



# 专题一 分类

---



# 目录

---

- 分类的定义
- 解决分类问题的一般方法和过程
- 三种常见的分类模型
  - 决策树
  - 基于规则的分类
  - 基于实例的分类



# 1. 分类的定义

- 给定一个记录集合（训练集合）
  - 每一个记录包含一组属性, 其中有一个为类属性.
- 发现一个以其它属性值为函数的类属性模型.
- 目标: 先前不可见（未知类标签）的记录能被尽可能精确地指派给一类.
  - 一个测试集常被用于确定模型的精度.  
通常, 给定的数据集被分为训练和测试集合, 训练集合通常用来建立模型, 而测试集合用来验证模型.



# 分类

---

- 分类的定义
- 解决分类问题的一般方法和过程
- 三种常见的模型
  - 决策树
  - 基于规则的分类
  - 基于实例的分类



## 2.1 解决分类问题的一般方法

- **分类技术/方法是一种根据输入数据集建立分类模型的系统方法。**
  - **用来建立分类模型的算法称为学习/训练算法**
  - **建立的模型应能很好地拟合输入数据中的类标号和属性集之间的联系。**
  - **训练算法的主要目标就是建立具有较好泛化能力的分类模型，即建立能够准确预测未知样本类标号的模型。**
  - **准确率是评价分类模型性能最主要的目标之一。**

## 2.2 建立分类模型的过程 (图示)

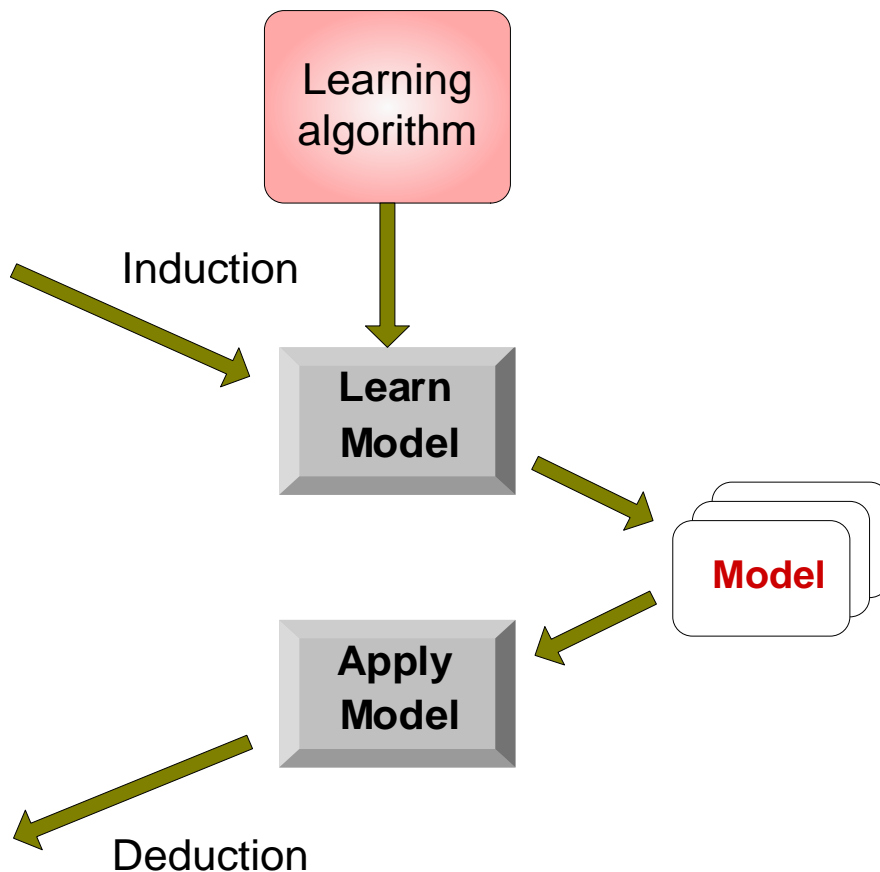
Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

2020/10/28

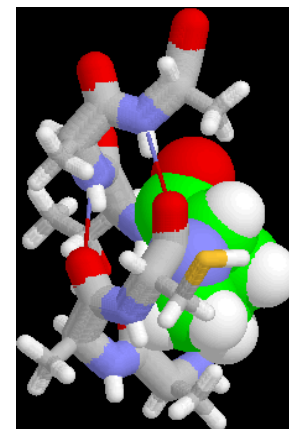


# 分类的应用场景举例

- 预测瘤细胞是良性还是恶性的
- 分类信用卡事务为合法的还是欺诈的



- 分类蛋白质的二级结构为  $\alpha$ -螺旋,  $\beta$ -薄片 还是任意卷



- 分类新闻故事为金融, 天气, 娱乐, 体育等



# 分类

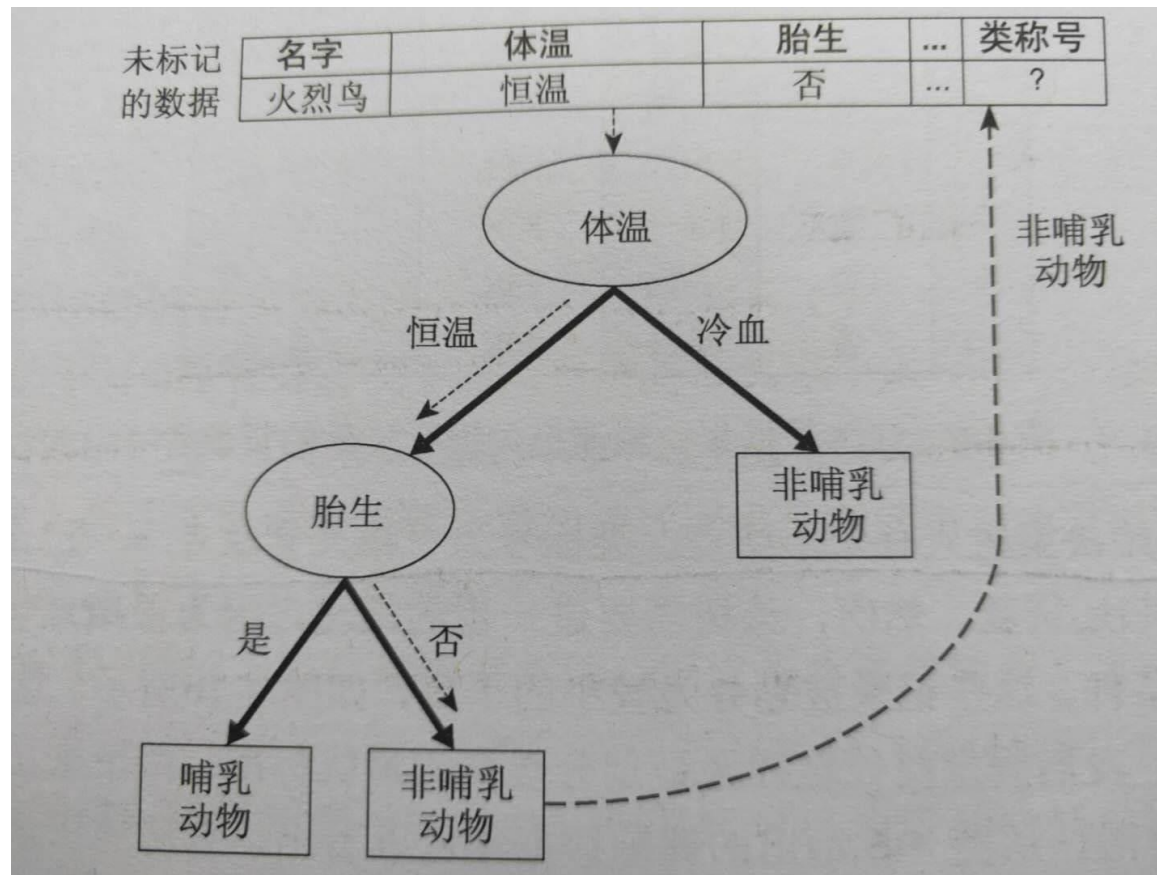
---

- 分类的定义
- 解决分类问题的一般方法和过程
- 三种常见的模型
  - 决策树
  - 基于规则的分类
  - 基于实例的分类



# 3.1 决策树——工作原理

- 判断一个新物种是否是哺乳动物。





## 3.1 决策树——工作原理

- 判断一个新物种是否是哺乳动物：
  - 该物种是冷血还是恒温动物？若是冷血的，不是哺乳动物；否则它或者是鸟类，或者是哺乳动物。
  - 是由雌性产崽进行繁殖吗？如果是，它肯定是哺乳动物，否则它有可能是非哺乳动物。
- 可以看出：通过提出一系列精心构思的关于**检验记录属性**的问题，可以解决分类问题。
- 这一系列的问题和这些问题的一系统回答可以组织成树的形式。



## 3.1 决策树——工作原理

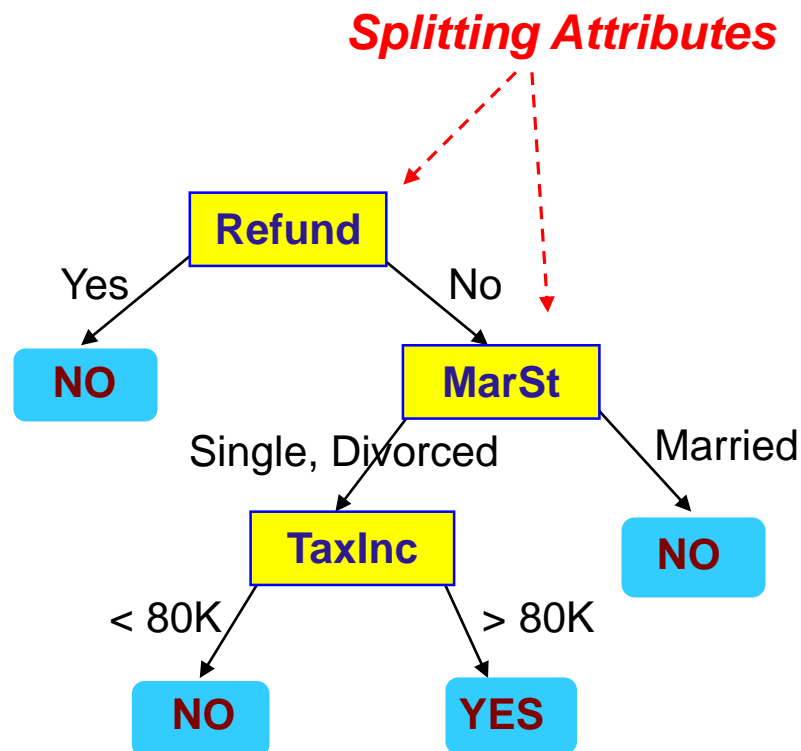
- 一旦建立了决策树，从根结点开始，将测试条件用于检验记录，根据测试结果选择适当的分支。沿着分支达到另一个内部结点或叶子结点。到达叶子结点后，叶子结点的类标号就被作为检验记录的类标号。

# 3.1 决策树——图例

Model: Decision Tree

二元的  
分类的  
连续的  
特征

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Training Data

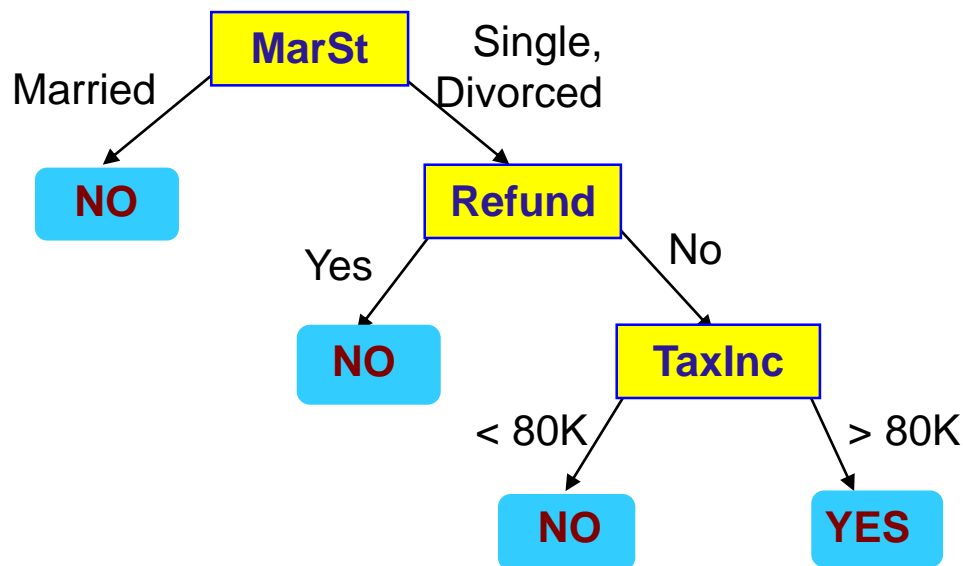
2020/10/28

是否拥有房产 婚姻状况 税收 年收入 有无偿还能力

# 3.1 决策树——图例

二元的  
分类的  
连续的  
类

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



有可能存在多个树适合同样的数据！

# 3.1 决策树——分类预测任务

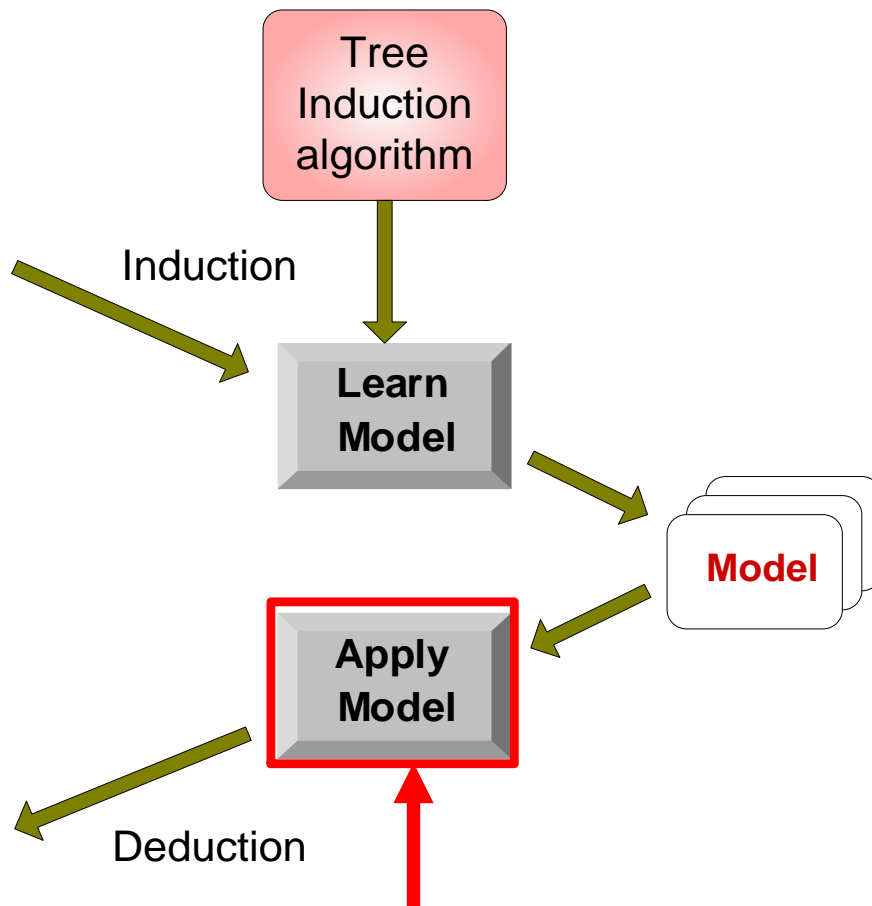
Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

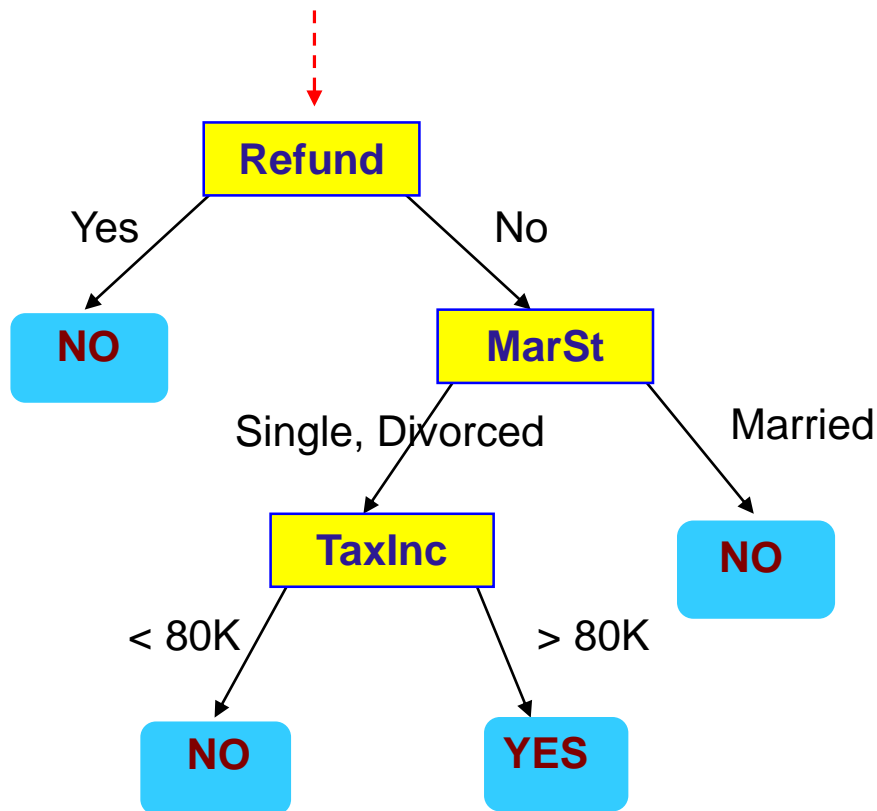
Test Set

2020/10/28



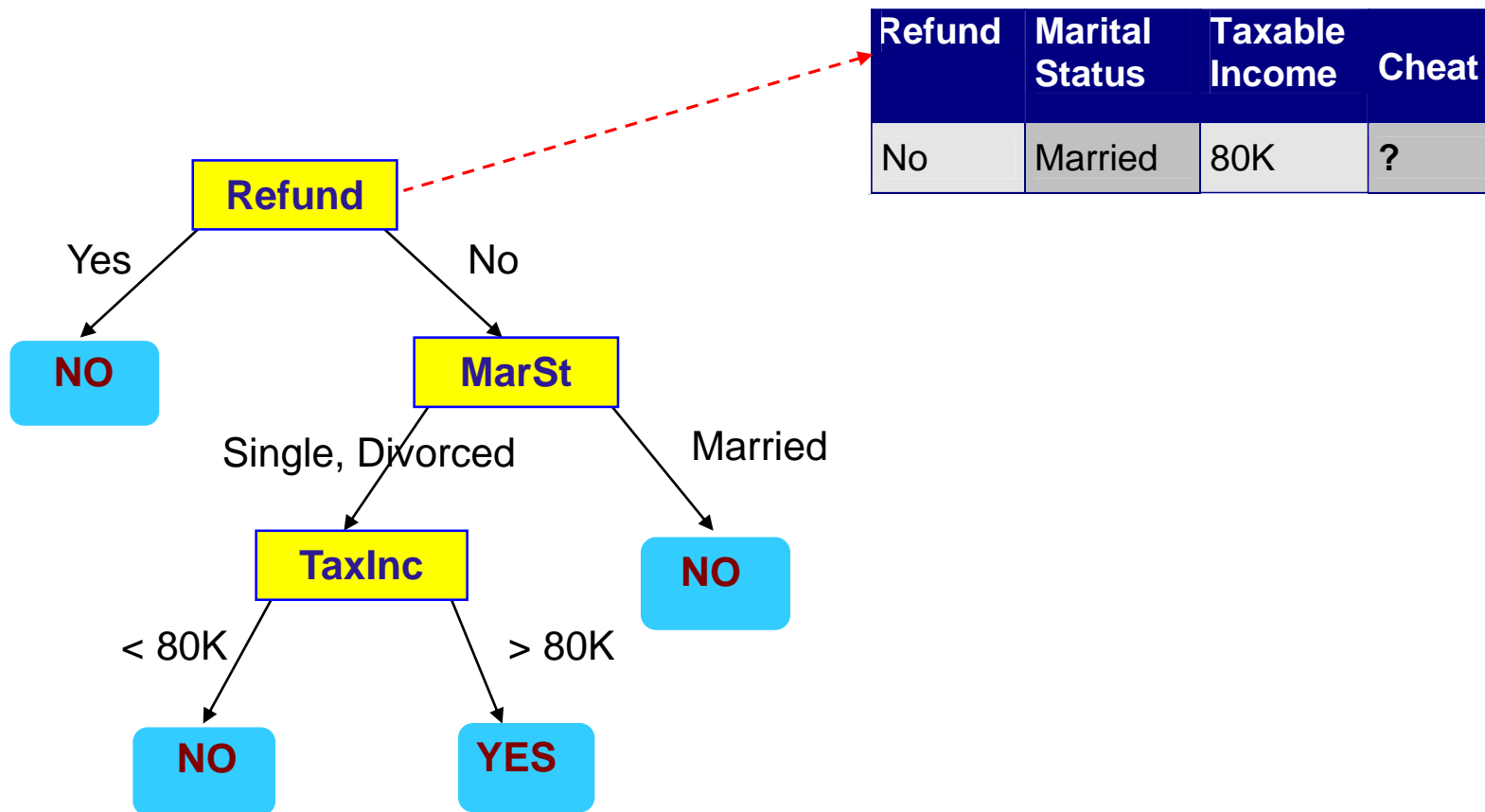
# 3.1 决策树——分类预测任务举例

Start from the root of tree.



Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

# 3.1 决策树——分类预测任务举例

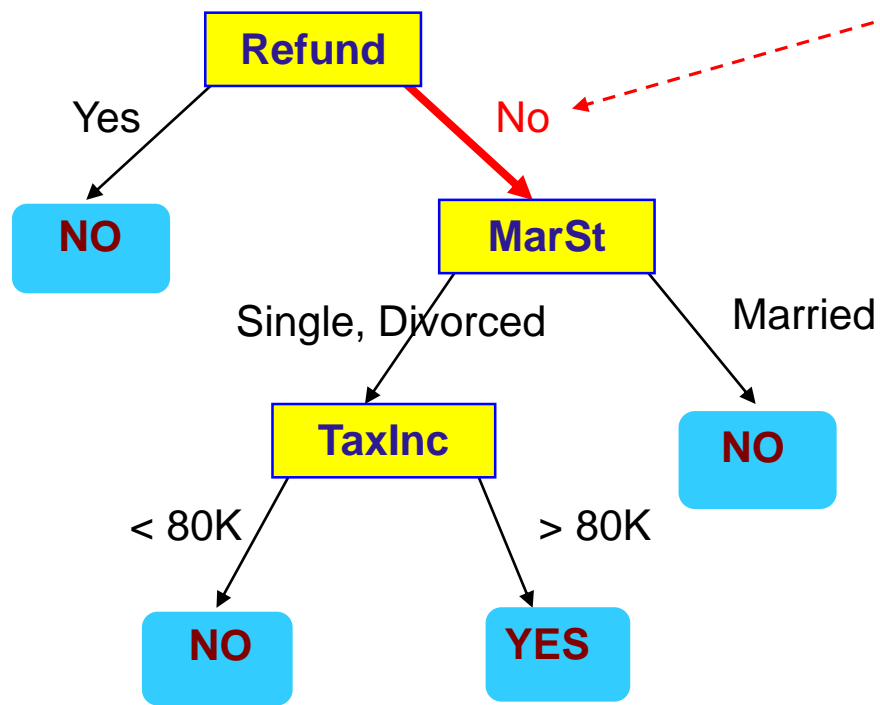




# 3.1 决策树——分类预测任务举例

Test Data

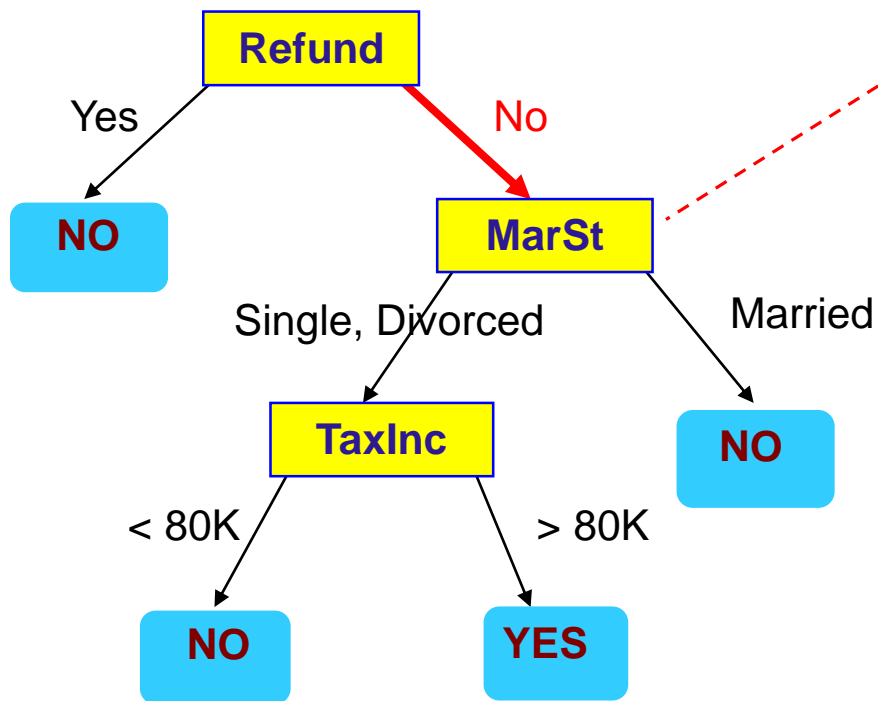
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



# 3.1 决策树——分类预测任务举例

Test Data

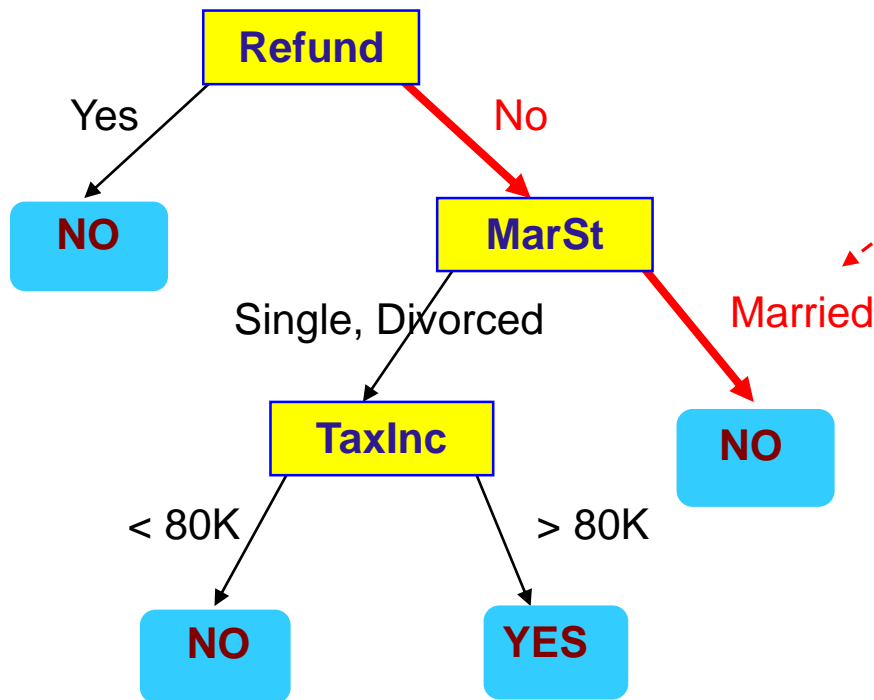
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



# 3.1 决策树——分类预测任务举例

Test Data

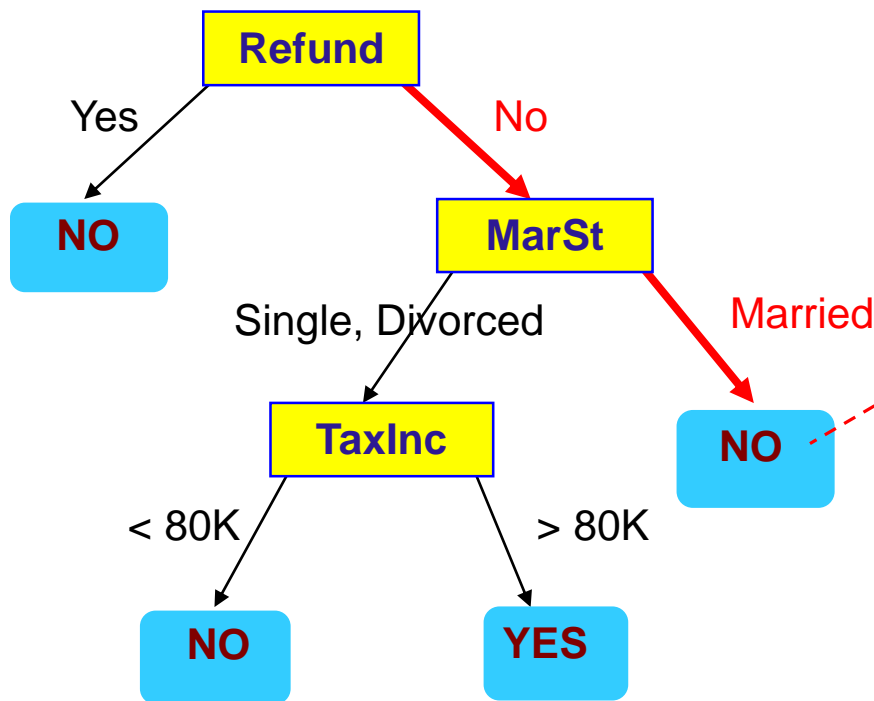
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



# 3.1 决策树——分类预测任务举例

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Assign Cheat to "No"

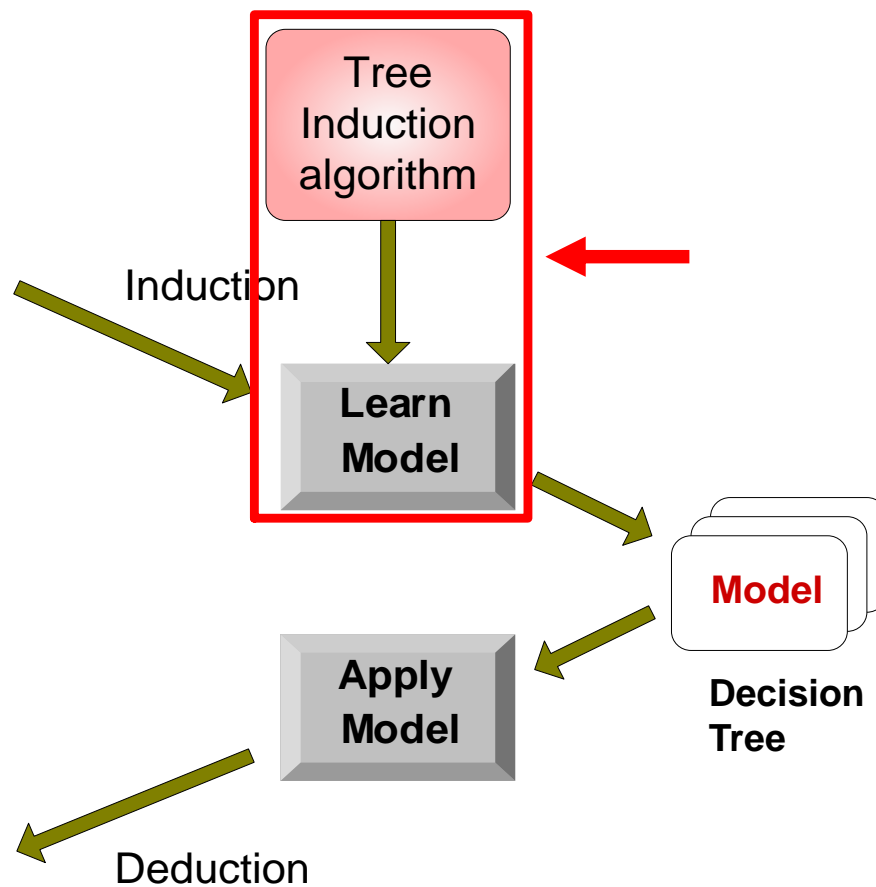
## 3.2 决策树——分类模型学习任务

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set





## 3.1 决策树——分类器学习任务

---

- 许多算法：

- Hunt's (亨特) 算法 (最早的算法之一)
- CART
- ID3, C4.5
- SLIQ, SPRINT

## 3.2 基于实例的分类——工作原理

Set of Stored Cases

Atr1	.....	AtrN	Class
			A
			B
			B
			C
			A
			C
			B

• 把存储的案例集合作为训练记录.

• 使用训练记录来预测未知案例的类标识.

Unseen Case

Atr1	.....	AtrN



## 3.2 基于实例的分类——表现形式

### ■ 例如：

#### ■ Rote-learner (机械的学习器)

- 记住整个训练数据，仅当测试实例的属性和某个训练实例完全匹配时才进行分类。

#### ■ Nearest neighbor (最近邻)

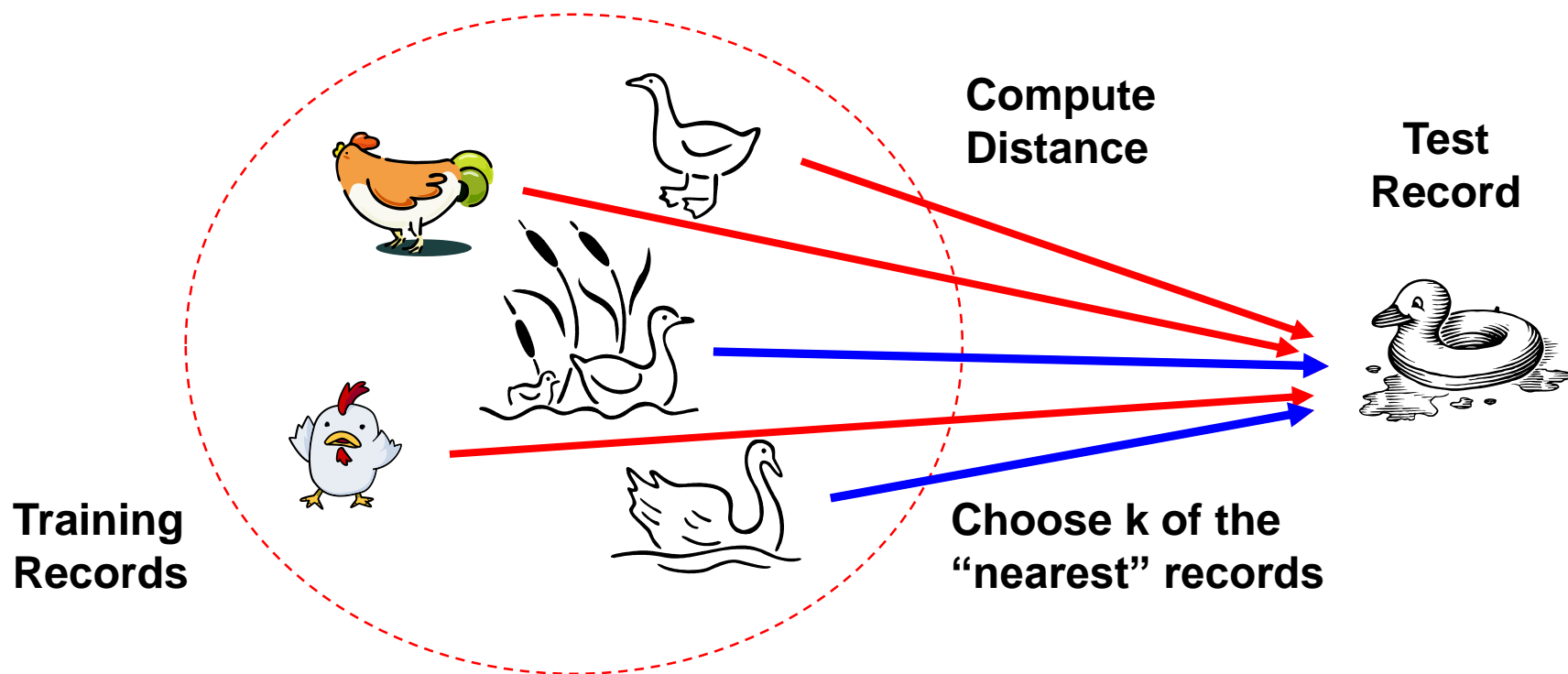
- 使用 $k$ 个最接近的点（最近邻）来进行分类。



## 3.2 基于实例的分类——最近邻分类

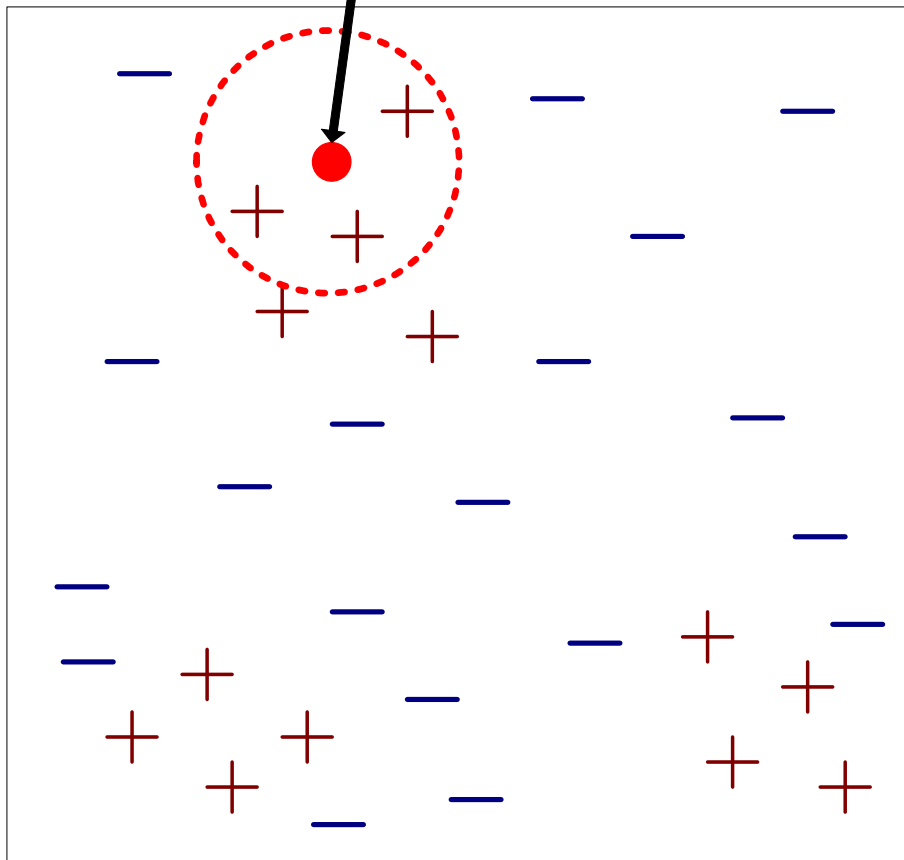
### ■ 基本思想：

- 如果行走姿态像鸭子，外观和声音也都像鸭子，那么它很可能就是一只鸭子。



## 3.2 最近邻分类过程

Unknown record



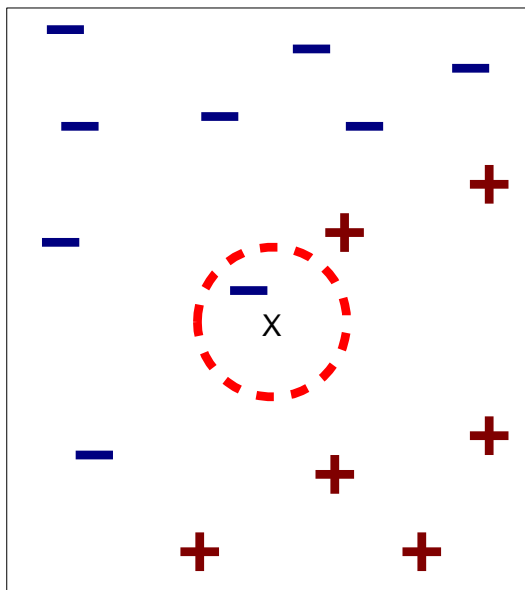
- 需要三件事情：

- 存储的记录集合.
- 计算记录之间距离的标准
- $K$ 的大小, 要得到的最近邻数目.

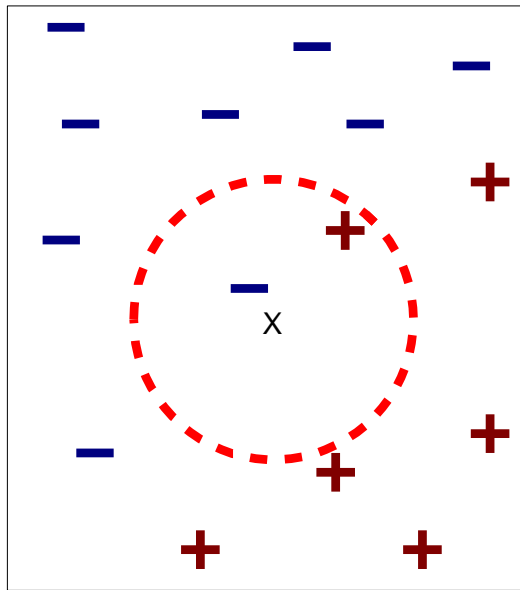
- 为分类一个未知记录：

- 计算与其它训练记录的距离.
- 识别出  $k$ 个最近邻居.
- 使用最近邻居的类标识来决定未知记录的类标识  
(可采取多数表决方案).

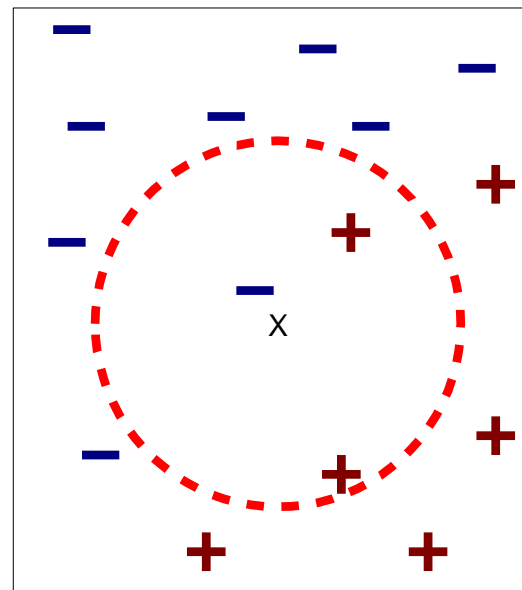
## 3.2 最近邻的定义



(a) 1-nearest neighbor



(b) 2-nearest neighbor



(c) 3-nearest neighbor

一个记录 $x$ 的 $K$ -最近邻是离 $x$ 距离最近的 $k$ 个数据点.



## 3.2 最近邻分类——距离

- 计算两点间的距离：

- 欧几里德距离：

$$d(p, q) = \sqrt{\sum_i (p_i - q_i)^2}$$

- 从最近邻的列表中决定该记录的类别.

- 在k-最近邻的点中取多数点的类标识来分类该点.

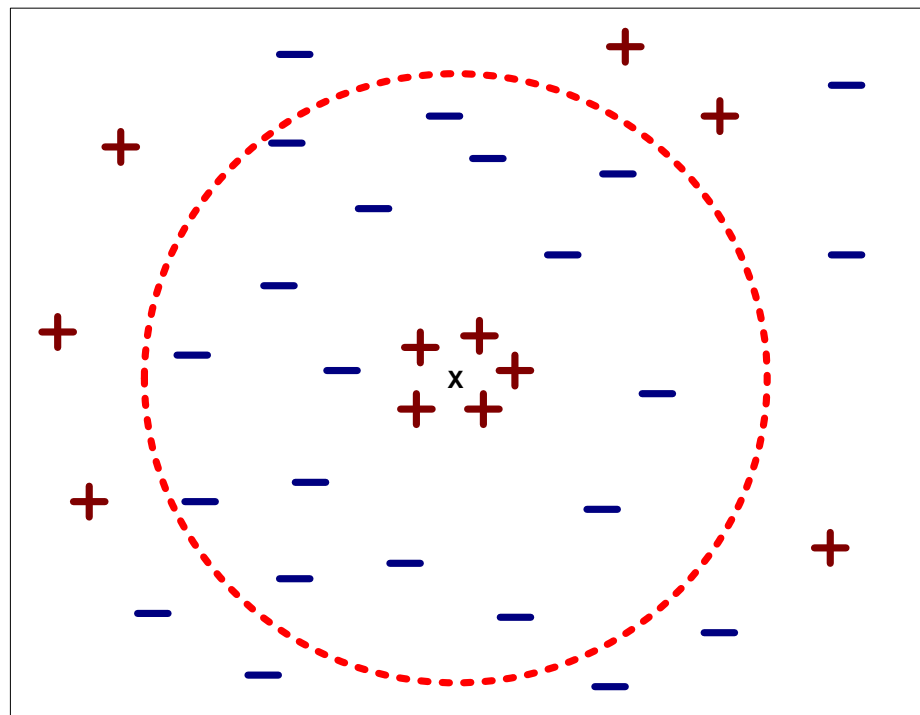
- 按照距离加权投票的结果：

- 加权因素：  $w = 1/d^2$  .

## 3.2 最近邻分类——K值

### ■ k 值的选择:

- 如果  $k$  太小, 则容易受到训练数据中噪音的影响, 而产生过分拟合的现象 (对噪音敏感) .
- 如果  $k$  太大, 可能会误分类测试样例, 因为最近邻列表中可能包含远离其近邻的数据点.





## 3.2 最近邻分类——讨论

---

### ■ 规范化问题

- 属性需要被标准化(统一度量尺度), 以防止距离度量受某一属性的支配.
- 例如:
  - 人的身高可能的变化范围为从 1.5m 到 2.5m
  - 人的体重可能的变化范围为从 90磅到 300磅
  - 一个人的收入可能的变化范围为\$10K to \$1M

## 3.2 最近邻分类——讨论

- 欧几里德度量的问题：

- 高维数据.

- 维数灾难.

- 可能产生直观上相反的结果. 如

1 1 1 1 1 1 1 1 1 1 0

0 1 1 1 1 1 1 1 1 1 1

VS

1 0 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 1

$$d = 1.4142$$

$$d = 1.4142$$

方法：把向量标准化为单位长度后再进行计算



## 3.2 最近邻分类——特点


---

- $k$ -NN 分类器是一种消极的学习方法.
  - 它不需要明确地建立模型.
  - 与积极学习方法（如决策树的归纳和基于规则的系统）不同.
  - 分类未知记录的花费相对昂贵.



## 3.3 基于规则的分类——概述

- 基于规则的分类器是使用一组 “if...then ...” 的规则(规则集合)来分类记录的技术.
- 每条规则的形式:  $(\text{条件}) \rightarrow y$ 
  - 其中
    - 条件是属性测试的合取,  $\wedge (A_k = v_k)$  .
    - $y$  是类标记.
  - 规则的左边: 规则的前件或条件.
  - 规则的右边: 规则的后件.
  - 示例:
    - $(\text{体温恒定} = \text{Yes}) \wedge (\text{胎生} = \text{Yes}) \rightarrow \text{哺乳类}$
    - $(\text{年收入} < 50\text{K}) \wedge (\text{归还} = \text{Yes}) \rightarrow \text{欺骗} = \text{No}$



Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
human	warm	yes	no	no	mammals
python	cold	no	no	no	reptiles
salmon	cold	no	no	yes	fishes
whale	warm	yes	no	yes	mammals
frog	cold	no	no	sometimes	amphibians
komodo	cold	no	no	no	reptiles
bat	warm	yes	yes	no	mammals
pigeon	warm	no	yes	no	birds
cat	warm	yes	no	no	mammals
leopard shark	cold	yes	no	yes	fishes
turtle	cold	no	no	sometimes	reptiles
penguin	warm	no	no	sometimes	birds
porcupine	warm	yes	no	no	mammals
eel	cold	no	no	yes	fishes
salamander	cold	no	no	sometimes	amphibians
gila monster	cold	no	no	no	reptiles
platypus	warm	no	no	no	mammals
owl	warm	no	yes	no	birds
dolphin	warm	yes	no	yes	mammals
eagle	warm	no	yes	no	birds

# 得到规则——示例

- R1: (胎生 = no)  $\wedge$  (会飞 = yes)  $\rightarrow$  鸟类
- R2: (胎生 = no)  $\wedge$  (水生 = yes)  $\rightarrow$  鱼类
- R3: (胎生 = yes)  $\wedge$  (体温 = 恒温)  $\rightarrow$  哺乳类
- R4: (胎生 = no)  $\wedge$  (会飞 = no)  $\rightarrow$  爬行类
- R5: (水生 = 有时)  $\rightarrow$  两栖类

## 3.3 基于规则的分类——预测

- 如果一个规则  $r$  的前件和事例  $x$  的属性匹配，则称  $r$  覆盖  $x$ 。当  $r$  覆盖给定的记录时，称  $r$  被激发或触发。
- $R1: (\text{胎生} = \text{no}) \wedge (\text{会飞} = \text{yes}) \rightarrow \text{鸟类}$
- $R2: (\text{胎生} = \text{no}) \wedge (\text{水生} = \text{yes}) \rightarrow \text{鱼类}$
- $R3: (\text{胎生} = \text{yes}) \wedge (\text{体温} = \text{恒温}) \rightarrow \text{哺乳类}$
- $R4: (\text{胎生} = \text{no}) \wedge (\text{会飞} = \text{no}) \rightarrow \text{爬行类}$
- $R5: (\text{水生} = \text{有时}) \rightarrow \text{两栖类}$

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
hawk	warm	no	yes	no	?
grizzly bear	warm	yes	no	no	?

规则 R1 覆盖第一种动物 (老鹰  $\Rightarrow$  鸟)

规则 R3 覆盖第二种动物 (大灰熊  $\Rightarrow$  哺乳类)

## 3.3 基于规则的分类——预测

R1: (胎生 = no)  $\wedge$  (会飞 = yes)  $\rightarrow$  鸟类

■ R2: (胎生 = no)  $\wedge$  (水生 = yes)  $\rightarrow$  鱼类

■ R3: (胎生 = yes)  $\wedge$  (体温 = 恒温)  $\rightarrow$  哺乳类

■ R4: (胎生 = no)  $\wedge$  (会飞 = no)  $\rightarrow$  爬行类

■ R5: (水生 = 有时)  $\rightarrow$  两栖类

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
lemur	warm	yes	no	no	?
turtle	cold	no	no	sometimes	?
dogfish shark	cold	yes	no	yes	?

根据测试记录所触发的规则来分类记录：

1. 狐猴是恒温动物, 能生育幼子, 这触发了规则 R3, 所以它被归于哺乳类.
2. 海龟同时触发了两个规则R4和R5.
3. 没有规则可以分类角鲨鱼.



## 3.3 基于规则的分类——规则的性质

### ■ 互斥规则

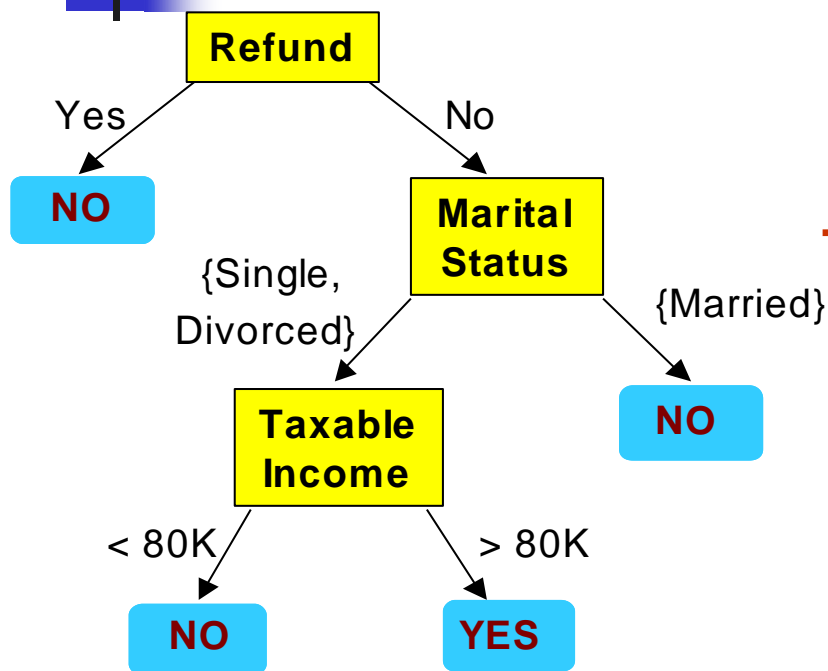
- 定义：如果规则集合 $R$ 中不存在两条规则被同一条记录同时触发, 则称 $R$ 中的规则是互斥的.
- 作用：每条记录至多被 $R$ 中的一条规则覆盖.

### ■ 穷举规则

- 定义：如果对属性值的任一组合,  $R$ 中都存在一个规则加以覆盖, 则称 $R$ 具有穷举覆盖.
- 作用：每条记录至少被 $R$ 中的一条规则覆盖.

- 2个性质共同作用：保证每条记录当且仅当被一条规则覆盖。

## 例如：从决策树得到的规则集合



### Classification Rules

(Refund=Yes) ==> No

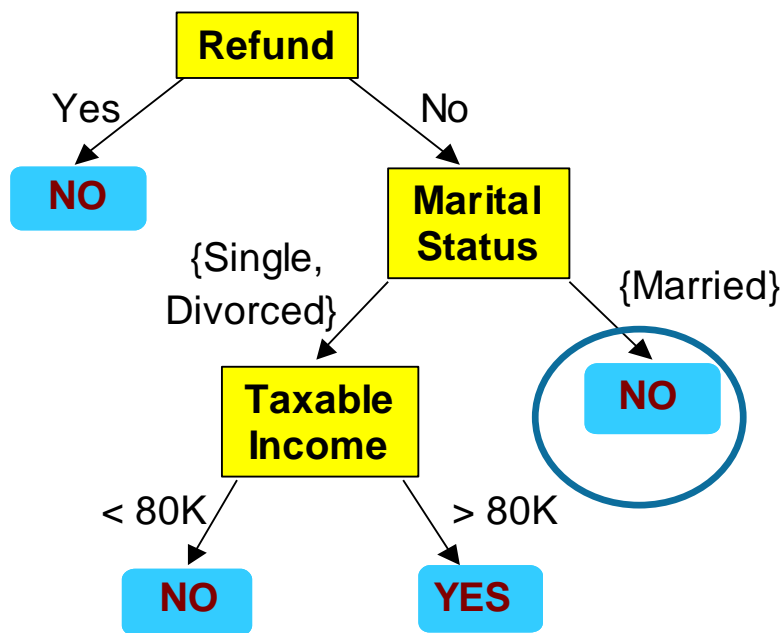
(Refund=No, Marital Status={Single, Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single, Divorced}, Taxable Income>80K) ==> Yes

(Refund=No, Marital Status={Married}) ==> No

1. 有多少个叶子就有多少条规则。
2. 规则是互斥的和穷举的。
3. 规则集合包含与决策树一样多的信息。

# 规则可以被简化



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Initial Rule:  $(\text{Refund}=\text{No}) \wedge (\text{Status}=\text{Married}) \rightarrow \text{No}$

Simplified Rule:  $(\text{Status}=\text{Married}) \rightarrow \text{No}$



# 规则简化的后果

- 规则不再是互斥的.
  - 一条记录可能触发多条规则,可能会引发预测冲突现象.
  - 解决方法?
    - 有序规则集.
    - 无序规则集— 使用投票(加权)确定测试记录的类标号.
- 规则不再具有穷举特征.
  - 一条记录可能不触发任何规则.
  - 解决方法?
    - 使用一条默认规则来覆盖那些未被覆盖的记录.
    - $R_d: ( ) \rightarrow y_d \cdot y_d$  没有被覆盖的训练记录中的多数类.





## 3.3 基于规则的分类——有序规则

---

- 规则集合中的规则按照优先级降序排列。
  - 优先级定义方法多样：准确率、覆盖率、描述长度、产生顺序等
  - 一个有序的规则集合也被称为一个**决策表**。
- 当一个测试记录出现时：
  - 由覆盖记录的最高等级的规则对其进行分类, 可避免类冲突。
  - 若无规则被触发, 它被指定为缺省类。



## 3.3 基于规则的分类——无序规则

■ 允许一条测试记录触发多条分类规则。

- 把每条被触发的规则后件作为对相应类的一次投票，投票最多的类作为该测试记录的类标号。
- 也可以在投票时用规则的准确率加权。

■ 特点：

- 测试一个记录时，不易受由于选择不当的规则而产生的影响。
- 由于不必维护规则的顺序，建立模型的相对开销较小。
- 对测试记录分类非常繁重，因为测试记录的属性要与规则集中的每一条规则的前件作比较。



## 3.3 基于规则的分类——学习方法

---

- 从训练集合中提取一组规则来识别数据集的属性与类标号之间的关键联系.
- 直接方法：
  - 直接从数据中提取规则.
  - 如：RIPPER, CN2
- 间接方法：
  - 从其它分类模型（例如. 决策树, 神经元网络等）中提取规则.
  - 如：C4.5



## 3.3 基于规则的分类——优点

---

- 具有与决策树一样好的表达能力.
- 易于理解.
- 易于产生.
- 能快速分类新的事例.
- 模型的性能可与决策树相媲美.