

# 人工智能与机器学习

## 第九讲 概念学习

倪东

工业控制研究所

杭州·浙江大学·2022



# 机器学习概论

- 什么是机器学习
- 机器学习研究的意义
- 机器学习简要发展历史回顾



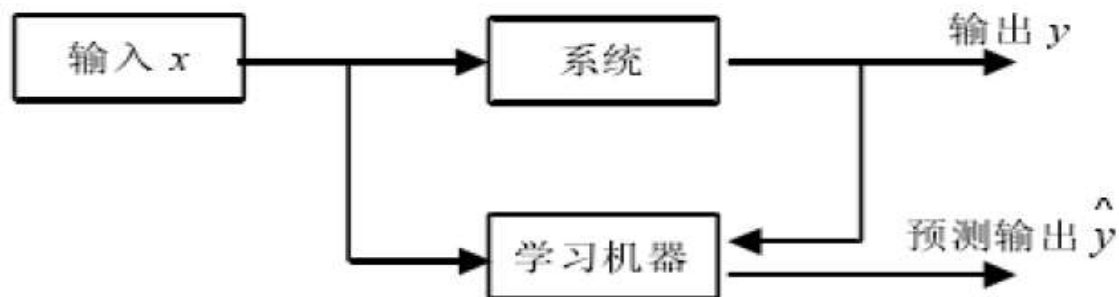
## 第九讲 概念学习

- 概念学习
  - 给定某一类别的若干正例和反例，从中获得该类别的一般定义。
- 搜索的观点
  - 在预定义的假设空间中搜索假设，使其与训练样例有最佳的拟合。
  - 利用假设空间的偏序结构
- 算法收敛到正确假设的条件？
- 归纳学习的本质，从训练数据中泛化的理由？



# 简介

- 许多机器学习涉及到从特殊训练样例中得到一般概念。
- 概念，可被看作一个对象或事件集合，它是从更大的集合中选取的子集，或在这个较大集合中定义的布尔函数。
- 概念学习问题的定义
  - 给定一个样例集合以及每个样例是否属于某个概念的标注，怎样推断出该概念的一般定义。又称从样例中逼近布尔函数。
  - 概念学习是指从有关某个布尔函数的输入输出训练样例中推断出该布尔函数。



# 概念学习任务

- 一个例子
  - 目标概念
    - **Aldo进行水上运动的日子，表示为布尔函数EnjoySport**
  - 任务目的
    - **基于某天的各属性，预测EnjoySport的值**
- 给定一个样例集D
  - **每个样例表示为6个属性的集合**

## 概念学习任务（2）

表2-1 目标概念EnjoySport的训练样例

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

## 概念学习任务 (3)

- 表示假设的形式（目标函数的表示）
  - 一个简单的形式，实例的各属性约束的合取式
  - 令每个假设为6个约束（或变量）的向量，每个约束对应一个属性可取值范围，为
    - ? 任意本属性可接受的值
    - 明确指定的属性值
    - $\phi$  不接受任何值
- 假设的例子
  - $\langle ?, \text{Cold}, \text{High}, ?, ?, ? \rangle$
  - $\langle ?, ?, ?, ?, ?, ? \rangle$  // 所有的样例都是正例
  - $\langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$  // 所有的样例都是反例

# 概念学习任务 (4)

## 形式化描述:

- 已知
  - 实例集X
    - **每个实例x由6个属性描述，每个属性的取值范围已确定**
  - 假设集H
    - **每个假设h描述为6个属性的取值约束的合取**
  - 目标概念c
    - **一个布尔函数，变量为实例**
  - 训练样例集D
    - **目标函数（或目标概念）的正例和反例**
- 求解
  - H中的一假设h，使对于X中任意x， $h(x)=c(x)$



# 术语定义

- 实例 $x$
- 实例集 $X$
- 概念
- 目标概念 $c$
- 训练样例 $x$
- 训练样例集 $D$
- 正例，目标概念成员
- 反例，非目标概念成员
- 假设 $h$
- 假设集 $H$

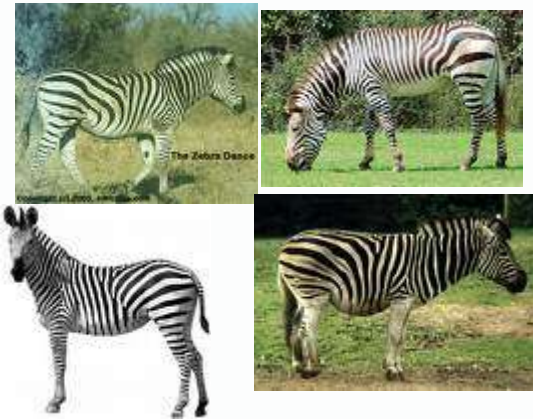
机器学习的目标就是寻找一个假设 $h$ ，使得对所有的 $x$ ，都有 $h(x)=c(x)$

# 归纳学习假设

- 什么是归纳学习？
  - 从特殊的样例得到普遍的规律（从特殊到一般）
- 归纳
  - 只能保证输出的假设能与训练样例相拟合
- 归纳假设的一个基本假定
  - 对于未见实例最好的假设就是与训练数据最佳拟合的假设
- 归纳学习假设
  - 任一假设如果在足够大的训练样例集中很好地逼近目标函数，它也能在未见实例中很好地逼近目标函数。

# Cannot Learn Exact Concepts from Limited Data, Only Approximations

Positive



Negative



 Learner



Classifier

Negative

**Wrong!**

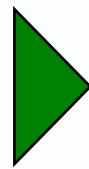


# Cannot Learn Even Approximate Concepts from Pathological Training Sets

Positive



Negative



Learner



Classifier



Negative

**Wrong!**



# 作为搜索的概念学习

- 概念学习可以看作一个 搜索的过程
  - 搜索范围：假设的表示所隐含定义的整个空间
  - 搜索目标：能够最好地拟合训练样例的假设
- 当假设的表示形式选定后，那么就隐含地为学习算法确定了所有假设的空间
  - 例子EnjoySport的假设空间，如果属性 *Sky* 有3种可能的值，而 *AirTemp*、*Humidity*、*Wind*、*Water*和 *Forecast*都只有两种可能值。
    - 实例空间  $X$ ：包含  $3 \times 2 \times 2 \times 2 \times 2 \times 2 = 96$  种不同的实例
    - 假设空间  $H$ 
      - 包含  $5 \times 4 \times 4 \times 4 \times 4 \times 4 = 5120$  种 **语法不同** 的假设
      - 由于：包含  $\emptyset$  符号的假设将每个实例都分类为反例。因此，**语义不同** 的假设只有  $1 + 4 \times 3 \times 3 \times 3 \times 3 \times 3 = 973$  个。

# 假设的一般到特殊序

- 假设的一般到特殊序关系
  - 考虑下面两个假设
    - $h_1 = \langle \text{Sunny}, ?, ?, \text{Strong}, ?, ? \rangle$
    - $h_2 = \langle \text{Sunny}, ?, ?, ?, ?, ? \rangle$
  - 任何被 $h_1$ 划分为正例的实例都会被 $h_2$ 划分为正例，因此 $h_2$ 比 $h_1$ 更一般。
- 利用这个关系，无需列举所有假设，就能在无限的假设空间中进行彻底的搜索



## 假设的一般到特殊序 (2)

- 关系 “更一般” 的精确定义
  - 任给实例  $x$  和假设  $h$ , 说  $x$  满足  $h$ , 当且仅当  $h(x)=1$
  - 令  $h_j$  和  $h_k$  是在  $X$  上定义的布尔函数, 称  $h_j$  比  $h_k$  更一般, 当且仅当  $(\forall x \in X)[(h_k(x)=1) \rightarrow (h_j(x)=1)]$
  - 记为  $h_j \text{ more\_general\_than\_or\_equal\_to } h_k$ , 或  $h_j \geq_g h_k$

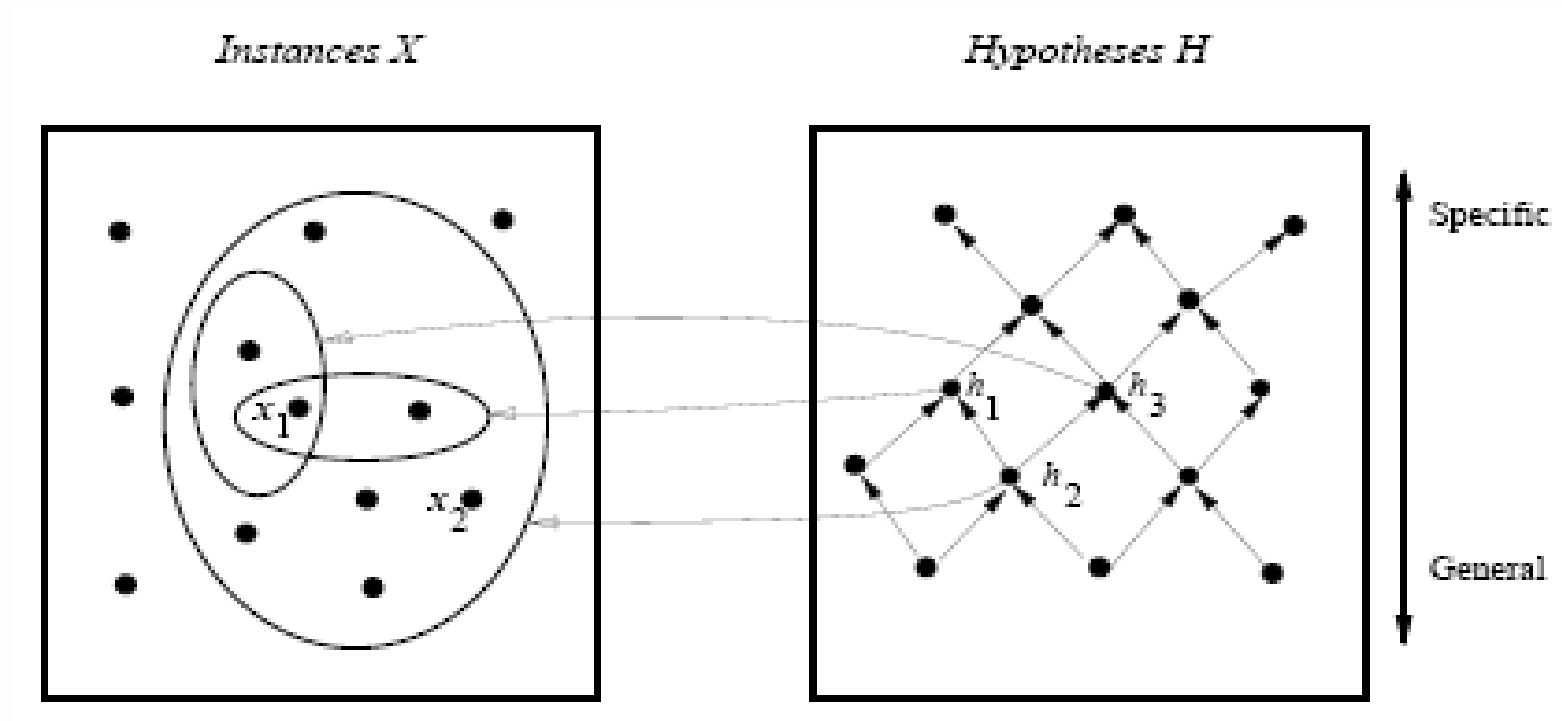


## 假设的一般到特殊序 (3)

- “更一般” 的严格情形
  - $h_j >_g h_k$ , 当且仅当,  $(h_j \geq_g h_k) \wedge \neg (h_k \geq_g h_j)$
- “更特殊” 关系的定义
  - $h_j \leq_g h_k$ , 当且仅当,  $h_k \geq_g h_j$
- 以EnjoySport为例说明上面的定义
- 偏序的特点（区别于全序），全序上的搜索可以是二分法，偏序的搜索比无序简单，比全序复杂。
- 这个偏序关系的定义与目标概念无关







$x_1 = \langle \text{Sunny Warm High Strong Cool Same} \rangle$

$x_2 = \langle \text{Sunny Warm High Light Warm Same} \rangle$

$h_1 = \langle \text{Sunny ? ? Strong ? ?} \rangle$

$h_2 = \langle \text{Sunny ? ? ? ? ?} \rangle$

$h_3 = \langle \text{Sunny ? ? ? ? Cool?} \rangle$



# Find-S: 寻找极大特殊假设

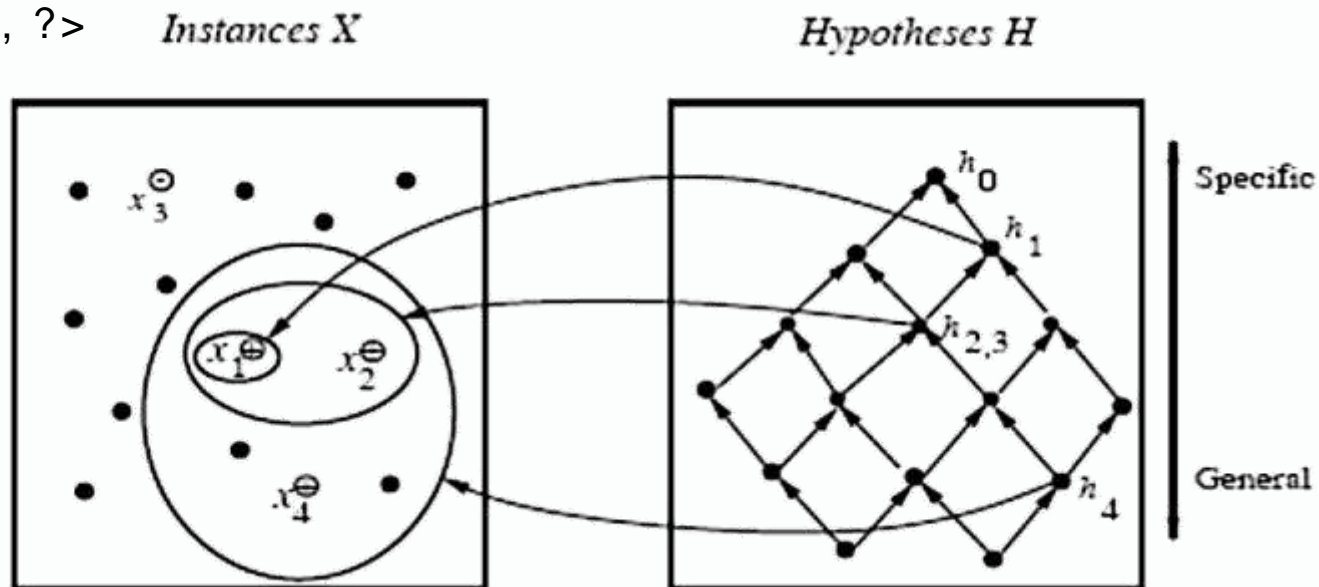
- 使用more\_general\_than偏序的搜索算法
  - 从H中最特殊假设开始，然后在假设覆盖正例失败时将其一般化

## Find-S算法

1. 将h初始化为H中最特殊假设
2. 对每个正例x
  - **对h的每个属性约束 $a_i$**   
**如果x满足 $a_i$**   
**那么不做任何处理**  
**否则将h中 $a_i$ 替换为x满足的另一个更一般约束**
3. 输出假设h

# Find-S: 寻找极大特殊假设 (2)

- Find-S算法在例子EnjoySport上的应用
  - $h \leftarrow \langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$
  - $h \leftarrow \langle \text{Sunny, Warm, Normal, Strong, Warm, Same} \rangle$
  - $h \leftarrow \langle \text{Sunny, Warm, ?, Strong, Warm, Same} \rangle$
  - 遇到反例,  $h$ 不变 (因为 $h$ 已经能够正确地识别反例)
  - $h \leftarrow \langle \text{Sunny, Warm, ?, Strong, ?, ?} \rangle$



$x_1 = \langle \text{Sunny Warm Normal Strong Warm Same} \rangle, +$   
 $x_2 = \langle \text{Sunny Warm High Strong Warm Same} \rangle, +$   
 $x_3 = \langle \text{Rainy Cold High Strong Warm Change} \rangle, -$   
 $x_4 = \langle \text{Sunny Warm High Strong Cool Change} \rangle, +$

$h_0 = \langle \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset \rangle$   
 $h_1 = \langle \text{Sunny Warm Normal Strong Warm Same} \rangle$   
 $h_2 = \langle \text{Sunny Warm ? Strong Warm Same} \rangle$   
 $h_3 = \langle \text{Sunny Warm ? Strong Warm Same} \rangle$   
 $h_4 = \langle \text{Sunny Warm ? Strong ? ?} \rangle$

## Find-S: 寻找极大特殊假设 (3)

- Find-S算法演示了一种利用more\_general\_than偏序来搜索假设空间的方法，沿着偏序链，从较特殊的假设逐渐转移到较一般的假设。因此，每一步得到的假设都是在那一点上与训练样例一致的最特殊的假设。
- Find-S的重要特点：对以属性约束的合取式描述的假设空间H，保证输出为H中与正例一致的最特殊的假设。
- 存在的问题
  - 是否收敛到了正确的目标概念？
  - 为什么要用最特殊的假设？
  - 训练样例是否相互一致？
  - 如果有多个极大特殊假设怎么办？



# 变型空间和候选消除算法

- 候选消除算法概说
  - 概念学习的另一种方法，候选消除算法 (candidate-elimination)
  - Find-S算法的不足，输出的假设只是H中能够拟合训练样例的多个假设中的一个
  - 候选消除算法输出与训练样例一致的所有假设的集合
  - 候选消除算法在描述这一集合时不需要明确列举所有成员
  - 利用more\_general\_than偏序结构，可以维护一个一致假设集合的简洁表示
  - 候选消除算法的应用：化学质谱分析、启发式搜索的控制规则
  - 候选消除算法的缺点：容错性能差



## 变型空间和候选消除算法 (2)

- “一致” 的定义
  - 一个假设  $h$  与训练样例集合  $D$  一致，当且仅当对  $D$  中每一个样例  $\langle x, c(x) \rangle$  都有  $h(x) = c(x)$ ，即
$$\text{Consistent}(h, D) \Leftrightarrow (\forall \langle x, c(x) \rangle \in D) h(x) = c(x)$$
  - “一致” 与 “满足” 的关系
- 变型空间 (Version Space)
  - 与训练样例一致的所有假设组成的集合
  - 表示了目标概念的所有合理的变型
- 关于  $H$  和  $D$  的变型空间，记为  $VS_{H,D}$ ，是  $H$  中与训练样例  $D$  一致的所有假设构成的子集
$$VS_{H,D} = \{h \in H \mid \text{Consistent}(h, D)\}$$

# 变型空间和候选消除算法 (3)

## • 列表后消除算法

表示变型空间的一种方法是列出其所有成员

- 变型空间  $\leftarrow$  包含  $H$  中所有假设的列表
- 对每个训练样例  $\langle x, c(x) \rangle$ , 从变型空间中移除所有  $h(x) \neq c(x)$  的假设
- 输出 Version Space 中的假设列表

## • 优点

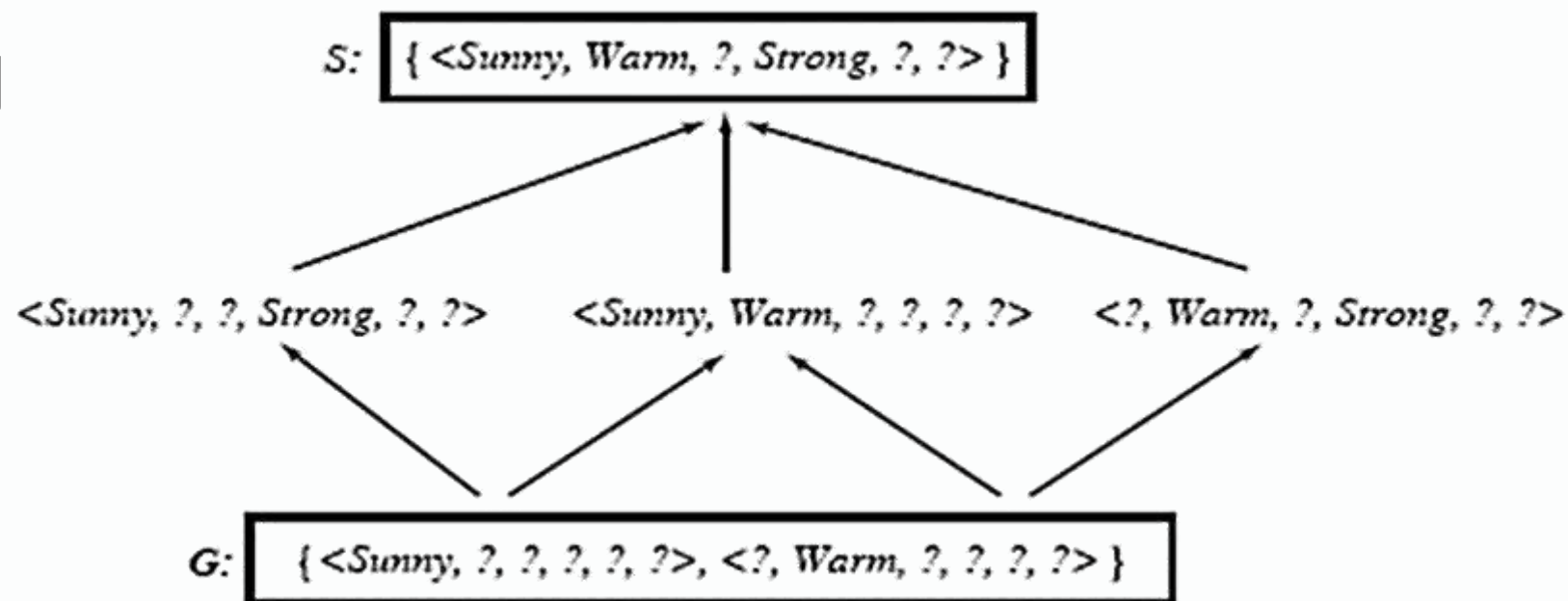
- 保证得到所有与训练数据一致的假设

## • 缺点

- 非常繁琐地列出  $H$  中的所有假设, 大多数实际的假设空间无法做到

## 变型空间和候选消除算法 (4)

- 变型空间的更简洁表示
  - 变型空间被表示为它的极大一般和极大特殊的成员
  - 这些成员形成了一般和特殊边界的集合，这些边界在整个偏序结构中划分出变型空间
  - 以EnjoySport为例





## 变型空间和候选消除算法 (5)

- 形式化定义
  - 极大一般
  - 极大特殊
  - 关于假设空间 $H$ 和训练数据 $D$ 的**一般边界** $G$ ，是在 $H$ 中与 $D$ 相一致的极大一般成员的集合
  - 关于假设空间 $H$ 和训练数据 $D$ 的**特殊边界** $S$ ，是在 $H$ 中与 $D$ 相一致的极大特殊成员的集合

## 变型空间和候选消除算法 (6)

**变型空间表示定理:** 令 $X$ 为一任意的实例集合,  $H$ 为 $X$ 上定义的布尔假设的集合。令 $c: X \rightarrow \{0,1\}$ 为 $X$ 上定义的任一目标概念, 并令 $D$ 为任一训练样例集合 $\{ \langle x, c(x) \rangle \}$ 。对所有的 $X, H, c, D$ 以及良好定义的 $S$ 和 $G$ :

$$VS_{H,D} = \{h \in H \mid (\exists s \in S)(\exists g \in G)(g \geq_g h \geq_g s)\}$$

证明: 只需证明: 1) 每一个满足上式右边的 $h$ 都在 $VS_{H,D}$ 中, 2)  $VS_{H,D}$ 的每个成员都满足都满足等式右边。...



# 变型空间和候选消除算法 (7)

## 候选消除算法

1. 初始化G和S

2. 如果d是一个正例

- 从G中移去所有与d不一致的假设
- 对S中每个与d不一致的假设s
  - 从S中移去s
  - 把s的所有的极小一般式h加入到S中, 其中h满足h与 d一致, 而且G的某个成员比h更一般
  - 从S中移去所有这样的假设: 它比S中另一个假设更一般

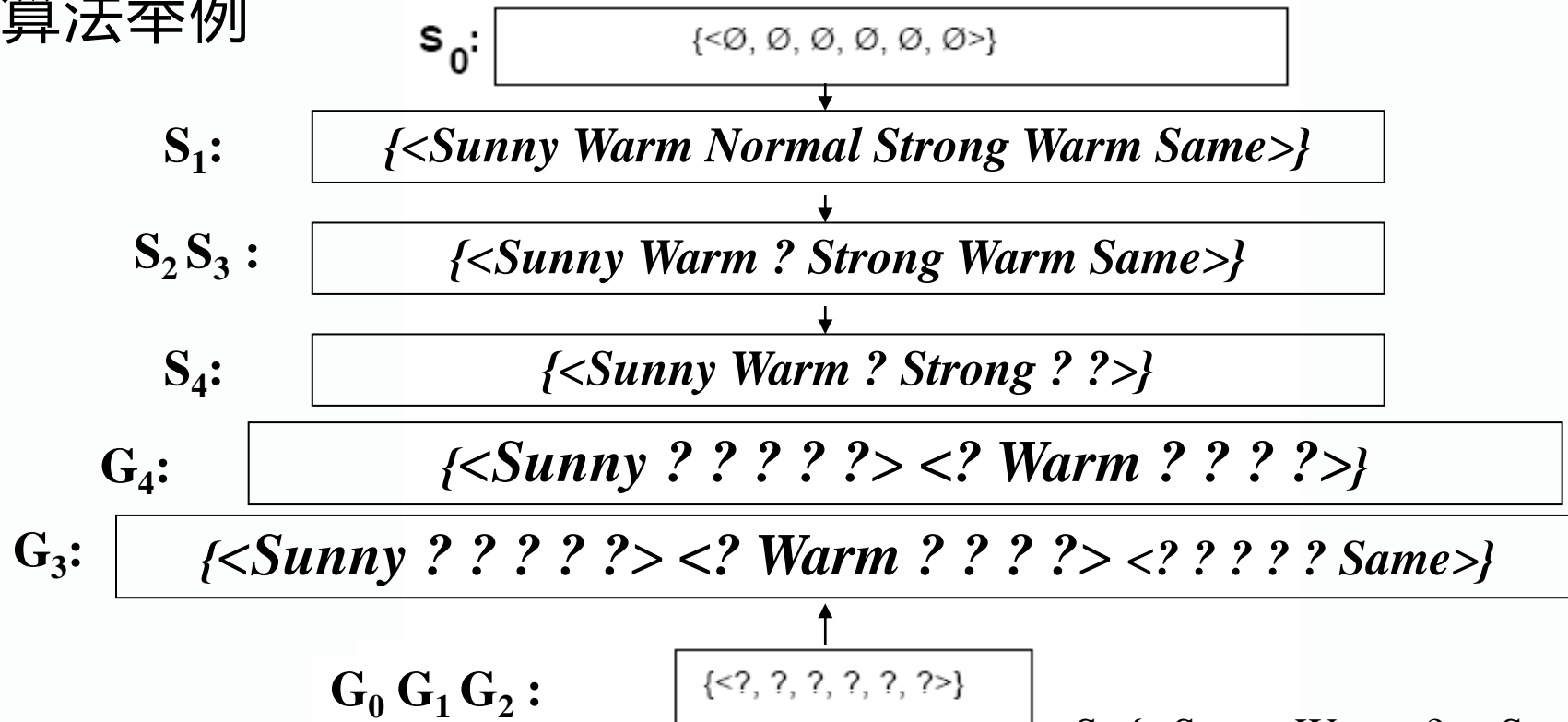
3. 如果d是一个反例

- 从S中移去所有与d不一致的假设
- 对G中每个与d不一致的假设g
  - 从G中移去g
  - 把g的所有的极小特殊化式h加入到G中, 其中h满足
    - h与d一致, 而且S的某个成员比h更特殊
  - 从G中移去所有这样的假设: 它比G中另一个假设更特殊



# 变型空间和候选消除算法 (8)

## • 算法举例



$x_1 = \langle \text{Sunny Warm Normal Strong Warm Same} \rangle, +$   
 $x_2 = \langle \text{Sunny Warm High Strong Warm Same} \rangle, +$   
 $x_3 = \langle \text{Rainy Cold High Strong Warm Change} \rangle, -$   
 $x_4 = \langle \text{Sunny Warm High Strong Cool Change} \rangle, +$

$S_3: \{\langle \text{Sunny Warm ? Strong Warm Same} \rangle\}$   
 $x_3 = \langle \text{Rainy Cold High Strong Warm Change} \rangle$   
 $\langle \text{Sunny ? ? ? ? ?} \rangle$   
 $\langle ? \text{ Warm ? ? ? ? ?} \rangle$   
 $\langle ? ? ? ? ? \text{ Same} \rangle$

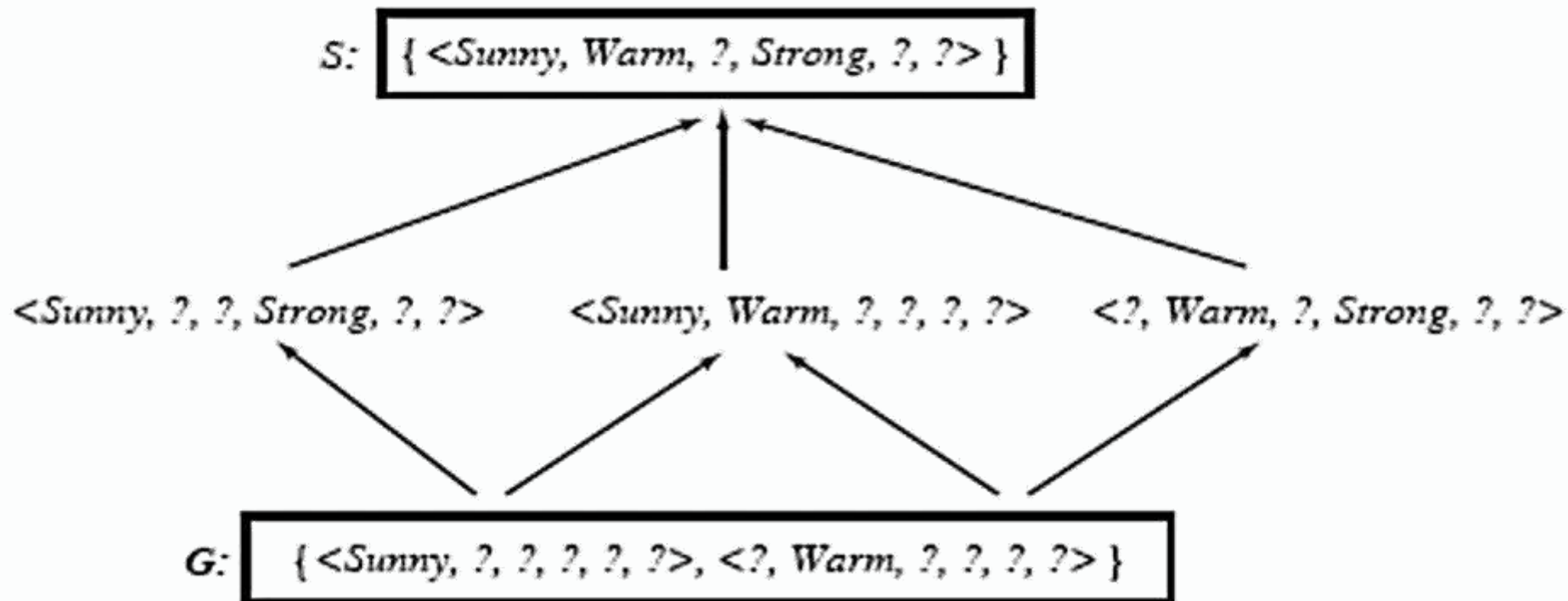


图7 最终变型空间

# 变型空间和候选消除的说明

- 候选消除算法收敛到正确的假设
  - 训练样例中没有错误
  - $H$ 中确实包含描述目标概念的正确假设
- 如果样例中存在错误
  - 如果给定足够的训练数据，我们会发现 $S$ 和 $G$ 边界收敛得到一个空的变型空间
- 如果目标概念不能由假设表示方式所描述
  - 比如是约束的析取
  - $\langle \text{Sunny}, ?, ?, ?, ?, ? \rangle \vee \langle \text{Cloudy}, ?, ?, ?, ?, ? \rangle$

## 变型空间和候选消除（2）

- 下一步需要什么样的训练样例
  - 一般来说，概念学习的最优查询策略，是产生实例以满足当前变型空间中大约半数的假设。这样，变型空间的大小可以在遇到每个新样例时减半，正确的目标概念就可在只用 $\log_2|VS|$ 次实验后得到。



## 变型空间和候选消除 (3)

- 怎样使用不完全学习概念
  - 虽然图7的变型空间中仍包含多个假设，即目标概念还未学习到，但是仍然有可能对新样例进行一定可信度的分类。
  - 待分类的新实例

*⟨Sunny Warm Normal Strong Cool Change⟩*

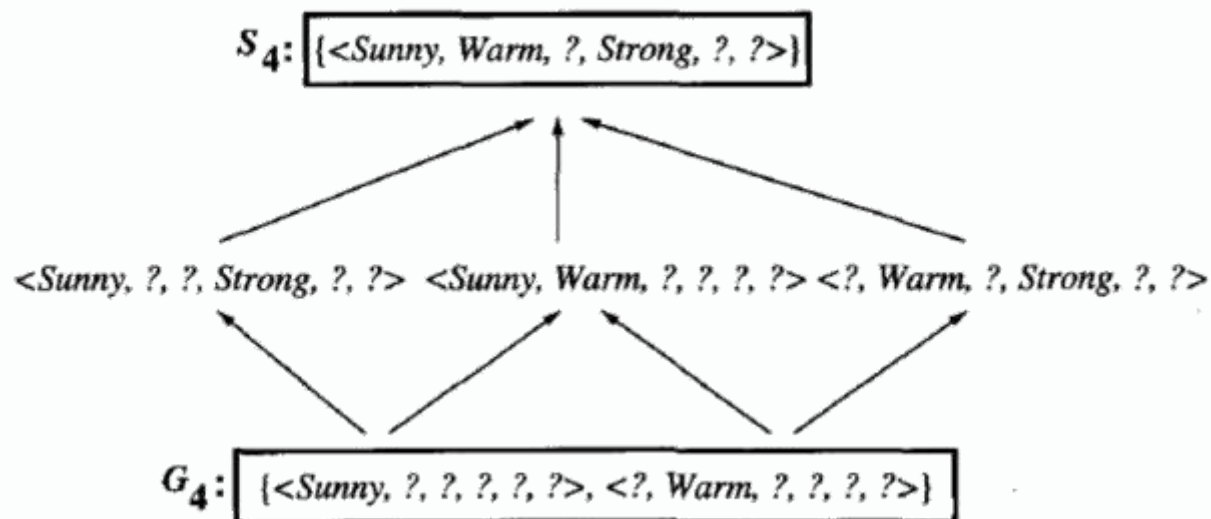
*⟨Rainy Cool Normal Light Warm Same⟩*

*⟨Sunny Warm Normal Light Warm Same⟩*





# 概念的应用



Instance	<i>Sky</i>	<i>AirTemp</i>	<i>Humidity</i>	<i>Wind</i>	<i>Water</i>	<i>Forecast</i>	<i>EnjoySport</i>
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?

# 概念的应用

- 判断是否是正例
  - 判断是否满足S中的每个假设
- 判断是否是反例
  - 判断是否不满足G中的每个假设

# 归纳偏置

- 有关候选消除算法的几个问题
  - 如果目标概念不在假设空间中怎么办？
  - 是否可设计一个包含所有假设的空间来解决这一困难？
  - 假设空间的大小对于算法推广到未见实例的能力有什么影响？
  - 假设空间的大小对所需训练样例的数量有什么影响？



## 归纳偏置 (2)

- 一个有偏的假设空间
  - 在EnjoySport这个例子中，假设空间限制为只包含属性值的合取。（有偏）
  - 这一限制，导致假设空间不能够表示最简单的析取形式的目标概念。



## 归纳偏置 (3)

- 无偏的学习器
  - 为了保证目标概念在假设空间中，需要提供一个假设空间，它能表达所有的可教授概念。换言之，它能表达实例集 $X$ 的所有子集。
  - 假设空间能够表达所有的可教授概念（比如任意析取、合取和否定式），范例中假设空间有 $2^{96}$ （实例空间大小为96）个不同的目标概念
- 问题：为什么范例中合取假设空间只能表示973个假设？



## 归纳偏置 (4)

- EnjoySport的无偏形式
  - 带来的问题：概念学习算法无法从训练样例中泛化。
  - 要想获得单个目标概念，就必须提供X中所有实例作为训练样例

假设学习器有3个正例 ( $x_1, x_2, x_3$ ) 以及两个反例 ( $x_4, x_5$ )

则变型空间S边界包含的假设是三个正例的析取

$S: \{ (x_1 \vee x_2 \vee x_3) \}$

G边界由刚好能够排除反例的假设组成

$G: \{ \text{NOT}(x_4 \vee x_5) \}$

只能对已知样本进行无歧义分类，未知样本投票也无效

## 归纳偏置 (5)

- 无偏学习的无用性
  - 归纳学习的一个基本属性：学习器如果不对目标概念的形式做预先的假定，它从根本上无法对未见实例进行分类
  - 归纳学习需要的预先假定，称为归纳偏置



## 归纳偏置 (6)

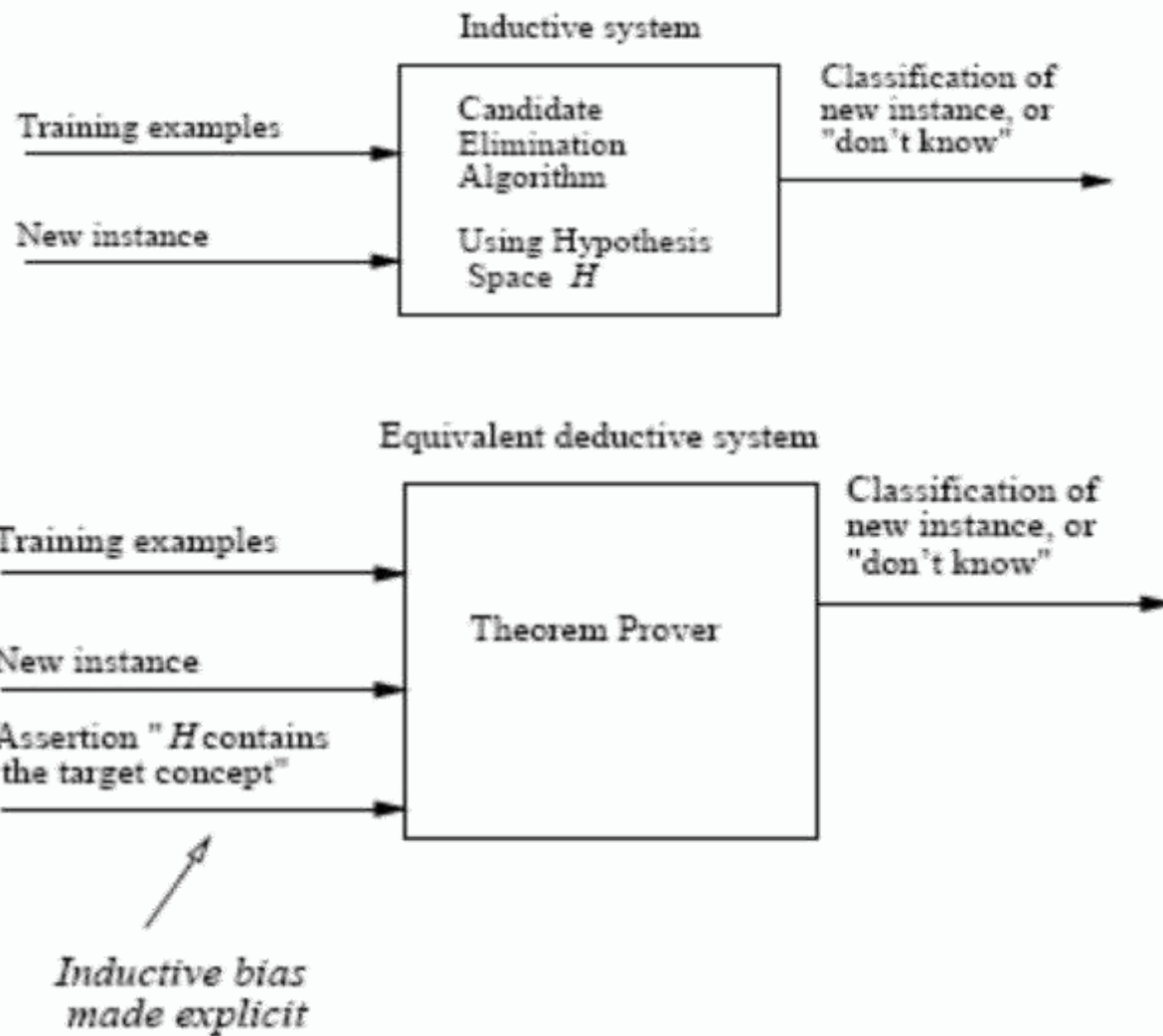
- 归纳偏置的精确定义
  - 考虑任意学习算法 $L$ ，以及任意目标概念提供的任意训练数据 $D_c = \{ \langle x, c(x) \rangle \}$
  - $L(x_i, D_c)$  表示在训练数据 $D_c$ 学习后 $L$ 赋予 $x_i$ 的分类，该推理过程描述为
  - $(D_c \wedge x_i) \succ L(x_i, D_c)$
  - $L$ 是归纳学习算法， $L(x_i, D_c)$ 这一推论出的结果正确性无法证明，无法演绎派生
  - 需要在 $D_c \wedge x_i$ 上附加怎样的前提，以使 $L(x_i, D_c)$ 能够演绎派生。
  - $L$ 的归纳偏置定义为前提集合 $B$ ，使所有的新实例满足：  
 $(B \wedge D_c \wedge x_i) \vdash L(x_i, D_c)$        $\vdash$  表示演绎派生
  - 考虑对于实例集合 $X$ 的概念学习算法 $L$ 。令 $c$ 为 $X$ 上定义的任一概念，并令 $D_c$ 为 $c$ 的任意训练样例集合， $L(x_i, D_c)$ 表示经过 $D_c$ 训练后 $L$ 赋予实例 $x_i$ 的分类。 $L$ 的归纳偏置是最小断言集合 $B$ ，它使任意目标概念 $c$ 和相应的训练样例 $D_c$ 满足：  
 $\forall x_i \in X [(B \wedge D_c \wedge x_i) \vdash L(x_i, D_c)]$



## 归纳偏置 (6)

- 候选消除算法的归纳偏置
  - $\{c \in H\}$

使用假设空间 $H$ 的候选消除算法的输入输出行为等价于利用了断言“ $H$ 包含目标概念”的演绎定理证明器



## 归纳偏置 (7)

- 3个有偏程度不同的归纳学习算法
  - 机械式
  - 候选消除算法
  - Find-S
- 一种算法的有偏性越强，它的归纳能力越强，可以分类更多的未见实例。
- 某些归纳偏置隐含在学习器中，有些表示为断言集合，可由学习器操作。

# 小结

- 主要内容

- 概念学习可看作**搜索预定义潜在假设空间**的过程;
- 假设的一般到特殊**偏序结构**可以定义在任何概念学习问题中, 这种结构便于假设空间的搜索;
- **Find-S算法**使用一般到特殊序, 在偏序结构的一个分支上执行一般到特殊搜索, 寻找一个与样例一致的最特殊假设;
- **候选消除算法**利用一般到特殊序, 通过渐近地计算极大特殊假设集合和极大一般假设集合发现变型空间;
- 候选消除算法**缺少健壮性**, 后面会描述一些学习算法, 它们能够处理有噪声的数据和目标概念无法在假设空间中表示的情况
- 归纳学习算法**隐含了归纳偏置**, 候选消除算法的偏置是: 目标概念可以在假设空间中找到。输出的假设和对新实例的分类可由归纳偏置和训练样例演绎推出



## 思考题

- 1. 解释为什么 *EnjoySport* 学习任务的假设空间的大小为 973。如果增加一属性 *WaterCurrent*，可取值 *Light*、*Moderate* 和 *Strong*，那么可能的实例数和可能的假设数将会增加多少？推广到一般，增加一新属性 *A*，有 *k* 种取值，实例数和假设数将会增加多少？



## 思考题

- 2 在候选消除算法中，如果训练样例按 *EnjoySport* 例子中的逆序出现，请分步给出 S 和 G 边界集合。尝试对训练样例排序，以使 *EnjoySport* 例子中的所有 S 和 G 集合的中间结果的大小之和为最小？

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

## 思考题

- 3 实现Find-S算法和候选消除算法。验证它是否可成功地产生*EnjoySport*例子中各步骤结果。