ЛЕКЦИЯ 7. Интеллектуальный анализ данных: задачи классификации.

Отношение прироста информации.

Энтропия как мера эффективности разбиения в узле может использоваться совместно с методологией разбиения путем создания отдельной ветви для каждого значения атрибута. Это так называемый алгоритм ID3 (Iterative Dihotomizer 3 – Итеративный дихотомайзер), разработанный австралийским ученым Дж. Р. Куинленом. Практическое применение данного алгоритма сопряжено с большой проблемой, а именно: только при разбиении набора данных на достаточное число небольших подмножеств количество классов, представленное в каждом узле, имеет тенденцию сокращаться и их энтропия, следовательно, тоже, поэтому деревья решений, которые строятся с использованием критерия уменьшения энтропии, предрасположены к сильной ветвистости.

Сложные деревья с большим числом ветвей имеют узлы с малым количеством примеров, трудно интерпретируются, ведут к снижению устойчивости модели в целом. Для решения этой проблемы были разработаны модификации алгоритма ID3, такие как алгоритмы C4,5, C5.0 и др. В них используется отношение полного прироста информации, полученного в результате данного разбиения, к приросту информации в данном узле. Такой метод называется отношением прироста информации. Он уменьшает ветвистость деревьев.

Алгоритм ID3

Одними из наиболее популярных алгоритмов построения деревьев решений являются алгоритм ID3 и его модификация C4,5. Алгоритм ID3 начинает работу со всеми обучающими примерами в корневом узле дерева. Для разделения множества примеров корневого узла выбирается один из атрибутов, и для каждого значения, принимаемого этим атрибутом, строится ветвь и создается дочерний узел. Затем все примеры распределяются по дочерним узлам в соответствии со значением атрибута (рис.1).

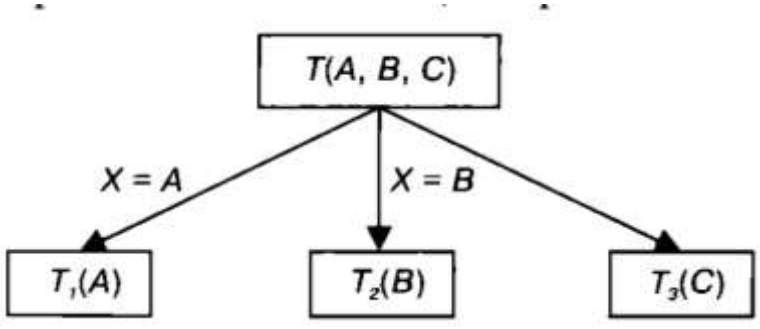

Рис. 1. Разбиение по атрибуту

Поясним это на рис. 1. Пусть атрибут X принимает три значения: A, B и C. Тогда при разбиении исходного множества T по атрибуту X алгоритм создаст три дочерних узла $T_1(A)$, $T_2(B)$ и $T_3(C)$, в первый из которых будут помещены все записи со значением A, во второй со значением B, а в третий со значением C.

Алгоритм повторяется рекурсивно до тех пор, пока в узлах не останутся только примеры одного класса, после чего узлы будут объявлены листами и разбиение прекратится. Наиболее проблемным этапом алгоритма является выбор атрибута, по которому будет производиться разбиение в каждом узле. Для выбора атрибута разбиения ID3 использует критерий, называемый приростом информации, или уменьшением энтропии.

Введем в рассмотрение меру прироста информации, вычисляемую как:

$$Gain(S) = Info(T) - Info_S(T),$$

где $Info(T)$ – энтропия множества T до разбиения;

$Info_S(T)$ – энтропия после разбиения S.

Данная мера представляет собой прирост количества информации, полученный в результате разделения множества T на подмножества $T_1, T_2, \dots, T_k$ с помощью разбиения S. В качестве наилучшего атрибута для использования вразбиении S выбирается тот атрибут, который обеспечивает наибольший прирост информации $Gain(S)$.

Рассмотрим пример. Пусть имеется набор данных T, содержащий 14 записей и 3 входных атрибута A_1, A_2, A_3, а также выходной атрибут (метку класса) C, который принимает два значения – C_1 и C_2 (табл. 1),

Таблица 1. Набор данных для иллюстрации работы алгоритма ID3

N п/п	A_1	A_2	A_3	C
1	A	70	Да	C_1
2	A	90	Да	C_2
3	A	85	Нет	C_2
4	A	95	Нет	C_2
5	A	70	Нет	C_1
6	B	90	Да	C_1
7	B	78	Нет	C_1
8	B	65	Да	C_1
9	B	75	Нет	C_1
10	C	80	Да	C_2
11	C	70	Да	C_2
12	C	80	Нет	C_1

13	C	80	Нет	C_1
14	C	96	Нет	C_1

В данном примере 9 записей относится к классу C_1, а пять к классу C_2. Энтропия множества T будет $Info(T) = -\left(\frac{9}{14}\right) log_2\left(\frac{9}{14}\right) - \left(\frac{5}{14}\right) log_2\left(\frac{5}{14}\right) = 0{,}94$ бит.

После использования атрибута A_1 для разбиения исходного множества на три подмножества: $T_1(A)$, $T_2(B)$ и $T_3(C)$ общая энтропия разбиения составит (условно полагаем, что $log_2(0) = 0$):

$$Info_{S_1}(T) = \frac{5}{14}\left(-\frac{2}{5} log_2\left(\frac{2}{5}\right) - \frac{3}{5} log_2\left(\frac{3}{5}\right)\right) + \frac{4}{14}\left(-\frac{4}{4} log_2\left(\frac{4}{4}\right) - \frac{0}{4} log_2\left(\frac{0}{4}\right)\right) + \frac{5}{14}\left(-\frac{3}{5} log_2\left(\frac{3}{5}\right) - \frac{2}{5} log_2\left(\frac{2}{5}\right)\right) = 0{,}694 \text{ бит},$$

Первое слагаемое формулы A – 5 шт. из них C_1 – 2 шт., C_2 – 3 шт.

Таким образом, прирост информации при использовании атрибута A_1 для разбиения исходного подмножества составит:

$$Gain(S_1) = 0{,}94 - 0{,}694 = 0{,}246 \text{ бит}$$

При использовании разбиения на основе атрибута A_3, проведя аналогичные вычисления, получим:

$$Info_{S_3}(T) = \frac{6}{14}\left(-\frac{3}{6} log_2\left(\frac{3}{6}\right) - \frac{3}{6} log_2\left(\frac{3}{6}\right)\right) + \frac{8}{14}\left(-\frac{6}{8} log_2\left(\frac{6}{8}\right) - \frac{2}{8} log_2\left(\frac{2}{8}\right)\right) = 0{,}892 \text{ бит}$$

Тогда прирост информации, обеспечиваемый разбиением исходного множества на основе атрибута A_3, будет

$$Gain(S_3) = 0{,}94 - 0{,}892 = 0{,}048 \text{ бит}$$

И так, для начального разделения множества T алгоритм выберет разбиение S_1 с использованием атрибута A_1 как обеспечивающее наибольший прирост информации, Для нахождения оптимальной проверки необходимо также рассмотреть атрибут A_2.

Ранее рассматривались разбиения по категориальным атрибутам. Но в текущем примере мы столкнулись с необходимостью проверки по числовому атрибуту A_2. Проверки по таким атрибутам сложнее, поскольку требуется установка порогов для разбиения, которая в общем случае может быть произвольной. Существует алгоритм для определения оптимального значения порога z. Сначала обучающие примеры сортируются в том порядке, в котором будут рассматриваться значения атрибута: $(v_1, v_2, \dots, v_m)$. Тогда существует только $m - 1$ порогов из возможных разбиений, каждое из которых должно быть

проверено на предмет максимального прироста информации, При этом в качестве порога могут выбираться срединные точки между соседними значениями $(v_i, v_{i+1})/2$ или наименьшие значения из каждого интервала (v_i, v_{i+1}).

Для иллюстрации данного способа нахождения порога возьмем числовой атрибут A_2. После сортировки примеров множества по возрастанию значений атрибута получим следующий порядок: $\{65, 70, 75, 78, 80, 85, 90, 95, 96\}$ и множество возможных пороговых значений $z = \{65, 70, 75, 78, 80, 85, 90, 95\}$, Из девяти значений оптимальным будет то, которое обеспечивает наибольший прирост информации. Если для каждого потенциального порога определить прирост информации подобно тому, как ранее это делалось для значений категориальной переменной, то можно обнаружить, что лучшим значением порога будет $z = 80$. Соответствующий процесс вычисления прироста информации для разбиения S_2 ($A_2 \leq 80$ или $A_2 \geq 80$), производится следующим образом:

$$Info_{S_2}(T) = \frac{9}{14}\left(-\frac{7}{9}log_2\left(\frac{7}{9}\right) - \frac{2}{9}log_2\left(\frac{2}{9}\right)\right) + \frac{5}{14}\left(-\frac{2}{5}log_2\left(\frac{2}{5}\right) - \frac{3}{5}log_2\left(\frac{3}{5}\right)\right)$$

$$= 0{,}837 \text{ бит}$$

$$Gain(S_2) = 0{,}94 - 0{,}837 = 0{,}103 \text{ бит}$$

Теперь можно сравнить прирост информации для всех трех атрибутов иллюстративного примера. Видим, что атрибут A_1 по-прежнему обеспечивает наибольший прирост информации $0{,}246$ и, следовательно, он должен быть выбран для использования в первом разбиении дерева решений.

Корневой узел произведет проверку для атрибута A_1, в результате чего будут созданы три ветви по одной для каждого значения атрибута (рис. 2).

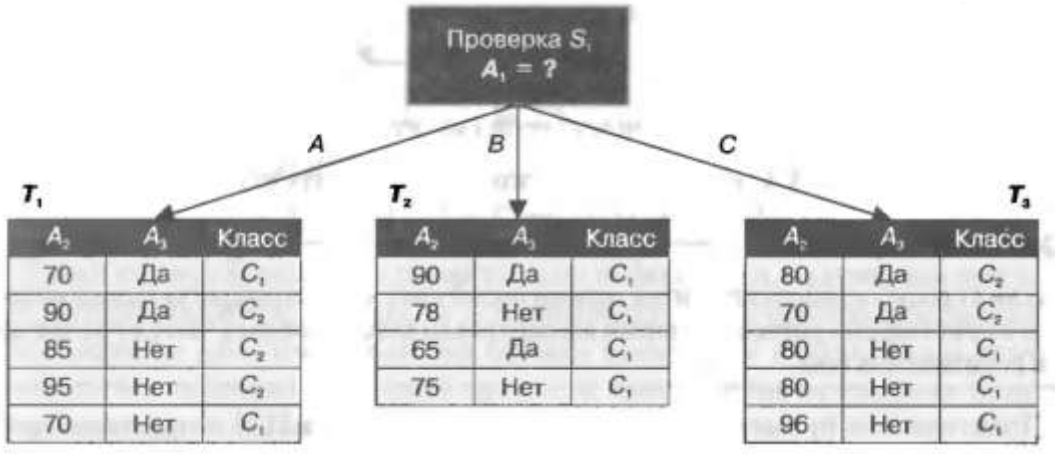

Рис. 2. Разбиение по первому атрибуту для примера из табл. 1

После начального разбиения все узлы потомки будут содержать по нескольку наблюдений из исходного множества, и для каждого узла затем будет

повторен процесс выбора атрибута разбиения и оптимизации порога (если для разбиения используется числовой атрибут). Поскольку узел потомок T_2, полученный в ветви для значения B, содержит 4 наблюдения, которые относятся к одному классу, то энтропия равна 0, узел объявляется листом и дальнейшее ветвление для него не проводится.

Для узла T_1, включающего 5 наблюдений, может быть сделана проверка по оставшимся атрибутам. Опуская промежуточные расчеты, констатируем, что оптимальное разбиение будет достигнуто с помощью атрибута A_2 для альтернативных вариантов $A_2 \leq 70$ или $A_2 \geq 70$. Поскольку множество T_1 содержит 5 наблюдений, из которых два относятся к классу C_1, а три к классу C_2, то в соответствии с формулой энтропии можно записать:

$$Info_{S_2}(T_1) = -\frac{2}{5} log_2\left(\frac{2}{5}\right) - \frac{3}{5} log_2\left(\frac{3}{5}\right) = 0{,}97 \text{ бит}$$

При использовании атрибута A_2 для разбиения T_1 с порогом 70 получим чистые узлы, поэтому $Info_{S_2}(T_1) = 0$. Прирост информации, обеспеченный данным разбиением, составит $Gain(S_3) = 0{,}94 - 0 = 0{,}94$ бит и будет максимальным.

Аналогичным образом могут быть выполнены расчеты и для подмножества T_3. В нем оптимальна проверка по значениям атрибута A_3. Следовательно, будут созданы две новые ветви, соответствующие значениям $A_3 = \text{Да}$ и $A_3 = \text{Нет}$. В результате получим два однородных подмножества, содержащих наблюдения одного класса. Результирующее дерево представлено на рис. 3.

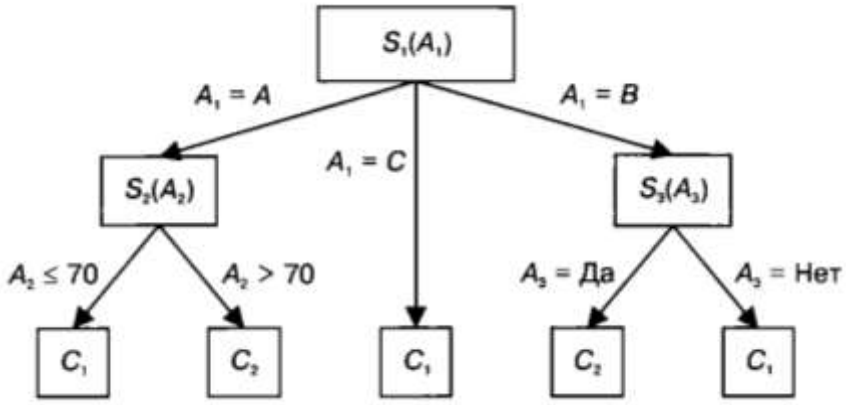

Рис. 3. Сгенерированное алгоритмом ID3 дерево решений

Практическое применение классического алгоритма ID3 сопряжено с рядом проблем, характерных для моделей, основанных на обучении вообще и деревьев решений в частности. Основными из них являются переобучение и наличие пропусков в данных. Для их эффективного преодоления ID3 был доработан, в результате чего появилось его расширение, названное C4,5.

Практический пример работы алгоритма C4.5

Чтобы про иллюстрировать работу алгоритма C4.5, рассмотрим задачу кредитного скоринга (оценки кредитоспособности). Воспользуемся набором данных для оценки кредитного риска, представленным в табл. 2.

Таблица 2. Кредитные истории клиентов

№ клиента	Сбережения	Другие активы (недвижимость, автомобиль и т.д.)	Годовой доход, тыс. у. е.	Кредитный риск
1	Средние	Высокие	75	Низкий
2	Низкие	Низкие	50	Высокий
3	Высокие	Средние	25	Высокий
4	Средние	Средние	50	Низкий
5	Низкие	Средние	100	Низкий
6	Высокие	Высокие	25	Низкий
7	Низкие	Низкие	25	Высокий
8	Средние	Средние	75	Низкий

Уровень риска, связанный с выдачей кредита клиенту, определяется на основе трех признаков: количества имеющихся у него сбережений, наличия собственности (автомобиль, недвижимость и т. д,), а также годового дохода. Первые два показателя представлены в модели категориальными переменными *Сбережения* и *Другие активы*, которые могут принимать три значения –Высокие, Низкие и Средние. Доход клиента представлен числовой переменной *Годовой доход*. Поскольку в 5 из 8 записей целевая переменная указывает на низкий кредитный риск, а в оставшихся 3 записях на высокий, энтропия исходного множества до разбиения составит:

$$Info(T) = -\sum_j p_j \, log_{\,2}(p_j) = -\frac{5}{8} log_{\,2}\left(\frac{5}{8}\right) - \frac{3}{8} log_{\,2}\left(\frac{3}{8}\right) = 0{,}954 \text{ бит}$$

Рассмотрим разбиение по атрибуту *Сбережения*. Две записи имеют значение данного атрибута *Высокие*, три – *Средние* и три – *Низкие*. Тогда соответствующие вероятности будут: $P_{\text{высокие}} = \frac{2}{8}, P_{\text{средние}} = \frac{3}{8}$ и $P_{\text{низкие}} = \frac{3}{8}$. Из двух записей, в которых присутствуют высокие сбережения, одна имеет значение целевой переменной, указывающее на высокий кредитный риск, а вторая – на низкий, Torда при случайном выборе из этих двух записей вероятность появления каждого класса составит 0,5. Следовательно, энтропия для значения *Высокие* атрибута *Сбережения* составит:

$$Info_{\text{высокие}}(T) = -\frac{1}{2}log_{\,2}\left(\frac{1}{2}\right) - \frac{1}{2}log_{\,2}\left(\frac{1}{2}\right) = 1 \text{ бит}$$

Три записи, в которых имеют место средние сбережения, содержат целевую переменную, указывающую на низкий кредитный риск, поэтому соответствующая энтропия будет:

$$Info_{\text{средние}}(T) = -\frac{3}{3}log_{\,2}\left(\frac{3}{3}\right) - \frac{0}{3}log_{\,2}\left(\frac{0}{3}\right) = 0$$

где условно полагаем, что $log_{\,2}(0) = 0$.

В технических приложениях информация рассматривается как аналог полезного сигнала, а энтропия как аналог шума. Отсюда понятно, почему энтропия для средних сбережений равна 0: мы получили чистый сигнал без шума. Таким образом, если потенциальный заемщик имеет средние сбережения, то кредитный риск будет низким с поддержкой 100 %.

Теперь приведем аналогичные рассуждения для значения *Низкие* атрибута *Сбережения*, Соответствующая энтропия будет:

$$Info_{\text{низкие}}(T) = -\frac{1}{3}log_{\,2}\left(\frac{1}{3}\right) - \frac{2}{3}log_{\,2}\left(\frac{2}{3}\right) = 0{,}918 \text{ бит}$$

Затем, используя выражение, рассчитаем полную энтропию разбиения:

$$Info(S) = \frac{N_1}{N}Info(T_1) + \frac{N_2}{N}Info(T_2) + \cdots + \frac{N_k}{N}Info(T_k) = \sum_{i=1}^{k}\frac{N_i}{N}Info(T_i)$$

$$Info_{\text{сбережения}}(T) = \frac{2}{8}\times 1 + \frac{3}{8}\times 0 + \frac{2}{8}\times 0{,}183 = 0{,}594 \text{ бита}$$

Тогда прирост информации, полученный в результате разбиения по атрибуту Сбережения, будет:

$$Gain(T) = Info(T) - Info_{\text{сбережения}}(T) = 0{,}954 - 0{,}594 = 0{,}36 \text{ бит}$$

Как интерпретировать полученное значение? Во-первых, $Info(T) = 0{,}954$ означает, что в среднем потребуется 0,954 бита для представления кредитного риска для 8 клиентов из табл. 2. Во-вторых, $Info_{\text{сбережения}}(T) = 0{,}594$ говорит о том, что разделение клиентов на три подмножества понизило среднее количество бит, требуемых для представления кредитного риска. Уменьшение энтропии после разбиения указывает на то, что в целом разбиение является полезным, то есть повышает чистоту дочерних узлов. В результате разбиения по переменной *Сбережения* мы получили прирост информации на 0,36 бита.

Теперь мы должны вычислить прирост информации для остальных возможных разбиений, представленных в табл. 3, и выбрать то из них, которое даст наибольший прирост информации (или, что одно и то же, уменьшение энтропии).

Таблица 3. Все варианты первого разбиения

№ разбиения	Дочерние узлы		
1	Сбережения = Низкие	Сбережения = Средние	Сбережения = Высокие
2	Активы = Низкие	Активы = Средние	Активы = Высокие
3	Доход ≤ 25 тыс. у. е.	Доход > 25 тыс. у. е.	
4	Доход ≤ 50 тыс. у. е.	Доход > 50 тыс. у. е.	
5	Доход ≤ 75 тыс. у. е.	Доход > 75 тыс. у. е.	

Потенциальное разбиение под номером 2 из табл. 3. использует атрибут *Активы*. В исходном множестве данных (табл. 2) две записи имеют значения данного атрибута *Высокие*, четыре записи – Средние и две – *Низкие*. Вероятности появления соответствующих значений будут следующие: $P_{\text{высокие}} = \frac{2}{8}, P_{\text{средние}} = \frac{4}{8}$ и $P_{\text{низкие}} = \frac{2}{8}$. Обе записи, в которых атрибут *Активы* принимает значение *Высокие* имеют значение целевой переменной, указывающее на низкий кредитный риск. В результате будет получен совершенно чистый узел с нулевой энтропией. Аналогичная ситуация ранее имела место для значения Высокие атрибута *Сбережения*.

Три записи из четырех, в которых Активы = Средние, указывают на низкий кредитный риск и одна запись на высокий. Энтропия соответствующего потомка рассчитывается как:

$$Info_{\text{средние}}(T) = -\frac{3}{4} log_2\left(\frac{3}{4}\right) - \frac{1}{4} log_2\left(\frac{1}{4}\right) = 0{,}811 \text{ бит}$$

В обеих записях, в которых клиенты имеют низкие активы, целевая переменная указывает на высокий кредитный риск, поэтому соответствующий дочерний узел также будет чистым с нулевой энтропией.

На основе результатов, полученных для каждого из трех узлов, вычислим полную энтропию разбиения по переменной *Активы*:

$$Info_{\text{активы}}(T) = \frac{2}{8} \times 0 + \frac{4}{8} \times 0{,}811 + \frac{2}{8} \times 0 = 0{,}406 \text{ бита}$$

Энтропия, полученная после разбиения по атрибуту *Активы*, меньше, чем после разбиения по атрибуту *Сбережения*, что указывает на более эффективное разделение записей различных классов. Чтобы убедиться в этом, вычислим прирост информации, полученный в результате разбиения по атрибуту *Активы*:

$$Gain(T) = Info(T) - Info_{\text{активы}}(T) = 0{,}954 - 0{,}406 = 0{,}548 \text{ бит}$$

Таким образом, прирост информации в результате разбиения по атрибуту *Активы* больше, чем при разбиении по атрибуту *Сбережения*, что выдвигает его на первое место среди возможных кандидатов.

При разбиении по числовому атрибуту *Доход* используются четыре порога – 25,50,75 и 100 тыс. Для возможного разбиения под номером 3 (табл. 3) три записи содержат значение переменной Доход меньше 25 тыс., а остальные пять записей – больше 25 тыс. Тогда соответствующие вероятности будут: $P_{\leq 25000} = \frac{3}{8}, P_{>25000} = \frac{5}{8}$.

Для одной из записей, в которых доход менее 25 тыс., указан низкий кредитный риск, а для двух других – высокий. Тогда энтропия узла, в который будут помещены клиенты с доходом, меньше или равным 25 тыс., составит:

$$Info_{\leq 25000}(T) = -\frac{1}{3} log_2\left(\frac{1}{3}\right) - \frac{2}{3} log_2\left(\frac{2}{3}\right) = 0{,}918 \text{ бит}$$

Четыре из пяти записей с доходом более 25 тыс. имеют низкий кредитный риск, и только одна запись – высокий. Тогда энтропия узла, куда попадут записи, сумма дохода клиента в которых превышает 25 тыс., составит:

$$Info_{>25000}(T) = -\frac{4}{5} log_2\left(\frac{4}{5}\right) - \frac{1}{5} log_2\left(\frac{1}{5}\right) = 0{,}722 \text{ бит}$$

Теперь вычислим энтропию для всего разбиения по условию Доход ≤ 25 тыс. у. е

$$Info_{\text{доход (25000)}}(T) = \frac{3}{8} \times 0{,}918 + \frac{5}{8} \times 0{,}722 = 0{,}796 \text{ бита}$$

Прирост информации, полученный в результате разбиения по атрибуту *Доход* с порогом 25 тыс.у.е.

$$Gain(T) = Info(T) - Info_{\text{доход (25000)}}(T) = 0{,}954 - 0{,}796 = 0{,}159 \text{ бит.}$$

Эти расчеты показывают, что наименее эффективным является разбиение 3.

Далее рассмотрим потенциальное разбиение 4 (табл. 3), в котором используется атрибут *Доход* с порогом 50 тыс. Вычислим энтропию:

$$Info_{\text{доход (50000)}}(T) = \frac{5}{8}\left(-\frac{2}{5} log_2\left(\frac{2}{5}\right) - \frac{3}{5} log_2\left(\frac{3}{5}\right)\right) + \frac{3}{8}\left(-\frac{3}{3} log_2\left(\frac{3}{3}\right) - \frac{0}{3} log_2\left(\frac{0}{3}\right)\right) = 0{,}607 \text{ бит}$$

Прирост информации составит:

$$Gain(T) = Info(T) - Info_{\text{доход (50000)}}(T) = 0{,}954 - 0{,}607 = 0{,}347 \text{ бит}$$

Данное значение прироста также существенно ниже, чем для атрибута *Активы*. И наконец, рассмотрим разбиение 5 по атрибуту Доход с использованием порога 75 тыс.:

$$Info_{\text{доход (75000)}}(T) = \frac{7}{8}\left(-\frac{4}{7}log_{\,2}\left(\frac{4}{7}\right) - \frac{3}{7}log_{\,2}\left(\frac{3}{7}\right)\right) + \frac{1}{8}\left(-\frac{1}{1}log_{\,2}\left(\frac{1}{1}\right) - \frac{0}{1}log_{\,2}\left(\frac{0}{1}\right)\right) = 0{,}862 \text{ бит}$$

$$Gain(T) = Info(T) - Info_{\text{доход (75000)}}(T) = 0{,}954 - 0{,}862 = 0{,}092 \text{ бит}$$

что является самым низким показателем среди всех потенциальных разбиений.

Таким образом, мы вычислили прирост информации, который обеспечивают все возможные разбиения корневого узла. Полученные результаты в компактной форме представлены в табл. 4.

Таблица 4. Варианты разбиений корневого узла с энтропией

№ разбиения	Дочерние узлы	Прирост информации
1	Сбережения = Низкие	0,36
	Сбережения = Средние	
	Сбережения = Высокие	
2	Активы = Низкие	0,548
	Активы = Средние	
	Активы = Высокие	
3	Доход ≤25 тыс. у. е.	0,159
	Доход > 25 тыс. у. е.	
4	Доход ≤50 тыс. у. е.	0,347
	Доход > 50 тыс. у. е.	
5	Доход ≤75 тыс. у. е.	0,092
	Доход > 75 тыс. у. е.	

Видно, что разбиение по атрибуту Активы обеспечило наибольший прирост информации, поэтому оно выбирается в качестве начального разбиения в корневом узле дерева. Схема начального разбиения представлена на рис. 4.

Рис. 4. Результат первого шага алгоритма C4.5

Получилось два листа для значений *Низкие* и *Высокие* атрибута Активы, дальнейшее разбиение которых производиться не будет, Для значения *Средние*

получен узел, содержащий одну запись со значением целевой переменной *Высокий* и три записи со значением *Низкий*. Поскольку в данном узле содержится смесь классов, алгоритм будет выполнять дальнейшее разбиение подмножества в нем (обозначим это подмножество T_1). Для этого вновь будет произведен поиск оптимального разбиения. Множество T_1 представлено в табл. 5.

Таблица 5. Множество T_1

№ клиента	Сбережения	Другие активы	Годовой доход, тыс. у. е.	Кредитный риск
3	Высокие	Средние	25	Высокий
4	Средние	Средние	50	Низкий
5	Низкие	Средние	100	Низкий
8	Средние	Средние	75	Низкий

Три из четырех записей в узле имеют значение целевой переменной *Низкий*, и одна *Высокий*. Энтропия узла до разбиения составит:

$$Info(T_1) = -\sum_j p_j \, log_2(p_j) = -\frac{3}{4} log_2\left(\frac{3}{4}\right) - \frac{1}{4} log_2\left(\frac{1}{4}\right) = 0{,}811 \text{ бит}$$

Возможные разбиения в узле показаны в табл. 6.

Таблица 6. Все варианты второго разбиения

№ разбиения	Дочерние узлы		
1	Сбережения = Низкие	Сбережения = Средние	Сбережения = Высокие
2	Доход ≤ 25 тыс. у. е.	Доход > 25 тыс. у. е.	
3	Доход ≤ 50 тыс. у. е.	Доход > 50 тыс. у. е.	
4	Доход ≤ 75 тыс. у. е.	Доход > 75 тыс. у. е.	

Для разбиения одна единственная запись с низкими сбережениями имеет и низкий кредитный риск наряду с двумя записями, содержащими средние сбережения. Единственная запись с высокими сбережениями имеет высокий кредитный риск. Поэтому энтропия узлов, полученных с помощью всех трех значений атрибута *Сбережения*, равна 0, то есть $Info_{\text{сбережения}}(T_1) = 0$. Все три потомка, полученные в результате разбиения, будут абсолютно чистыми. Ясно, что такое разбиение обеспечивает максимальный прирост информации:

$$Gain(T_1) = Info(T_1) - Info_{\text{сбережения}}(T_1) = 0{,}811 - 0 = 0{,}811 \text{ бит}$$

Аналогично можно показать, что энтропия для разбиения по атрибуту *Доход* с порогом 25 тыс. также равна 0 и такое разбиение обеспечивает максимальный прирост информации. Но остановим выбор на разбиении по атрибуту *Сбережения*. Полное дерево, полученное в результате данного разбиения, представлено на рис. 5.

Рис. 5. Полное дерево решений

Таблицы частот

Напомним, что ключевым моментом в работе алгоритмов ID3 и C4.5 является вычисление энтропии

$$Info(T) = -\sum_{j} p_j \, log_{\,2}(p_j)$$

где $p_j -$ вероятность того, что случайно выбранная из множества T запись относится к классу j.

Данная вероятность определяется как отношение числа записей j-го класса к общему числу записей. Это означает, что для вычисления энтропии совершенно не обязательно рассматривать все записи множества данных достаточно использовать информацию о частоте появления классов в этом множестве. Такая информация представляется в специальных таблицах, называемых таблицами частот. В столбцах таблицы частот приводятся атрибуты, которые могут использоваться для разбиения, и принимаемые ими значения, а в строках для каждого значения указывается количество записей, в которых атрибут принимает данное значение при определенном состоянии переменной класса. Например, для множества из табл. 1 может быть построена таблица частот следующего вида (табл. 7). Из табл. 9.9 можно увидеть, что из четырех записей, в которых атрибут

Активы принимает значение *Средние*, одна связана с высоким кредитным риском, а три – с низким.

Таблица 7. Пример таблицы частот

Сбережения			Активы			Годовой доход, тыс. у. е				Кредитный риск
Высокие	Средние	Низкие	Высокие	Средние	Низкие	≤ 25	$25-50$	$50-75$	>75	
1	0	2	0	1	2	2	1	1	0	Высокий
1	3	1	2	3	0	1	1	2	1	Низкий

Таблица частот сама по себе является очень информативным представлением, например, для разведочного анализа. Даже простой визуальный анализ табл. 7. позволяет ответить на вопрос: что критичнее с точки зрения кредитного риска низкие сбережения или низкие активы? Поскольку все клиенты, имеющие низкие активы, связаны с высоким кредитным риском, а клиенты с низкими сбережениями распределились поровну, можно предположить, что низкие активы более важны. Но риск зависит от состояний нескольких входных атрибутов, поэтому выводы, которые могут быть сделаны на основе визуального анализа таблицы частот, являются приближенными. Чтобы получить более достоверные результаты, необходимо использовать формальный подход.

Кроме того, таблица частот дает возможность сразу оценить эффективность разбиений по тому или иному атрибуту. Так, если в столбце для определённого значения атрибута присутствует 0, то соответствующее разбиение позволит получить максимальный прирост информации и чистый узел с нулевой энтропией, который будет объявлен листом.

Проблема переобучения

Основной недостаток алгоритма ID3 тенденция к переобучению. Например, если данные содержат шум, то число уникальных значений атрибутов увеличится, а в крайнем случае для каждого примера обучающего множества значения атрибутов окажутся уникальными. Следуя логике ID3, можно предположить, что при разбиении по такому атрибуту будет создано количество узлов, равное числу примеров, так как в каждом узле окажется по одному примеру. После этого каждый узел будет объявлен листом и дерево даст число правил, равное числу примеров обучающего набора, С точки зрения классифицирующей способности такая модель бесполезна.

В качестве примера описанной ситуации возьмем случай, когда атрибут A_1 в выборке, представленной в табл. 1, содержит не три значения A, B и C, а 14

значений от A до N. Тогда в результате разбиения будет сформировано 14 узлов, в каждый из которых попадет только один пример, и на этом построение дерева закончится. Дополнительно усложняет проблему тот факт, что критерий прироста информации всегда будет приводить к выбору атрибутов с наибольшим количеством уникальных значений. Действительно, если в результате разбиения будут получены множества, содержащие по одному объекту, образующему один класс, то частота появления этого класса окажется равной числу примеров, то есть 1. Таким образом, в выражении для оценки количества информации появится $log_2(1) = 0$. Следовательно, и все выражение обратится в ноль, то есть $Info_S(T) = 0$. Всегда будет обеспечен максимальный прирост информации $Gain(S) = max$, что приведет к выбору алгоритмом соответствующего атрибута.

Данная проблема решается введением нормировки. В рассмотрение включается дополнительный показатель, который представляет собой оценку потенциальной информации, созданной при разбиении множества T на n подмножеств T_i:

$$Split - Info(S) = -\sum_{i=1}^{n}\left(\left(\frac{|T_i|}{|T|}\right) log_2\left(\frac{|T_i|}{|T|}\right)\right)$$

С помощью этого показателя можно модифицировать критерий прироста информации, перейдя к отношению:

$$Split - Info(S) = \frac{Gain(T)}{Split - Info(T)}$$

Новый критерий позволяет оценить долю информации, полученной при разбиении, которая является полезной, то есть способствует улучшению классификации. Использование данного отношения обычно приводит к выбору более удачного атрибута, чем обычный критерий прироста. Вычисление отношения прироста проиллюстрируем с помощью при м ера из табл. 1.

Найдем прирост информации для разбиения S_1:

$$Split - Info(S_1) = -\frac{5}{14} log_2\left(\frac{5}{14}\right) - \frac{4}{14} log_2\left(\frac{4}{14}\right) - \frac{5}{14} log_2\left(\frac{5}{14}\right) = 1{,}577$$

Тогда отношение прироста будет:

$$Split - Info(S_1) = \frac{0{,}246}{1{,}577} = 0{,}156$$

Аналогичная процедура должна быть выполнена для остальных разбиений в дереве решений.

Смысл этой модификации прост. T_i/T – отношение числа примеров в i-м подмножестве, полученном в результате разбиения S, к числу примеров в

родительском множестве T. Если в результате разбиения получается большое число подмножеств с небольшим числом примеров, что характерно для переобучения, то параметр $Split - Info$ растет. Поскольку он стоит в знаменателе выражения, то $Gain - ratio$ есть обычный прирост информации, «оштрафованный» с помощью параметра $Split - Info$. Благодаря этому атрибут, для которого параметр $Split - Info$ растет, имеет меньше шансов быть выбранным для разбиений, чем при использовании обычного $Gain$-критерия.

Неизвестные значения атрибутов

Рассматривая алгоритмы, мы предполагали, что все значения атрибутов, используемых при разбиении, известны. Но типичной проблемой реальных наборов является отсутствие значений атрибутов в отдельных наблюдениях. Если в наборе данных, для которого строится дерево решений, имеют место отсутствующие значения, то можно выбрать один из вариантов решения данной проблемы:

- исключить все наблюдения, содержащие пропущенные значения;
- создать новый или модифицировать существующий алгоритм таким образом, чтобы он мог работать с пропусками в данных.

Первое решение простое, но оно часто неприемлемо из-за того, что в наборе данных может присутствовать большое количество записей с пропущенными значениями. И если все такие записи будут исключены из рассмотрения, то оставшихся записей окажется недостаточно для формирования обучающего множества.

Различные классификационные алгоритмы обычно основаны на заполнении пропусков наиболее вероятными значениями.

В алгоритме C4.5 предполагается, что наблюдения с неизвестными значениями имеют статистическое распределение соответствующего атрибута согласно относительной частоте появления известных значений. Например, рассмотрим множество наблюдений из табл. 1, в котором отсутствует значение атрибута A_1 в строке 6.

Введем в рассмотрение параметр F, который представляет собой число наблюдений в наборе данных с известным значением данного атрибута, отнесенное к общему числу наблюдений. Тогда модифицированный для работы с пропущенными значениями критерий прироста информации будет иметь вид:

$$Gain(S) = F(Info(T) - Info_S(T))$$

Аналогично параметр $Split - Info(S)$ может быть видоизменен путем отнесения числа наблюдений, содержащих пропущенные значения, к отдельным дополнительно создаваемым подмножествам. Если разбиение S дает n

подмножеств, его параметр $Split - Info(S)$ вычисляется, как если бы в результате разбиения было получено $n + 1$ подмножество.

Рассмотрим работу алгоритма C4.5, модифицированного для обработки пропущенных значений, на примере из табл. 1.

Таблица 1. Набор данных для иллюстрации работы алгоритма C4.5

N п/п	A_1	A_2	A_3	C
1	A	70	Да	C_1
2	A	90	Да	C_2
3	A	85	Нет	C_2
4	A	95	Нет	C_2
5	A	70	Нет	C_1
6		90	Да	C_1
7	B	78	Нет	C_1
8	B	65	Да	C_1
9	B	75	Нет	C_1
10	C	80	Да	C_2
11	C	70	Да	C_2
12	C	80	Нет	C_1
13	C	80	Нет	C_1
14	C	96	Нет	C_1

Из 13 наблюдений, в которых значения атрибута A_1 известны, 8 относятся к классу C_1 и 5 к классу C_2. Поэтому в результате разбиения по атрибуту A_1 получим:

$$Info_{S_1}(T) = -\left(\frac{8}{13}\right) log_2\left(\frac{8}{13}\right) - \left(\frac{5}{13}\right) log_2\left(\frac{5}{13}\right) = 0{,}961 \text{ бит}$$

После разделения исходного множества T с помощью атрибута A_1 на подмножества в соответствии со значениями атрибута A, B и C результирующая информация определяется следующим образом:

$$Info_{S_1}(T) = \frac{5}{13}\left(-\frac{2}{5} log_2\left(\frac{2}{5}\right) - \frac{3}{5} log_2\left(\frac{3}{5}\right)\right) + \frac{3}{13}\left(-\frac{3}{3} log_2\left(\frac{3}{3}\right) - \frac{0}{3} log_2\left(\frac{0}{3}\right)\right)$$
$$+ \frac{5}{14}\left(-\frac{3}{5} log_2\left(\frac{3}{5}\right) - \frac{2}{5} log_2\left(\frac{2}{5}\right)\right) = 0{,}747 \text{ бит}$$

Прирост информации, полученный в результате данного разбиения, корректируется с помощью фактора $F = 13/14 = 0{,}93$. Тогда:

$$Gain(S_1) = F(0{,}961 - 0{,}747) = 0{,}93(0{,}961 - 0{,}747) = 0{,}199 \text{ бит}$$

Прирост, полученный в результате данного разбиения, несколько ниже, чем для множества из табл. 1 (то есть без пропусков), где он составлял 0,246:

$$Split - Info(S_1) = -\left(\frac{5}{13}\right)log_2\left(\frac{5}{13}\right) - \left(\frac{3}{13}\right)log_2\left(\frac{3}{13}\right) - \left(\frac{5}{13}\right)log_2\left(\frac{5}{13}\right) - \left(\frac{1}{13}\right)log_2\left(\frac{1}{13}\right) = 1{,}876$$

Когда наблюдение с известным значением атрибута переходит из множества T в подмножество T_i вероятность его отношения к T_i, равна 1, а ко всем остальным подмножествам – 0. Если в примере имеется пропущенное значение, то относительно данного примера может быть сделано только некоторое вероятностное предположение. При этом очевидно, что вероятность принадлежности примера ко всем подмножествам, полученным в результате разбиения, будет равна 1 (то есть к какому-либо из них он будет отнесен обязательно).

После разбиения множества T на подмножества по атрибуту A_1 запись, содержащая пустое значение, будет распределена во все три полученных подмножества. Данная ситуация проиллюстрирована на рис. 4.

Веса w_i для примеров с пропусками будут равны вероятностям $5/13, 3/13$ и $5/13$ соответственно. Теперь T_i может быть интерпретировано алгоритмом не как число записей в соответствующем подмножестве, а как сумма весов w для данного подмножества.

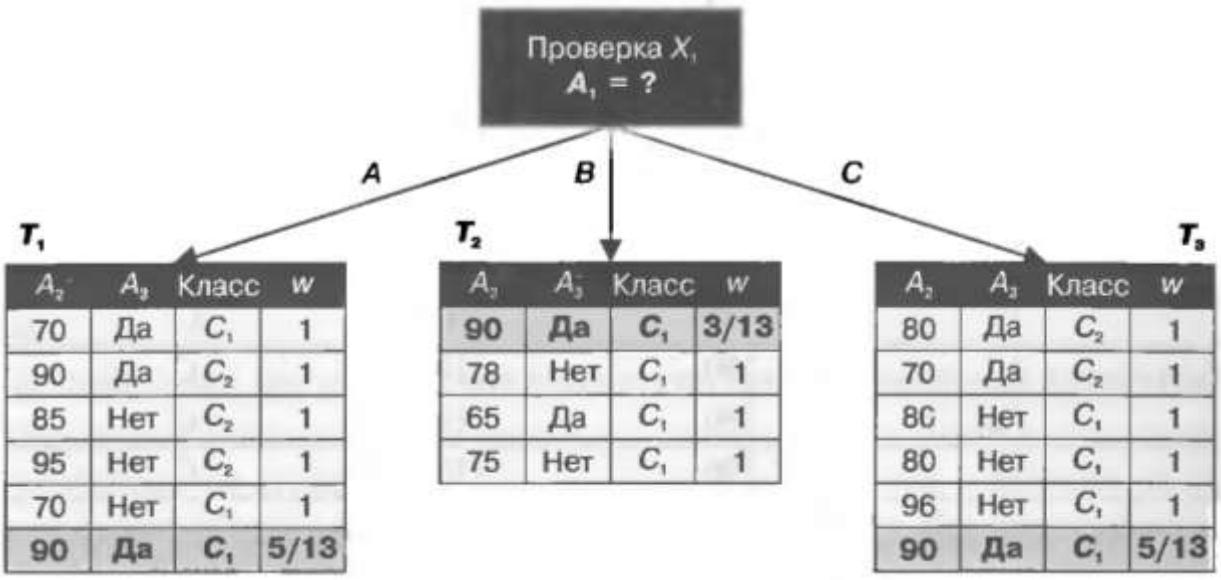

Рис. 4. Пример разбиения множества