

第十五章 规则学习

汇报人：杜昭武
时间:2025/3/23

1.基本概念

规则定义：

- 语义明确的逻辑规则，形如“若...则...” (If-Then)

示例：好瓜 $\leftarrow$ (色泽=乌黑) $\wedge$ (根蒂=硬挺)

$$\oplus \leftarrow \mathbf{f}_1 \wedge \mathbf{f}_2 \wedge \cdots \wedge \mathbf{f}_L$$

优势：

对比黑箱模型（如神经网络、SVM）：

- 可解释性高：规则直接展示决策逻辑
- 知识表达强：支持领域知识注入，因此规则学习能更自然地在学习过程中引入领域知识
 - 示例：爷爷(X,Y) $\leftarrow$ 父亲(X,Z) $\wedge$ 父亲(Z,Y)
- 处理复杂任务：抽象推理能力

规则1: 好瓜 $\leftarrow$ (根蒂=蜷缩) $\wedge$ (脐部=凹陷)；

规则2: $\neg$ 好瓜 $\leftarrow$ (纹理=模糊)。

显然，规则集合中的每条规则都可以看作一个子模型，规则集合是这些模型的一个集成，当一个示例被判别结果不同的多条规则覆盖时，则称发生了**冲突**。

冲突消解：

- 投票法：将判别相同的规则数最多的结果作为最终结果
- 排序法：在规则集合上定义一个**排序**，在发生冲突时使用**排序最前**的规则
- 元规则法：根据领域知识事先设定一些“元规则”，即关于规则的规则，例如“发生冲突时使用长度最小的规则”，根据元规则的指导来使用规则集。

1.基本概念

命题规则 (Propositional Rule)

- 定义：由原子命题（命题原子）与逻辑连接词（ $\wedge$, $\vee$, $\neg$, $\leftarrow$ ）构成的简单陈述句
- 组成：
 - 原子命题：描述单个属性值的判断，如根蒂=蜷缩、脐部=凹陷
 - 逻辑连接词：组合原子命题形成规则体
- 示例：
 - 好瓜 $\leftarrow$ 根蒂=蜷缩 $\wedge$ 脐部=凹陷
- 特点：
 - 表达能力有限，仅适用于结构化表格数据（如西瓜数据集）
 - 规则头与规则体均为具体属性值的组合，无变量或关系描述。

一阶规则 (First-Order Rule)

- 定义：由原子公式（含变量、谓词、函数）和量词（ $\forall, \exists$ ）构成的关系型规则
- 组成：
 - 原子公式：
 - 谓词：描述属性或关系（如自然数(X)、父亲(X,Y)）
 - 函数：定义变量间的操作（如 $\sigma(X)=X+1$ ）。
 - 量词：限定变量范围（ $\forall$ 表示“任意”， $\exists$ 表示“存在”）
- 示例：
 - $\forall X (\text{自然数}(\sigma(X)) \leftarrow \text{自然数}(X))$ $\text{好瓜}(X) \leftarrow \text{根蒂}(X, \text{蜷缩}) \wedge \text{脐部}(X, \text{凹陷})$
- 特点：
 - 支持变量和关系推理，可表达复杂逻辑（如“所有自然数加1仍是自然数”）
 - 适用于知识图谱、数学定理证明等需抽象推理的场景

2.序贯覆盖

规则学习的目标是产生一个能覆盖尽可能多的样例的规则集。最直接的做法是序贯覆盖。
在训练集上每学习到一条规则，就将该规则的训练样例去除，然后以剩下的训练样例组成训练集重复上述过程。
由于每次只处理一部分数据，因此也被称为“分治”策略。

以西瓜数据集2.0为例，
“当前规则仅覆盖正例”

好瓜 $\leftarrow$ (色泽=青绿). 1, 6, 10, 17.包含两个正例，两个反例

好瓜 $\leftarrow$ (色泽=青绿) $\wedge$ (根蒂=蜷缩). 1.6.10.17

好瓜 $\leftarrow$ (色泽=青绿) $\wedge$ (根蒂=稍蜷). 6仅包含一个正例

规则 1: 好瓜 $\leftarrow$ (色泽=青绿) $\wedge$ (根蒂=稍蜷);

规则 2: 好瓜 $\leftarrow$ (色泽=青绿) $\wedge$ (敲声=浊响);

规则 3: 好瓜 $\leftarrow$ (色泽=乌黑) $\wedge$ (根蒂=蜷缩);

规则 4: 好瓜 $\leftarrow$ (色泽=乌黑) $\wedge$ (纹理=稍糊).

编号	色泽	根蒂	敲声	纹理	脐部	触感	好瓜
1	青绿	蜷缩	浊响	清晰	凹陷	硬滑	是
2	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是
3	乌黑	蜷缩	浊响	清晰	凹陷	硬滑	是
4	青绿	蜷缩	沉闷	清晰	凹陷	硬滑	是
5	浅白	蜷缩	浊响	清晰	凹陷	硬滑	是
6	青绿	稍蜷	浊响	清晰	稍凹	软粘	是
7	乌黑	稍蜷	浊响	稍糊	稍凹	软粘	是
8	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是
9	乌黑	稍蜷	沉闷	稍糊	稍凹	硬滑	否
10	青绿	硬挺	清脆	清晰	平坦	软粘	否
11	浅白	硬挺	清脆	模糊	平坦	硬滑	否
12	浅白	蜷缩	浊响	模糊	平坦	软粘	否
13	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否
14	浅白	稍蜷	沉闷	稍糊	凹陷	硬滑	否
15	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否
16	浅白	蜷缩	浊响	模糊	平坦	硬滑	否
17	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否

2.序贯覆盖

基于以上穷尽搜索的做法在属性和候选值较多时由于**组合爆炸**而不可行，所以现实任务中一般有两种策略来产生规则：

- 自顶向下（生成-测试法）
- 自底向上（数据驱动）

好瓜 $\leftarrow$ (色泽=乌黑).

先考虑规则准确率，准确率相同的时候考虑覆盖样例数

好瓜 $\leftarrow$ (色泽=乌黑) $\wedge$ (根蒂=蜷缩).



3.剪枝优化

规则生成的本质是一个**贪心搜索过程**，需要有一定的机制来缓解过拟合的风险，最常见的方法就是“剪枝”

- 预剪枝：剪枝发生在**规则生长过程中** – CN2算法
- 后剪枝：发生在规则**产生后** – 减错剪枝
- **CN2算法：剪枝可以借助显著性检验来进行。用规则集进行预测必须显著优于直接基于训练样例集后验概率分布进行预测**，为了方便计算，CN2使用了似然率统计量LRS

$$LRS = 2 \cdot \left(\hat{m}_+ \log_2 \frac{\left(\frac{\hat{m}_+}{\hat{m}_+ + \hat{m}_-} \right)}{\left(\frac{m_+}{m_+ + m_-} \right)} + \hat{m}_- \log_2 \frac{\left(\frac{\hat{m}_-}{\hat{m}_+ + \hat{m}_-} \right)}{\left(\frac{m_-}{m_+ + m_-} \right)} \right),$$

LRS越大：说明采用规则（集）进行预测与直接使用训练集正，反例比率进行猜测的差别越大

LRS越小：说明规则（集）的效果越可能仅是偶然现象

在数据量比较大的现实任务中，通常设置为在LRS很大（例如0.99）时CN2算法才停止规则（集）生长

3.剪枝优化

- **减错剪枝 (REP)**：将样例集划分为训练集和验证集
 - 训练集：从训练集上学得规则集R后进行多轮剪枝，在每一轮穷举所有可能得剪枝操作，包括删除规则中的某个文字，删除规则尾部多个文字，删除整条规则等等
 - 验证集：用验证集对剪枝产生的**所有候选规则集**进行评估，保留最好的那个规则集进行下一轮剪枝
 - 如此继续，知道无法通过剪枝提高验证集性能复杂度： $O(M^4)$ m为训练样例数
- **IREP** :生成每条规则前，先将当前样例集分为训练集和验证集
 - 训练集：在训练集上生成一条规则r
 - 验证集：**立即在验证集上**对其进行REP剪枝得到规则r1
 - 将r1的覆盖的样例去除，在更新的样例集上重复上述过程复杂度： $O(m \log 2m)$
- REP：针对**规则集**进行剪枝
- IREP：针对**单条规则**进行剪枝

3. 剪枝优化

将剪枝机制与其他一些后处理手段结合起来对规则集进行优化，则往往会有更好的效果——**规则学习算法 ROPPER**

其核心思想是通过**增量学习**和**后剪枝**优化规则集，逐步覆盖更多训练样例并减少分类错误

输入：训练样例集 D ;
重复次数 k .

过程：

- 1: $\mathcal{R} = \text{IREP}^*(D)$;
- 2: $i = 0$;
- 3: **repeat**
- 4: $\mathcal{R}' = \text{PostOpt}(\mathcal{R})$;
- 5: $D_i = \text{NotCovered}(\mathcal{R}', D)$;
- 6: $\mathcal{R}_i = \text{IREP}^*(D_i)$;
- 7: $\mathcal{R} = \mathcal{R}' \cup \mathcal{R}_i$;
- 8: $i = i + 1$;
- 9: **until** $i = k$

输出：规则集 $\mathcal{R}$

4. 一阶规则学习

对西瓜数据, 我们不妨定义:

- 色泽深度: 乌黑 > 青绿 > 浅白;
- 根蒂蜷度: 蜷缩 > 稍蜷 > 硬挺;
- 敲声沉度: 沉闷 > 浊响 > 清脆;
- 纹理清晰度: 清晰 > 稍糊 > 模糊;
- 脐部凹陷度: 凹陷 > 稍凹 > 平坦;
- 触感硬度: 硬滑 > 软粘.

关系数据

色泽更深(2, 1)	色泽更深(2, 6)	色泽更深(2, 10)	色泽更深(2, 14)
色泽更深(2, 16)	色泽更深(2, 17)	色泽更深(3, 1)	色泽更深(3, 6)
...	...	...	...
色泽更深(15, 16)	色泽更深(15, 17)	色泽更深(17, 14)	色泽更深(17, 16)
根蒂更蜷(1, 6)	根蒂更蜷(1, 7)	根蒂更蜷(1, 10)	根蒂更蜷(1, 14)
...	...	...	...
根蒂更蜷(17, 7)	根蒂更蜷(17, 10)	根蒂更蜷(17, 14)	根蒂更蜷(17, 15)
敲声更沉(2, 1)	敲声更沉(2, 3)	敲声更沉(2, 6)	敲声更沉(2, 7)
...	...	...	...
敲声更沉(17, 7)	敲声更沉(17, 10)	敲声更沉(17, 15)	敲声更沉(17, 16)
纹理更清(1, 7)	纹理更清(1, 14)	纹理更清(1, 16)	纹理更清(1, 17)
...	...	...	...
纹理更清(15, 14)	纹理更清(15, 16)	纹理更清(15, 17)	纹理更清(17, 16)
脐部更凹(1, 6)	脐部更凹(1, 7)	脐部更凹(1, 10)	脐部更凹(1, 15)
...	...	...	...
脐部更凹(15, 10)	脐部更凹(15, 16)	脐部更凹(17, 10)	脐部更凹(17, 16)
触感更硬(1, 6)	触感更硬(1, 7)	触感更硬(1, 10)	触感更硬(1, 15)
...	...	...	...
触感更硬(17, 6)	触感更硬(17, 7)	触感更硬(17, 10)	触感更硬(17, 15)
更好(1, 10)	更好(1, 14)	更好(1, 15)	更好(1, 16)
...	...	...	...
更好(7, 14)	更好(7, 15)	更好(7, 16)	更好(7, 17)
¬更好(10, 1)	¬更好(10, 2)	¬更好(10, 3)	¬更好(10, 6)
...	...	...	...
¬更好(17, 2)	¬更好(17, 3)	¬更好(17, 6)	¬更好(17, 7)

$$(\forall X, \forall Y)(\text{更好}(X, Y) \leftarrow \text{根蒂更蜷}(X, Y) \wedge \text{脐部更凹}(X, Y)).$$

4. 一阶规则学习

- FOLL-一种著名的一阶规则学习算法，它遵循序贯覆盖且采用**自顶向下**的规则归纳策略

1. 最初的空规则 $\text{更好}(X, Y) \leftarrow .$

2. 考虑数据中所有其他谓词以及各种变量搭配作为候选文字。新加入的文字应至少包含一个已出现的变量，否则没有任何实际意义，考虑下列候选文字

色泽更深(X, Y), 色泽更深(Y, X), 色泽更深(X, Z), 色泽更深(Z, X),
色泽更深(Y, Z), 色泽更深(Z, Y), 色泽更深(X, X), 色泽更深(Y, Y),
根蒂更蜷(X, Y),
敲声更沉(X, Y),

3. 使用FOLL增益来选择文字
$$\text{F_Gain} = \hat{m}_+ \times \left(\log_2 \frac{\hat{m}_+}{\hat{m}_+ + \hat{m}_-} - \log_2 \frac{m_+}{m_+ + m_-} \right),$$

4. FOLL像命题规则学习那样继续增加规则体长度，最终生成合适的单条规则加入规则集

5.归纳逻辑程序设计

- **最小一般泛化:**
更好(1, 10) $\leftarrow$ 根蒂更蜷(1, 10) $\wedge$ 声音更沉(1, 10) $\wedge$ 脐部更凹(1, 10)
 $\wedge$ 触感更硬(1, 10);
更好(1, 15) $\leftarrow$ 根蒂更蜷(1, 15) $\wedge$ 脐部更凹(1, 15) $\wedge$ 触感更硬(1, 15).

这两条规则只对应了特殊的关系数据样例，难以具有泛化能力。因此希望把这样的“特殊”规则转变为更“一般”的规则。为了达到这个目的采用最小一般泛化，简称LGG.

- 1. 给定一阶公式 r_1 和 r_2 , LGG先找出涉及**相同谓词的文字**
- 2. 对文字中每个位置的常量逐一进行考察
 - 若常量在两个文字中相同则保持不变记为 $LGG(t, t) = V$
 - 否则将他们替换为一个新变量，并将该替换应用于公式中的所有其他位置:
 - 假定这两个不同的常量分别为 s, t ，新变量为 V ，则记为 $LGG(s, t) = V$,并在以后所有出现 $LGG(s, t)$ 的位置用 V 来替代。

更好(1, Y) $\leftarrow$ 根蒂更蜷(1, Y) $\wedge$ 声音更沉(1, 10) $\wedge$ 脐部更凹(1, Y)

$\wedge$ 触感更硬(1, Y);

更好(1, Y) $\leftarrow$ 根蒂更蜷(1, Y) $\wedge$ 脐部更凹(1, Y) $\wedge$ 触感更硬(1, Y).

5.归纳逻辑程序设计

3. LGG忽略r1和r2中不含共同谓词的文字，因为若LGG包含某条公式所没有的谓词，则LGG无法特化为那条公式。例子中需要忽略“声音更沉”这个文字，于是：

$$\text{更好}(1, Y) \leftarrow \text{根蒂更蜷}(1, Y) \wedge \text{脐部更凹}(1, Y) \wedge \text{触感更硬}(1, Y).$$

4.上式仅能判断瓜1是否比其他瓜更好。为了提升其泛化能力，假定另有一条关于瓜2的初始规则：

$$\begin{aligned} \text{更好}(2, 10) \leftarrow & \text{颜色更深}(2, 10) \wedge \text{根蒂更蜷}(2, 10) \wedge \text{敲声更沉}(2, 10) \\ & \wedge \text{脐部更凹}(2, 10) \wedge \text{触感更硬}(2, 10), \end{aligned}$$

5.令 $\text{LGG}(10, Y) = Y2$, 将所有的“10”与“Y”成对出现的位置均替换为Y2
令 $\text{LGG}(2, 1) = X$ 并删去谓词不同的文字，就得到如下这条不包含常量的一般规则

$$\text{更好}(X, Y_2) \leftarrow \text{根蒂更蜷}(X, Y_2) \wedge \text{脐部更凹}(X, Y_2) \wedge \text{触感更硬}(X, Y_2).$$

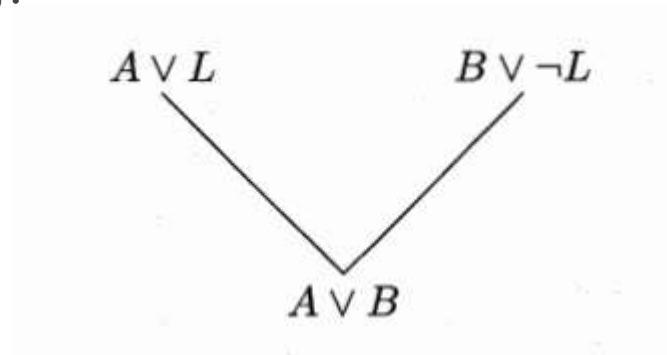
5.逻辑归纳程序设计

• 归结原理

• 数学定义:

- 给定子句 $C1=A \vee L$ 和 $C2=B \vee \neg L$, 归结项 $C=A \vee B$
- 公式: $C=(C1-\{L\}) \vee (C2-\{\neg L\})$
- 简记为 $C=C1*C2$

• 直观示例:



逆归结的目标

- 定义: 已知归结结果 C 和子句 Ci , 推导另一子句 Cj 。
- 公式:
 - $C2=(C-(C1-\{L\})) \vee \{\neg L\}$
- 核心价值: 发明新概念 (如谓词)、精化逻辑规则。

5. 逻辑归纳程序设计

吸收 (Absorption)

作用：合并共同前件，简化规则结构。

- 原规则：更好(X) ← 根蒂更蜷(X) ∧ 纹理更清(X)
- 吸收后：更好(X) ← q(X) ∧ 纹理更清(X)，其中 q(X) ← 根蒂更蜷(X)

辨识 (Identification)

作用：通过共享中间项生成新规则。

- 原规则：更好(X) ← 根蒂更蜷(X) ∧ 纹理更清(X)
- 辨识后：质量(X) ← 纹理更清(X)

内构 (Intra-construction)

作用：引入新谓词统一多规则（如西瓜数据集的 q(M,N)q(M,N)）。

- 输入规则：
 - C1: 更好(X) ← 根蒂更蜷(X) ∧ 纹理更清(X)
 - C2: 更好(X) ← 根蒂更蜷(X) ∧ 敲声更沉(X)
- 内构结果：
 - C': 更好(X) ← 根蒂更蜷(X) ∧ q(X)
 - q(X) ← 纹理更清(X)
 - q(X) ← 敲声更沉(X)

吸收(absorption) :

辨识(identification) :

内构(intra-construction) :

互构(inter-construction) :

$$\begin{array}{l} \text{吸收(absorption):} \quad \frac{p \leftarrow A \wedge B \quad q \leftarrow A}{p \leftarrow q \wedge B \quad q \leftarrow A} \\ \text{辨识(identification):} \quad \frac{p \leftarrow A \wedge B \quad p \leftarrow A \wedge q}{q \leftarrow B \quad p \leftarrow A \wedge q} \\ \text{内构(intra-construction):} \quad \frac{p \leftarrow A \wedge B \quad p \leftarrow A \wedge C}{q \leftarrow B \quad p \leftarrow A \wedge q \quad q \leftarrow C} \\ \text{互构(inter-construction):} \quad \frac{p \leftarrow A \wedge B \quad q \leftarrow A \wedge C}{p \leftarrow r \wedge B \quad r \leftarrow A \quad q \leftarrow r \wedge C} \end{array}$$

互构 (Inter-construction)

作用：通过共享新谓词重构规则。

- 原规则：
 - 更好(X) ← 根蒂更蜷(X) ∧ 纹理更清(X)
 - 更甜(X) ← 根蒂更蜷(X) ∧ 敲声更沉(X)
- 互构结果：
 - 更好(X) ← r(X) ∧ 纹理更清(X)
 - 更甜(X) ← r(X) ∧ 敲声更沉(X)
 - r(X) ← 根蒂更蜷(X)