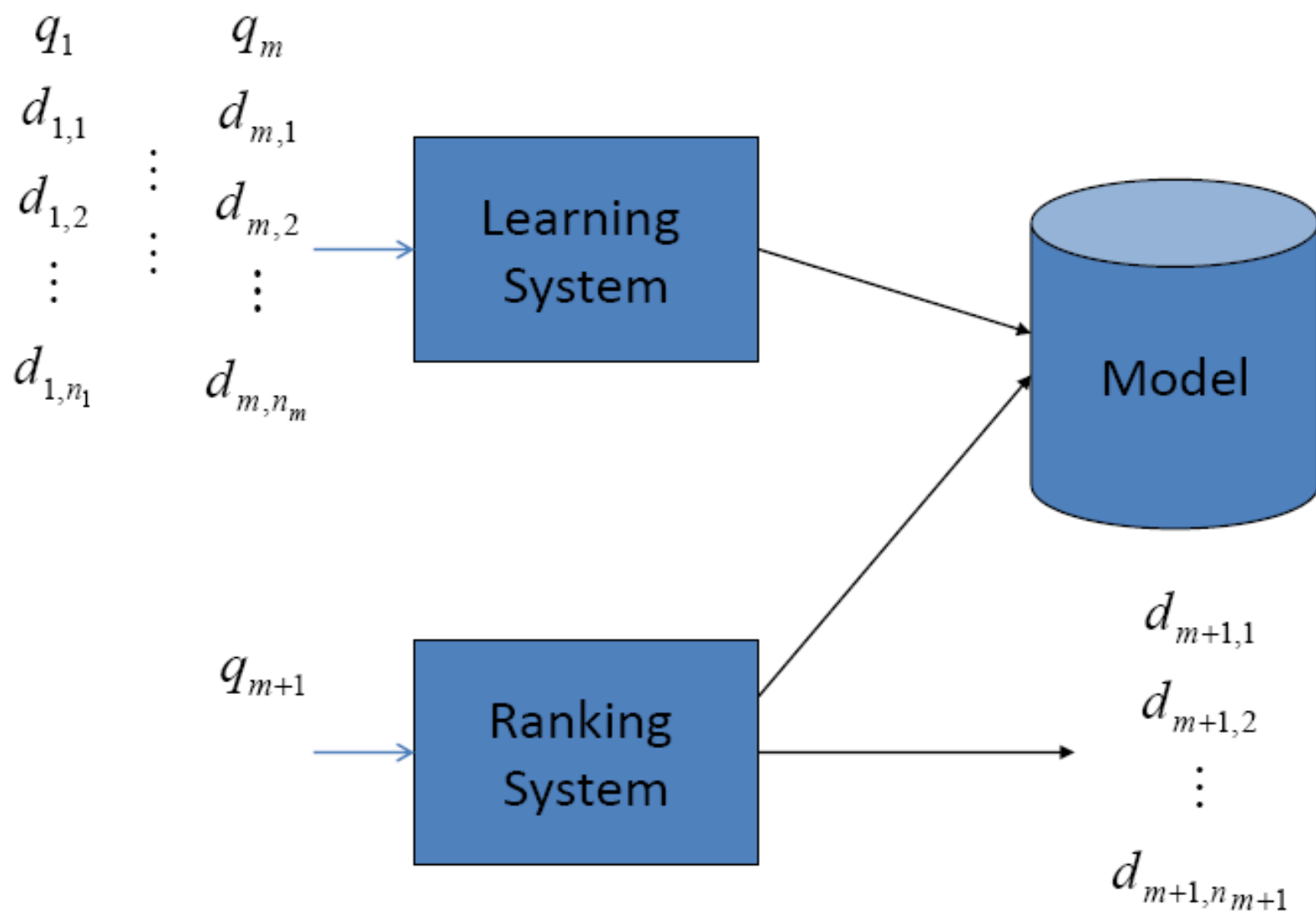




# Ranking机器学习

- 其原始说法是learning for ranking
- 问题主要来自信息检索，假设用户的需求不能简单地表示为“喜欢”或“不喜欢”，而需要将“喜欢”表示为一个顺序，问题是如何通过学习，获得关于这个“喜欢”顺序的模型。

# Learning to Rank

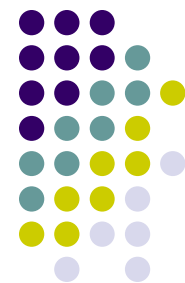




# 数据流机器学习

- 在网络数据分析与处理中，有一类问题，从一个用户节点上流过的数据，大多数是无意义的，由于数据量极大，不能全部存储，因此，只能简单判断流过的文件是否有用，而无法细致分析
- 如何学习一个模型可以完成这个任务，同时可以增量学习，以保证可以从数据流中不断改善(或适应)用户需求的模型





# 研究现状

- 主要以任务为驱动力,学习方法有待创新
- 以上这些机器学习方式还处于实验观察阶段, 缺乏坚实的理论基础
- 实际应用效果仍有待研究

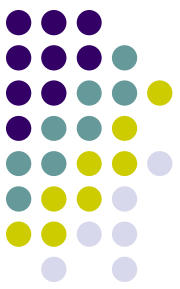


# 当前机器学习所面临情况是：

- 数据复杂、海量，用户需求多样化。从而，要求：
  - (1) 需要科学和高效的问题表示，以便将其学习建立在科学的基础上
  - (2) 应用驱动成为必然，从而针对某个或某类应用给出特定的学习方法将不断涌现
  - (3) 对机器学习的检验问题只能在应用中检验自己
  - (4) 对机器学习的结果的解释，将逐渐受到重视



现在我们逐一讨论几种比较常用的学习算法.....

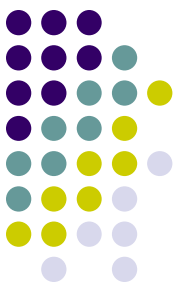


## 7.3 机械学习

- 机械学习 (Rote Learning)

又称为记忆学习或死记硬背式的学习。这种学习方法直接**记忆或存储**环境提供的新知识，并在以后通过对知识库的**检索**来直接使用这些知识，而不再需要进行任何的**计算和推导**。



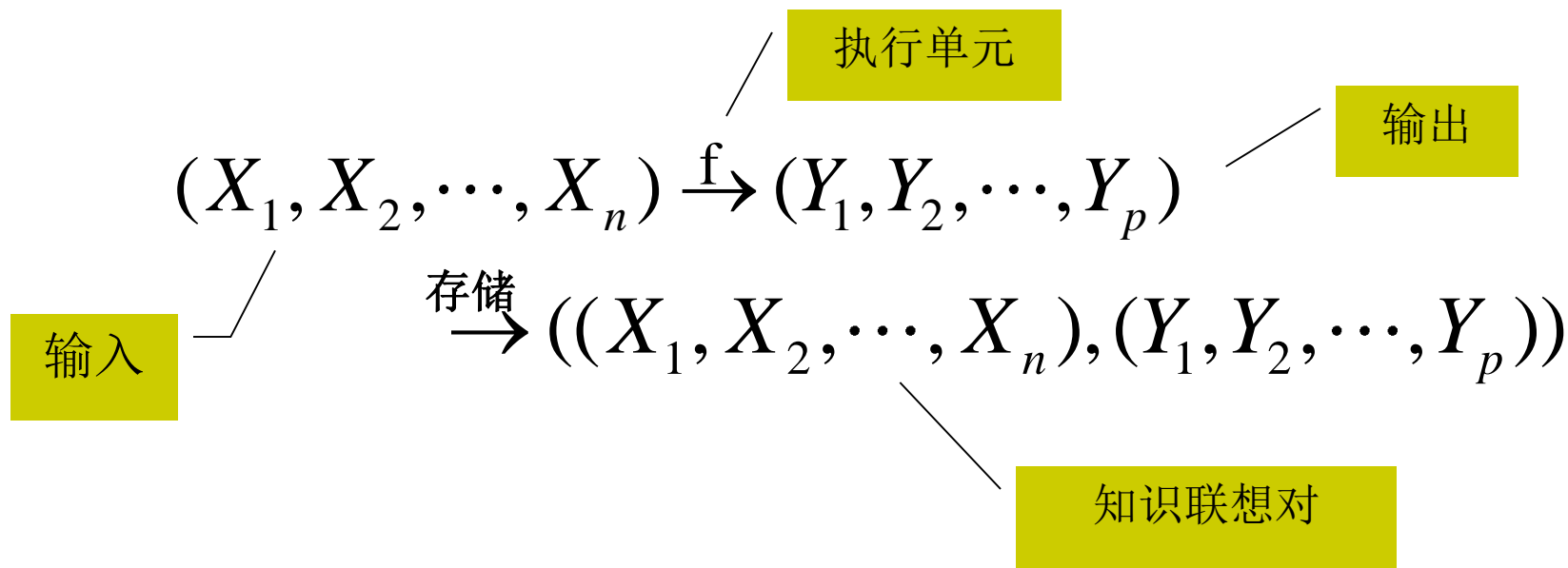


- 机械学习是一种基本的学习过程，虽然它没有足够的能力独立完成智能学习，但存储对于任何智能型的程序来说，都是必要的和基本的。记忆学习是任何学习系统的一部分，任何学习系统都要将它所获取的知识存储在知识库中，以便使用这些知识。



# 机械学习的过程

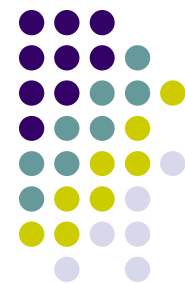
- 执行机构每解决一个问题，系统就记住这个问题和它的解。简单的机械学习模型：





# 例子：汽车修理成本估算系统

- 输入：有关待修理汽车的描述，包括制造厂家、出厂日期、车型、汽车损坏的部位以及它的损坏程度
- 输出：该汽车的修理成本



## 例子：汽车修理成本估算系统

- 为了进行估算，系统必须在其知识库中查找同一厂家，同一出厂日期、同一车型、同样损坏情况的汽车，然后把知识库中对应的数据作为修理成本的估算数据输出给用户。如果在系统的知识库中没有找到这样的汽车，则系统将请求用户给出大致的费用并进行确认，系统则会将该车的描述和经过确认的估算费用存储到知识库中，以便将来查找使用。



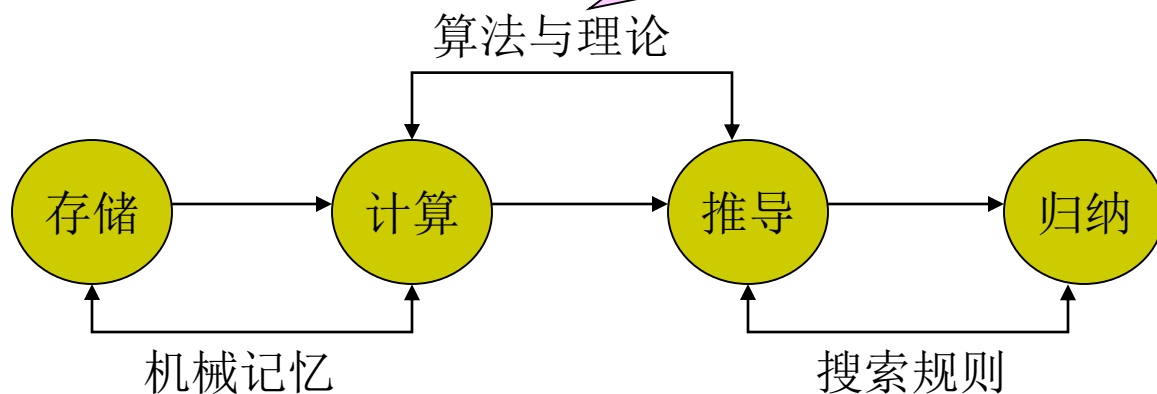
# 数据化简级别图

- 莱纳特(Lenat), 海斯·罗思(Hayes Roth)和克拉尔(Klahr)等人于1979年提出: 可以把机械学习看成是数据化简分级中的第一级。机械学习与计算、归纳和推理之间的关系如下图所示。



# 数据化简级别图

例如，第一次解一个一元二次方程的时候，必须使用很长的一段推导才能得出解方程的求根公式。但是一旦有了求根公式，以后再解一元二次方程时，就不必重复以前的推导过程，可以直接使用求根公式计算出根，这样就把推导问题简化成计算问题



在机械学习中，我们忽略计算过程，只记忆计算的输入输出，这样就把计算问题化简成另外存储问题。

可以在大量病例的基础上归纳总结出治疗的一般规律，形成规则，当遇见一个新病例时，就使用规则去处理它，而不必再重新推断解决办法，提高了工作效率。



# 机械学习要注意的问题

- 存储组织信息
  - 如何存储？使得 检索时间  $<$  计算时间
- 环境的稳定性和存储信息的适应性
- 存储与计算之间的权衡
  - 存储空间+检索时间 vs 计算时间
  - 只存储最常使用信息，忘记不常使用信息



## 7.5 类比学习

- 类比学习是利用二个不同领域（**源域**、**目标域**）中的知识相似性，可以通过类比，从源域的知识（包括相似的特征和其它性质）推导出目标域的相应知识，从而实现学习。
- 例如：
  1. 一个从未开过truck的司机，只要他有开car的知识就可完成开truck的任务。
  2. 若把某个人比喻为很像消防车，则可通过观察消防车的行为，推断出这个人的性格。
- 所以，类比学习系统可以使一个已有的计算机应用系统转变为适应于新的领域，来完成原先没有设计的相类似的功能。



# 类比推理

- 类比推理是在两个相似域之间进行的：
  - 源域S（已经认识的域）
  - 目标域T（当前尚未完全完全认识的域）
- 推理目的：

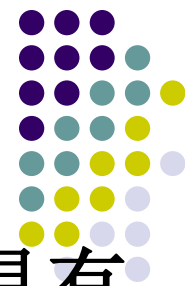
从S中选出与当前问题最近似的问题及其求解方法以解决当前的问题，或者建立起目标域中已有命题间的联系，形成新知识。



# 类比推理过程

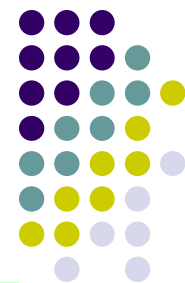
1. 回忆与联想
2. 选择: 从找出的相似情况中选出与当前情况最相似的情况及其知识。
3. 建立对应关系: 在S与T间建立映射关系。
4. 转换: 将S中的知识引到T中来, 建立求解当前问题的方法会学习到关于T的新知识。

# 类比学习形式



- 已知:  $a \in S$  与  $b \in T$  具有相似的性质  $P$ ,  $a$  还具有性质  $Q$
- 推理:  $b$  也具有性质  $Q$
- 步骤:
  - (1) 找出  $S$  和  $T$  的相似性质  $P$ , 找出  $S$  的性质  $Q$  和性质  $P$  对  $a$  的关系:  $P(a) \rightarrow Q(a)$
  - (2) 推广:  $\forall x \in S, P(x) \rightarrow Q(x)$
  - (3)  $\forall x \in T, P(x) \rightarrow Q(x)$
  - (4) 利用假言推理,  $P(b), P(x) \rightarrow Q(x)$ , 得出  $b$  具有性质  $Q$ 。





# 类比学习研究类型

- 问题求解型

求解一个新问题时，先回忆以前是否求解过类似问题，若是，则以此为依据求解新问题。

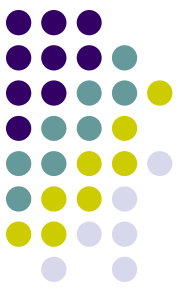
- 预测推理型

- 传统的类比法

用来推断一个不完全确定的事物可能还有的其他属性

- 因果关系型

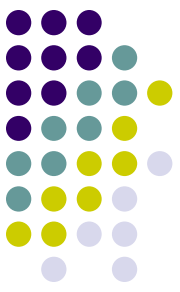
已知因果关系 $S1:A \rightarrow B$ ,  
现有 $A' \cong A$ , 则可能有 $B'$   
满足 $A' \rightarrow B'$



## 7.6 解释学习

- 基于解释的学习(Explanation-based learning, EBL)

解释学习兴起于20世纪80年代中期，根据任务所在领域知识和正在学习的概念知识，对当前实例进行分析和求解，得出一个表征求解过程的因果解释树，以获取新的知识。



- 例如，学生根据教师提供的目标概念、该概念的一个例子、领域理论及可操作准则，首先构造一个解释来说明为什么该例子满足目标概念，然后将解释推广为目标概念的一个满足可操作准则的充分条件。
- EBL已被广泛应用于知识库求精和改善系统的性能。著名的EBL系统有迪乔恩（G.DeJong）的GENESIS, 米切尔（T.Mitchell）的LEXII和LEAP, 以及明顿（S.Minton）等的PRODIGY。

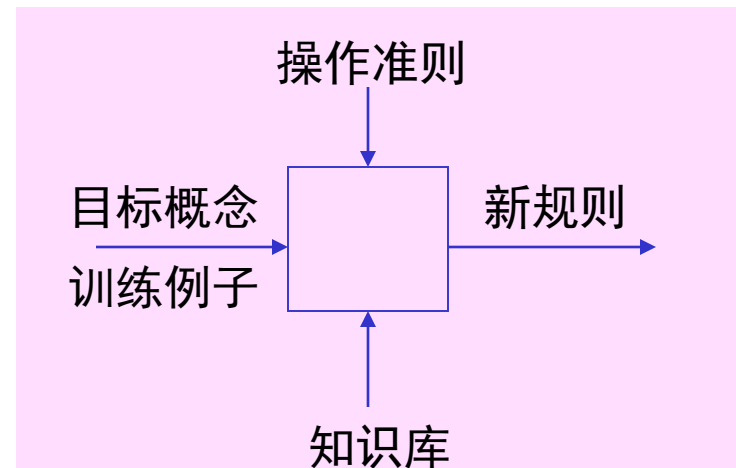
# 解释学习的一般性描述



- 米切尔提出了一个解释学习的统一算法EBG, 建立了基于解释的概括过程, 并用知识的逻辑表示和演绎推理进行问题求解。其一般性描述为:

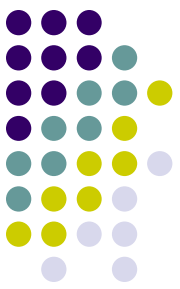
给定: 领域知识DT  
目标概念TC  
训练实例TE  
操作性准则OC

找出: 满足OC的关于TC的充分条件





- 系统进行学习时，首先运用领域知识DT找出训练实例TE为什么是目标概念TC的实例的解释，然后根据操作性准则OC对解释进行推广，从而得到关于目标概念TC的一个一般性描述，即一个可供以后使用的形式化表示的一般性知识。



# 解释学习的学习过程与算法

EBG算法可概括为两步：

## 1. 构造解释

运用领域知识进行演绎，证明提供给系统的训练实例为什么是满足目标概念的一个实例。

例如：设要学习的目标概念是“一个物体（Obj1）可以安全地放置在另一个物体（Obj2）上”，即

$\text{Safe-To-Stack}(\text{Obj1}, \text{obj2})$





- 训练实例为描述物体Obj1与Obj2的下述事实：

On(Obj1,Obj2)

Isa(Obj1,book of AI)

Isa(Obj2,table)

Volume(Obj1,1)

Density(Obj1,0.1)

.....

- 领域知识是把一个物体放置在另一个物体上面的安全性准则：

$\neg \text{Fragile}(y) \rightarrow \text{Safe-To-Stack}(x, y)$

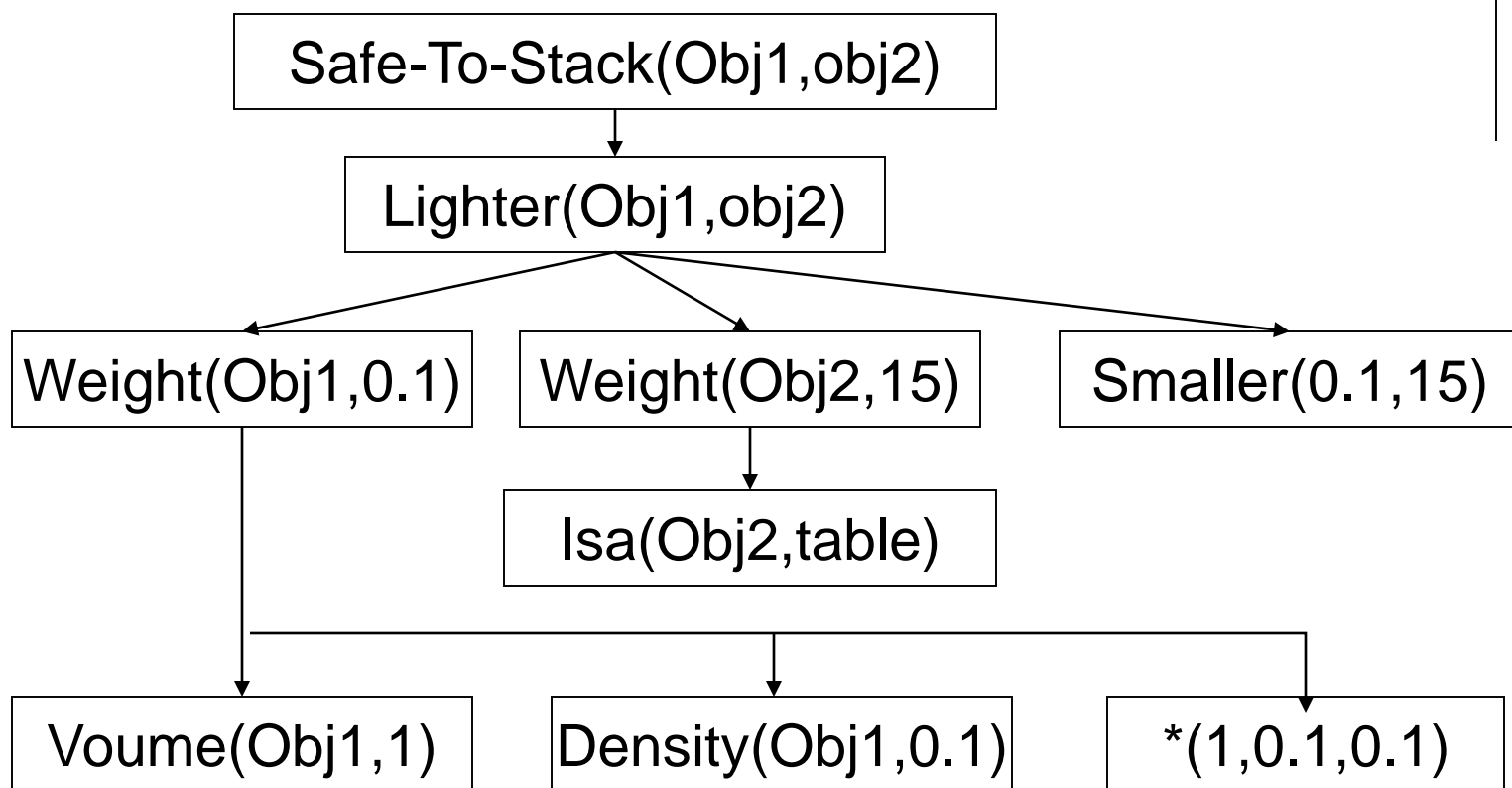
$\text{Lighter}(x, y) \rightarrow \text{Safe-To-Stack}(x, y)$

$\text{Volume}(p, v) \wedge \text{Density}(p, d) \wedge (v, d, w) \rightarrow \text{Weight}(p, w)$

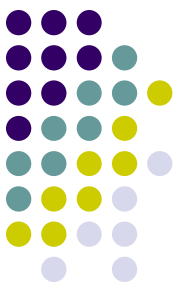
$\text{Isa}(p, \text{table}) \rightarrow \text{Weight}(p, 15)$

$\text{Weight}(p1, w1) \wedge \text{Weight}(p2, w2) \wedge \text{Smaller}(w1, w2) \rightarrow \text{Lighter}(p1, p2)$

图1 Safe-To-Stack(Obj1,obj2)解释结构



这是一个由目标概念引导的逆向推理，最终得到了一个解释结构。



## 2. 获取一般性的知识

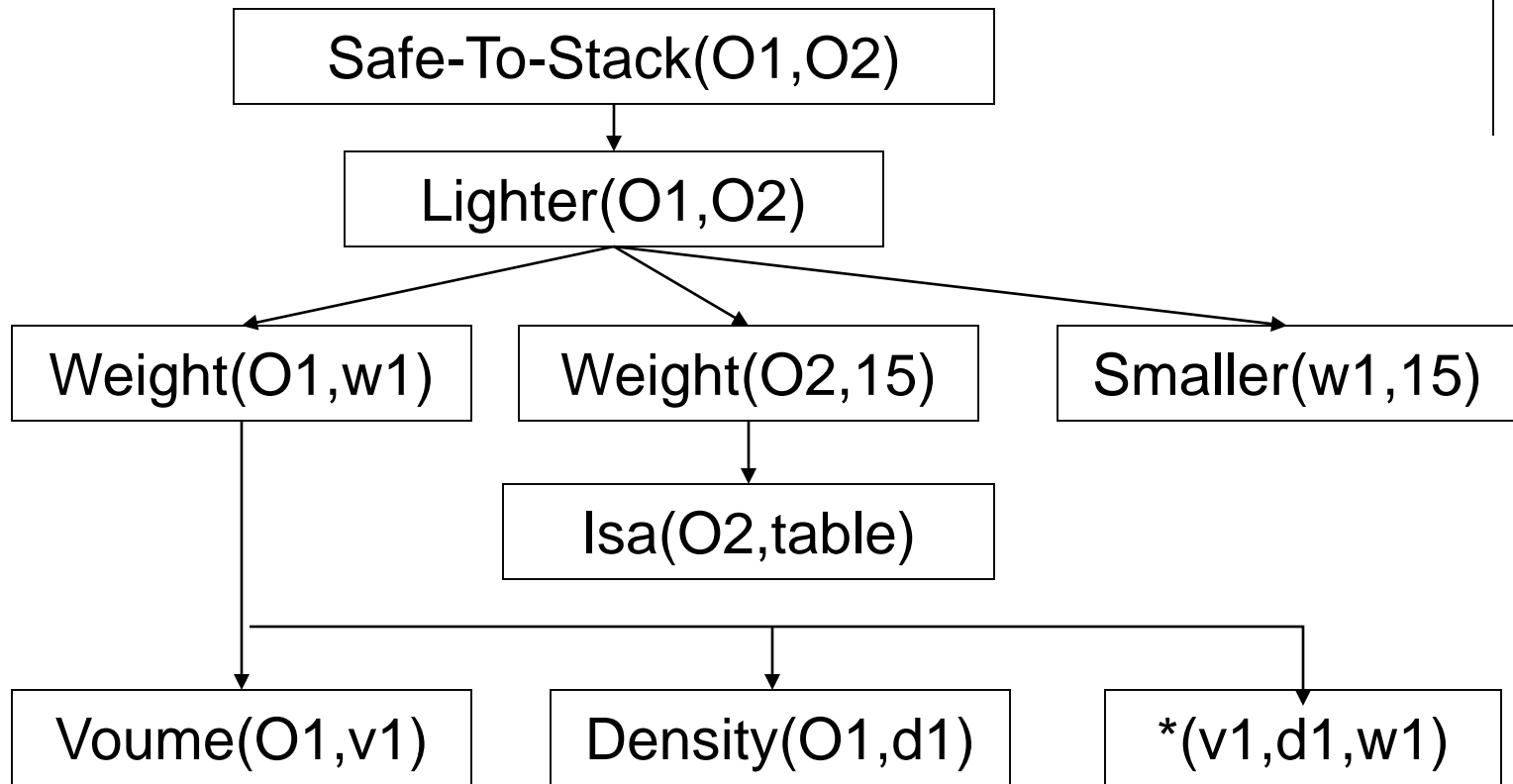
任务：对上一步得到的解释结构进行一般化的处理，从而得到关于目标概念的一般性知识。

方法：将常量换成变量，并把某些不重要的信息去掉，只保留求解问题必须的关键信息。图2为图1的一般化解释结构，可以得到如下一般性知识：

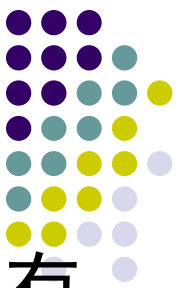
$\text{Volume}(\text{O1}, \text{v1}) \wedge \text{Density}(\text{O1}, \text{d1}) \wedge (\text{v1}, \text{d1}, \text{w1}) \wedge \text{Isa}(\text{O2}, \text{table}) \wedge \text{Smaller}(\text{w1}, 15) \rightarrow \text{Safe-To-Stack}(\text{Obj1}, \text{obj2})$



图2 Safe-To-Stack(O1,O2)一般化解释结构



以后求解类似问题时，就可以直接利用这个知识进行求解，提到了系统求解问题的效率。



# 领域知识的完善性

- 领域知识对证明的形成起着重要的作用，只有完善的领域知识才能产生正确的学习描述。但是，不完善是难以避免的，此时有可能出现如下两种极端情况：
  1. 构造不出解释
  2. 构造出了多种解释
- 解决办法：
  1. 最根本的办法是提供完善的领域知识
  2. 学习系统也应具有测试和修正不完善知识的能力，使问题能尽早地被发现，尽快地被修正。

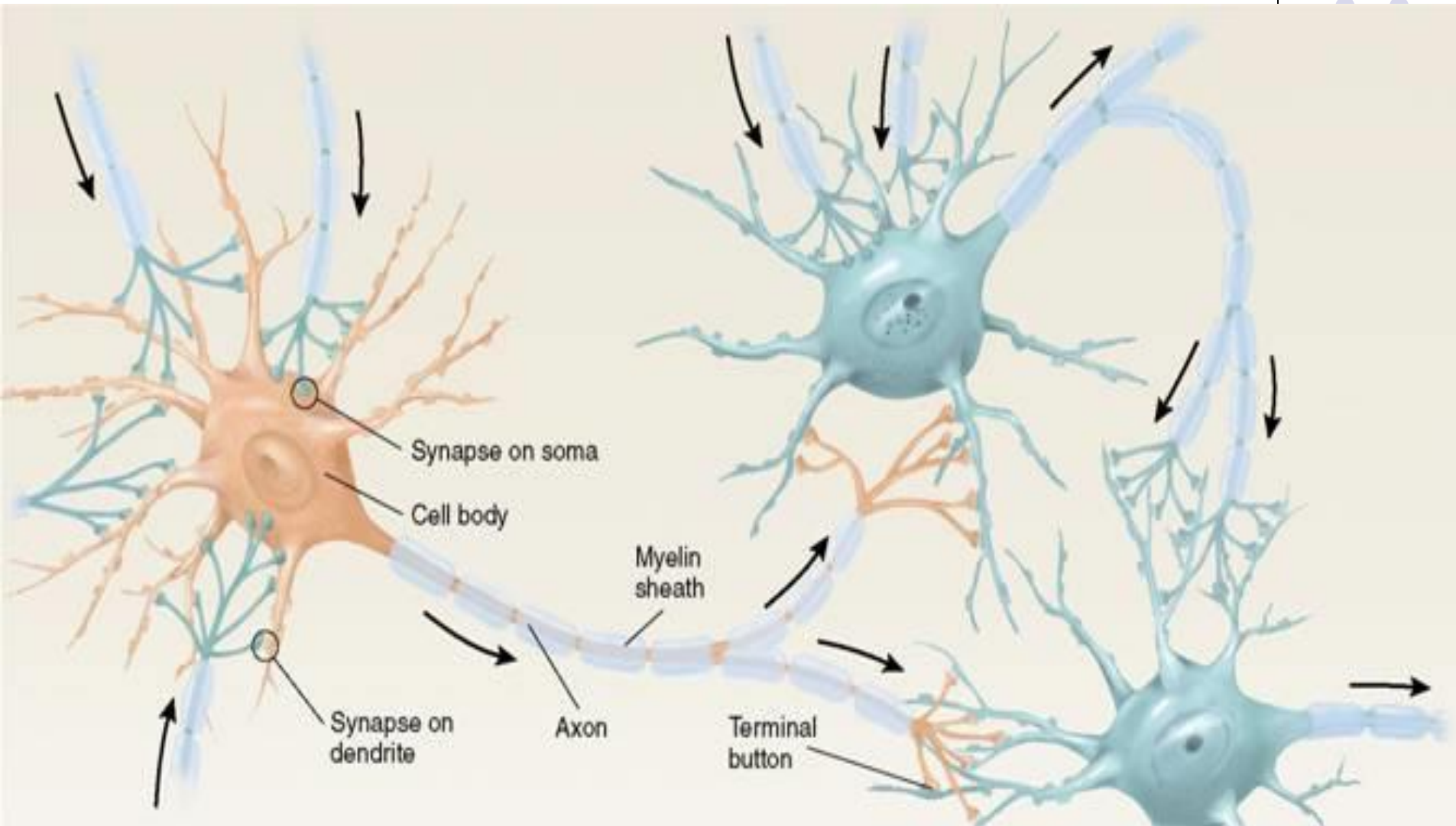


## 7.7 神经学习

- 我们先简单回顾一下神经网络的基本原理。



# 人工神经元的基本构成





1 一组突触和联结，联结具有权值

$$W_1, W_2, \dots, W_n$$

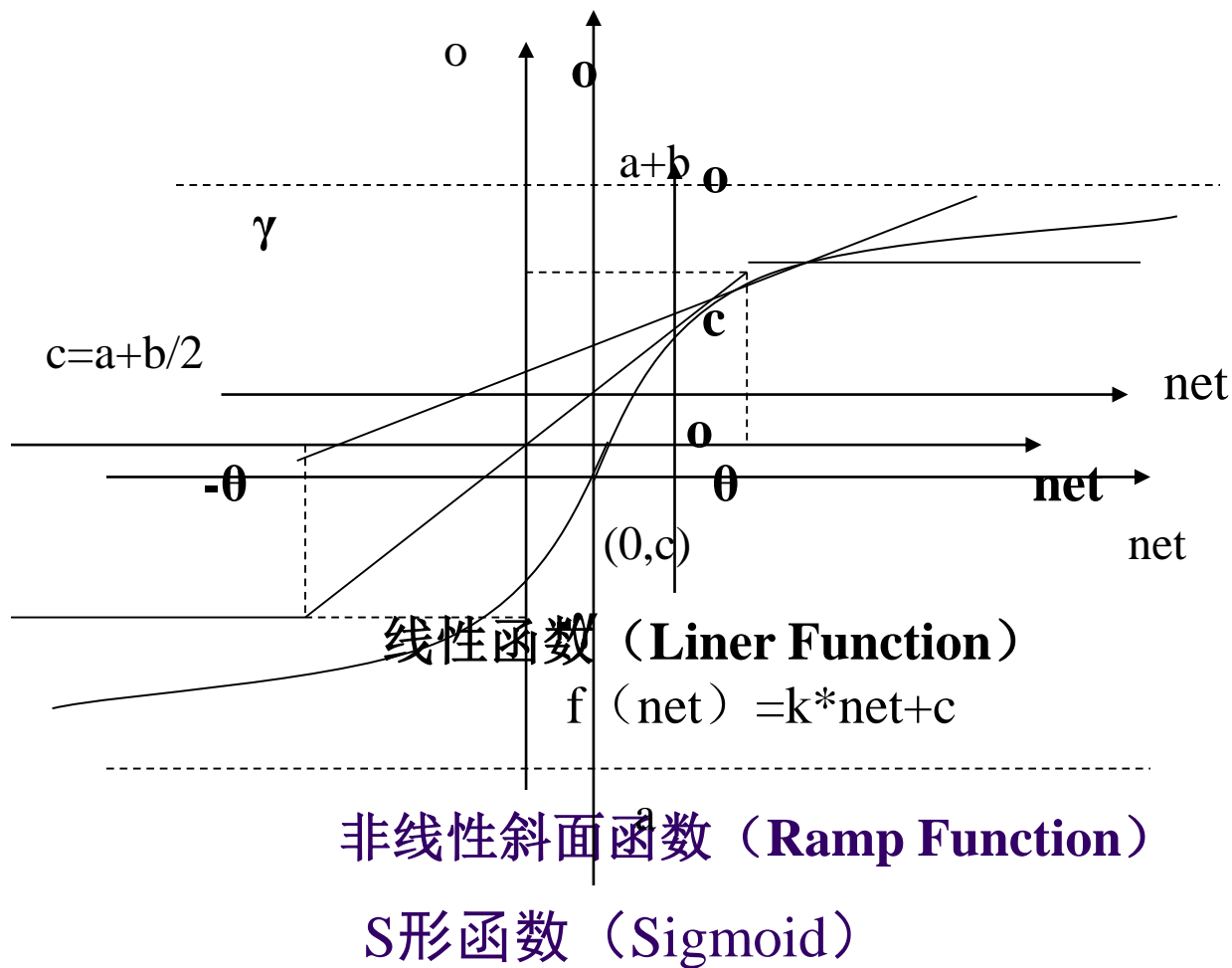
2 通过加法器功能，将计算输入的权值之和

$$net = \sum x_i w_i$$

3 激励函数限制神经元输出的幅度

$$o = f(net)$$

# 典型激励函数





# 神经网络常见三大类模型

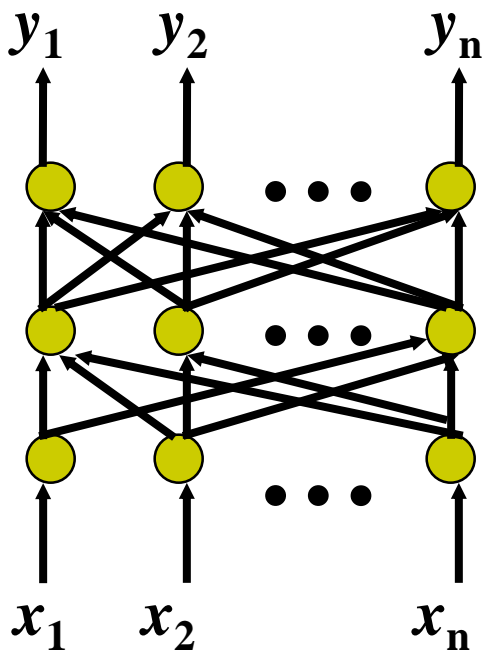
神经网络中常见三大类模型：前向神经网络；反馈神经网络；自组织神经网络。

- **前向神经网络**各神经元接受前一层的输入并输出给下一层，无反馈，常见的有BP神经网络和RBF径向基神经网络。
- **反馈神经网络**，其输入数据决定反馈系统的初始状态，然后系统经过一系列的状态转移后逐渐收敛于平衡状态，即为反馈神经网络经过计算后的输出结果。

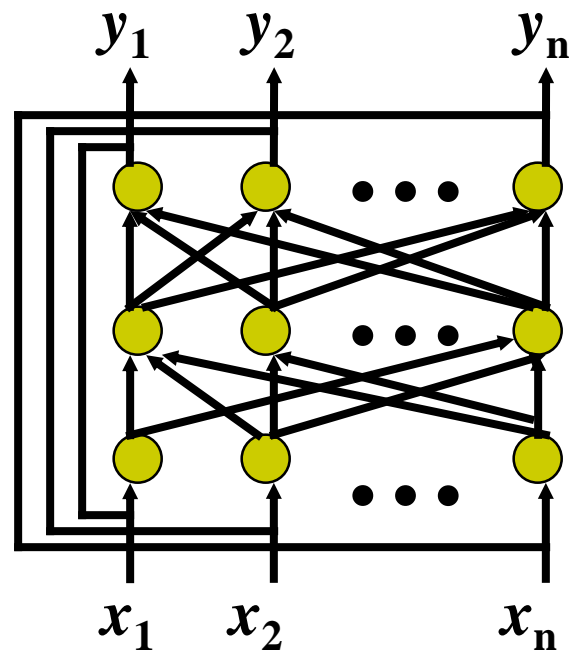
**自组织神经网络**是一种无教师学习神经网络，可以模拟人类根据过去经验自动适应无法预测的环境变化，通常利用竞争原则进行学习，可以通过学习提取一组数据中的重要特征或某种内在规律性。对生物神经系统的观察可以发现局部相关作用可以导致整体的某种有序性，自组织神经网络。



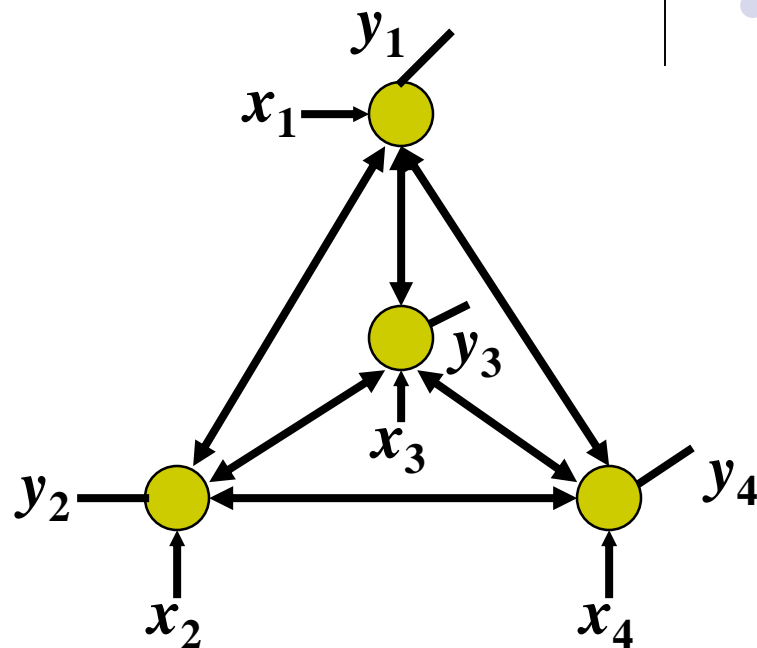
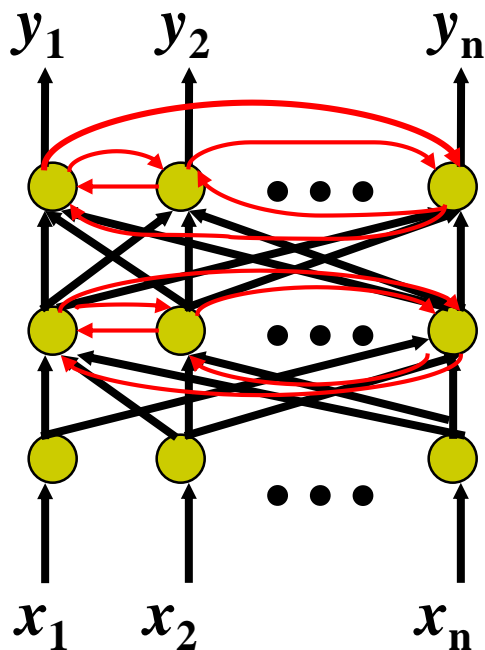
单纯前馈网络



有反馈的前馈网络



# 前馈内层互联网络



# 反馈型全互联网络







# 学习算法上的分类

在人工神经网络中，权是一个反映信息存储的关键量，在结构和转换函数定了以后，如何设计权使网络达到一定的要求这是人工神经网络必不可少的部分，大多数神经网络权的设计是通过学习得到的，目前可分为下列几种。

## 1.死记式学习

网络的权是事先设计的，值是固定的。

## 2. $\delta$ 学习律

这种方法是用已知例子作为教师对网络的权进行学习。

设  $(x^i, y^i)$   $i = 1, \dots, P$  为已知的输入、输出例子，  
 $x^i$ ,  $y^i$  为  $n$  和  $m$  维矢量，  $y^i = (y_1, y_2, \dots, y_m)^T$



$\mathbf{x}^i = (x_1, x_2, \dots, x_n)^T$ , 把  $\mathbf{x}^i$  作为神经网络的输入,  
在权的作用下, 可计算出实际神经网络的

输出为  $\mathbf{y}^{i'} = (y'_1, y'_2, \dots, y'_j, \dots, y'_m)^T$

设任一输入神经元  $q$  到  $y'_j$  的权为  $w_{qj}$  则其权的改变量为

$$\Delta w_{qj} = \eta \delta_j V_q$$
$$\delta_j = F(y_j - y'_j)$$

其中  $\eta$  为步长,  $(y_j - y'_j)$  为误差,  $V_q$  为第  $q$  个神经元的  
输出,  $F(\cdot)$  函数是根据不同的情况而定, 多数人工神经网络  
 $\delta_j = (y_j - y'_j)$ ;  $F(x) = x$ 。



### 3. 自组织的学习和Hebbian学习律

两个神经元之间的连接权，正比于  
两个神经元的活动值，如  $V_i$ ,  $V_j$

表示两个神经元的输出值，则他们之间的权的变化为

$$\Delta w_{ij} = \eta V_i V_j$$

这里  $\eta$  为步长或常数。



## 4.相近学习

设  $w_{ij}$  为从神经元  $i$  到神经元  $j$  的权,

$V_i$  为  $i$  神经元的输出, 则

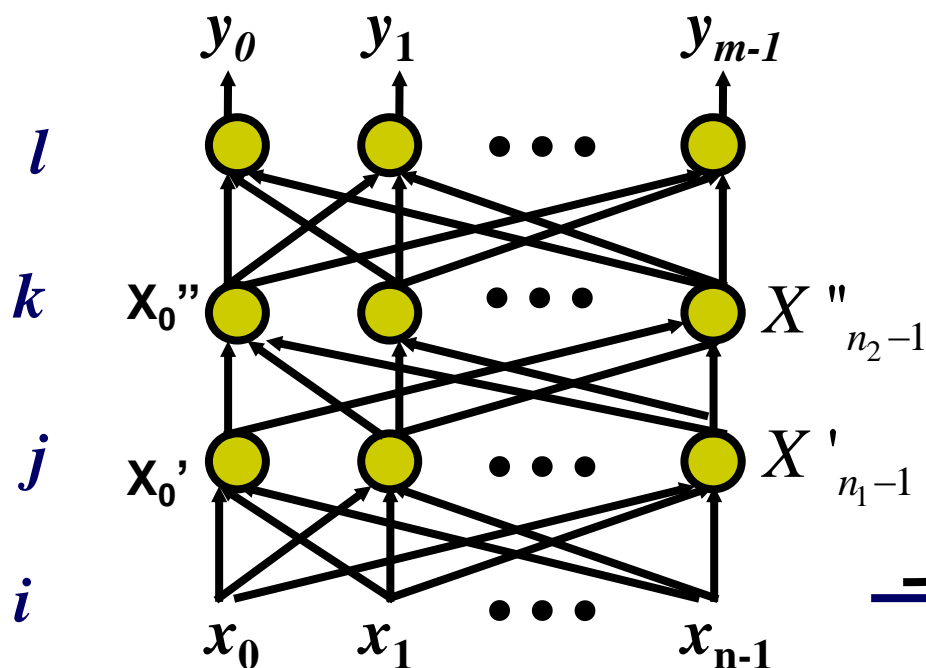
$$\Delta w_{ij} = \alpha(V_i - w_{ij})$$

在这个学习中, 使  $w_{ij}$  十分逼近  $V_i$  的值。  
如Kohonen和ART等都采用这类学习方法。

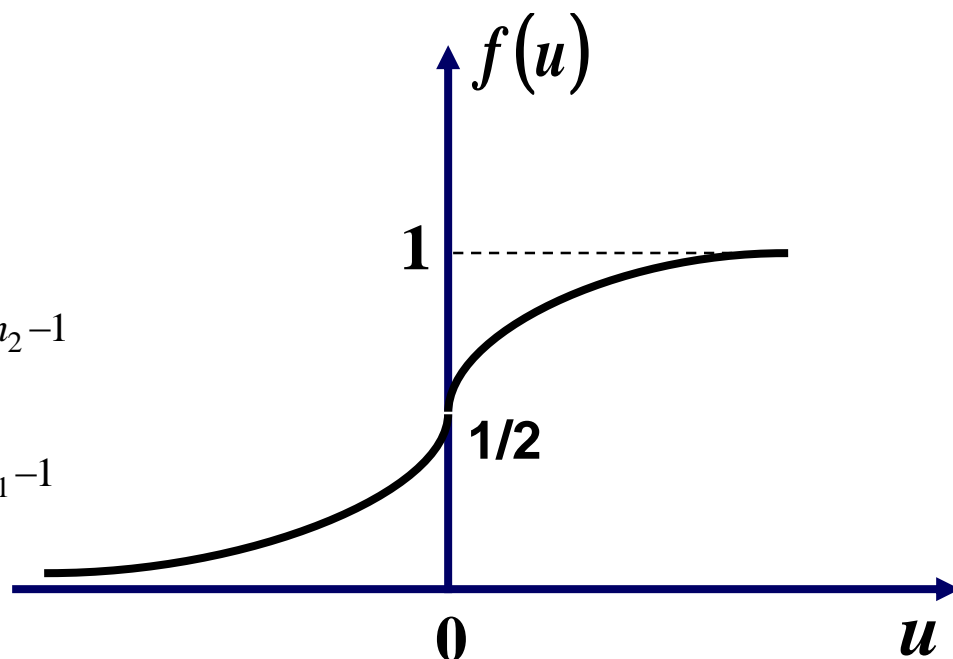
# 非线性变换单元组成的前馈网络 (简称B-P网络)



## (一) 网络的结构与数学描述



图a: 多层B-P网络



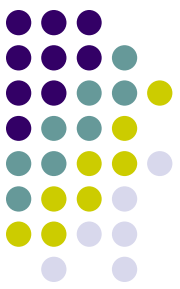
图b: 输入输出非线性函数

$$f(u_j) = \frac{1}{1 + e^{-u_j}} \quad (1)$$



$$\begin{cases} y_l = f\left(\sum_{k=0}^{n_2-1} w''_{kl} x''_k - \theta''_l\right) \\ x''_k = f\left(\sum_{j=0}^{n_1-1} w'_{jk} x'_j - \theta'_k\right) \\ x'_j = f\left(\sum_{i=0}^{n-1} w_{ij} x_i - \theta_j\right) \end{cases} \quad (2)$$

**B-P网络是完成n维空间向量对m维空间的近似映照。  
(F为近似映照函数)**



如果输入第 $P_1$ 个样本对  $(x^{P_1}, y^{P_1})$

通过一定方式训练后，得到一组权  $W^{P_1}$

$W^{P_1}$  包括网络中所有的权和阈值，此时

$W^{P_1}$  的解不是唯一的，而是在权空间中的一个范围，也可为几个范围。对于所有的学习样本 $P_1=1,2,...,P$ 都可以满足：

$$y^{P_1} = F(x^{P_1}, W^{P_1})$$

各自的解为 $W^1, W^2, ..., W^P$ ，通过对样本集的学习，得到满足所有样本正确映照的解为

$$W = \bigcap_{P_1=1}^P W^{P_1}$$





学习的过程就是求解 $W$ 的过程，因为学习不一定要求很精确，所以得到的是一种近似解。

## (二) B-P的学习算法

为了方便起见，在图a的网络中，把阈值写入连接权中去，令： $\theta_l'' = w_{n_2 l}''$ ； $\theta_k' = w_{n_1 k}'$ ； $\theta_j' = w_{n j}$ ； $x_{n_2} = -1$ ； $x_{n_1} = -1$ ； $x_n = -1$ ；则方程（2）改为

$$y_l = f\left(\sum_{k=0}^{n_2} w_{kl}'' x_k''\right) \quad (2a)$$

$$x_k'' = f\left(\sum_{j=0}^{n_1} w_{jk}' x_j'\right) \quad (2b)$$

$$x_j' = f\left(\sum_{i=0}^n w_{ij} x_i\right) \quad (2c)$$



**B-P**算法属于 $\delta$ 学习律，是一种有教师的学习算法。

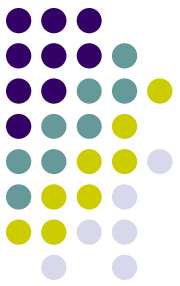
第 $P_l$ 样本输入到图a所示的网络，得到输出 $y_l$ ， $l=0,1,\dots,m-1$ ，其误差为各输出单元误差之和，满足：

$$E_{P_l} = \frac{1}{2} \sum_{l=0}^{m-1} \left( t_l^{P_l} - y_l^{P_l} \right)^2$$

对于 $P$ 个样本的学习，其总误差为

$$E_{\text{总}} = \frac{1}{2} \sum_{P_l=1}^P \sum_{l=0}^{m-1} \left( t_l^{P_l} - y_l^{P_l} \right)^2 \quad (3)$$

这里用梯度法可以使总的误差向减小的方向变化，直到 $\Delta E_{\text{总}}=0$ 为止，这种学习方式其矢量 $W$ 能够稳定到一个解，但并不保证是 $E_{\text{总}}$ 的全局最小解，可能是一个局部极小解。



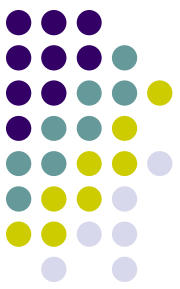
# 具体学习算法的解析式推导如下：

令 $n_0$ 为迭代次数，根据（3）式和  
梯度算法，可得到每一层的权的  
迭代公式为

$$w''_{kl}(n_0 + 1) = w''_{kl}(n_0) - \eta \frac{\partial E_{\text{总}}}{\partial w''_{kl}} \quad (4a)$$

$$w'_{jk}(n_0 + 1) = w'_{jk}(n_0) - \eta \frac{\partial E_{\text{总}}}{\partial w'_{jk}} \quad (4b)$$

$$w_{ij}(n_0 + 1) = w_{ij}(n_0) - \eta \frac{\partial E_{\text{总}}}{\partial w_{ij}} \quad (4c)$$



从(4a)式可以看出,  $w_{kl}''$  是第  $k$  个神经元与输出层第  $l$  个神经元之间的连接权, 它只与输出层中一个神经元有关, 将(3)式代入(4a)中的第二项, 利用公式(1)得:

$$\begin{aligned} \frac{\partial E_{\text{总}}}{\partial w_{kl}''} &= \sum_{P_l=1}^P \frac{\partial E_{P_l}}{\partial y_l^{P_l}} \frac{\partial y_l^{P_l}}{\partial u_l''^{P_l}} \frac{\partial u_l''^{P_l}}{\partial w_{kl}''} \quad (5) \\ &= \sum_{P_l=1}^P (t_l^{P_l} - y_l^{P_l}) f'(u_l''^{P_l}) x_k''^{P_l} \end{aligned}$$

$$\text{这里 } u_l''^{P_l} = \sum_{k=0}^{n_2} w_{kl}'' x_k''^{P_l}$$



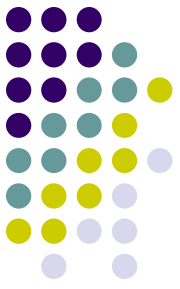
$\mathbf{x}_k''^{P_1}$  为  $P_1$  样本输入网络时,  $\mathbf{x}_k''$  的输出值。

$$f'(u_l''^{P_1}) = \frac{e^{-u_l''^{P_1}}}{(1 + e^{-u_l''^{P_1}})^2} \quad (6)$$
$$= f(u_l''^{P_1})[1 - f(u_l''^{P_1})] = y_l^{P_1}(1 - y_l^{P_1})$$

将(6)、(5)代入(4a), 得:

$$w_{kl}''(n_0 + 1) = w_{kl}''(n_0) + \eta \sum_{P_1=1}^P \delta_{kl}^{P_1} \mathbf{x}_k''^{P_1}$$

这里的  $\delta_{kl}^{P_1} = (t_l^{P_1} - y_l^{P_1}) y_l^{P_1} (1 - y_l^{P_1})$



对于中间隐层，根据(4b)式有：

$$\Delta w'_{jk} = -\eta \frac{\partial E_{\text{总}}}{\partial w'_{jk}}$$

而

$$\begin{aligned} \frac{\partial E_{\text{总}}}{\partial w'_{jk}} &= - \sum_{P_l=1}^P \sum_{l=0}^{m-1} (t_l^{P_l} - y_l^{P_l}) \frac{\partial y_l^{P_l}}{\partial u_l^{''P_l}} \frac{\partial u_l^{''P_l}}{\partial x_k^{''P_l}} \frac{\partial x_k^{''P_l}}{\partial u_k^{'P_l}} \frac{\partial u_k^{'P_l}}{\partial w'_{jk}} \\ &= - \sum_{P_l=1}^P \sum_{l=0}^{m-1} (t_l^{P_l} - y_l^{P_l}) f'(u_l^{''P_l}) w_{kl}^{''} x_k^{''P_l} (1 - x_k^{''P_l}) x_j^{'P_l} \\ &= - \sum_{P_l=1}^P \sum_{l=0}^{m-1} \delta_{kl}^{P_l} w_{kl}^{''} x_k^{''P_l} (1 - x_k^{''P_l}) x_j^{'P_l} \\ &= - \sum_{P_l=1}^P \delta_{jk}^{P_l} x_j^{'P_l} \end{aligned}$$

其中：

$$\delta_{jk}^{P_1} = \sum_{l=0}^{m-1} \delta_{kl}^{P_1} w_{kl}'' x_k''^{P_1} (1 - x_k''^{P_1})$$

所以

$$w'_{jk}(n_0 + 1) = w'_{jk}(n_0) + \eta \sum_{P_1=1}^P \delta_{jk}^{P_1} x_j'^{P_1}$$

**注意：**  $w'_{jk}$  是与输出层中每个神经元都有关系。

$$y_l^{P_1} = f(u_l''^{P_1})$$

$$u_l''^{P_1} = \sum_{k=0}^{n_2} w_{kl}'' x_k''$$

$$x_k''^{P_1} = f(u_k'^{P_1})$$

$$u_k'^{P_1} = \sum_{j=0}^{n_1} w'_{jk} x_j'$$







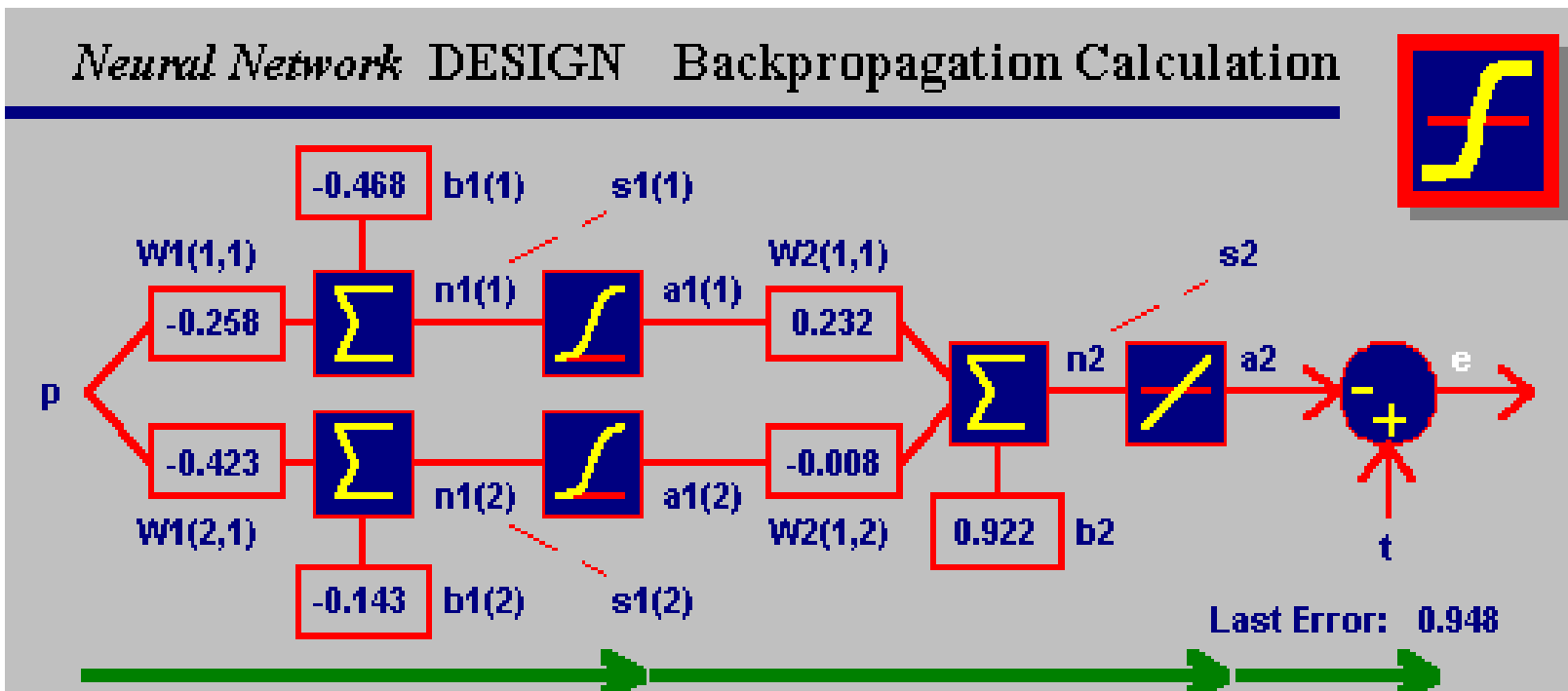
同理可得

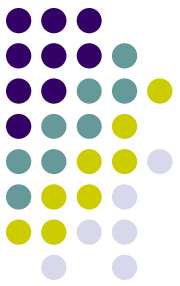
$$w_{ij}(n_0 + 1) = w_{ij}(n_0) + \eta \sum_{P_i=1}^P \delta_{ij}^{P_i} x_i^{P_i}$$

其中：

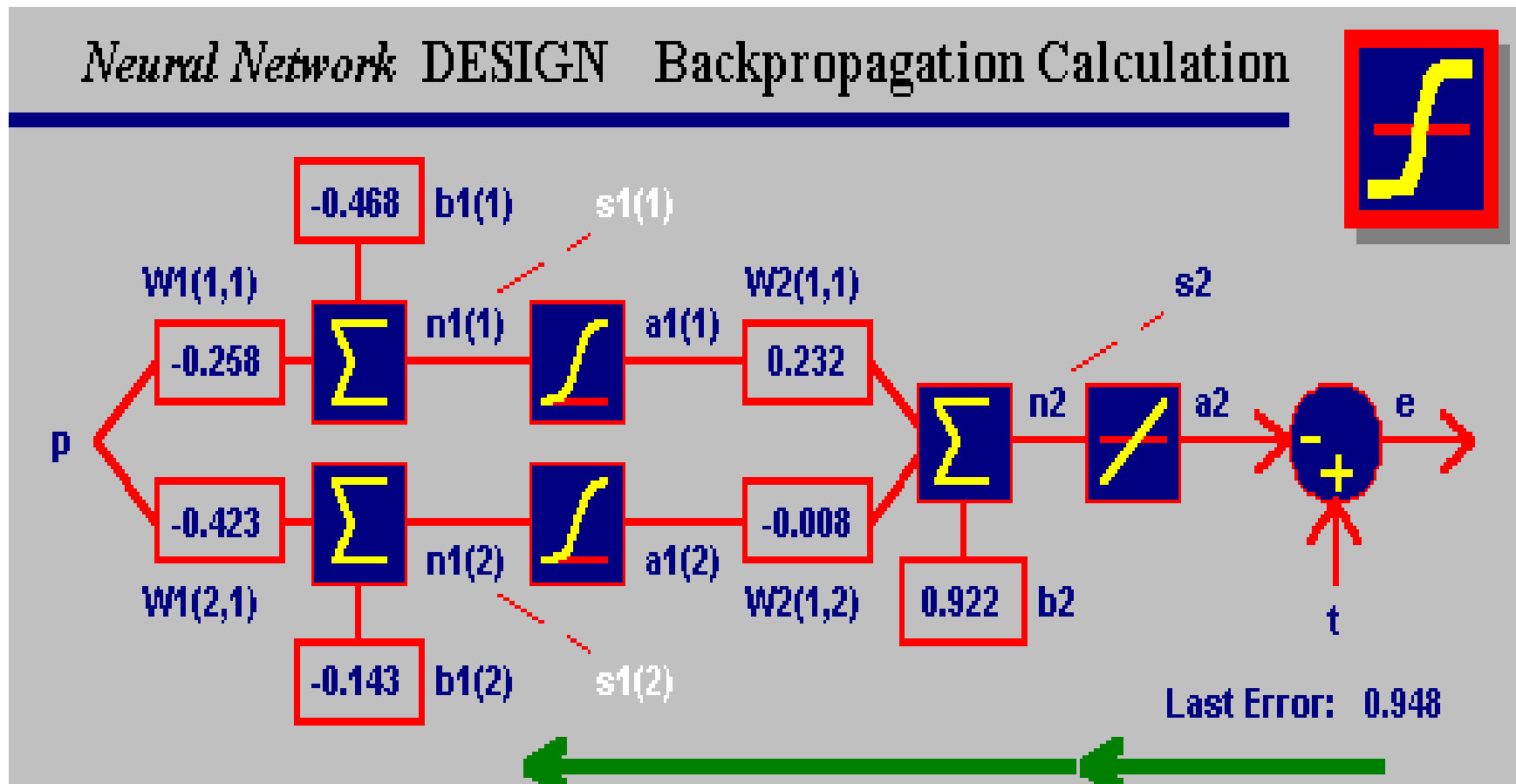
$$\delta_{ij}^{P_i} = \sum_{k=0}^{n_2} \delta_{jk}^{P_i} w'_{jk} x_j'^{P_i} (1 - x_j'^{P_i})$$

1985年，发展了BP网络学习算法，包括正向传播和反向传播，正向传播过程中，输入信息从输入层经隐单元逐层处理传向输出层，每层神经元状态仅影响下一层神经元的状态，如图：





如在输出层得不到期望的输出，则转入反向传播，将误差信号沿原来的通路返回，修改各层神经元的权值使误差信号最小，如图：



# 反向传播学习示例

## 熟人学习问题



- 假设正好有两个输入为1值，而其余输入为0值。H1和H2为隐节点，具有与门作用。该网络的目标是要确定对应于输入的两个人是否相识。该网络的任务为学习上面一组3人中的任何一人是下面一组3人中任何一人的熟人。如果网络的输出大于0.9，那么就判断这两个人为相识；如果输出小于0.1，就判断为不相识；其它结果被认为是模糊不定的。节点A为“熟人”。

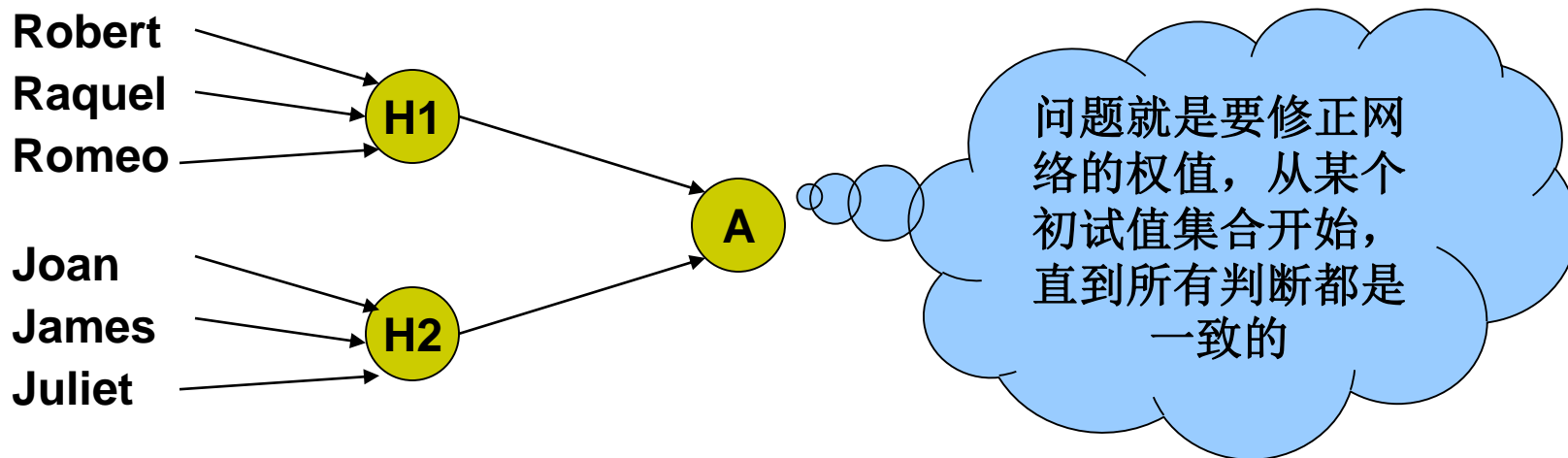




表1 BP网络学习经验数据

Robert	Raquel	Romeo	Joan	James	Juliet	熟人	同胞
1	1	0	0	0	0	0	1
1	0	1	0	0	0	1	1
0	0	1	0	0	1	0	1
0	0	0	1	0	1	0	1
0	0	0	0	1	1	0	0
1	1	0	0	0	0	1	0
1	0	1	0	0	1	0	0
1	0	0	1	0	1	0	0
1	0	0	0	1	1	0	
0	0	1	1	0	0	1	0
0	0	1	0	1	0	1	0
0	0	1	0	0	1	1	0
0	0	0	1	1	0	0	1
0	0	0	1	0	1	0	1
0	0	0	0	1	1	0	1

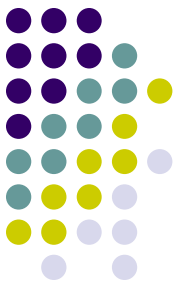


表2 训练NN时观察到的权值变化

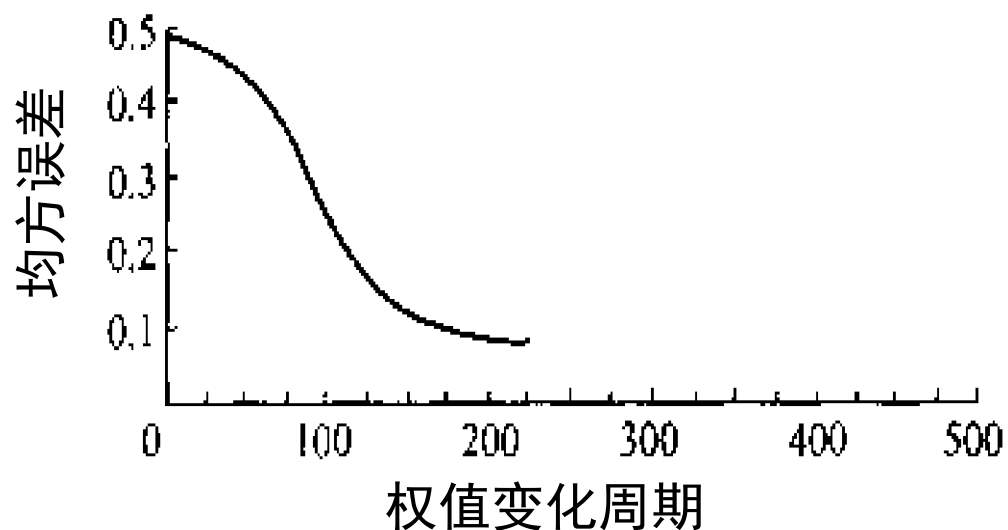
第一个初始阈值为0.1，第二个为0.2，其他的每次递增0.1，直到1.1。通过BP算法改变初始值，直到所有输出均方误差在0.1内为止。

当所有采样输入产生一个合适的输出值时的阈值和权值

权值	初始值	结束第一个任务	权值	初始值	结束第一个任务
$t_{H1}$	0.1	1.99	$w_{Jenny \rightarrow H2}$	0.7	5.28
$w_{Robert \rightarrow H1}$	0.2	4.65	$w_{Juliet \rightarrow H2}$	0.8	5.28
$w_{Raquel \rightarrow H1}$	0.3	4.65	$t_A$	0.9	9.07
$w_{Romeo \rightarrow H1}$	0.4	4.65	$w_{H1 \rightarrow A}$	1.0	6.27
$t_{H2}$	0.5	2.28	$w_{H2 \rightarrow A}$	1.1	6.12
$w_{Jenny \rightarrow H2}$	0.6	5.28			



图1 认识熟人问题的学习试验结果

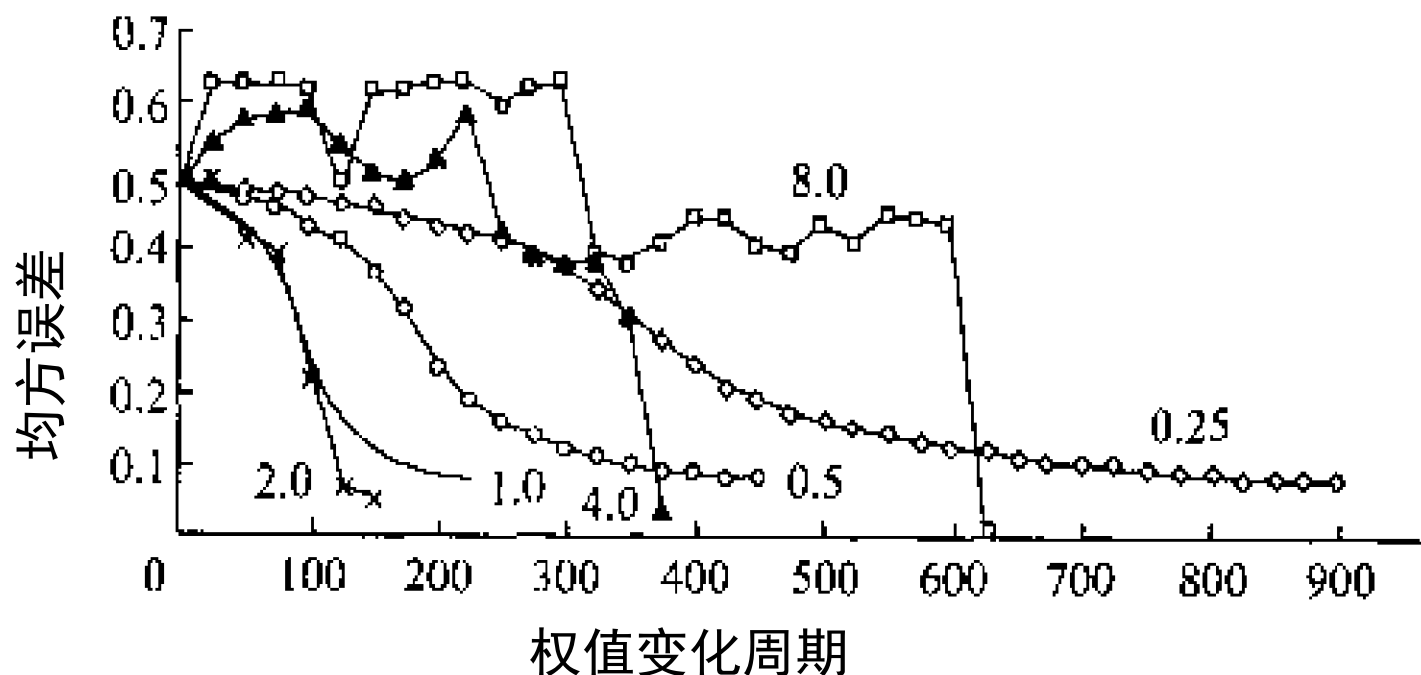


这个网络大约经过了255次权值修正后，网络性能才变得满意，即均方误差 $<0.1$ 。





图2 学习行为与比率参数的关系

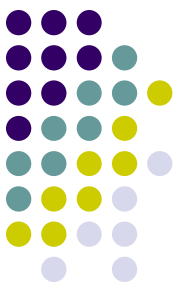


比率参数 $r$ 是用来**调节权值变化**的一个参数。 $r$ 值越大越有利于提高学习速度，但又不能太大，以免使输出过分地超出期望值而引起超调。



# BP算法的不足:

- 收敛速度非常慢，且随着训练样例维数增加，网络性能变差。
- 不完备的算法，可能出现局部极小问题
- 网络中隐节点个数的选取尚无理论指导
- 新样例的加入会影响已学习过的样例



# 基于Hopfield网络的学习

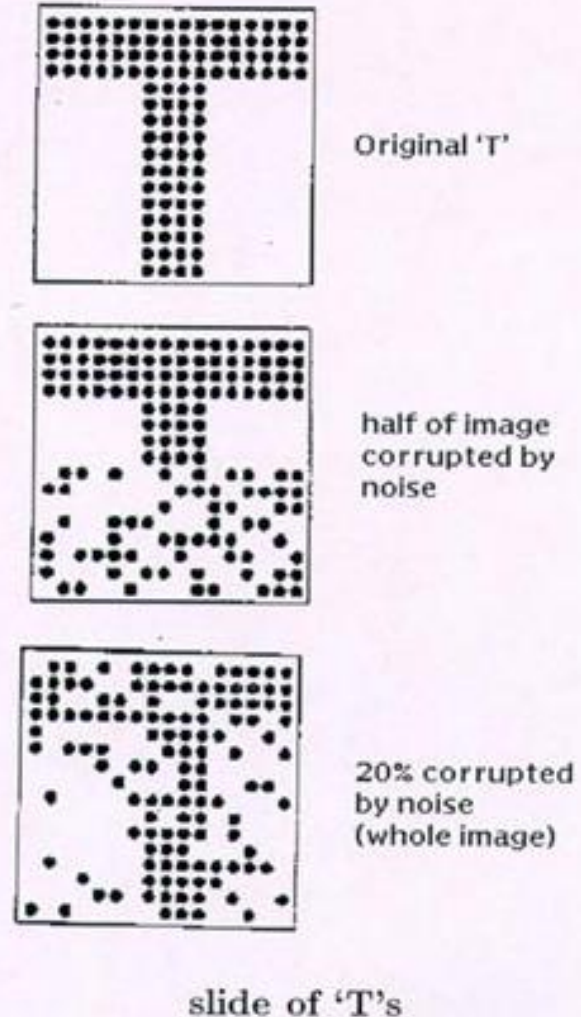
- 1982年，J. Hopfield提出了可用作联想存储器<sup>器</sup>的互连网络，这个网络称为Hopfield网络模型，也称Hopfield模型。1984年，他又提出连续时间神经网络模型。这两种模型的许多重要特性是密切相关的。一般在进行计算机仿真时采用离散模型，而在用硬件实现时则采用连续模型。

# 联想存储器

## Associative Memory



- Nature of associative memory
  - part of information given
  - the rest of the pattern is recalled





# 联想存储器网络的几点要素

- Hopfield提出，如果把神经网络的各平衡点设想为存储于该网络的信息，而且网络的收敛性保证系统的动态特性随时间而达到稳定，那么这种网络称为**联想存储器**。
  - 1. 联想存储网络可以由一个状态向量 $\mathbf{v} = (v_1, v_2, \dots, v_m)$ 来表示
  - 2. 网络有一组稳定的状态向量 $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ .
  - 3. 网络总是从任意的初始状态 $\mathbf{v}$ ，随着能量函数 $E$ 的减少吸引到某个稳定状态



# 优化问题（能量函数E）

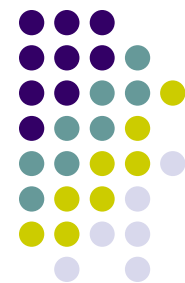
$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} v_i v_j + \sum_i \frac{1}{\tau_i} \int_0^{v_i} f_i^{-1}(v) - \sum I_i v_i$$

对E求微分（省略推导过程）

得

$$\frac{dE}{dt} \leq 0$$

这表明若函数E是有界函数，Hopfield网络总是吸引到E函数的局部最小值上。通过适当地选取 $w_{ij}$ 的值和外部输入信号 $I_i$ ，就可以将优化问题匹配到神经网络上。

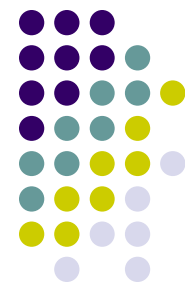


# 优化问题（能量函数 $E$ ）

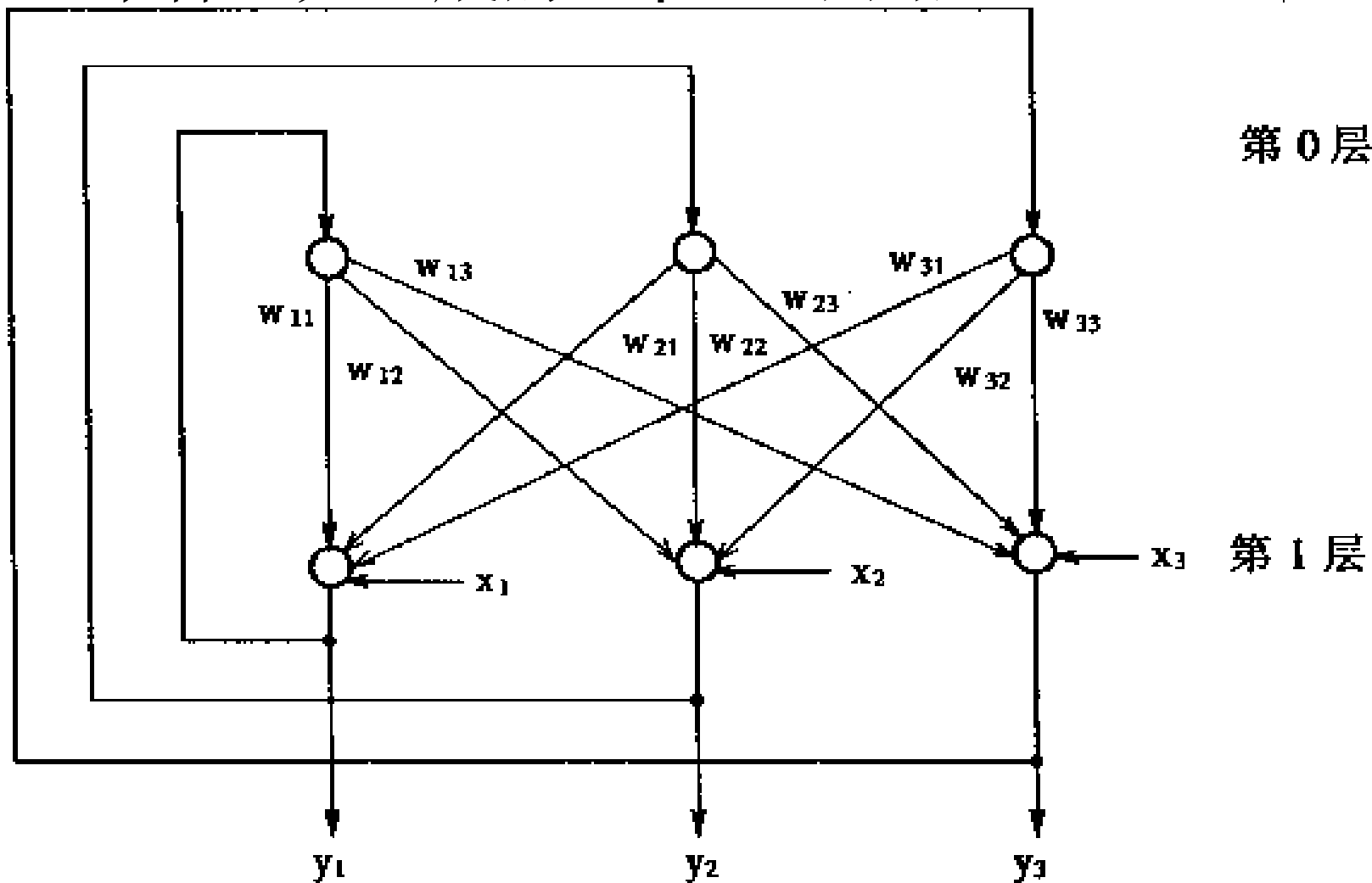
- 在实际应用中，任何一个系统，如果其优化问题可以用能量函数 $E(t)$ 作为目标函数，那么，总可以用连续Hopfield网络对其进行求解。由于引入能量函数 $E(t)$ ，Hopfield使神经网络和问题优化直接对应。利用神经网络进行优化计算，就是在神经网络这一动力系统给出初始的估计点，即初始条件；然后随网络的运动传递而找到相应极小点。这样，大量的优化问题都可以用连续的Hopfield网来求解。



# 离散Hopfield神经网络



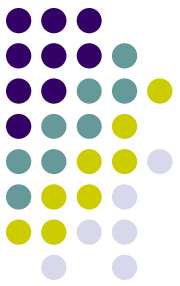
- 三个神经元组成的Hopfield网络





- 霍普菲尔特提出的离散网络模型是一个离散时间系统，每个神经元只有两种状态，可用1和-1，或者1和0表示，由连接权值 $w_{ij}$ 所构成的矩阵是一个零对角的对称矩阵，即

$$w_{ij} = \begin{cases} w_{ji}, & i \neq j \\ 0, & i = j \end{cases}$$



- 在该网络中，每当有信息进入输入层时，在输入层不做任何计算，直接将输入信息分布地传递给下一层各有关节点。若用 $X_j(t)$ 表示节点 $j$ 在时刻 $t$ 的状态。则该节点在下一时刻（即 $t+1$ ）的状态由下式决定：

$$X_j(t+1) = \begin{cases} 1, & H_j(t) \geq 0 \\ -1(\text{或}0), & H_j(t) < 0 \end{cases}$$

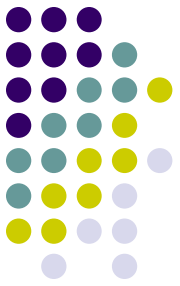
这里 
$$H_j(t) = \sum_{i=1}^n w_{ij} X_i(t) - \theta_j$$

$w_{ij}$ 为从节点 $i$ 到节点 $j$ 的连接权值,  $\theta_j$ 为节点 $j$ 的阈值



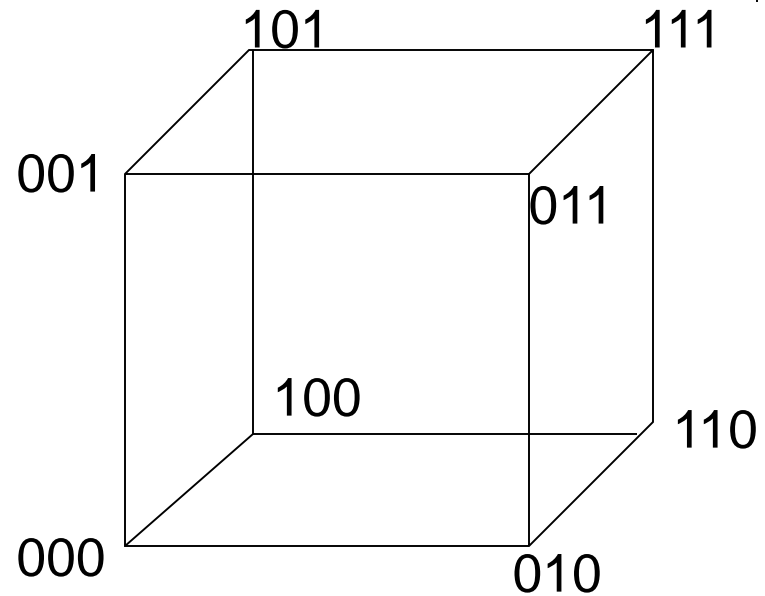
- 整个网络的状态用 $X(t)$ 表示，它是由各节点的状态所构成的向量。对于上图，若假设输出层只有两个节点，并用1和0分别表示每个节点的状态，则整个网络共有四种状态，分别为：

00, 01, 10, 11

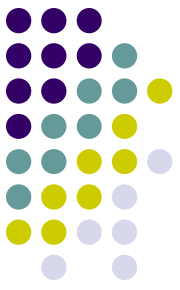


- 如果假设输出层有三个节点，  
则整个网络共有八

种状态，每个状态是一个三位的二进制数，如图所示。在该图中，立方体的每一个顶角代表一个网络状态。



三个神经元的八个状态



- 一般来说，如果在输出层有 $n$ 个神经元，则网络就有 $2^n$ 个状态，它可以与一个 $n$ 维超立体的顶角相联系。当有一个输入向量输入到网络后，网络的迭代过程就不断地从一个顶角转向另一个顶角，直至稳定于一个顶角为止。如果网络的输入不完全或只有部分正确，则网络将稳定于所期望顶角附近的一个顶角那里。

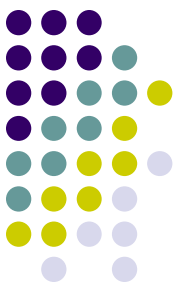


# Hopfield模型算法

## 1. 设置互连权值

$$W_{ij} = \begin{cases} \sum_{s=0}^{m-1} x_i^s x_j^s, & i \neq j \\ 0, & i = j, 0 \leq i, j \leq n-1 \end{cases}$$

其中， $x_i^s$ 为s类样例的第i个分量，它可以为+1或-1(0)，样例类别数为m，节点数为n



## 2. 未知类别样本初始化

$$y_i(0)=x_i \quad 0 \leq i \leq n-1$$

其中,  $y_i(t)$  为节点  $i$  在  $t$  时刻的输出, 当  $t=0$  时,  $y_i(0)$  就是节点  $i$  的初始值,  $x_i$  为输入样本的第  $i$  个分量。

## 3. 迭代直到收敛

$$y_i(t+1) = f\left(\sum_{i=0}^{n-1} W_{ij} y_i(t)\right), \quad 0 \leq j \leq n-1$$

该过程将一直重复进行, 直到进一步的迭代不再改变节点的输出为止。

## 4. 转2继续。





# Hopfield网络的不足

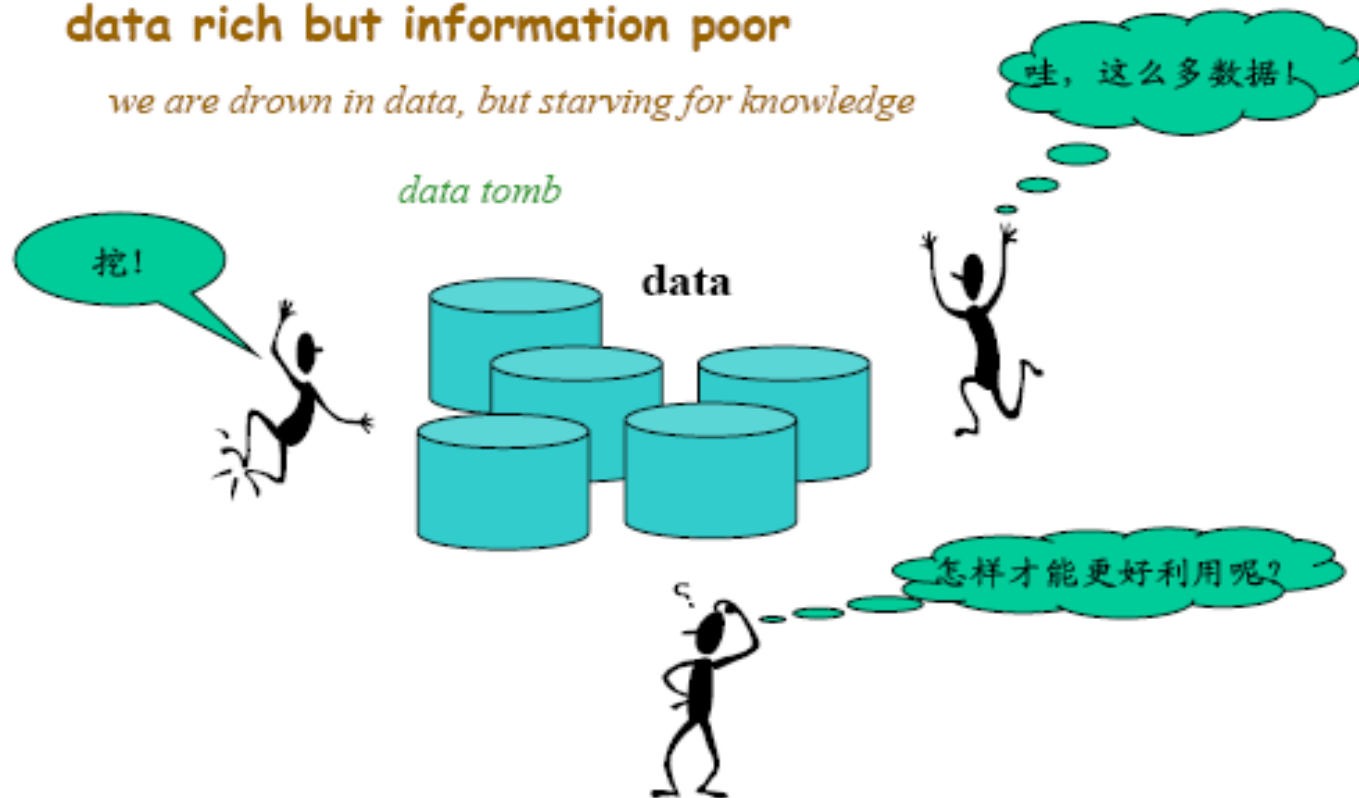
- 很难精确分析网络的性能
- 其动力学行为比较简单

## 7.8 知识发现

**产生的前提：**从大量数据中找出隐藏在其中的、有用的信息和规律；计算机技术和信息技术的发展使其有能力处理这样大量的数据。

**data rich but information poor**

*we are drown in data, but starving for knowledge*

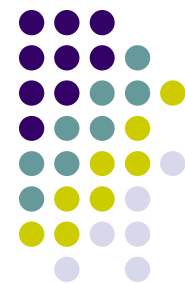


# 成功例子



- 零售业的沃尔玛“啤酒与尿布”规律
- Google
- 中国宝山钢铁公司1995年开始利用数据挖掘技术提高热轧产品质量、合理配矿
- 中国铁道部1998年开始利用数据挖掘技术分析春运期间的铁路客流量，挖掘影响铁路客运总量的关键因素

.....



# 知识发现的发展过程

- 八十年代人工智能（AI）研究项目失败后，转入实际应用时提出的。
- 1989年国际上第一次关于数据挖掘与知识发现的研讨会在美国的底特律召开，在此次会议上第一次提出了知识发现（Knowledge Discovery in Database, KDD）一词。
- 1995年，在加拿大召开了第一届KDD和DM国际学术会议。会议对KDD做了确切的定义。
- 从九十年代中后期开始，KDD和DM已成为研究的热点和焦点。

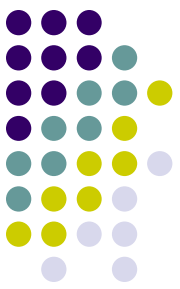


# 知识发现的定义

- 数据库中的知识发现是从大量数据中辨识出有效的、新颖的、潜在行用的、并可被理解的模式的高级处理过程。

经知识发现提取出的知识必须是新颖的，这可以通过两个途径来衡量：①通过比较新得到的数据和以前的数据或期望得到的数据的差异来衡量新颖程度②二是通过其内部所包含的未知信息来衡量。提取出的模式应该是有意义的，这可以通过某些函数的值来衡量。

如果成绩为81，82. 83, 84, 85, 86. 87, 88, 89或9D，则成绩优良”就不能称为个模式。



# 知识发现处理过程

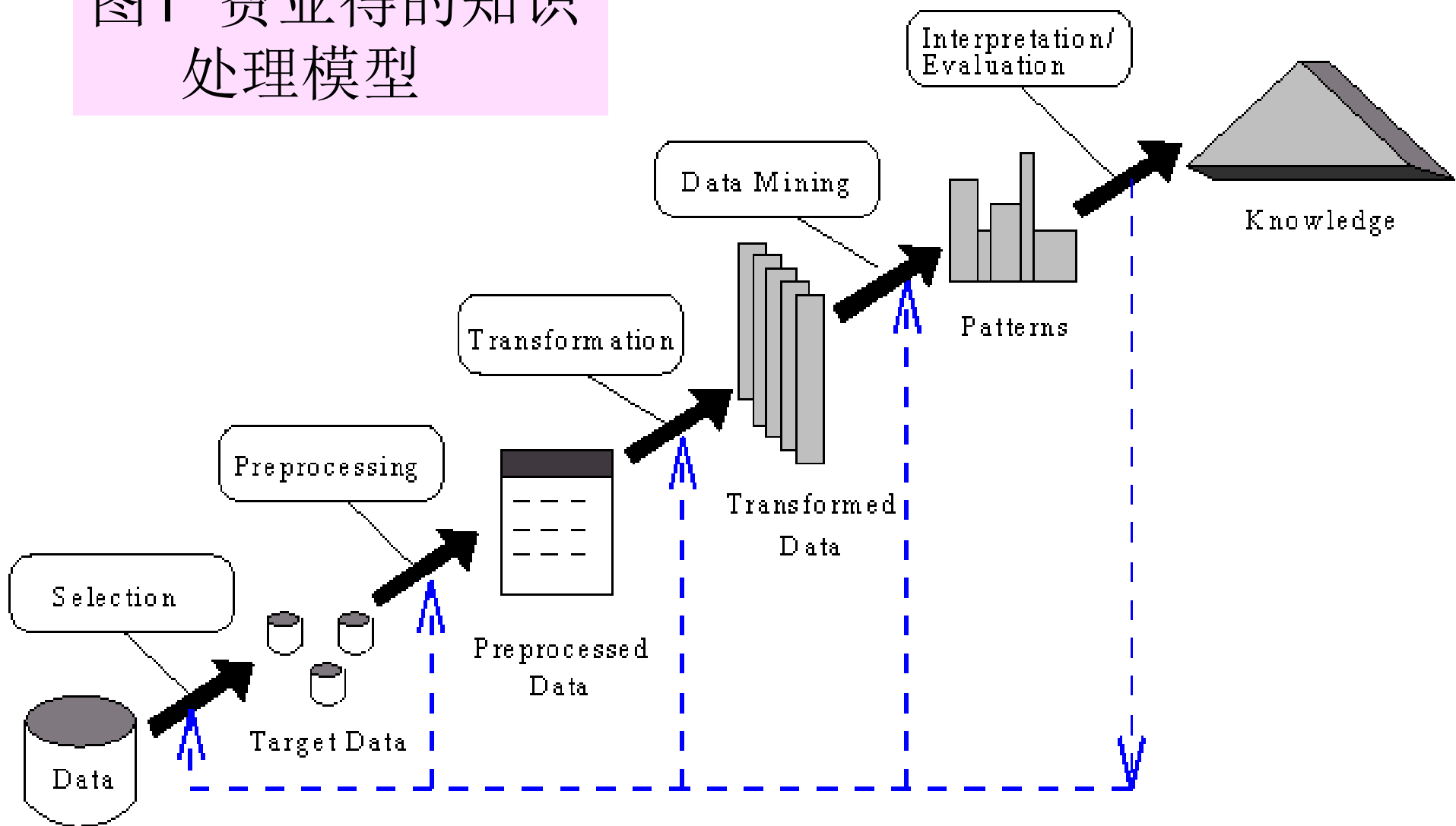
下面我们介绍两类较具代表性的处理过程模型

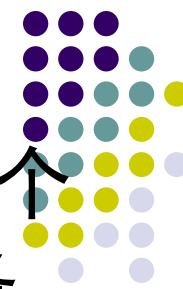
- 多处理阶段过程模型（1）

多处理阶段模型将数据库中的知识发现看做是一个多阶段的处理过程，图1是费亚得 Usama M.Fayyad 等人给出的处理模型。



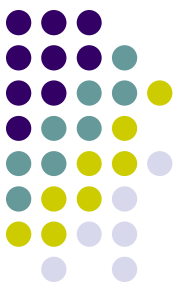
图1 费亚得的知识处理模型





- 在图1处理模型中，KDD处理过程共分为九个处理阶段，这九个处理阶段分别是数据准备、数据选择、数据预处理、数据缩减、KDD目标确定、挖掘算法确定、数据挖掘、模式解释及知识评价。每个处理阶段KDD系统会提供处理工具完成相应的工作。在对挖掘的知识进行评测后，根据结果可以决定是否重新进行某些处理过程，在处理的任意阶段都可以返回以前的阶段进行再处理。



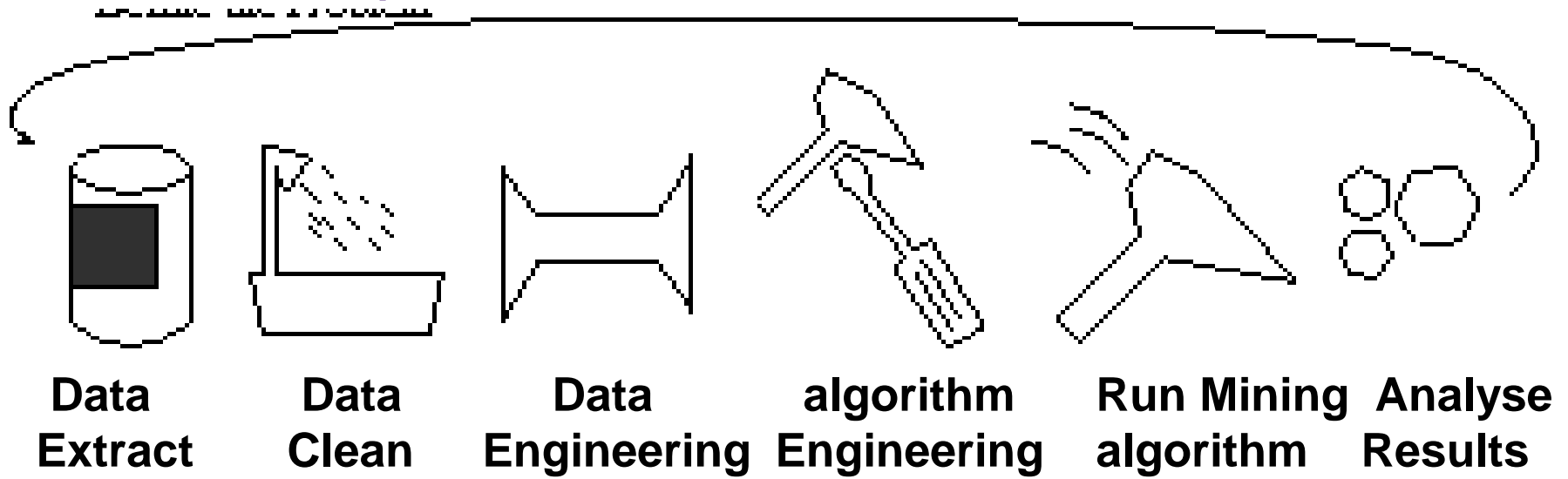


- 多处理阶段过程模型（2）
- 下面我们介绍的数据挖掘处理过程模型是 George H.John 在他的博士论文中提出的，虽然在某些地方与费亚得给出的处理模型有一些区别，但这种区别主要表现在对整个处理过程的组织和表达方式上，在内容上两者并没有非常本质的区别。



图2 John的知识处理模型

## Define the problem





- 以用户为中心的处理模型

Brachman & Anand从用户的角度对KDD处理过程进行了分析。他们认为数据库中的知识发现应该更着重于对用户进行知识发现的整个过程的支持，而不是仅仅限于在数据挖掘的一个阶段上。通过对很多KDD用户在实际工作中遇到的问题了解，他们发现用户的很大一部分工作量是与数据库的交互上。所以他们在开发数据挖掘系统IMACS (Interactive Marketing Analysis and Classification System) 时特别强调对用户与数据库交互的支持。图3给出了该模型的框图。

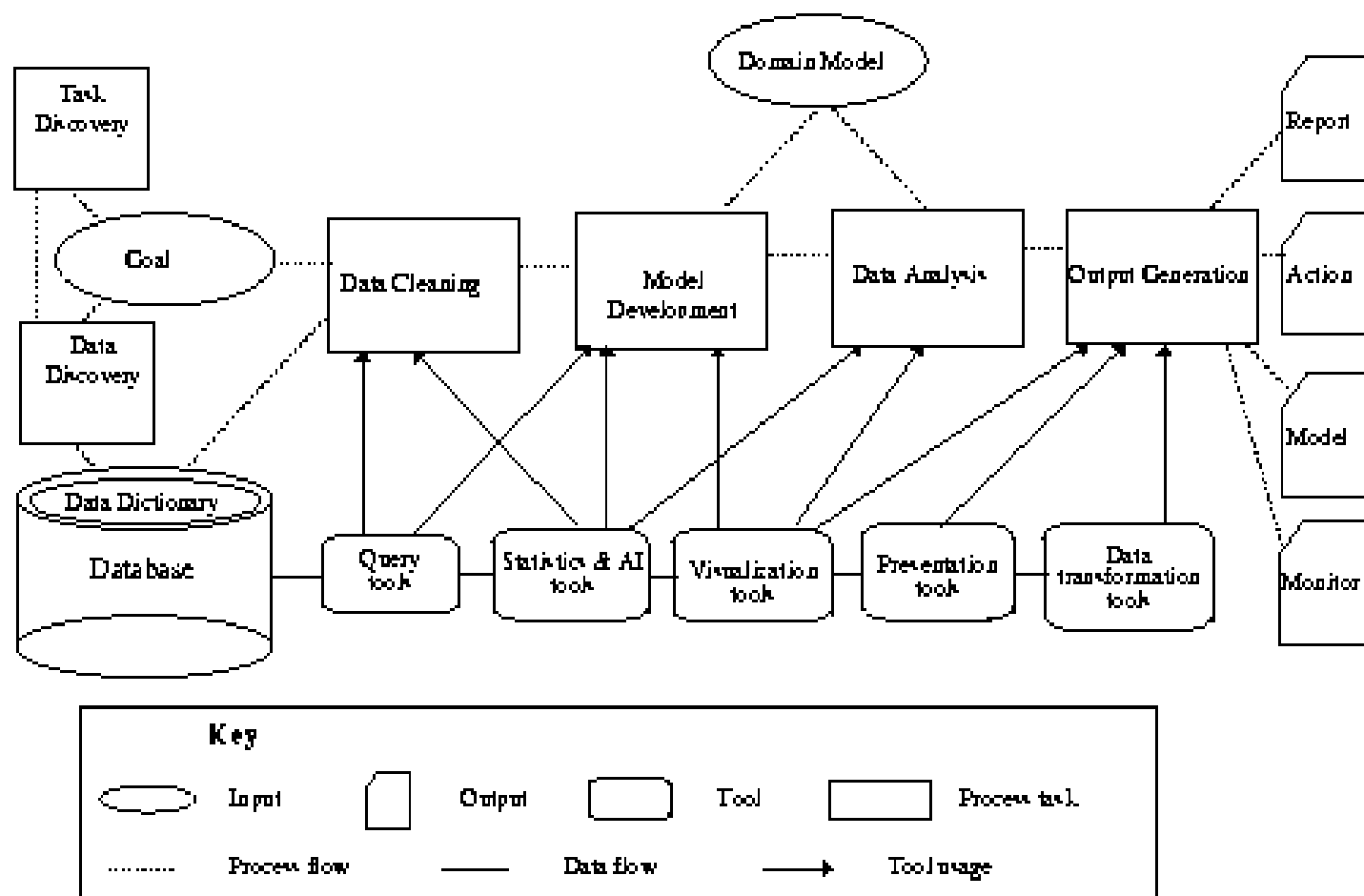
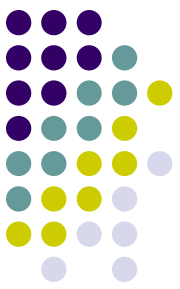


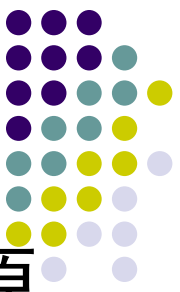
图3 以用户为中心的处理模型



# 知识发现的方法

## 1. 统计方法

- (1) **传统方法**：主要研究渐近理论，即当样本趋于无穷多时的统计性质。统计方法处理过程分三个阶段：搜索数据、分析数据、进行推理
- (2) **模糊集**：表示和处理不确定性数据的方法。不但可以处理不完全数据、噪声或不精确数据，还可以提供比传统方法更灵巧、更平滑的性能。

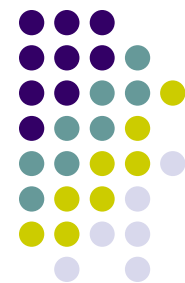


### (3) 支撑向量机SVM (support vector machine)

建立在计算学习理论的结构风险最小化原则之上，其主要思想是针对两类分类问题，在高维空间中寻找一个超平面作为两类的分割，以保证最小的分类错误率。

### (4) 粗糙集 (rough set)

由波拉克 (Pawlak) 在1982年提出。它是一种新的数学工具，用来处理含糊性和不确定性。粗糙集是由集合的下近似、上近似来定义的。边界区的成员可能是该集合的成员，但不是确定的成员。它常与规则归纳、分类和聚类方法结合起来使用，很少单独使用。



## 2. 机器学习方法

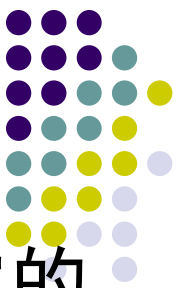
### (1) 规则归纳

规则反映数据项中某些属性或数据集中某些数据项之间的统计相关性。

### (2) 决策树

应用于分类和预测，提供了一种展示在什么条件下会得到什么值这类规则的方法，一个决策树表示一系列的问题，每个问题决定了继续下去的问题会是什么。

适合于处理非数值型数据，需要在生成决策树后再对决策树进行剪枝处理，最后将决策树转化为规则，用于对新事例进行分类。



### (3) 范例推理

直接使用过去的经验或解法来求解给定的问题。当给定一个特定问题时，范例推理就检索范例库，寻找相似的范例。如果存在相似的范例，它们的解法就可以用来求解新的问题。该新问题被加到范例库，以便将来参考。

### (4) 贝叶斯信念网络

是概率分布的图的表示，是一种直接的、非循环的图。节点表示属性变量，边表示属性变量之间的依赖关系。





## (5) 科学发现

在试验环境下发现科学定理。

## (6) 遗传算法

模仿人工选择培育良种的思路，从一个初始规则集合开始，迭代的通过交换对象成员产生群体（繁殖），评估并择优复制逐代积累计算，最终得到最有价值的知识集。

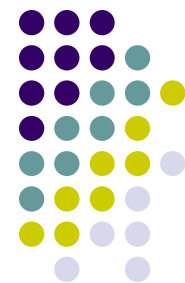


### 3. 神经网络方法

人工神经网络方法：前馈式网络、反馈式网络和自组织网络。主要用于分类、聚类、特征挖掘、预测等方面。通过向一个训练数据集学习和应用所学知识生成分类和预测的模式。

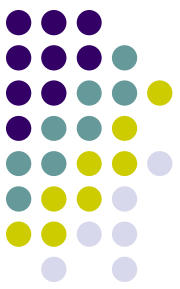
### 4. 可视化方法

可视化就是把数据、信息和知识转化为可视的表示形式的过程。信息的可视化就是要处理这些数据种类以及它们在商业和信息技术领域的相关的分析任务，以发现信息中的模式、聚类、区别、联系和趋势等。



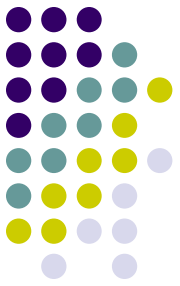
# 知识发现的应用

- 应用领域
  - 1. 金融业
  - 2. 保险业
  - 3. 零售业
  - 4. 科学研究
  - 5. 其他领域
- 典型案例



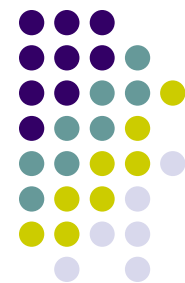
# 市场分析和管理的

- 数据从哪里来？
  - 信用卡交易，会员卡，商家的优惠券，消费者投诉电话，公众生活方式研究
- 目标市场
  - 构建一系列的“客户群模型”，这些顾客具有相同特征：兴趣爱好，收入水平，消费习惯，等等
  - 确定顾客的购买模式
- 应用1：交叉市场分析
  - 货物销售之间的相互联系和相关性，以及基于这种联系上的预测

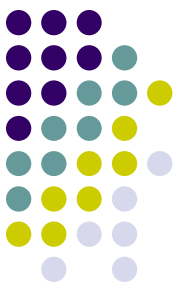


- 应用2：顾客分析
  - 哪类顾客购买那种商品（聚类分析或分类预测）
- 应用3：客户需求分析
  - 确定适合不同顾客的最佳商品
  - 预测何种因素能够吸引新顾客
- 应用4：提供概要信息
  - 多维度的综合报告
  - 统计概要信息（数据的集中趋势和变化）

# 公司分析和风险管理

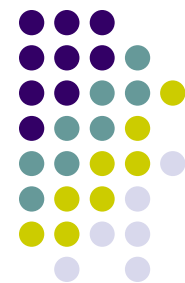


- 财务计划
  - 现金流转分析和预测
  - 交叉区域分析和时间序列分析（财务资金比率，趋势分析等等）
- 资源计划
  - 总结和比较资源和花费
- 竞争
  - 对竞争者和市场趋势的监控
  - 将顾客按等级分组和基于等级的定价过程
  - 将定价策略应用于竞争更激烈的市场中



# 欺诈行为检测和异常模式的发现

- 方法：对欺骗行为进行聚类 and 建模，并进行孤立点分析
- 应用：保险、卫生保健、零售业、信用卡服务、电信等
  - 汽车保险：对相撞事件的分析，发掘索赔是否是欺诈行为
  - 洗钱：发现可疑的货币交易行为
  - 医疗保险
    - 职业病人，医生以及相关数据分析
    - 不必要的或相关的测试
  - 电信：电话呼叫欺骗行为
    - 电话呼叫模型：呼叫目的地，持续时间，日或周呼叫次数. 分析该模型发现与期待标准的偏差
  - 零售产业
    - 分析师估计有38%的零售额下降是由于雇员的不诚实行为造成的



# 机器学习和模式识别的区别

- 机器学习
- 要使计算机具有知识一般有两种方法：一种是由知识工程师将有关的知识归纳、整理，并且表示为计算机可以接受、处理的方式输入计算机。另一种是使计算机本身有获得知识的能力，它可以学习人类已有的知识，并且在实践过程中不总结、完善，这种方式称为机器学习。
- 机器学习的研究，主要在以下三个方面进行：一是研究人类学习的机理、人脑思维的过程；和机器学习的方法；以及建立针对具体任务的学习系统。
- 机器学习的研究是在信息科学、脑科学、神经心理学、逻辑学、模糊数学等多种学科基础上的。依赖于这些学科而共同发展。目前已经取得很大的进展，但还没有能完全解决问题。
- 模式识别
- 模式识别是研究如何使机器具有感知能力，主要研究视觉模式和听觉模式的识别。如识别物体、地形、图象、字体（如签字）等。在日常生活各方面以及军事上都有广大的用途。近年来迅速发展起来应用模糊数学模式、人工神经网络模式的方法逐渐取代传统的用统计模式和结构模式的识别方法。特别神经网络方法在模式识别中取得较大进展。