Критерии ветвления, синтез бинарных решающих деревьев и алгоритм LISTBB

Донской В. И.
Таврический национальный университет им. В. И. Вернадского
vidonskoy@mail.ru

Donskoy V. I. Splitting criteria, binary decision tree synthesis, and algorithm LISTBB
Abstract

In our days, interest to the class of inductors on the basis of decision trees does not weaken, especially in the context of paradigm of Data Mining. At the same time most widespread Quinlan algorithms ID3 and C4.5, as we show in the paper, are not the best. It is therefore possible to see the successful attempts of creation another heuristic splitting criteria for the algorithms of synthesis of decision trees. Comparative definition of different splitting criteria used for the synthesis of binary decision trees is the purpose of the paper. We include the criteria D, Ω, Zı which were developed by an author yet at 1979-80 years. These criteria define splitting principle in algorithm LISTBB.

Keywords*: Decision Trees, Splitting Criteria, algorithm LISTBB*

Введение

Идеи построения и применения деревьев решений в машинном обучении и распознавании впервые появились в статьях Ханта и Ховленда в конце 50-х годов XX века. Но центральной работой, привлекшей внимание математиков и программистов к этому направлению во всем мире, явилась книга Ханта, Марина и Стоуна [20], увидевшая свет в 1966г. В Советском союзе научное направление, связанное с решающими деревьями, начало развиваться примерно в то же время в научной школе А. Ш. Блоха [1]. Из многочисленных работ этой школы (см. обзор в работе [6]) следует обратить особое внимание на исследование В. А. Орлова [15], который первым, еще в начале 70-х годов прошлого века, – более чем на 10 лет раньше Росса Куинлана – предложил энтропийный критерий ветвления и алгоритм синтеза решающих деревьев, который принципиально не отличался от широко используемого в настоящее время алгоритма *ID3*.

В настоящее время интерес к классу индукторов на основе решающих деревьев не ослабевает, особенно в контексте парадигмы Data Mining. В то же время наиболее распространенные алгоритмы *ID3* и *C*4.5 Росса Куинлана, как можно увидеть ниже, отнюдь не являются лучшими. Поэтому можно обнаружить успешные попытки создания других эвристических алгоритма синтеза решающих деревьев по прецедентной информации [27,28].

Целью настоящей работы является сравнительное описание различных критериев ветвления при синтезе бинарных решающих деревьев (БРД), включая критерии, разработанные автором еще в 1979-80 годах, которые легли в основу алгоритма LISTBB.

Название LISTBB указанного алгоритма синтеза БРД объясняется тем, что его первая реализация на автокоде компьютера М222 была осуществлена на основе спискового представления (LIST); Branching (B) – обозначало ветвление, а Boolean – второе B – обозначало случай булевых переменных.

Алгоритм *LISTBB* и его модификации *LISTD*, *LISTBB*(*P*) многократно применялись при решении практических задач и использовались при создании программных комплексов *РАДИУС*-222, *ТРИОЛЬ, ИНТМАН* [5,9,10]. Главная особенность алгоритма *LISTBB* состоит в том, что он «заточен» именно на минимизацию отыскиваемого БРД индуктора по числу листьев. Такой подход дает результаты в среднем (по числу проведенных испытаний) превышающие результаты использования других подходов к синтезу решающих деревьев.

Синтез бинарных решающих деревьев, вообще говоря, состоит из двух этапов: выбора признаковых предикатов и собственно построения дерева решений. Эти этапы могут быть совмещены, что часто реализуется при синтезе деревьев, соответствующих разбиениям вещественного признакового пространства гиперпараллелепипедами. Далее предполагается, что применяется именно двухэтапный подход, причем все рассмотрение сосредоточено на вопросе синтеза БРД при уже найденном наборе признаковых предикатов и их значений, зафиксированных в логических таблицах обучения.

Каждой внутренней вершине БРД ставится в соответствие некоторый (признаковый) предикат. Из каждой внутренней вершины БРД исходят два ребра, соответствующие нулевому и единичному значению предиката, приписанного этой вершине. Все ветви БРД не содержат одинаковых предикатов в своих внутренних вершинах и заканчиваются листьями, которые помечены номерами классов. К указанным классам алгоритм распознавания, определяемый БРД, относит такие объекты (точки признакового пространства), для которых все предикаты в ветви дерева, заканчивающейся этим листом, обращаются в единицу (выполняются).

Легко проверяется известное свойство БРД: число его внутренних вершин всегда на единицу меньше числа листьев. Поэтому минимизация числа листьев, числа тестов в вершинах, числа внутренних вершин – эквивалентны.

Длиной ветви называют число содержащейся в ней вершин. Высотой БРД называют длину его ветви, содержащей наибольшее число вершин. Дерево называют равномерным (сбалансированным), если все его ветви имеют равную длину.

Будем полагать, что на основе анализа предметной области выбрано n признаковых предикатов для синтеза БРД. Отождествим эти признаковые предикаты с булевыми переменными $x_1,...,x_n$.

Класс булевых функций, представимых БРД, полон: при помощи бинарного дерева можно реализовать алгоритм реализации любой булевой

функции. Это важное свойство легко доказывается путем последовательного разложения Шеннона по одной переменной. Но класс булевых функций, определяемых БРД с числом листьев, не превышающим заданной константы μ, достаточно узок [4].

Разложение по r переменным вдоль любой ветви БРД определяет интервал ранга r в разбиении множества вершин единичного n-мерного куба B^n на совокупность непересекающихся интервалов, помеченных номерами классов, к которым БРД относит эти интервалы. Кодами интервалов являются наборы значений предикатов, размещенных во внутренних вершинах соответствующих ветвей, а их размерность равна 2^{n-r}.

Рассмотрение процесса ветвления как последовательного разбиения B^n на интервалы использует теоретико-множественный подход, развитый в работах Ю. И. Журавлева [14]. Этот подход оказался очень плодотворным и послужил толчком к разработке ряда критериев ветвления на основе понятия отделимости [13]. Синтез БРД с минимальным числом листьев равносилен синтезу кратчайшего ортогонального покрытия, корректного относительно размещения точек из обучающей выборки по интервалам разбиения.

Число листьев μ БРД является естественной мерой его сложности, поскольку число внутренних вершин $\mu-1$ определяет количество однотипных шагов, выполняемых при «наращивании» дерева в процессе синтеза.

Обозначим q – заданное число классов объектов, а $\mathfrak{D}(n,q,\mu)$ – семейство БРД, имеющих ровно μ листьев. Точная формула для числа $d = |\mathfrak{D}(n,q,\mu)|$ неизвестна. Произвольная булева функция представима БРД, вообще говоря, не единственным образом.

В работе [4] получена асимптотическая оценка

$$d(n,q,\mu) \sim (\mu-1)![q(q-1)]^{\mu-1} n(n-1)^{\mu-2} \text{ при } n \to \infty,$$

и доказано, что число $b(n,2,\mu)$ булевых функций (случай $q=2$), представимых БРД с ровно μ листьями, удовлетворяет неравенству

$$b(n,2,\mu) < (\mu-1)!2^{\mu-1} n^{\mu-1}.$$

Для размерности Вапника-Червоненкиса (VCD) [2] конечного класса $\mathfrak{B}(n,2,\mu)$ решающих функций, представимых в виде БРД с числом листьев, не превышающим μ, в случае двух классов, использование метода $pVCD$ [12] позволило получить оценку [7]

$$VCD(\mathfrak{B}(n,2,\mu)) < (\mu-1)(\log(n+1) + \log\mu + 1). \qquad (0.1)$$

1. Подходы к оцениванию качества деревьев решений как эмпирических индукторов

Машинное обучение по прецедентам, о котором идет речь в данной статье, реализует принцип эмпирической индукции, состоящий в том, что

заключение строится путем обобщения наблюдений, перехода от частных свойств наблюдаемых примеров к их общему свойству. В рассматриваемом нами случае найденное в процессе обучения общее свойство имеет вид набора составляющих *условных свойств* (*концептов* – по Ханту). В совокупности эти найденные свойства представляются решающим деревом. Условными свойствами или концептами являются конъюнкции, соответствующие ветвям построенного по обучающей информации дерева, и определяющие набор логических функций. Обучение должно строиться так, чтобы найденное общее свойство было присуще не только как можно большему числу наблюдаемых примеров, но и как можно большему числу допустимых объектов, которые не участвовали в обучении. Только тогда можно говорить о том, что найденное при обучении общее свойство (закономерность, совокупность функций, правило) будет иметь малую вероятность ошибки. Если это действительно так, то говорят, что для применяемого метода обучения имеет место *обучаемость* [8].

Целью данного параграфа является обоснование того, что задача синтеза БРД по обучающей выборке должна ставиться как задача поиска дерева с минимальным числом листьев, которое правильно классифицирует как можно большее число примеров.

Существуют по крайней мерее три подхода к обоснованию этого положения.

- Класс решающих деревьев, применяемый при обучении, тем у́же, чем меньше параметр μ, ограничивающий число листьев. Тогда его емкость (VCD) – меньше, что обеспечивает обучаемость по теории Вапника-Червоненкиса.
- Другие, не связанные непосредственно с VCD, статистические оценки надежности распознавания при помощи обученных классификаторов – деревьев решений тем выше, чем меньшее число листьев они имеют.
- Длина описания классифицирующего дерева тем короче, чем меньше число его листьев, что определяет надежность распознавания на основе принципа *MDL* – минимальной длины описания.

Опишем кратко эти три подхода.

1. Сложность оценивания вероятностей ошибок решающих деревьев связана с тем, что эмпирические частоты событий, найденные путем подсчета числа ошибок на обучающей информации, являются смещенными. Достаточным условием равномерной сходимости эмпирических частот ошибок к соответствующим вероятностям является конечность размерности Вапника-Червоненкиса класса применяемых эмпирических гипотез [2]. Чем меньше VCD класса, тем быстрее скорость равномерной сходимости, и тем меньшую длину обучающей выборки можно использовать для получения заданной точности.

Из оценки (0.1) следует, что *чем меньше число листьев* μ *эмпирического индуктора – БРД, тем меньше VCD класса* $\mathcal{B}(n,2,\mu)$ *решающих функций, представимых в виде БРД.*

2. Точность БРД как эмпирического индуктора можно оценить по контрольной выборке. В этом случае используется следующая вероятностная схема. Точки контрольной выборки извлекаются из генеральной совокупности случайно и независимо. Контрольная выборка не содержит общих примеров с использованной обучающей выборкой. Контрольные точки снабжены точной информацией о принадлежности классам, и их появление не зависит и от того, какое дерево было построено при обучении. Тогда частота ошибок на контрольной выборке будет несмещенной оценкой вероятности ошибки построенного решающего дерева на генеральной совокупности.

Будем рассматривать случай бинарных переменных, полагая, что исходное признаковое пространство при описании каждого объекта n признаковыми предикатами порождает отображение признакового пространства в $B^n = \{0,1\}^n$ и вероятностную меру P на множестве B^n; $\sum_{\tilde{x} \in B^n} P(\tilde{x}) = 1$. Обозначим $P(E)-$ вероятность ошибки любого БРД с μ листьями при распознавании произвольного объекта $\tilde{x} \in B^n$, а $\Pr(U)$ – вероятность выполнения некоторого условия U.

Если классификатор БРД, имеющий μ листьев, на контрольной последовательности длины l_C допустил δl_C ошибок, где $0 \le \delta < 1$, то для любого ε такого, что $1 > \varepsilon > \delta$, имеют место неравенства (см. приложение):

$$\Pr(P(E) \ge \varepsilon) \le \frac{\mu}{4 l_C (\varepsilon - \delta)^2};$$

$$\Pr(P(E) \ge \varepsilon) < \exp\{-\frac{(\varepsilon - \delta)^2 l_c}{\mu}\},$$

из которых следует, что статистическая надежность БРД классификаторов тем выше, чем меньше число листьев.

3. Не теряя общности, рассмотрим случай двух классов. Программирование слова p для декомпрессии любого БРД с μ листьями с целью получения оценки сложности $KS(h_\mu)$ основано на представлении каждой из $\mu - 1$ вершин ветвления словом-атомом, состоящим из двух частей [7]:

Код номера переменной или значение решающей функции (0 или 1)	Номер следующего атома в конкатенации или значение решающей функции (0 или 1)

Префикс атома может иметь $n+1$ значение, поскольку 0 и 1 резервируются для значений классифицирующей функции, а значениями 2,3,…, $n+1$ кодируются номера признаков 1,2,…,n. Окончание атома может иметь μ значений: 0 и 1 резервируются как в префиксе. Остальные $\mu - 2$ значений соответствуют направленным рёбрам дерева, являющимися указателями на

решающие вершины дерева (атомы списка). Указатель на одну (начальную вершину дерева) не требуется: нужны указатели только на $\mu - 2$ внутренних вершин. Всего получается μ значений для окончания атома.

Использование стандартного самоограничивающего кода позволяет получить оценку

$$KP(h_\mu) < 2(]\log\log n[+]\log\log \mu[) + (\mu - 1)(]\log(n+1)[+]\log \mu[),$$

$$KP(h_\mu) \approx 2(\log\log n + \log\log \mu) + (\mu - 1)(\log(n+1) + \log \mu).$$

Очевидно, что чем меньше число листьев, тем короче описание БРД индуктора.

2. Критерии ветвления (***splitting criteria***)

Главным элементом алгоритмов синтеза БРД по заданным бинарным обучающим таблицам является выбор на каждом шаге переменной для ветвления или, что равносильно, для разбиения некоторого интервала N_t. Интервал разбивается на два интервала N_t^1 и N_t^2 таких, что $N_t^1 \cup N_t^2 = N_t$; $N_t^1 \cap N_t^2 = \emptyset$; при условии, что в интервале N_t непременно содержатся точки различных классов.

Обозначим k – номер переменной, выбранной для разбиения интервала N_t. Поскольку именно выбранная переменная определяет разбиение, будем обозначать интервалы $N_t^1(k)$ и $N_t^2(k)$. Определим $A(k) = N_t^1(k) \cap T_{l,n}$ – множество точек из обучающей выборки, попавших в интервал $N_t^1(k)$; $B(k) = N_t^2(k) \cap T_{l,n}$ – множество точек из обучающей выборки, попавших в интервал $N_t^2(k)$. Пусть $|A(k)| = m_1(k)$; $|B(k)| = m_2(k)$.

Будем говорить, что некоторый предикат $S(k)$ является критерием ветвления, если переменная x_k выбирается для ветвления в том и только в том случае, когда этот предикат принимает истинное (единичное) значение. Критерии ветвления могут быть различными.

Рассмотрим следующие критерии ветвления.

Критерий S_2 (*полной отделимости*). $S_2(k) = 1$, если множество $A(k)$ содержит точки только одного класса, множество $B(k)$ содержит точки только одного класса и классы наборов в $A(k)$ и $B(k)$ различны; иначе – $S_2(k) = 0$.

Критерий S_1 [6,11] (*частичной отделимости*). $S_1(k) = 1$, если множество $A(k)$ содержит точки только одного класса или множество $B(k)$ содержит точки только одного класса; иначе – $S_1(k) = 0$. Легко видеть, что событие «$S_2(k) = 1$» влечет событие «$S_1(k) = 1$».

Критерий D [6,11] (*равномерного разделения пар*). Пусть $T_{m_t,n} = T_{l,n} \cap N_t$ – подмножество точек обучающей выборки, попавших в

интервал N_t, а $K_t(k)$ – число пар наборов разных классов в подмножестве $T_{m_t,n}$, которые различаются по переменной x_k. Если $k^* = \arg\max_k K_t(k)$ и для разбиения используется переменная x_{k^*}, то будем говорить, что для ветвления используется критерий D.

Свойства критерия D.

1° Пусть число точек, подлежащих разбиению, зафиксировано. Пусть возможны любые размещения этих точек и их пометок номерами классов в разбиваемом интервале N_t. Для того, чтобы при заданной обучающей выборке и заданном интервале, подлежащем разбиению, величина $D(k^*) = \max_k K_t(k)$ имела максимальное возможное значение, необходимо и достаточно одновременное выполнение двух следующих условий:

(i) Класс любой точки множества $A(k^*)$ отличен от класса любой точки множества $B(k^*)$.

(ii) Равномерность разбиения: $m_1(k^*) = m_2(k^*)$ при четном значении $m_{1,2}$ или $|m_1(k^*) - m_2(k^*)| = 1$ при нечетном $m_{1,2}$, где $m_{1,2} = m_1(k^*) + m_2(k^*)$ – число точек обучающей выборки, попавших в разбиваемый интервал N_t.

Достаточность. Предположим, что величина $D(k)$ может быть увеличена. Следовательно, можно увеличить число пар точек разных классов в интервалах разбиения (при зафиксированной величине $m_{1,2}$). Тогда: либо существуют точки одного и того же класса во множестве $A(k)$ (или в $B(k)$), и тогда такие точки можно переносить в соседний интервал разбиения $N_t^2(k)$ (или $N_t^1(k)$); либо, если условие (i) выполнено, величина $m_1(k)(m_{1,2} - m_1(k))$ не достигает максимума. Но тогда не выполняется условие (ii).

Необходимость. Если в разбиваемом интервале число пар наборов разных классов, которые различаются по переменной x_k, является максимально возможным, то наборов одного и того же класса ни во множестве $A(k)$, ни во множестве $B(k)$ быть не может (i). При этом условие (ii) является необходимым условием экстремума при целочисленных величинах $m_1(k)$ и $m_2(k)$.

2° Критерий D может применяться в случаях любых признаковых пространств и любых разделяющих предикатах.

Критерий DKM (Dietterich, Kearns, Mansour). Этот критерий был предложен в работах [23], и был рассчитан на случай двух классов. Если в двух интервалах разбиения $N_t^1(k)$ и $N_t^2(k)$ соответственно s_{11} точек первого класса и s_{22} точек второго класса, то $DKM(k) = 2\sqrt{\frac{s_{11}s_{22}}{m_{1,2}}} = 2\sqrt{\hat{p}_{11}\hat{p}_{22}}$. Здесь $\hat{p}_{11}$ и $\hat{p}_{22}$ - оценки вероятностей появления точек первого класса в интервале $N_t^1(k)$ и второго класса – в интервале $N_t^2(k)$. В работе показано, что

использование критерия DKM в задачах синтеза БРД предпочтительнее, чем использование энтропийного критерия E и критерия Джини G (см. ниже).

Свойства критерия DKM.

1° $DKM(k)=1$, если в каждом из интервалов разбиения содержатся точки только одного класса.

2° Критерий DKM обладает таким же свойством равномерности, как и критерий D.

3° Критерий D обладает преимуществом перед критерием DKM: может использоваться при числе классов, большем двух.

Критерий TWO (Twoing).

Пусть для случая двух классов, как обозначалось выше, в интервале разбиения $N_t^1(k)$ содержатся s_{11} точек первого и s_{21} точек второго классов, а в интервале $N_t^2(k)$ - s_{12} точек первого и s_{22} точек второго класса; всего в интервале $N_t^1(k)$ содержится $m_1 = s_{11} + s_{21}$ точек из обучающей выборки, а в интервале $N_t^2(k)$ - $m_2 = s_{12} + s_{22}$ точек. Разбиению подлежат $m_{1,2} = m_1 + m_2$ точек. Тогда критерий *Twoing* определяется выражением

$$TWO = \frac{m_1 m_2}{m_{1,2}^2}\left(\left| \frac{s_{11}}{m_1} - \frac{s_{12}}{m_2} \right| + \left| \frac{s_{21}}{m_1} - \frac{s_{22}}{m_2} \right| \right)^2.$$

$$TWO = \hat{p}\hat{q}\left(\left| \hat{p}_{11} - \hat{p}_{12} \right| + \left| \hat{p}_{21} - \hat{p}_{22} \right| \right)^2,$$

где $\hat{p} = \frac{m_1}{m_{1,2}}$, $\hat{q} = \frac{m_2}{m_{1,2}}$, $\hat{p} + \hat{q} = 1$. При безошибочном разделении $s_{12} = s_{21} = 0$ и тогда $TWO = 4\hat{p}\hat{q}$. Если при этом имеет место равномерное распределение точек выборки по интервалам разбиения – т.е. $\hat{p} = \hat{q} = \frac{1}{2}$, то $TWO = 1$.

Свойства критерия TWO в основном совпадают со свойствами критерия DKM.

Критерий Ω. [6,11] Пусть при разбиении по переменной x_k в интервале $N_t^1(k)$ оказались точки $J_1(k)$ разных классов, а в интервале $N_t^2(k)$ – точки $J_2(k)$ разных классов. Обозначим $\Omega(k^*) = \min\limits_k (J_1(k) + J_2(k))$. Будем говорить, что используется критерий Ω, если для разбиения выбирается переменная k^* и при этом классы хотя бы одной пары точек из разных интервалов разбиения $N_t^1(k^*)$ и $N_t^2(k^*)$ различны.

Свойства критерия Ω.

1° Имеет место эквивалентность $(\Omega(k) = 2) \Leftrightarrow (S_2(k) = 1)$.

2° Если значение $\Omega(k)$ равно числу q классов объектов в исходной задаче, то разбиение по переменной x_k приводит к тому, что объекты каждого из классов попадут только в один из двух интервалов разбиения. Назовем это свойство *чувствительностью к иерархическому разделению классов.*

Критерий E (*энтропийный*). Пусть $s_{i,j}$ – количество точек класса i в интервалах разбиения $N_t^j(k)$, $j=1,2$, полученных при разбиении интервала N_t по переменной x_k. В общем случае $m_{1,2}$ точек обучающей выборки распределятся по двум полученным в результате разбиения интервалам так, как показано на рис. 2.1 (где для наглядности полагается, что число классов в выборке равно двум).

$N_t^1(k)$	$N_t^2(k)$
содержит $m_1(k)$ точек;	содержит $m_2(k)$ точек;
из них $s_{1,1}$ точек – класса 1	из них $s_{1,2}$ точек – класса 1
и $s_{2,1}$ точек – класса 2.	и $s_{2,2}$ точек – класса 2.

Рис. 2.1. Разбиение на два интервала при ветвлении

Вероятность того, что произвольный объект из $N_t^j(k)$ принадлежит классу i, может быть оценена как $\hat{p}_{i,j} = s_{i,j} / m_j(k)$, где $m_j(k)$ – число точек выборки, попавших в интервал $N_t^j(k)$. Заметим, что оценка условной вероятности $\hat{p}_{i,j}$ – смещенная.

Оценкой энтропии интервала $N_t^j(k)$ будет $I_j(k) = -\sum_i \hat{p}_{i,j} \log_2 \hat{p}_{i,j}$. А оценкой средней энтропии по двум интервалам $N_t^1(k)$ и $N_t^2(k)$ будет величина $E(k) = \frac{m_1(k)}{m_{1,2}} I_1(k) + \frac{m_2(k)}{m_{1,2}} I_2(k)$, поскольку $\frac{m_j(k)}{m_{1,2}}$ является оценкой вероятностной меры интервала $N_t^j(k)$, и тогда $E(k)$ – среднестатистическая оценка.

Критерий E выбора переменной для разбиения (ветвления) интервала N_t состоит в выборе переменной с номером

$$k^* = \arg\min_k E(k),$$

что соответствует минимизации неопределенности в результате разбиения текущего интервала.

Свойства критерия E.

1° Энтропийный критерий E не чувствителен к равномерности разбиения – может давать одинаковые значения в случаях, когда количество объектов в интервалах равно и когда различается вплоть до 1 и $m_{1,2} - 1$.

Действительно, если в каком либо интервале j содержатся объекты только одного класса i, то оценка вероятности $\hat{p}_{i,j} = s_{i,j} / m_j(k)$ будет равна единице независимо от величины $m_j(k)$. В частности, рассмотрим две таблицы на рис. 2.2

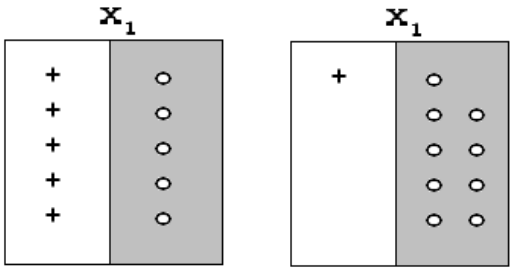

Рис. 2.2. Неравномерное распределение объектов по интервалам.

И в одном, и в другом случае критерий E принимает нулевое значение. Заметим, что критерий D в этих случаях примет различные значения: 25 и 9.

2° Критерий E нечувствителен к иерархическому разделению классов. Это свойство иллюстрируется следующим рис. 2.3.

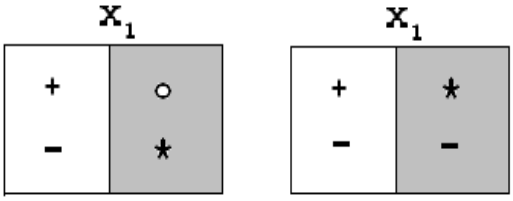

Рис. 2.3. Два случая, когда значения критерия E совпадают и равны 1.

Критерий информационного выигрыша (*Information gain, IGain*) [32] рассчитан на выбор переменной для ветвления на основе энтроприйного подхода. Критерий усовершенствован так, чтобы оценивать средний прирост информации (выигрыш) от выполнения шага ветвления.

Начальное среднее количество информации, необходимое для определения класса произвольного объекта определяется как

$$Info(T) = -\sum_{j=1}^{q} \frac{s_j}{l} \log \frac{s_j}{l} = -\sum_{j=1}^{q} \hat{p}_j \log \hat{p}_j ,$$

где T – обучающая выборка; l – число примеров в обучающей выборке; q – число различных классов (значений целевой переменной); s_j – число точек из обучающей выборки, помеченных классом j; $\hat{p}_j$ – оценка вероятности появления класса j, вычисленная по данной обучающей выборке.

Критерий выбора переменной x_k – *по максимуму информационного выигрыша* $Gain(k) = Info(T) - Info(k) = Info(T) - E(k)$, где $E(k)$ - величина определенного выше критерия E – есть средняя энтропия по интервалам разбиения при выборе для ветвления переменной x_k.

Критерий MEE (***M****inimum* ***E****rror* ***E****ntropy*)[28].

Сначала рассмотрим случай двух классов – ω_1 и ω_2. Пусть x_k - кандидат на переменную ветвления, а ω_1 - номер класса – кандидат для пометки интервала разбиения $N_t^1(k)$ (левой ветви) в случае разбиения по переменной x_k. Тогда правая ветвь (и интервал $N_t^2(k)$) предположительно

помечается оставшимся классом – ω_2. Если считать такое ветвление правильным, то любая точка из обучающей выборки, попадающая в интервал $N_t^1(k)$ и принадлежащая классу ω_2, будет классифицироваться неверно. Обозначим соответственно число таких ошибочных точек в интервалах $N_t^1(k)$ и $N_t^2(k)$ как r_{12} и r_{21}. Тогда оценками вероятностей ошибок типа «перепутывания классов» в разбиваемом интервале $N_t = N_t^1 \cup N_t^2$ будут $\hat{P}_{12} = \frac{r_{12}}{m_{1,2}}$ и $\hat{P}_{21} = \frac{r_{21}}{m_{1,2}}$, где $m_{1,2}$ - число точек выборки, попадающих в интервал N_t. Величина $1 - \hat{P}_{12} - \hat{P}_{21}$ будет оценкой вероятности правильного вычисления классов вершиной с распознавателем x_k и метками ω_1 и ω_2. Числовая оценка для рассматриваемого критерия *MEE* задается формулой

$$EE = EE(N_t, k, \hat{P}_{12}, \hat{P}_{21}) = -\hat{P}_{12}\log\hat{P}_{12} - \hat{P}_{21}\log\hat{P}_{21} - (1-\hat{P}_{12}-\hat{P}_{21})\ln(1-\hat{P}_{12}-\hat{P}_{21})$$

и называется энтропией ошибки. Правило ветвления *MEE* состоит в выборе для разбиения допустимого интервала N_t и допустимой переменной с таким номером k, чтобы достигалось минимальное значение энтропии ошибки

$$\min_{N_t,k} EE(N_t, k, \hat{P}_{12}, \hat{P}_{21}).$$

Свойства критерия MEE.

1° Минимальное значение оценки $EE = 0$ имеет место в случае правильной классификации вершиной всех точек выборки, попавших в интервал разбиения. Максимальное – $EE = 1$ имеет место при «полном перепутывании» точек в интервалах разбиения, когда $\hat{P}_{12} = \hat{P}_{21} = \frac{1}{2}$.

2° *С ростом «перепутывания» классов оценка EE возрастает. Заметим, что в этом случае и значение критерия* Ω *возрастает.*

3° В случае частичной отделимости, например, при $\hat{P}_{12} = 0$, если при этом $\hat{P}_{21} = \frac{1}{2}$, вычисления также дают $EE = 1$. Поэтому критерий MEE в иногда может не различать случаи частичной и полной разделимости классов.

Критерий G (*основанный на индексе Джини*). Индекс Джини интервала $N_t^j(k)$ равен $g(N_t^j(k)) = 1 - \sum_i \hat{p}_{i,j}^2 = 1 - \sum_i (s_{i,j}/m_j(k))^2$. Суммируются квадраты оценок условных вероятностей всех классов в данном интервале. Если в интервале содержатся точки только одного класса, то его индекс достигает минимального значения, равного нулю. Критерий G для ветвления определяется по формуле

$$G(k) = g(N_t^1(k)) + g(N_t^2(k)).$$

Выбор переменной осуществляется по правилу $k^* = \arg\min_k G(k)$.

Свойства критерия G.

1° Если в интервале содержатся точки только одного класса, то его индекс достигает минимального значения, равного нулю. Поэтому критерий G определяет частичную отделимость.

2° $(G(k)=0)\Leftrightarrow(S_2(k)=1)$, что означает способность критерия G определять полную отделимость.

В работах [37, p.7] показано, что применение критерия Джини может привести к неразличению иерархической отделимости классов, и приведен пример (рис. 2.4). На рис. 2.4 представлены два случая разбиений. Случай A соответствует полной отделимости двух пар классов. Но по критерию Джини более предпочтительным оказывается разбиение B.

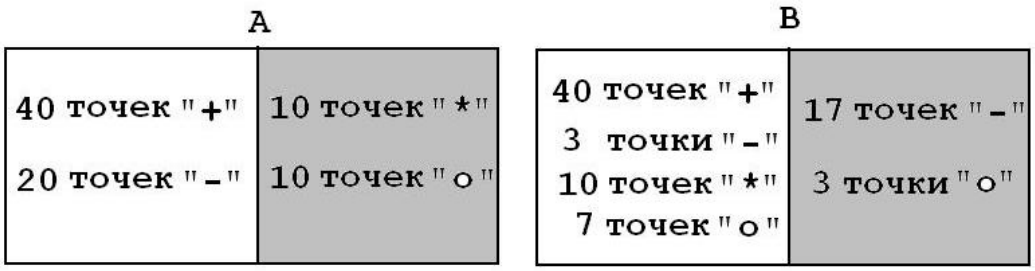

Рис. 2.4. Два случая расположения точек

Сравним результаты использования различных критериев.

Пример. Дан интервал размерности 5, в котором содержатся 9 точек трех различных классов, обозначенных метками $+, -, *$ (рис. 2.5).

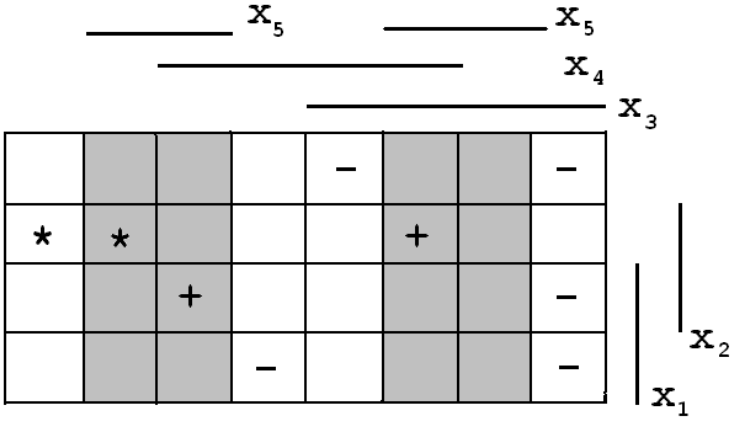

Рис. 2.5. Точки выборки в интервале размерности 5

Значения критериев ветвления при выборе переменных $x_1,\ldots,x_5$ представлены на рис. 2.6.

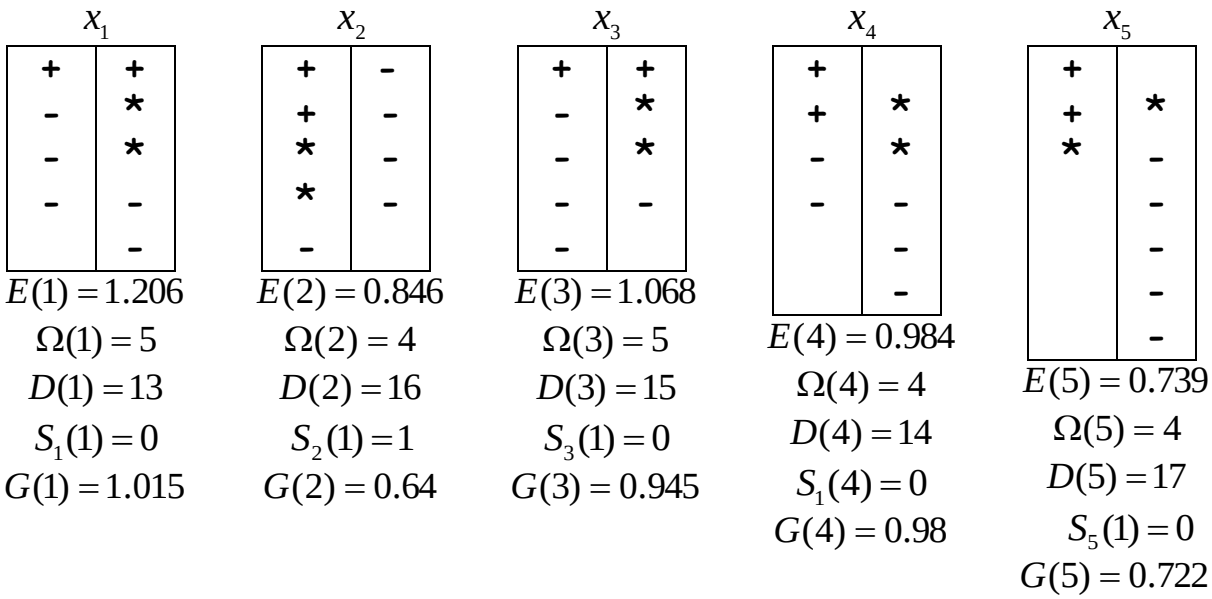

Рис. 2.6. Значения критериев в разных случаях распределения точек в интервале

Сравнение значений критериев показывает, что они, за исключением критериев S_1 и G, согласованы: определяют один и тот же выбор переменной – x_5. Критерии S_1 и G, в свою очередь, согласованы друг с другом и выделяют случай частичной отделимости. Если упорядочить переменные по убыванию значения критерия E, то значения критерия D, как видно из таблицы 2.1 и рис. 2.7, будут возрастать, но монотонность роста нарушается: для переменной x_3 увеличенное значение $D(3)=15$ объясняется большей «чувствительностью» критерия D к частичной отделимости по сравнению с критерием E.

Таблица 2.1

Критерии	x_1	x_3	x_4	x_2	x_5
E	1.206	1.068	0.984	0.846	0.739
D	13	15	14	16	17

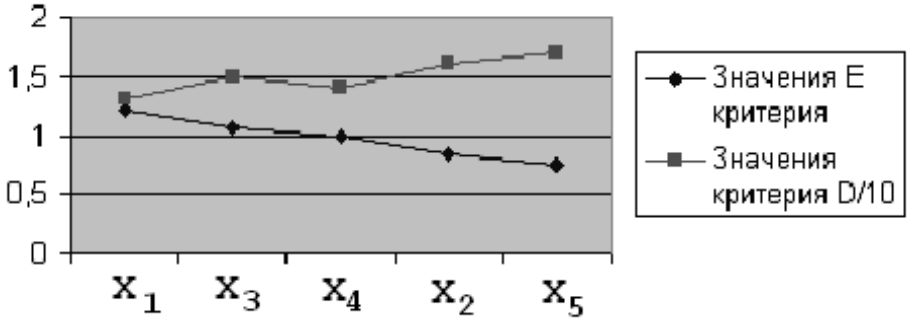

Рис.2.7.

Пример. Дан интервал размерности 4, в котором содержатся 10 точек пяти различных классов (рис. 2.8).

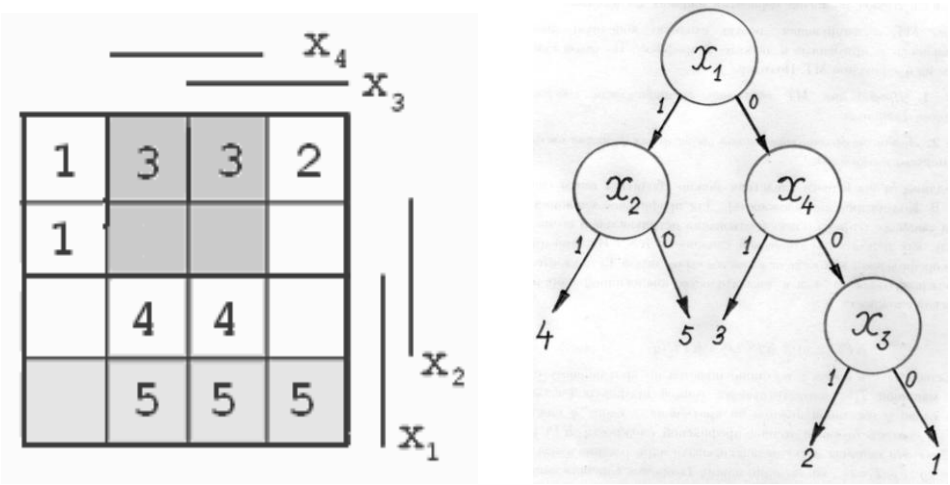

Рис. 2.8. Распределение точек и оптимальное дерево для примера

В этом примере на всех шагах синтеза значения критериям E и D совпадают. Приведем их значения только для первого шага разбиения (табл. 2.2).

Таблица 2.2

Номер переменной / *Значение критерия*	x_1	x_2	x_3	x_4
D	25	20	21	22
E	1.246	1.565	1.922	1.551

Легко видеть, что в случае, когда в каждом интервале разбиения будут содержаться точки только одного класса, критерий E будет давать нулевое значение. Вследствие утверждения, в этом случае выбор по критериям E и D всегда будет совпадать.

Согласно многократным экспериментам по применению различных критериев ветвления, в работе [28] представлены сравнительные результаты. В частности, сравнивалось число листьев в полученных в результате синтеза решающих деревьях. Оценивание производилось на 36 реальных задачах. В таблице 2.3 приведены данные: сколько раз использование каждого из пяти сравниваемых критериев приводило к получению деревьев с наименьшим по сравнению со всеми другими алгоритмами листьев (лучшие результаты) и наибольшим числом листьев (худшие результаты).

Таблица 2.3

Алгоритмы	*Gini*	*Info Gain*	*Twoing*	*C4.5*	*MEE*
Число выигрышей	11	9	8	1	18
Число проигрышей	4	3	3	24	7

Данные, приведенные в таблице 2.3, подтверждают, прежде всего, что *нельзя указать критерий ветвления, который дает лучшие результаты во всех случаях – при любых допустимых входных данных*. Но, тем не менее, согласно таблице 2.3, алгоритм *MEE* побеждает как минимум вдвое чаще

других. Несколько неожиданным представляется то, что по результатам рассматриваемых экспериментов алгоритм *C4.5*, который очень часто используют в приложениях, оказался худшим.

В работе [6] проводились экспериментальные исследования алгоритмов синтеза БРД. В статистических экспериментах точки – вершины единичного n-мерного куба $(n = 25)$ генерировались равновероятно; также равновероятно каждой сгенерированной точке присваивался номер одного из заданного числа классов (таблица 2.4).

Таблица 2.4. Статистические испытания трех алгоритмов ветвления

Алгоритмы	*Среднее по 15 экспериментам число листьев*		
	25 признаков 5 классов 50 объектов в выборке	*25 признаков 2 класса 50 объектов в выборке*	*25 признаков 5 классов 100 объектов в выборке*
LISTBB	23,1	13,3	44,7
LISTD	24,5	14,1	46,7
LISTB	44,9	34,9	—

Алгоритм *LISTBB*, показавший лучшие в этом эксперименте результаты (см. ниже), является гибридной процедурой ситуативного выбора критерия ветвления, зависящего от начального значения критерия Ω и наличия полной или частичной отделимости. Алгоритм *LISTD* использует только критерий *D*; алгоритм *LISTB* реализует произвольный порядок выбора признаков для ветвления, полученный в результате случайной генерации.

Алгоритм *LISTBB* в первую очередь вычисляет значение критерия Ω, который логически наиболее близок к критерию *MEE*.

3. Правила остановки при обучении и подрезание решающих деревьев

Решающее дерево называют *корректным* (относительно данной обучающей выборки), если все примеры этой выборки классифицируется деревом правильно. Разбиение пространства признаков, порождаемое корректным решающим деревом таково, что каждое *терминальное* множество, входящее в полученное разбиение, содержит точки, принадлежащие только одному классу. Терминальные множества соответствуют листьям дерева. Каждое из них наследует номер класса, которым помечен соответствующий лист.

***Правило* 1**. Процесс синтеза решающего дерева (ветвление) продолжается до тех пор, пока оно не станет корректным. Это возможно только в том случае, когда предикатные описания всех пар объектов обучающей выборки, принадлежащих различным классом, не совпадают.

***Правило* 2**. Процесс синтеза прекращается, когда число листьев достигает заданной пороговой величины.

***Правило* 3**. Процесс синтеза прекращается, когда информационный выигрыш (*Information gain*) невозможно увеличить за счет замены ни одного листа новой внутренней вершиной.

***Правило* 4**. Процесс синтеза прекращается, когда длины всех ветвей достигли заданной величины.

***Правило* 5**. Процесс синтеза прекращается, когда терминальные множества, подлежащие ветвлению, содержат число точек, меньшее заданного порогового значения.

***Правило* 6**. Момент остановки при синтезе дерева определяется на основе принципа минимальной длины описания (*Minimum Description Length*), согласованного с выбором наиболее вероятных гипотез по правилу Байеса. Этот подход соответствует парадигме *Ideal MDL* [38]. Он является одной из формализаций «бритвы Оккама»: *наилучшей гипотезой является та, которая минимизирует сумму длины описания кода гипотезы* (*называемой моделью*) *и длины описания множества данных относительно этой гипотезы*. В рассматриваемом случае кодом модели является бинарное описание решающего дерева (в виде некоторой строки), а описанием данных – бинарное строковое описание некоторой совокупности обучающих примеров. Это правило для случая БРД подробно описано в [7].

Правило* 7**. Момент остановки процесса синтеза решающего дерева определяется на основе оценки *VCD* класса деревьев не более чем с μ листьями по правилу «Плюс пять***»[12].

***Правило* 8**. Остановка на основе теоретической оценки вероятности ошибки происходит тогда, когда добавление любой дополнительной вершины к строящемуся дереву уже не приводит к уменьшению ошибки. Такой подход описан во многих работах, в частности, в [7].

Последние два правила остановки являются теоретически наиболее обоснованными.

Любое из перечисленных правил может быть применено с некоторым одним или совокупностью критериев ветвления и дать «новый» алгоритм машинного обучения, основанный на построении дерева решений. Что и наблюдается в многочисленных публикациях, посвященных синтезу эмпирических индукторов рассматриваемого класса.

Правила подрезания (***редуцирования***) определяют максимально возможную длину ветвей дерева. Если какая-нибудь ветвь имеет длину, больше заданного ограничения, то она укорачивается, и вместо последней вершины ветвления в редуцированной ветви ставится метка класса. Эта метка чаще всего определяется тем, точек какого класса содержится больше в интервале, соответствующем редуцированной ветви.

Редуцирование приходится применять, когда попытка синтезировать корректное решающее дерево приводит к его неоправданной сложности.

4. Алгоритмы синтеза деревьев решений по прецедентной информации

Алгоритм CLS (***C****oncept* ***L****earning* ***S****ystem*). Это – классический алгоритм Ханта [20], который явился основой для подавляющего большинства разработок в области синтеза решающих деревьев в процессе машинного обучения. Алгоритм *CLS* циклически разбивает точки обучающей выборки на подмножества в соответствии со значениями переменных, имеющих наибольшую разделяющую способность. Разбиение заканчивается, когда в подмножестве оказываются объекты лишь одного класса. В ходе процесса разбиений формируется дерево решений.

Алгоритм ID3 [32] был предложен Россом Куинланом в 1986 г. и основывался на алгоритме Ханта, учеником которого был Куинлан. Алгоритм ID3 был основан на использовании критерия информационного выигрыша для выбора вершины и переменной для ветвления. Синтез решающего дерева завершался либо в случае достижения его корректности относительно выборки, либо когда ветвление ни в одной некорректной вершине не приводило к увеличению информационного выигрыша.

Алгоритм C4.5 [31]. Этот алгоритм явился развитием идей, реализованных в ID3, был разработан Р. Куинланом в 1993 г. и использовал отношение выигрыша (gain ratio) как критерий ветвления. Процесс синтеза (добавления вершин) в алгоритме C4.5 прекращался, когда число точек для разбиения становилось меньше некоторого порога.

Алгоритм CART [16]. Аббревиатура CART взята из названия «***C****lassification* ***A****nd* ***R****egression* ***T****rees*». Алгоритм предназначен для синтеза бинарных решающих деревьев. Для ветвления используется критерий *Twoing*. CART рассчитан, кроме прочего, на построение деревьев регрессии, в корневых вершинах которых вместо меток классов помещаются вещественные числа. В этих случаях ветвление осуществляется по минимуму среднеквадратической ошибки.

Алгоритм CHAID [22] (***CH****isquare–****A****utomatic–****I****nteraction–****D****etection* – *интерактивное обнаружение на основе критерия* χ^2). Применение методов прикладной статистики для реализации ветвления при синтезе решающих деревьев получило развитие в начале 70-х годов. *CHAID* являлся «развитием» алгоритма *AID* (***A****utomatic* ***I****ntegration* ***D****etection*) [35] и был ориентирован на выбор групп значений переменных для ветвления следующим образом. Для каждой переменной находились такие пары ее значений, которые незначительно изменялись при изменении целевого признака во входных данных. В зависимости от типов переменных-признаков незначительность такого изменения оценивалась разными статистическими критериями: Пирсона (χ^2) – для номинальных переменных, Фишера – для непрерывных переменных, критерием правдоподобия – для ранговых переменных. Статистически значимо неразличимы пары значений переменных объединялись в однородную группу значений, и процесс

повторялся, пока находились «неразличимые» пары. Для ветвления (построения текущей вершины) интерактивно выбиралась такая переменная, которая разделяла группы однородных значений. Синтез дерева прекращался при выполнении любого из следующих условий:

1. Достижение максимальной заданной глубины дерева;

2. Число точек выборки для дальнейшего разбиения в терминальных вершинах или в любой получаемой дочерней вершине меньше заданного значения.

При этом пропущенные значения переменных (если таковые имелись в начальной информации) выделялись в отдельные группы значений.

Алгоритм QUEST (*Quick Unbiased Efficient Statistical Tree*) [25]. Для осуществления ветвления связь между каждой входной переменной и целевой переменной оценивалась на основе F-критерия *ANOVA* (*Analysis Of Variances*) или теста Левене [24] однородности дисперсий для порядковых или непрерывных переменных или критерия χ^2 для номинальных переменных. Для многоклассовых целевых переменных применялся кластерный анализ для объединения в два «сверхкласса». Для ветвления использовалась переменная, имеющая наибольшую оценку статистической связи с целевым признаком. Для подрезания деревьев использовался скользящий контроль, применение которого давало основание авторам говорить о несмещенности статистических оценок. Здесь отмечена только часть особенностей алгоритма *QUEST*, касающихся выбора переменных для ветвления. *QUEST* можно классифицировать как сложную систему анализа данных, дающую возможность исследовать различные варианты предикторов и применять оптимизационные процедуры для их выбора.

Алгоритм SLIQ (***S****upervised* ***L****earning* ***I****n* ***Q****UEST*) [29]. Этот алгоритм рассчитан на применение в области Data Mining и работу с большими объемами исходных данных. Для ветвления используется индекс Джини и специальные методы быстрой сортировки.

Алгоритм PUBLIC (***Pru****ning and* ***Buil****ding* ***I****ntegrate* ***C****lassifier*) [33]. Выбор порогового значения переменной для ветвления осуществляется на основе построения гистограмм распределения классов. Каждая точка на гистограмме, рассматриваемая как кандидат для определения порога ветвления, оценивается энтропийным критерием, который используется для окончательного выбора переменной и порога.

Алгоритмы – CAL5 [30]***, FACT*** (ранняя версия алгоритма *QUEST*)***, LMDT*** [17]***, T1*** [21]***, MARS*** [18] и многие другие – принципиально не отличаются от рассмотренных выше.

Ниже приведена таблица 4.1, в которой приведены данные об использовании алгоритмов синтеза деревьев решений в медицинских задачах.

Таблица 4.1. Частота использования алгоритмов в медицинских приложениях [36]

Алгоритм	*Частота использования* (%)
ID3	68
C4.5	54.55
CART	40.9
SLIQ	27.27
Public	13.6
CLS	9

5. Гибридный алгоритм LISTBB, основанный на использовании совокупности критериев ветвления

Алгоритм LISTBB выбора переменной для ветвления (разбиения интервала)

1^o Вычислить множество номеров переменных, для которых достигается минимум критерия Ω:

$$\tilde{\kappa}_\Omega = \{k_0 : k_0 = \arg\min_k \Omega(k)\},$$

где k пробегает номера свободных переменных разбиваемого интервала.

2^o Если $|\tilde{\kappa}_\Omega| = 1$, т.е. минимум критерия Ω достигается только для одной переменной, выбрать эту переменную k_0 и завершить алгоритм выбора.

3^o Если $\min_k \Omega(k) = q$, где q – число классов, то выбрать для разбиения любую переменную k^* такую, что $k^* = \arg\max_{k \in \tilde{\kappa}_\Omega} D(k)$, и завершить алгоритм выбора.

4^o Если частичная отделимость не имеет места, т.е. $\forall k \in \tilde{\kappa}_\Omega (S_1(k) = 0)$, то выбрать для разбиения любую переменную k^* такую, что $k^* = \arg\max_{k \in \tilde{\kappa}_\Omega} D(k)$, и завершить алгоритм выбора.

5^o Если частичная отделимость имеет место, то выбрать для разбиения любую переменную k^* по максимуму частичной отделимости: такую, что $k^* = \arg\max_{k \in \tilde{\kappa}_\Omega} Z_1(k)$, и завершить алгоритм выбора. □

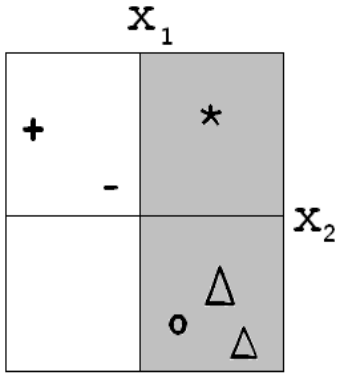

Рис. 2.9

Для пояснения пункта 3^o алгоритма *LISTBB* можно привести следующий пример (рис. 2.9). Шесть точек в разбиваемой области принадлежат пяти различным классам, обозначенным символами **+, –, *, o, Δ.** Разбиение по переменным, условно обозначенным как x_1 и x_2, дает $\Omega(1)=\Omega(2)=5$; $D(1)=8$, но $D(2)=9$. Из этого примера следует, что при равных значениях критерия Ω для двух разных переменных, значение критерия D для этих переменных в то же время может отличаться.

В процессе построения БРД выполняются шаги ветвления, и поэтому число листьев синтезируемого дерева растет. При этом существует нижняя оценка числа листьев БРД, которое получится в итоге процедуры синтеза. В зависимости от выбора стратегий ветвления и по мере приближения к завершению синтеза эта нижняя оценка может изменяться. Поэтому будем назвать ее *текущей*.

Утверждение 5.1. Текущей оценкой снизу для числа листьев синтезируемого корректного БРД является величина $\mu_t+\Omega(k^*)-1$, где μ_t - текущее число листьев БРД до выполнения шага ветвления очередного интервала, $\Omega(k^*)$ – минимальное значение критерия Ω, достигаемое при выборе для ветвления переменной x_k.

Доказательство. Действительно, на шаге t построена некоторая часть дерева, концевые вершины которого (листья) могут содержать объекты различных классов и соответствовать некоторым интервалам. Пусть построенная часть дерева имеет μ_t листьев. Интервал N_t, соответствующий одному такому листу, разбивается на два интервала, поэтому к μ_t-1 оставшимся листьям будут добавлено не менее $\Omega(k^*)$ листьев, поскольку все точки различных классов в интервалах разбиения $N_t^1(k^*)$ и $N_t^2(k^*)$ для достижения корректности БРД должны быть разделены.

Замечание. Поскольку $q\le\Omega(k)\le 2q$, где q – изначальное число классов в обучающей выборке, то при малых величинах q, равных двум или трем, полезность оценки, полученной в утверждении, небольшая. Но с увеличением значения q эта оценка может действительно стать полезной.

Утверждение 5.2. При выборе в алгоритме *LISTBB* переменной для ветвления согласно шагу 5^o имеет место оценка

$$\min_k \Omega(k)-1\le\Delta\mu_t\le m_{1,2}-Z_1(k^*)+1,$$

где $\Delta\mu_t$ - приращение числа листьев БРД после выполнения ветвления по переменной x_{k^*}.

Доказательство. Левая часть неравенства доказана в предыдущем утверждении, а правая часть неравенства становится очевидной, если

заметить, что разделению подлежат $m_{1,2}$ точек разбиваемого интервала, и в худшем случае пришлось бы отделять каждую точку отдельным листом дерева. Заметим, что на шаге 5^o для ветвления выбирается переменная с номером $k^* \in \tilde{\kappa}_\Omega = \{k_0 : k_0 = \arg\min_k \Omega(k)\}$. Но при частичной отделимости $Z_1(k^*)$ точек появится один интервал, не подлежащий дальнейшему дроблению, и к синтезируемому дереву добавится один соответствующий лист. А второй интервал разбиения будет содержать $m_{1,2} - Z_1(k^*)$ точек, которые в худшем случае в дальнейшем будут разделены интервалами по одной точке в каждом. □

Согласно утверждениям, *алгоритм LISTBB*, являясь эвристическим, *направлен на выбор переменной для ветвления так, чтобы минимизировать и нижнюю, и верхнюю оценку приращения число листьев*. Но его «пристрастие» к частичной отделимости может приводить к случаям, когда $Z_1(k^*)$ слишком мало, например, $Z_1(k^*)$=1, и тогда выигрыш от выбора переменной для ветвления по частичной отделимости может оказаться невыгодным.

Параметрический вариант алгоритма $LISTBB(p)$ содержит параметр p, который определяет ветвление в пункте 5^o следующим образом:

5^o Если частичная отделимость имеет место и $Z_1(k^*) > p$, то выбрать для разбиения любую переменную k^* по максимуму частичной отделимости: такую, что $k^* = \arg\max_{k \in \tilde{\kappa}_\Omega} Z_1(k)$, и завершить алгоритм выбора; иначе – выбрать любую переменную k^* такую, что $k^* = \arg\max_{k \in \tilde{\kappa}_\Omega} D(k)$, и завершить алгоритм выбора.

Завершение синтеза БРД (ветвления) при выполнении алгоритма *LISTBB* происходит либо когда получено безошибочное разбиение, либо когда число листьев превышает заданный порог. Это порог сначала был эвристическим параметром, а в настоящее время определяется на основе принципа идеального *MDL* – по оптимальному значению суммарной длины описания дерева и ошибочно классифицируемых им точек [7].

6. Приложение

Будем рассматривать случай бинарных переменных, полагая, что исходное признаковое пространство при описании каждого объекта n признаковыми предикатами порождает отображение признакового пространства в $B^n = \{0,1\}^n$ и вероятностную меру P на множестве B^n; $\sum_{\tilde{x} \in B^n} P(\tilde{x}) = 1$. Обозначим $P(E)$ – вероятность ошибки любого БРД с μ листьями при распознавании произвольного объекта $\tilde{x} \in B^n$, а $\Pr(U)$ – вероятность выполнения некоторого условия U.

Теорема 6.1. Если классификатор БРД, имеющий μ листьев, на контрольной последовательности длины l_C допустил δl_C ошибок, где $0 \le \delta < 1$, то для любого ε такого, что $1 > \varepsilon > \delta$, имеет место неравенство

$$\Pr(P(E) \ge \varepsilon) \le \frac{\mu}{4l_C(\varepsilon - \delta)^2}.$$

Доказательство. Обозначим пометки листьев БРД $\omega_1,...,\omega_s,...,\omega_\mu$ так, чтобы пометка ω_s определяла класс точек, попавших в интервал разбиения N_s, который соответствует s-той ветви дерева, заканчивающейся листом ω_s. Вероятностную меру интервала N_s обозначим $P(N_s) = \Pr\{\tilde{x} \in N_s\}$. Для упрощения записи будем обозначать N_s и интервал, и событие «$\tilde{x} \in N_s$», а ω_s – и номер класса, и событие, заключающееся в появлении точки именно этого класса. Интервалы $N_1,...,N_s,..,N_\mu$ образуют разбиение множества B^n, поэтому

$$\sum_{s=1}^{\mu} P(N_s) = 1; \quad P(E) = \sum_{s=1}^{\mu} P(E/N_s)P(N_s);$$

$$P(E/N_s) = 1 - P(\omega_s/N_s); \quad P(\omega_s, N_s) = P(\omega_s/N_s)P(N_s);$$

$$P(E) = \sum_{s=1}^{\mu}(1 - P(\omega_s/N_s)P(N_s) = \sum_{s=1}^{\mu}(P(N_s) - \sum_{s=1}^{\mu} P(\omega_s, N_s) = 1 - \sum_{s=1}^{\mu} P(\omega_s, N_s).$$

Для каждого интервала разбиения частоты

$$\nu(\omega_s, N_s) = \frac{n(\omega_s, N_s)}{l_C}$$

определяются числами $n(\omega_s, N_s)$ точек из контрольной выборки, попавших в интервал N_s и отнесенных к классу ω_s. Эти точки классифицируется деревом правильно. Обозначим число точек контрольной выборки, попавших в интервал N_s и классифицируемых неправильно, как k_s. Тогда

$$\sum_{s=1}^{\mu}(n(\omega_s, N_s) + k_s) = l_C; \quad \sum_{s=1}^{\mu}\frac{n(\omega_s, N_s)}{l_C} + \sum_{s=1}^{\mu}\frac{k_s}{l_C} = 1;$$

$$\sum_{s=1}^{\mu}\nu(\omega_s, N_s) + \delta = 1, \tag{6.1}$$

где $\delta = \frac{1}{l_C}\sum_{s=1}^{\mu} k_s$ – доля ошибок на контрольной выборке.

Подставим левую часть равенства (6.1) вместо единицы в формулу, определяющую ошибку БРД:

$$P(E) = 1 - \sum_{s=1}^{\mu} P(\omega_s, N_s) = \sum_{s=1}^{\mu}\nu(\omega_s, N_s) - \sum_{s=1}^{\mu} P(\omega_s, N_s) + \delta.$$

Событие «$P(E) \ge \varepsilon$» равносильно событию

$$\sum_{s=1}^{\mu}\nu(\omega_s, N_s) - \sum_{s=1}^{\mu}P(\omega_s, N_s) \geq \varepsilon - \delta .$$

Найдем математическое ожидание и дисперсию случайной величины $\xi = \sum_{s=1}^{\mu}\nu(\omega_s, N_s)$ – суммы независимых случайных величин.

$$\mathrm{M}[\xi] = \sum_{s=1}^{\mu}\mathrm{M}[\nu(\omega_s, N_s)] = \sum_{s=1}^{\mu}P(\omega_s, N_s);$$

$$\mathrm{D}[\xi] = \sum_{s=1}^{\mu}\mathrm{D}[\nu(\omega_s, N_s)] = \sum_{s=1}^{\mu}\mathrm{M}[(\frac{n(\omega_s, N_s)}{l_C} - P(\omega_s, N_s))^2] =$$

$$= l_C^{-2}\sum_{s=1}^{\mu}\mathrm{M}[(n(\omega_s, N_s) - l_C P(\omega_s, N_s))^2].$$

Здесь $l_C P(\omega_s, N_s)$ – математическое ожидание случайной величины «ω_s, N_s», а $\mathrm{M}[(n(\omega_s, N_s) - l_C P(\omega_s, N_s))^2]$ – дисперсия этой случайной величины, равная $l_C P(\omega_s, N_s)(1 - P(\omega_s, N_s)) \leq \frac{l_C}{4}$. Отсюда получаем неравенство $\mathrm{D}[\xi] \leq \frac{\mu}{4l_s}$.

Используя неравенство Чебышёва

$$(\forall \epsilon > 0)\ \Pr(|\varsigma - \mathrm{M}[\varsigma]| \geq \epsilon) \leq \mathrm{D}[\varsigma]/\epsilon^2,$$

Получаем

$$\Pr(P(E) \geq \varepsilon) \leq \frac{\mu}{4l_C(\varepsilon - \delta)^2}.$$

Следствие. Чем меньше число листьев решающего дерева, тем выше его статистическая надежность.

Применение вместо неравенства Чебышёва неравенства Бернштейна дает оценку

$$\Pr(P(E) \geq \varepsilon) < \exp\{-\frac{(\varepsilon - \delta)^2 l_c}{\mu}\}.$$

Список литературы

1. Блох А. Ш. Об одном алгоритме обучения для задач по распознаванию образов / А. Ш. Блох // Вычислительная техника в машиностроении. – Минск: 1966. - №10. – С. 37 – 43.
2. Вапник В. Н., Червоненкис А. Я. Теория распознавания образов / В. Н. Вапник, А. Я. Червоненкис // Теория распознавания образов. – М.: Наука, 1974. – 416 с.
3. Воронцов К. В. Логические алгоритмы классификации (курс лекций «Машинное обучение») / К. В. Воронцов. – М.: 2012. – 53 с. http://www.machinelearning.ru/wiki/images/9/97/Voron-ML-Logic-slides.pdf

4. Донской В.И. Асимптотика числа бинарных решающих деревьев / В. И. Донской // Ученые записки Таврического национального ун-та им. В. И. Вернадского, серия «Информатика и кибернетика». – 2001. – №1. – С.36–38.
5. Донской В. И. Интеллектуализированная программная система IntMan поддержки принятия решений в задачах планирования и управления / В.И.Донской, В. Ф. Блыщик, А. А. Минин, Г. А. Махина // Искусственный интеллект. – 2002. – №2. – С.406–415.
6. Донской В.И. Исследование алгоритмов распознавания, основанных на построении решающих деревьев: автореф. дисс. на соиск. уч. степени канд. физ.-мат. наук: спец. 01.01.09 «Математическая кибернетика» / В.И.Донской. – М. , 1982. – 16 с.
7. Донской В. И. Колмогоровская сложность и ее применение в машинном обучении / В. И. Донской // Таврический вестник информатики и математики. – 2012. – №2. – С. 4 – 35.
8. Донской В. И. Машинное обучение и обучаемость: сравнительный отбор / В.И.Донской // Intellectual Archive. – 2012. – №.933. – 19 с. http://www.sciteclibrary.ru/texsts/rus/stat/st4820.pdf
9. Донской В.И. О построении программного обеспечения распознающих систем / В. И. Донской // Программирование. – 1980. – № 2. – С. 87 – 90.
10. Донской В.И. О совместном использовании абдукции, аналогии, дедукции и индукции при синтезе решений / В.И. Донской // Искусственный интеллект. – №2. – 2000. – С. 59 – 66.
11. Донской В. И., Страхов С. Б. Выбор признаков при синтезе решающих деревьев. – Симферополь: Симферопольский ун-т, 1982. – 12 с. (Рукопись деп. в ВИНИТИ, № 1765-82).
12. Донской В. И. Сложность семейств алгоритмов обучения и оценивание неслучайности извлечения эмпирических закономерностей / В. И. Донской // Кибернетика и системный анализ. – 2012. – №2. – С. 86 – 96.
13. Журавлев Ю. И. Об отделимости подмножеств вершин n-мерного куба / Юрий Иванович Журавлев // Науч. Труды Матеем. ин-та им. В. А. Стеклова. – 1958. – Т.1. – С. 143 – 157.
14. Журавлев Ю. И. Теоретико-множественные методы в алгебре логики / Юрий Иванович Журавлев // Проблемы кибернетики. – 1962. – Вып.2. – С. 5 – 44.
15. Орлов В.А. Применение граф-схемного метода распознавания образов: автореф. дисс. на соиск. уч. степени канд. техн. наук: спец.. 05.13.01 «Техническая кибернетика и теория информации» / В.А.Орлов. – Владивосток, 1974. – 23 с.
16. Breiman L. Classification and regression trees / L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone.- Calif.: Wadsworth: 1984. – 58 p.
17. Brodley C. E., Utgoff. P. E. Multivariate decision trees / C. E. Brodley, P. E. Utgoff // Machine Learning. – 1995. – Vol. 19. – P. 45 –77.
18. Friedman J. H. Multivariate Adaptive Regression Splines / J. H. Friedman // The Annual of Statistics. – 1991. – Vol. 19. – P. 1 –141.
19. Hyafil L, Rivest R. L. Constructing Optimal Binary Decision Trees is NP-Complete / L. Hyafil, R.L. Rivest // Information Proc. Letters. – 1976. – Vol. 3. – №1. – P. 15 –17.
20. Hunt E. B. Experiments in Induction / Earl B. Hunt, Janet Marin, Philip J. Stone. – N. Y.: Academic Press, 1966. – 247 p.
21. Holte R. C. Very simple classification rules perform well on most commonly used datasets / R. C. Holte // Machine Learning. – 1993. – Vol.11. – P.63-90.
22. Kass G. V. An exploratory technique for investigating large quantities of categorical data / G. V. Kaas // Applied Statistics. – 1980. – Vol.29(2). – P.119-127.
23. Kearns M., Mansour Y. On the boosting ability of top-down decision tree learning algorithms / M. Kearns, Y. Mansour // Journal of Computer and Systems Sciences. – 1999. – Vol. 58(1). – P.109 –128.

24. Levene H. Robust tests for equality of variances / H. Levene // Contributions to Probability and Statistics / Ed. I. Olkin, Palo Alto. – Stanford University Press: 1960. – P. 278-292.
25. Loh W.-Y., Shin Y.-S. Split Selection Methods for Classi_cation Trees / Wei-Yin Loh and Yu-Shan Shih // Statistica Sinica. – 1997. – Vol. 7. – P. 815 – 840.
26. Maimon O., Rokach L. Data Mining and Knowledge Discovery. Handbook, 2nd ed.// Oded Maimon, Lior Rokach Springer: New York, 2010. – 1285 p.
27. Marques de Sa J. P. New Results on Minimum Error Entropy Decision Trees / Joaquim P. Marques de Sa, Raquel Sebastiao, and Joao Gama, Tanja Fontes // CIAPR'11 Proceedings of the 16th Iberoamerican Congress conference on Progress in Pattern Recognition, Image Analysis, Computer vision, and Applications. Chile, Pucon. – 2011. – P. 355 – 362.
28. Marques de Sa J. P. Tree Classifiers Based on Minimum Error Entropy Decisions / Joaquim P. Marques de Sa, Raquel Sebastiao, and Joao Gama // Canadian Journal on Artificial Intelligence, Machine Learning & Pattern Recognition. – 2011. – Vol. 2. – № 3. – P. 41 – 55.
29. Mehta M., Agrawal R., Rissanen J. SLIQ: A fast scalable classifier for data mining / Manish Mehta, Rakesh Agrawal, Jorma Rissanen / In Advances in Database Technology – EDBT '96 .Avignon, France, March 1996 // Lecture Notes in Computer Science. – 1996. – Vol. 1057. – P. 18-32.
30. Muller W., Wysotzki F. Automatic construction of decision trees for classification / W. Muller, F. Wysotzki // Annals of Operations Research. – 1994. – Vol. 52. – P. 231-247.
31. Quinlan J.R. C4.5: Programs for Machine Learning / John Ross Quinlan. – Morgan Kaufmann: 1993. – 302 c.
32. Quinlan J.R. Induction of decision trees // Machine Learning. – 1986. – Vol. 1. P. 81–106.
33. Rastogi R., Shim K. PUBLIC: A Decision Tree Classifier that Integrates Building and Pruning / Rajeev Rastogi, Kyuseok Shim // Proceedings of the 24th VLDB Conference August 1998, USA. – New York:1998. – P. 404 – 415.
34. Shih Yu-Shan. Families of splitting criteria for classification trees / Yu-Shan Shih // Statistics and Computing. – 1999. – Vol. 9. – P. 309-315.
35. Sonquist J. A. Searching for structure (alias-AID-III) // John A. Sonquist, Elizabeth Lauh Baker, James N. Morgan. – Institute for Social Research, University of Michigan: 1971. – 287 P.
36. Stasis A.C. Using decision tree algorithms as a basis for a heart sound diagnosis decision support system / A.C.Stasis, E.N.Loukis, S.A. Pavlopoulos, D.Koutsouris // Information Technology Applications in Biomedicine, 2003. 4th International IEEE EMBS Special Topic Conference, April 2003. 354 - 357
37. Taylor P. C., Silverman B. W. Block diagrams and splitting criteria for classification trees / P. C. Taylor, B. W. Silverman // Statistics and Computing. – 1993. Vol.3. – P. 147–161.
38. Vitanyi P., Li M. Ideal MDL and Its Relation to Bayesianism / Paul M.B. Vitanyi, Ming Li // In Proc. ISIS: Information, Statistic and Induction in Science. – Singapore: World Scientist, 1996. – P. 282 – 291.