大数据导论
Introduction to Big Data

第6讲 分类：基础概念与算法

叶允明

计算机科学与技术学院

哈尔滨工业大学（深圳）

目录

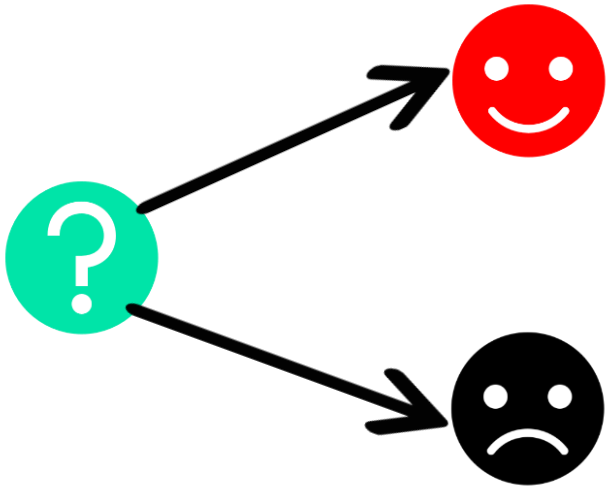
主要参考资料

- Jiawei Han, MIcheline Kamber, Jian Pei著；范明，孟小峰等译. 数据挖掘：概念与技术. 机械工业出版社, ISBN: 9787111391401, 2012.
- 第8章：8.1、8.4

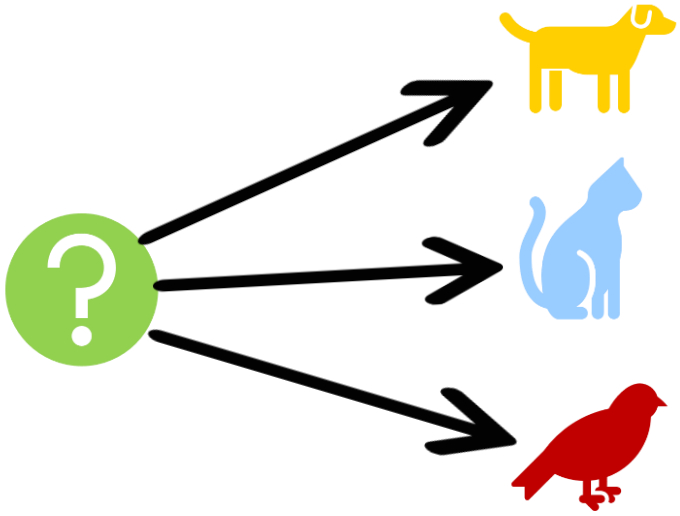
分类的基本概念

分类的基本概念

- 分类（classification）：判断（预测）给定数据对象所属的类别



二分类



多分类

分类的应用领域

- 分类是人类认识世界的最基本方法
- 几乎每个人工智能应用领域都涉及到分类问题
  - 信用评估
  - 图像识别
  - 目标市场营销
  - 医学诊断
  - 欺诈检测
  - 文本分类
  - ……

分类任务的定义

- 分类任务可以用一个形式化函数表示：

$$y = f(\boldsymbol{x}),$$

$$其中\boldsymbol{x} \in \boldsymbol{D}, y \in \{c_1, c_2, \dots, c_k\}$$

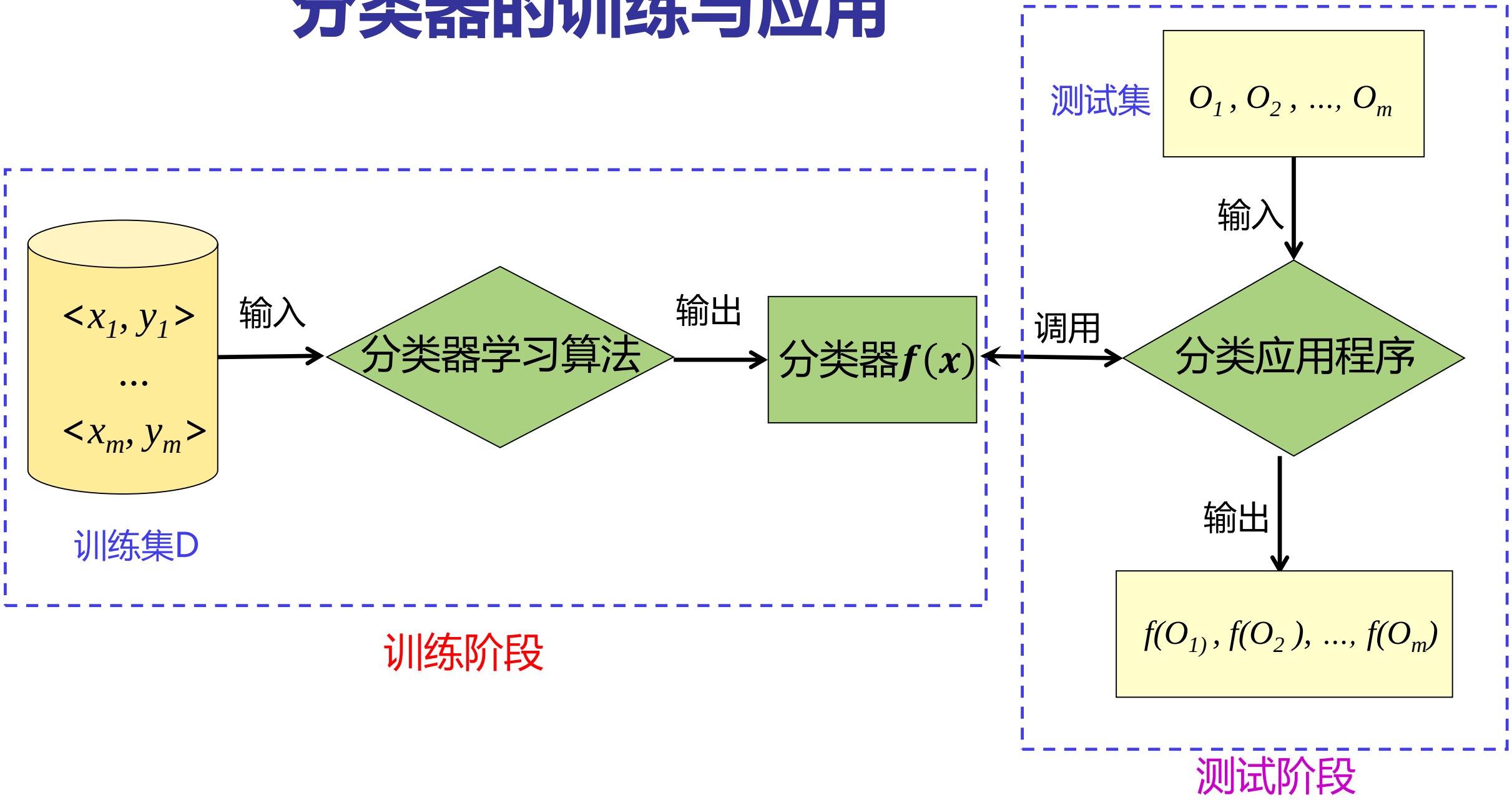
  - 分类函数$f(\boldsymbol{x})$经过运算可以输出一个离散值y，又称为“分类器”（classifier）
  - 给定数据集$\boldsymbol{D}$中的一个数据对象向量$\boldsymbol{x}$（称为“实例”）
  - y的取值范围是类别的数字编码集合（$c_1, c_2, \dots, c_k$）

如何构造函数$\boldsymbol{f}(\boldsymbol{x})$呢?

完成分类任务的“两阶段”流程

- **分类器构建（训练阶段）：即学习阶段**
  - 从已知类标（class label）的训练数据集中学习，生成分类器 $f(\boldsymbol{x})$
  - 分类器又称为分类模型，可表示成分类规则、决策树或者数学公式
- **分类器应用（测试阶段）：**
  - 用分类器 $f(\boldsymbol{x})$来判断未知类标数据对象的类别

分类器的训练与应用


测试集
$O_1, O_2, ..., O_m$
输入
$<x_1, y_1>$
...
$<x_m, y_m>$
输入
分类器学习算法
输出
分类器$f(x)$
调用
分类应用程序
输出
$f(O_1), f(O_2), ..., f(O_m)$
训练集D
训练阶段
测试阶段

“贷款审批”的分类应用案例

训练集D

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
10	中年	正常	低	通过
11	中年	一般	较高	通过
12	中年	正常	低	不通过
13	中年	超出	中	不通过
14	中年	正常	高	不通过
15	中年	正常	低	不通过

输入

分类器学习算法

输出

分类器$f(x)$

训练阶段

测试集

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	超出	高	?
2	青年	一般	中	?
3	老年	一般	较高	?
4	中年	正常	低	?

调用

输入

分类应用程序

输出

$f(O_1), f(O_2), \ldots, f(O_m)$

测试阶段

数据挖掘模型“学习”的三要素

- **定义模型空间：**
  - 假定模型$f(\boldsymbol{x})$的可能“取值”空间，即通常需要假定模型的表示形式
  - 例如，可假定分类模型（分类器） $f(\boldsymbol{x})$的形式为线性分段函数
- **评价模型$f(\boldsymbol{x})$“好坏”的标准**
  - 有监督学习：基于类标/目标值
  - 无监督学习：常基于数据的空间分布或统计指标
- **搜索“最优”模型f^*的算法**
  - 大数据条件下求最优解通常在计算上很困难，只能求近似最优、次优

基于规则的分类方法
——顺序覆盖算法

基于规则的分类模型

- 基于规则的分类模型：是指由一组规则构成的分类模型（分类器）。
- 规则（rule）是指语义清晰、能描述客观事实中所隐含的规律或概念的逻辑准则
- 通常一条规则可以用“$if\ \ldots, then\ \ldots$”语句的形式来表达。

“if 年龄 $=$ '中年' 且 未偿还贷款 $=$ '高', $then$ 类别 $=$ '不通过'”

规则的形式化表示

- 分类规则可以用以下逻辑表达式来形式化表示：

$$r_i \in T, \qquad r_i : (b_i) \rightarrow \tilde{y}_i,$$

$$\text{其中} \tilde{y}_i \in \{c_1, c_2, \ldots, c_k\},\ b_i = t_1 \wedge t_2 \wedge \cdots \wedge t_n$$

规则覆盖的定义

规则 r：

“*if* 年龄 = '中年' 且 消费收入比 = '一般', *then* 类别 = '通过'”

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过

衡量规则的质量

- 覆盖率（coverage）：规则覆盖记录数占数据集D的比例。

$$coverage(r) = \frac{n_{covers}}{|D|}$$

- 准确率（accuracy）：规则正确分类的元组数与它所覆盖的元组数之比。

$$accuracy(r) = \frac{n_{correct}}{n_{covers}}$$

衡量规则的质量

- 例：数据集中共有1000条记录，所学得的规则r覆盖了其中的600条记录（指记录符合规则中的所有合取条件），但这600条中只有300条是分类正确的，则规则r的覆盖率、准确率分别为多少?

$$coverage(r) = \frac{600}{1000} = 0.6$$

$$accuracy(r) = \frac{300}{600} = 0.5$$

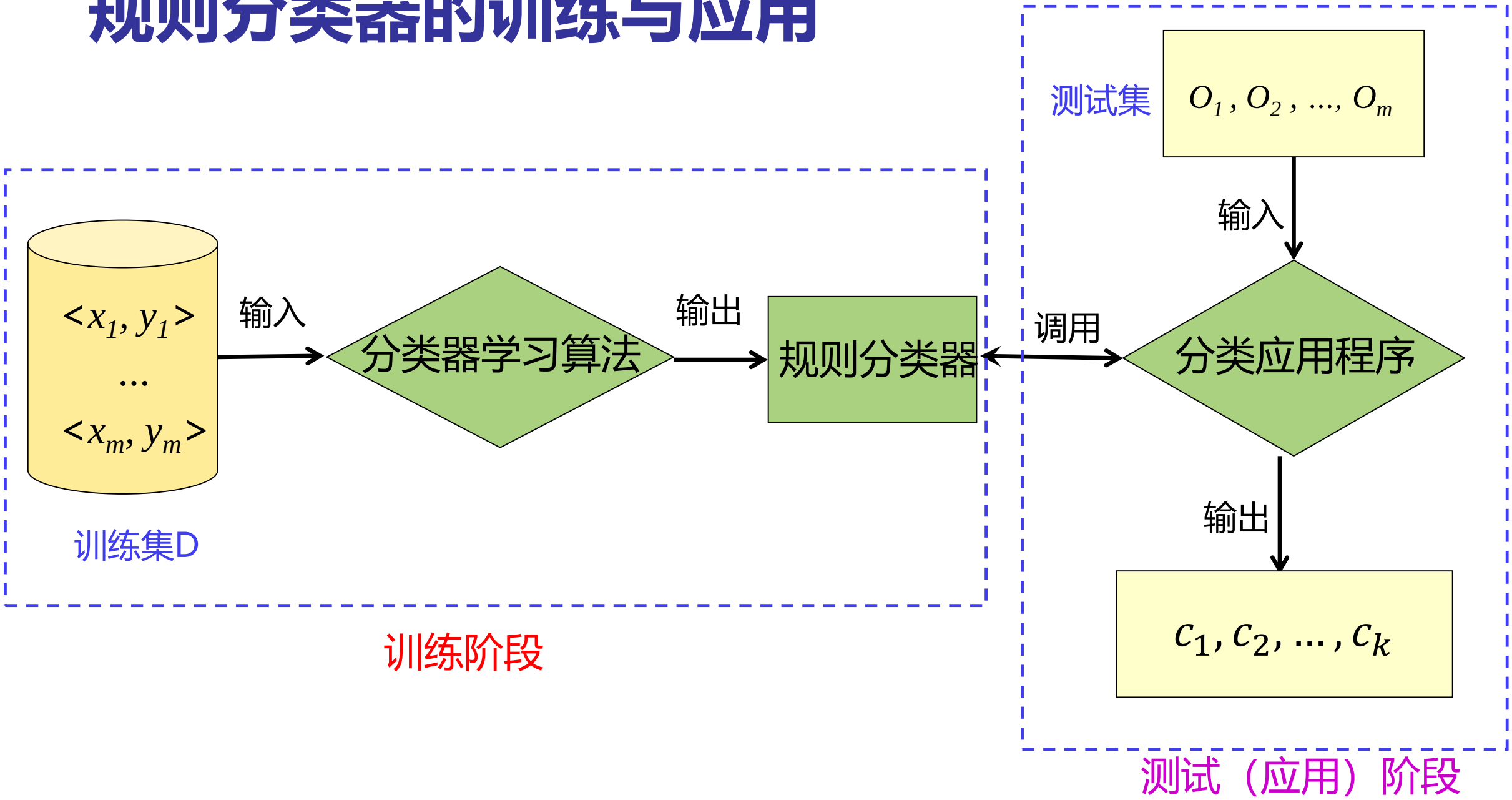
规则触发与规则激活

规则 r　“*if* 年龄 = '中年' 且 消费收入比 = '一般', *then* 类别 = '通过'”

记录 x

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过

规则分类器的训练与应用

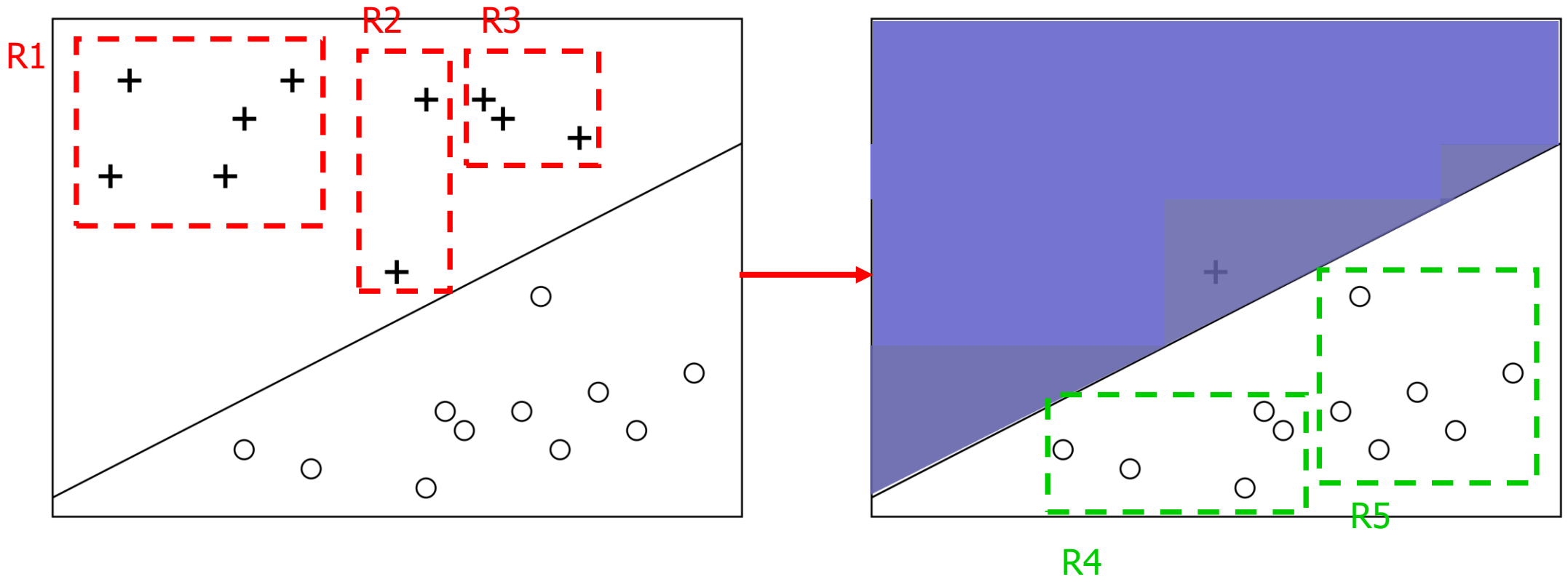


怎么从数据中学习规则分类器?

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
10	中年	正常	低	通过
11	中年	一般	较高	通过
12	中年	正常	低	不通过
13	中年	超出	中	不通过
14	中年	正常	高	不通过
15	中年	正常	低	不通过

规则分类器的学习：顺序覆盖算法

- 算法思想：逐类归纳生成规则集合；每个类依次生成n条规则，直到该类样本被完全覆盖。



顺序覆盖算法

输入： (1) 训练样本集 $D \in \{C_1, C_2, ..., C_k\}$，其中 C_i 为样本所属的类；

(2) 样本所有属性及属性可能取值的集合 $A_v = \{Attribute - Values\}$

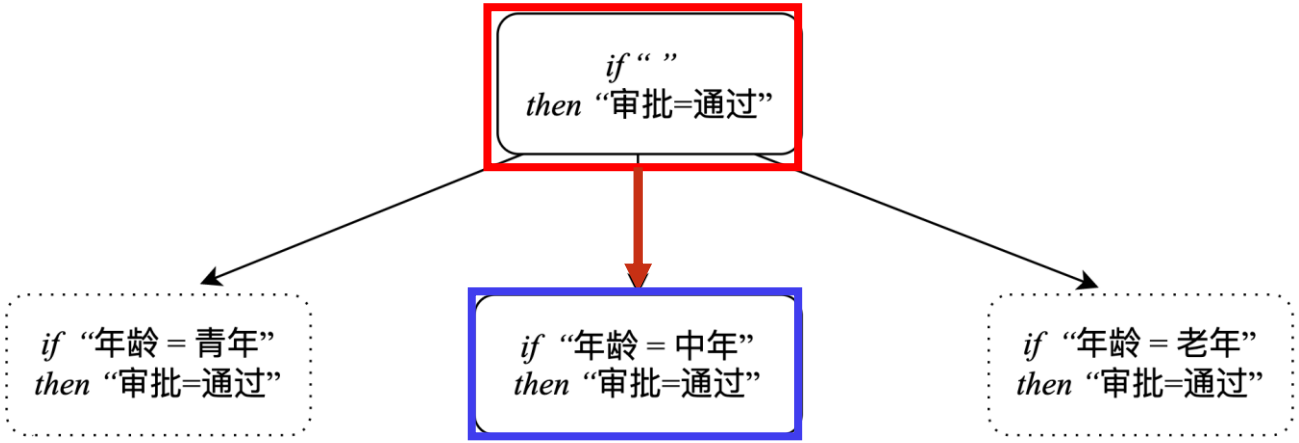
输出： 顺序覆盖算法所学到的规则集 T

1: 规则集在初始时设置为空：$T = \{\}$;
2: **for** C_i **do**
3: **repeat**
4: 在当前训练集对类 C_i 找出一条最优的分类规则：$r = \text{Learn\_One\_Rule(D, } C_i, A_v)$
5: 将新生成的规则 r 加入规则集 T：T += r;
6: 删除 D 中被规则覆盖的样例;
7: **until** 满足终止条件
8: **end for**
9: **return** T;

单条规则的归纳学习算法（Learn_One_Rule）

- Step1：从当前类最简单的规则开始学起：规则前件为空
- Step2：采用一种贪心的深度优先策略添加新的合取条件，使新的规则的“质量”能得到最大提高
  - 依次添加合取条件（属性-值）
  - 衡量“规则质量”常采用规则的准确率
- Step3：不断重复Step2，直到新的规则无法提升原规则的“质量”为止

Learn_One_Rule算法



0.46

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……

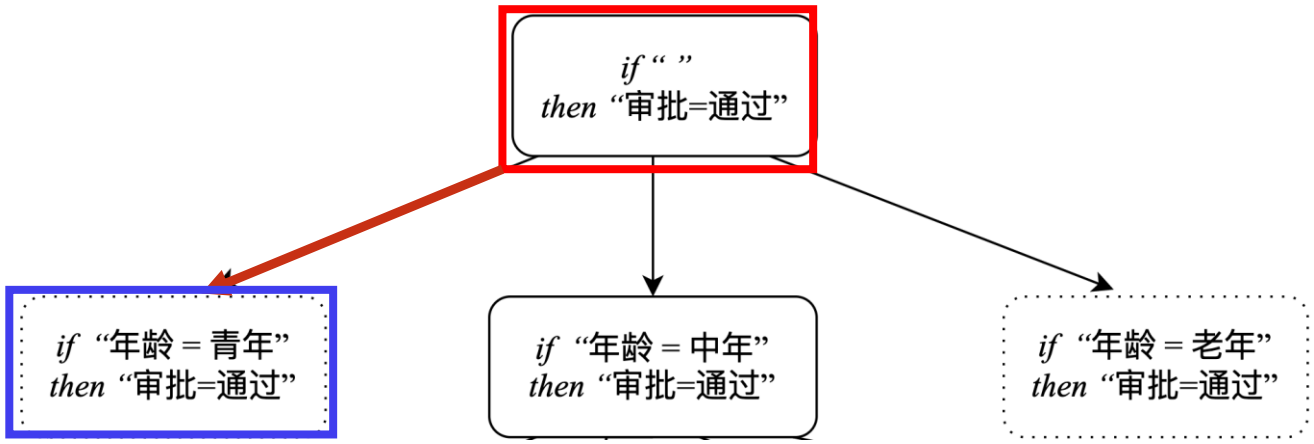
$$accuracy(r) = \frac{219}{475} = 0.46$$

$$\max[accuracy(r)] = 0.46$$

最大准确率对应的规则为：

"if '年龄' = '中年', then '类别' = '通过'"

Learn_One_Rule算法



0.51 **0.46**

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……

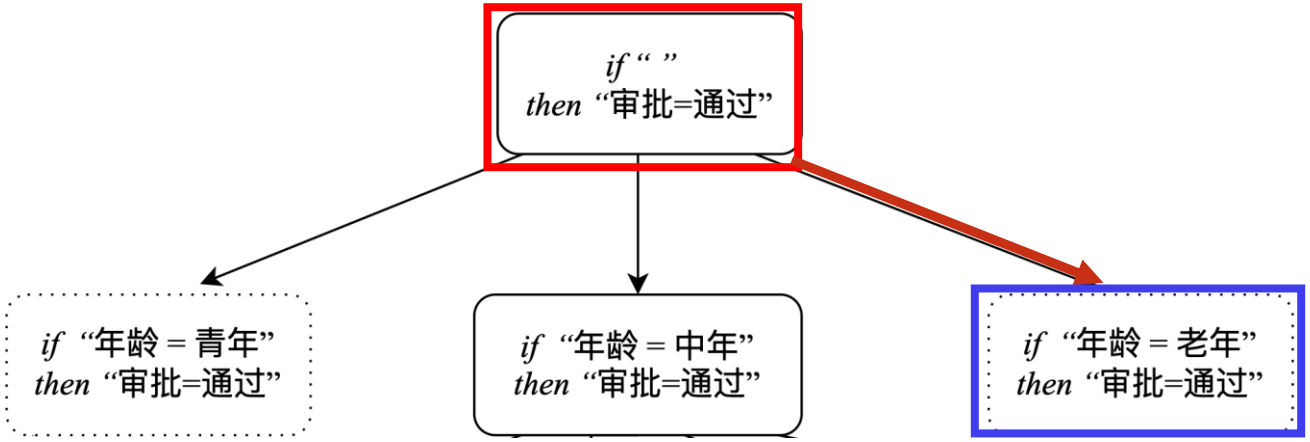
$$accuracy(r) = \frac{112}{220} = 0.51$$

$$\max[accuracy(r)] = 0.51$$

最大准确率对应的规则为：

"*if* 年龄 =' 青年', *then* 类别 = '通过'"

Learn_One_Rule算法



0.51 **0.46** **0.34**

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……

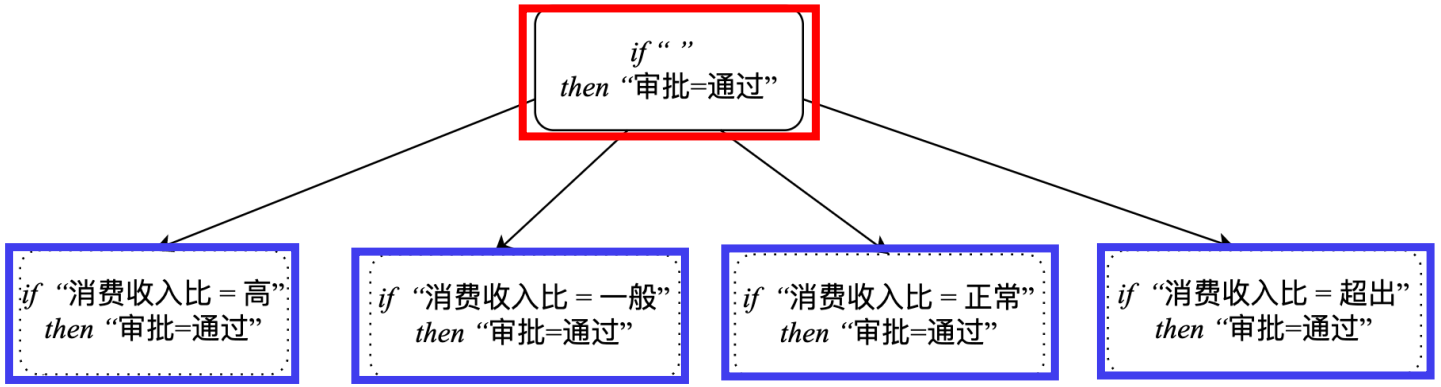
$$accuracy(r) = \frac{105}{305} = 0.34$$

$$\max[accuracy(r)] = 0.51$$

最大准确率对应的规则为：

"if 年龄 =' 青年', then 类别 = '通过'"

Learn_One_Rule算法



0.33 **0.67** **0.59** **0.62**

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……

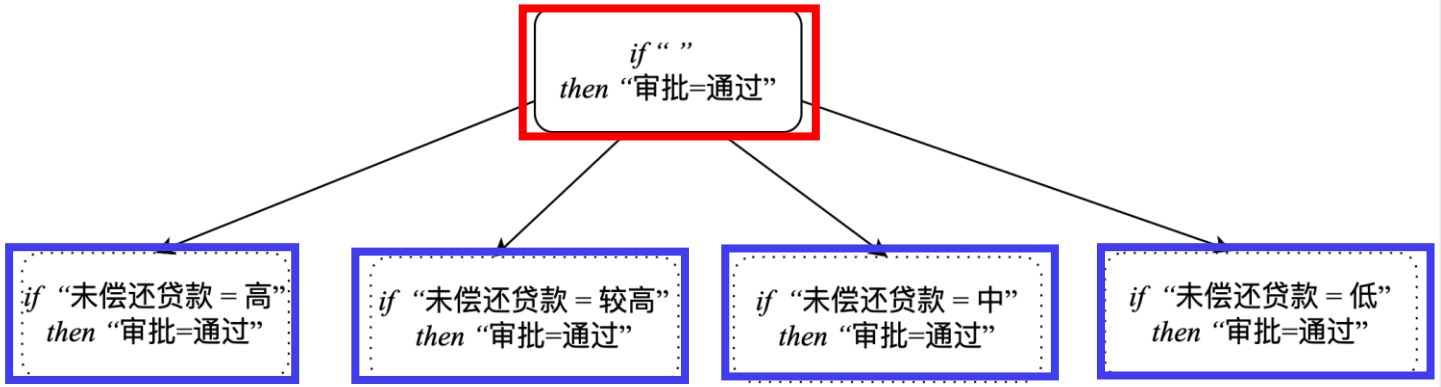
$$accuracy(r) = \frac{2}{3} = 0.67$$

$$\max[accuracy(r)] = 0.67$$

最大准确率对应的规则为：

""*if* 消费收入比 = '高'般*then*类别 ='通过' "

Learn_One_Rule算法



0.51 **0.44** **0.37** **0.60**

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……

$$accuracy(r) = \frac{1574}{2850} = 0.51$$

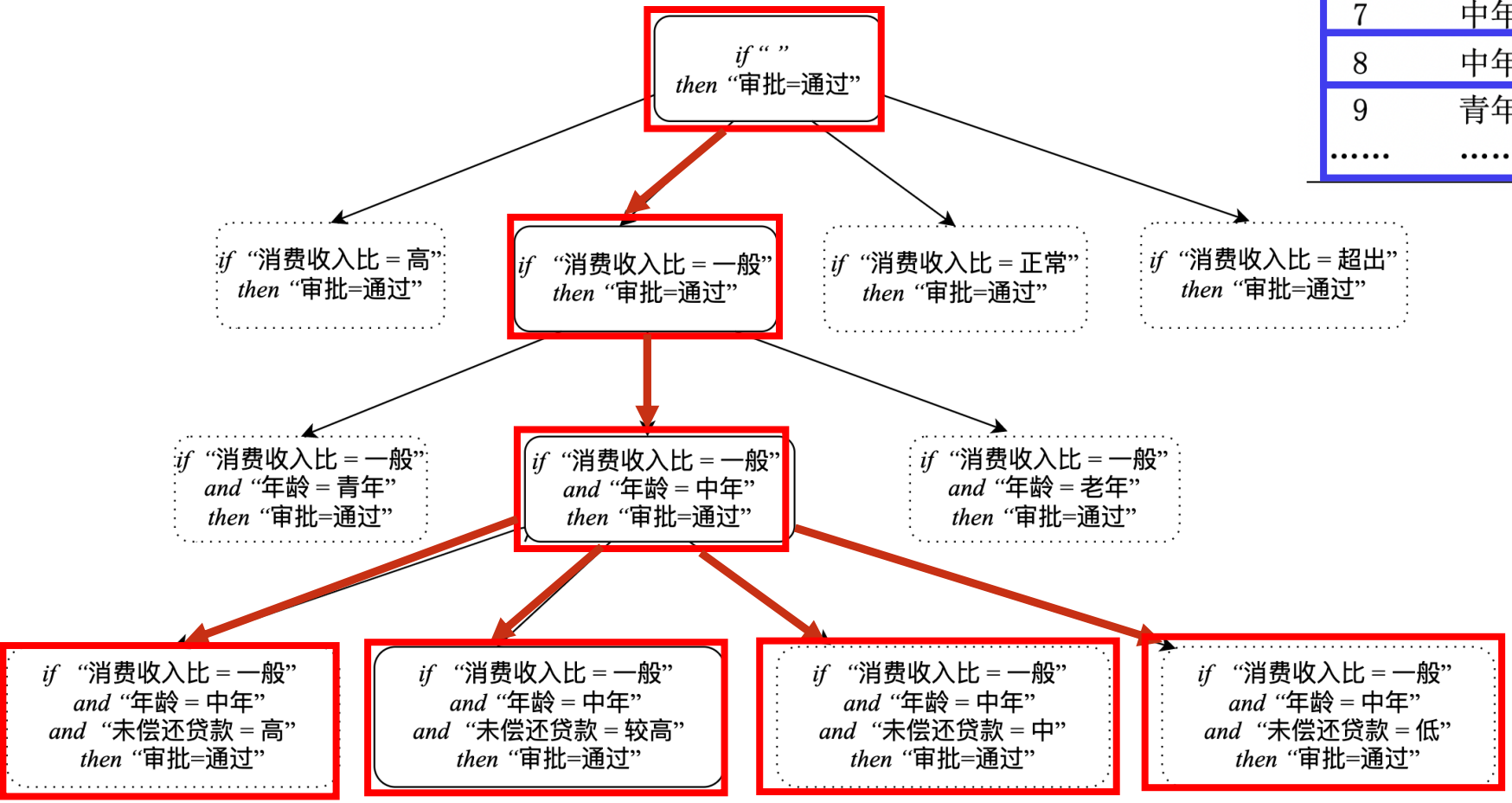
$$\max[accuracy(r)] = 0.51$$

最大准确率对应的规则为：

“*if* 未偿还贷款 = '高', *then* 类别 = '通过'”

Learn_One_Rule算法

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	高	高	不通过
2	中年	一般	高	通过
3	中年	一般	较高	通过
4	中年	一般	低	通过
5	中年	一般	高	通过
6	老年	正常	低	通过
7	中年	超出	中	通过
8	中年	一般	较高	通过
9	青年	超出	低	通过
……	……	……	……	……



$accuracy(r) = 0.31$

$accuracy(r') = 0.67$

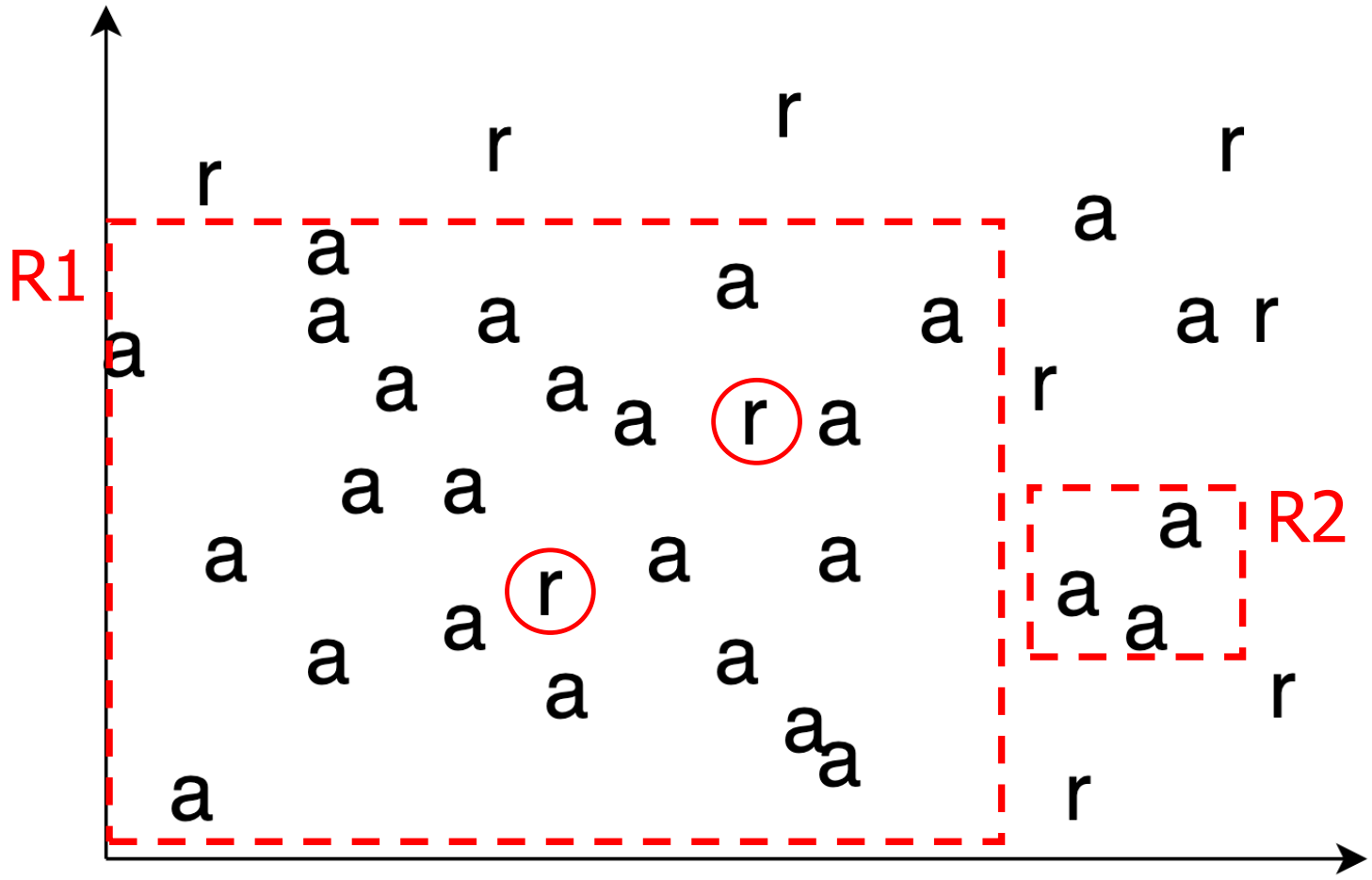
$accuracy(r'') = 0.83$

$accuracy(r''') = 0.72$

$\max[accuracy(r''')] < 0.83$

规则质量不再提升,算法终止。返回规则 (r'')

衡量规则的质量：问题



R1: accuracy<100%

R2: accuracy=100%

R2的质量比R1好吗?

衡量规则的质量：改进方法

- FOIL增益: 同时考虑覆盖率和准确率，综合衡量规则的质量。

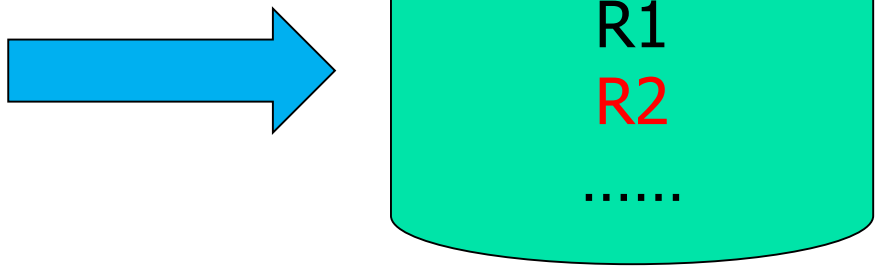
$$FOIL\_Gain = pos' \times (\log_2 \frac{pos'}{pos'+neg'} - \log_2 \frac{pos}{pos+neg})$$

  - pos/neg 为规则 R覆盖的正、负样本个数。
  - 通过FOIL增益衡量规则质量，会优先留取覆盖率和准确率都较为理想的规则。

规则分类器的应用（测试阶段）

- 规则的激活：当元组 X 满足规则 R 的触发条件，且按激活规则的次序应以规则 R 判定元组 X 的类别时，称为规则 R 被元组 X 激活。

序号	年龄	消费收入比	未偿还贷款	审批
1	中年	超出	高	?
2	青年	一般	中	?
3	老年	一般	较高	?
4	中年	正常	低	?



应用规则分类器的常见问题

- “记录不符合任一条规则”（无激活规则）
  - ✓ 默认类（一般是训练集中占比最大的类）
- “记录同时满足多条规则”（规则冲突问题）

规则“冲突问题”

- “规则 R 被 X 触发” ≠ “规则 R 被 X 激活”

规则 r1 “if 年龄 = '中年' 且 消费收入比 = '正常', $then$ 类别 = '不通过'”

规则 r2 “if 年龄 = '中年' 且 未偿还贷款 = '高', $then$ 类别 = '通过'”

记录 x

序号	年龄	消费收入比	未偿还贷款	审批
14	中年	正常	高	未通过

- “规则 r1, r2 被 x 触发”;
- 但只有 “规则 r1 被 x 激活”!

规则“冲突问题”的解决方法

- 规则“冲突”问题：元组同时满足多条规则，即多条规则可被触发
- 问题可转化为：多条被触发规则的选择问题
- 主要方法两类：
  - ✓ 基于规则复杂度的激活准则
  - ✓ 基于规则优先级的激活准则

基于规则复杂度的激活准则

- 不需要提前对规则进行整体排序；
- 规则前件包含的条件越多，规则的激活优先级越高；
- 多条规则同时被触发，优先按条件更苛刻的规则分类。

基于规则优先级的激活准则

- 预先对规则集合排序；
- 排序之后的有序规则集又称为决策表，实际预测时用该表中的规则按次序与待预测元组匹配；
- 一旦某条规则被激活，后面的规则将无法再被该记录激活。
- 排序方法可分为两种：
  - 基于类重要性的排序方法
  - 基于规则质量的排序方法