**Алгоритм индуцирования знаний из БД**

Алгоритм генерирует продукционные правила.

В алгоритме используется представле-ние знаний в виде деревьев решений.

*Рассмотрим пример.*

Пусть необходимо построить базу знаний для получения ответа: «Как поступить, чтобы при-быль росла?».

**Исходная база данных, из которой извлекаются знания**

|  |  |  |  |
| --- | --- | --- | --- |
| **ПРИБЫЛЬ** | ВОЗРАСТ | КОНКУРЕНЦИЯ | ТИП |
| **падает** | старый | нет | ПО |
| **падает** | средний | есть | ПО |
| **растёт** | средний | нет | ЭВМ |
| **падает** | старый | нет | ЭВМ |
| **растёт** | новый | нет | ЭВМ |
| **растёт** | новый | нет | ПО |
| **растёт** | средний | нет | ПО |
| **растёт** | новый | есть | ПО |
| **падает** | средний | есть | ЭВМ |
| **падает** | старый | есть | ПО |

Искомый атрибут «Прибыль» будем называть *атрибутом класса*.

Для построения дерева решений нужно взять один из атрибутов таблицы в качестве *основного (корневого) атрибута*. Пусть это будет «Возраст».

Преобразуем исходную таблицу к следующему виду (сортируем по графе Возраст):





Поскольку теперь для атрибута класса наше дерево решений выводит однозначный ответ, то дерево решений построено.

Порождаем правила:

**1.** **ЕСЛИ** Возраст = новый

**ТО** Прибыль = растёт.

**2.** **ЕСЛИ** Возраст = старый

**ТО** Прибыль = падает.

**3.** **ЕСЛИ** Возраст = средний

**И** Конкуренция = нет

**ТО** Прибыль = растёт.

**4.** **ЕСЛИ** Возраст = средний

**И** Конкуренция = есть

**ТО** Прибыль = падает.

**Алгоритм C4.5**

Улучшает базовый алгоритм индуцирования знаний.

**Основнoе отличие:** следующий условный атрибут, по которому проводится разбиение, определяется по критерию минимизации энтропии.

Теперь алгоритм не зависит от порядка следования атрибутов таблицы данных.

**Общее описание  
алгоритма C4.5**

**Обозначения**

*T* — множество примеров (таблица или подтаблица данных);

*m* — количество условных атрибутов (столбцов таблицы)

|*T* | — мощность множества примеров (количество строк в таблице или подтаблице данных);

*C***1** , *C***2** , …, *C****k*** — значения, принимаемые атрибутом класса;

X — текущий условный атрибут, по которому мы хотим провести разбиение

***A*1** , *A***2** , …, *A****n*** — значения, принимаемые текущим условным атрибутом;

**Выбор условного атрибута для разбиения**

Пусть рассматриваем условный атрибут *X*, принимающий *n* значений *A*1, *A*2 ... *An*. Тогда разбиение множества (таблицы) *T* по атрибуту *X* даст нам подмножества (подтаблицы) *T*1, *T*2 ... *Tn*.

Пусть freq(*Cj*,*T* ) — количество примеров из множества *T*, в которых атрибут класса равен *Cj*

Тогда вероятность того, что случайно выбранная строка из таблицы *T* будет принадлежать классу *Cj*, равна



Например, вероятность того, что прибыль будет расти, составляет *P* = 5 / 10 = 0,5

Согласно теории информации, количество содержащейся в сообщении информации зависит от её вероятности log2(1/*P*) = - log2(P).

В качестве единицы энтропии принят бит, что соответствует логарифмам при основании 2.

Энтропия таблицы T, то есть среднее количество информации, необходимое для определения класса, к которому относится строка из таблицы *T*:



Энтропия таблицы *T* после её разбиения по атрибуту *X* на *n* подтаблиц:



Критерий для выбора атрибута *X* – следующего атрибута для разбиения:



**Шаги алгоритма C4.5**

**Шаг 1.** Для всех условных атрибутов *X*1, … *Xm* таблицы *T* вычисляем критерий разбиения Gain(*Xi*). Выбираем такой атрибут *X*, для которого Gain(*Xi*) максимально.

**Шаг 2.** Разбиваем таблицу по выбранному атрибуту на *N* подтаблиц. Проверяем каждую подтаблицу следующим образом.

**2.1.** Если подтаблица монотонна (все строки относятся к одному классу), то порождаемправило.

**2.2.** В противном случае рекурсивно применяем алгоритм C4.5 к полученной подтаблице

**Пример работы алгоритма C4.5**

В качестве примера возьмём уже известную нам задачу о построении базы знаний для получения ответа: «Как поступить, чтобы прибыль росла?».

Рассмотрим поведение алгоритма C4.5.

1. Рассчитаем Gain(*X*) для всех условных атрибутов исходной таблицы.

Info(*T*) = -(0,5·log2(0.5) + 0,5·log2(0.5)) = -(-0,5-0,5) = 1

Расчёт критерия разбиения для атрибута «**ВОЗРАСТ**»

Info(*T*1) = -(3/3 · log2(3/3)) = 0.

Info(*T*2) = -(3/3 · log2(3/3)) = 0.

Info(*T*3) = -(2/4 · log2(2/4) + 2/4 · log2(2/4))= 1.

InfoВОЗРАСТ(*T*) = 3/10 · 0 + 3/10 · 0 + 4/10 · 1= 0,4;

**Gain(ВОЗРАСТ) = 1 – 0,4 = 0,6.**

Расчёт критерия разбиения для атрибута «**КОНКУРЕЦИЯ**»

Info(*T*1) = -(1/4 · log2(1/4) + 3/4 · log2(3/4)) = 1.

Info(*T*2) = -(2/6 · log2(2/6) + 4/6 · log2(4/6))= 1.59.

InfoКОНКУРЕНЦИЯ(*T*) = 4/10 · 1 + 6/10 · 1,59 = 1.354

**Gain(КОНКУРЕНЦИЯ) = 1 – 1,354 = -0,354.**

Расчёт критерия разбиения для атрибута «**ТИП**»

Info(*T*1) = -(2/4 · log2(2/4) + 2/4 · log2(2/4))= = 1.

Info(*T*2) = -(3/6 · log2(3/6) + 3/6 · log2(3/6))= = 1.

InfoТИП(*T*) = 4/10 · 1 + 6/10 · 1 = 1

**Gain(ТИП) = 1 – 1 = 0.**

Критерий разбиения Gain(*Xi*) максимально для условного атрибута «**ВОЗРАСТ**»

Разбиваем таблицу по выбранному атрибуту на *N* подтаблиц.

Повторяем алгоритм для полученных подтаблиц (см. Шаги алгоритма C4.5).