О критериях ветвления, используемых при синтезе решающих деревьев*

И. Е. Генрихов
ingvar1485@rambler.ru
«Рейс ИТ», г. Москва

Предложен новый критерий ветвления — критерий максимизации доли объектов различных классов (Maximum Differences of Classes (MDC)). На модельных данных проанализированы особенности критерия MDC в сравнении с такими известными критериями, как: Gain, GainRatio, Gini Index, Twoing и критерий равномерного разбиения. На большом числе прикладных задач проведено исследование структурных и распознающих свойств решающего дерева в зависимости от применяемого критерия ветвления: глубина дерева, средняя глубина листьев дерева, «сбалансированность» дерева (разница между глубиной и средней глубиной листьев дерева), взвешенная глубина распределения описаний обучающих объектов по листьям дерева, «оптимальность» распределения обучающих объектов по листьям дерева (абсолютная разница между средней глубиной листьев дерева и взвешенной глубиной распределения описаний обучающих объектов по листьям дерева), качество дерева (с помощью метода «leave-one-out» и анализа распределения отступов обучающих объектов), число листьев дерева. Показано, что новый критерий ветвления позволяет получить более оптимальное решающее дерево по сравнению с рассмотренными критериями.

Ключевые слова: *критерий ветвления; распознавание образов; решающее дерево*

About splitting criteria used for synthesis of decision trees*

I. E. Genrikhov
Race IT, Moscow

A new splitting criterion is proposed — criterion of maximization share objects of different classes (Maximum Differences of Classes (MDC)). On the model data, the particular qualities of criterion MDC are analyzed in comparison with such famous criteria as: Gain, GainRatio, Gini Index, Twoing, and criteria of uniform partition. On a large number of real world tasks, the structural and recognizable properties of a decision tree are investigated to depending on the criteria branching: depth of tree, average depth of tree leaves, “balance” of tree (differences between depth and average depth of tree leaves), weighted depth of descriptions distribution of training objects on the leaves of tree, “optimal” distribution of training objects on tree leaves (absolute difference between average depth of tree leaves and weighted depth of the descriptions distribution of training objects on the leaves of tree), quality of tree (with method “leave-one-out” and analysis margins distribution of training objects), and number of tree leaves. It is shown that the new splitting criterion allows to receive more optimal decision tree in comparison with the considered criteria.

Keywords: *splitting criteria; pattern of recognition; decision tree*

Введение

Деревья решений (РД) являются известным инструментом, используемым при решении задач распознавания. Процедура построения классического дерева решений представляет собой итерационный процесс. Как правило, для построения очередной вершины дерева выбирается признак, наилучшим образом удовлетворяющий некоторому критерию ветвления. По значениям этого признака и осуществляется ветвление, далее указанная процедура повторяется для каждой из ветвей. При этом в зависимости от выбранного признака построенные деревья могут существенно отличаться как по структуре, так и по своим распознающим качествам.

Основная проблема заключается в синтезе «оптимального» РД — наиболее простого РД с высокой обобщающей способностью. В основу данной проблемы положен принцип «бритвы Оккама» (Occam's Razor): «Глупо прилагать больше усилий, чем нужно для достижения цели... Не стоит приумножать сущности сверх необходимого» [1]. Исследованию этой задачи посвящены много работ ([2, 3], обзор работ в [4]), в которых обосновывается NP-полнота задачи поиска оптимального РД. Решение данной проблемы обычно сводится к поиску оптимального баланса между размером дерева и его качеством с помощью различных методов редукции дерева [5, 1, 6, 4, 7].

В данной работе предлагается новый критерий ветвления — Maximum Differences of Classes (MDC), предназначенный для обработки вещественнозначной информации с наличием пропусков в признаковых описаниях объектов. Отличие указанного критерия от известных критериев таких, как: Gain [8], GainRatio [9], Gini Index [10, 11], Twoing [10, 11] и критерия равномерного разбиения (Dcrit) [12], заключается в способе оценивания на текущем шаге синтеза РД наилучшего разделения множества обучающих объектов на подмножества, образуемых после выбора признака при ветвлении из внутренней вершины дерева.

Очень часто в качестве оценки оптимальности РД рассматривается число листьев дерева ([12, 13, 1, 14, 15, 4]). При этом в качестве функции оптимальности РД также могут быть использованы и другие характеристики РД [16, 4] такие, как глубина и различные функционалы на основе взвешенной глубины листьев дерева. Поэтому, в данной работе для исследования структуры и качества РД в зависимости от применяемого критерия ветвления используются следующие характеристики дерева: число листьев дерева, глубина дерева, средняя глубина листьев дерева, «сбалансированность» дерева (разница между глубиной и средней глубиной листьев дерева), взвешенная глубина распределения описаний обучающих объектов по листьям дерева, «оптимальность» распределения обучающих объектов по листьям дерева (абсолютная разница между средней глубиной листьев дерева и взвешенной глубиной распределения описаний обучающих объектов по листьям дерева), оценка качества дерева с помощью метода «leave-one-out» и анализа распределения отступов обучающих объектов.

В разд. 1 введены основные понятия и описаны исследуемые критерии ветвления (Gain, GainRatio, Gini Index, Twoing, Dcrit и MDC) для случая вещественнозначной информации с пропусками в признаковых описаниях объектов.

В разд. 2 описан алгоритм синтеза РД, используемый в данной работе, для случая неравномерного распределения обучающих объектов по классам (в этом случае можно указать пару классов таких, что число обучающих объектов в одном из них существенно

Работа выполнена при финансовой поддержке РФФИ, проект № 13-01-00787.

больше числа обучающих объектов в другом) в предположении, что информация вещественнозначная и в признаковых описаниях объектов могут встречаться пропуски.

В разделе 3 приведены результаты анализа особенностей критериев ветвления, указанных в разделе 1, на модельных данных и описаны результаты исследования качества РД и структурных свойств РД в зависимости от применяемого критерия ветвления на реальных задачах из репозитория UCI и из коллекции задач, собранной в отделе Математических проблем распознавания и методов комбинаторного анализа Вычислительного центра им. А. А. Дородницына Российской академии наук (ВЦ РАН).

Основные понятия

Рассматривается задача распознавания по прецедентам с системой признаков $\{x_1, \ldots, x_n\}$, с непересекающимися классами $K_i,\ i \in I = \{1, \ldots, l\}$, и множеством обучающих объектов $T = \{S_1, \ldots, S_m\}$, где $S_r = (a_{r1}, \ldots, a_{rn}),\ a_{rj} \in \{\mathbb{R}, «*»\},\ r \in \{1, \ldots, m\}$, $j = 1, \ldots, n$. Если $a_{rj} =$«*», то значение признака x_j для объекта S_r не определено. Пусть далее $S = (b_1, \ldots, b_n)$ — распознаваемый объект, и $b_j \in \{\mathbb{R}, «*»\},\ j = 1, \ldots, n$.

Опишем структуру РД. Пусть $\hat{T}$ — подмножество обучающих объектов и $X(\hat{T})$ — подмножество признаков, рассматриваемые на текущем шаге построения РД. На первом шаге $\hat{T} = T,\ X(\hat{T}) = \{x_1, \ldots, x_n\}$.

При построении дерева могут встречаться два типа вершин: висячие и обычные вершины.

Определение 1. *Вершины РД, не имеющие выходящих дуг, называются висячими вершинами или листьями.*

Определение 2. *Обычная вершина в РД называется внутренняя вершина РД, для которой выполняются следующие условия:*

1. *данной вершине соответствует ровно один признак $x \in \{x_1, \ldots, x_n\}$;*
2. *в данную вершину входит одна дуга и выходит не менее двух дуг, помеченных разными числами;*
3. *каждая дуга, выходящая из данной вершины, входит либо в висячую вершину, либо в обычную вершину РД.*

Определение 3. *Глубиной ветви в РД называется число обычных вершин, которые содержит эта ветвь.*

Определение 4. *Глубиной РД называется максимальная глубина среди всех построенных ветвей дерева.*

Пусть v — висячая вершина, ей может быть приписана пара $\left(B_v, \left\{\omega_v^1, \ldots, \omega_v^l\right\}\right)$ [17], где B_v — элементарная конъюнкция (э.к.) над переменными $x_1, \ldots, x_n,\ \omega_v^i$ — оценка принадлежности объекта S классу K_i, $i \in I$, вносимая вершиной v.

В данной работе используется следующий способ вычисления вектора оценок в висячей вершине v [17]. Пусть m_v^i — число объектов класса K_i, описание которых попадает в интервал истинности конъюнкции $B_v,\ m^i$ — число объектов класса K_i в исходной обучающей выборке. Тогда $\omega_v^i = (m_v^i + 1)/(m^i + l), i \in I$.

Замечание 1. Причина применения указанного способа вычисления вектора оценок в висячей вершине v заключается в том, что в этом случае повышается качество распознавания в задачах с неравномерным распределением обучающих объектов по классам (в этом случае можно указать пару классов таких, что число обучающих объектов в одном из них существенно больше числа обучающих объектов в другом) [17].

При синтезе РД обычная вершина, соответствующая признаку x_t, $x_t \in X(\hat{T})$, помечается парой $(x_t, d(x_t))$, где $d(x_t)$ — «оптимальный» порог перекодировки [17]. Спуск из вершины $(x_t, d(x_t))$ происходит по двум ветвям, при этом левая ветвь помечается 0, а правая — 1. При спуске из вершины $(x_t, d(x_t))$ по левой (правой) ветви удаляются те объекты из $\hat{T}$, для которых значение признака x_t больше (не больше) $d(x_t)$.

Замечание 2. В данной работе применена стандартная процедура построения обычной вершины дерева в случае вещественнозначных значений признаков. Также, например, могут быть применены методы понижения значности до начала синтеза РД [18, 19], методы перекодировки значений признака с использованием более одного порога в обычной вершине дерева (перекодировка на основе интервалов) [20], построение линейной комбинации признаков в обычной вершине дерева без перекодировки значений [4] и другие методы [21, 22, 7].

Пусть висячая вершина v порождена ветвью дерева с обычными вершинами $x_{j_1}, \ldots, x_{j_r}$ и $\sigma_i, i \in 1, \ldots, r,$ — метка дуги, выходящая из вершины x_{j_i}. Под э.к. B_v для висячей вершины v подразумевается конъюнкция вида $[x_{j_1} > d(x_{j_1})]^{\sigma_1} \ldots [x_{j_r} > d(x_{j_r})]^{\sigma_r}$, где $[x_{j_i} > d(x_{j_i})]^{\sigma_i} = 1$, если $x_{j_i} > d(x_{j_i})$ при $\sigma_i = 1$ или $x_{j_i} \leqslant d(x_{j_i})$ при $\sigma_i = 0$, иначе $[x_{j_i} > d(x_{j_i})]^{\sigma_i} = 0$, $i \in 1, \ldots, r$.

Под интервалом истинности N_v э.к. B_v будем понимать множество наборов вида $(\alpha_1, \ldots, \alpha_n)$, где $\alpha_{j_i} = \sigma_i$ при $i = 1, \ldots, r$, и $\alpha_j \in \{0, 1\}$, $j \notin \{j_1, \ldots, j_r\}$.

Описанием объекта $S = (b_1, \ldots, b_n)$ в вершине v будем называть вектор $S(v) = (\beta_1, \ldots, \beta_n)$, в котором $\beta_{j_i} = 1$, если $b_{j_i} > d(x_{j_i})$, иначе $\beta_{j_i} = 0$ при $i = 1, \ldots, r$, и $\beta_j = 0$ при $j \notin \{j_1, \ldots, j_r\}$.

Определение 5. *Висячая вершина v называется голосующей для S, если $S(v) \in N_v$.*

При синтезе классического РД описание распознаваемого объекта S может попасть только в один лист дерева. В данной работе также строится РД, в котором описание распознаваемого объекта может попасть только в один лист дерева.

Пусть v — голосующая висячая вершина v для S. Для каждого $i \in I$ вычисляется оценка принадлежности объекта S классу K_i, имеющая вид

$\Gamma(S, K_i) = \omega_v^i$, $i \in I$.

Объект S зачисляется в класс K_i, если

$\Gamma(S, K_i) = \max\limits_{j \in I} \Gamma(S, K_j)$, $i \in I$,

$\Gamma(S, K_i) \neq \Gamma(S, K_j)$ при $i \neq j$, $j \in I$.

Если классов с максимальной оценкой несколько, то среди них выбирается только один, а именно тот, который имеет наибольшее число объектов в обучающей выборке, иначе происходит отказ алгоритма от классификации объекта S.

В случае вещественнозначной информации важной является задача нахождения такого порога $d(x_t)$, который наилучшим образом разделяет объекты из $\hat{T}$ по признаку x_t, принадлежащие разным классам. Опишем применяемый в данной работе способ выбора порога для перекодировки текущих значений признака $x_t \in X(\hat{T})$.

Пусть $\{c_1, \ldots, c_u\}$, $u \leqslant m,$ — множество различных значений по признаку x_t, $c_{i+1} > c_i$, $1 \leqslant i \leqslant u - 1$. Пусть объекты $S_{i_1} = (a_{i_1 1}, \ldots, a_{i_1 n})$, $S_{i_2} = (a_{i_2 1}, \ldots, a_{i_2 n})$ из $\hat{T}$ принадлежат разным классам. Если $a_{i_1 t} = c_i$ и $a_{i_2 t} = c_{i+1}$, тогда число $k_{t_i} = (c_i + c_{i+1})/2$, $1 \leqslant i \leqslant u - 1$, является порогом признака x_t.

Обозначим через $G_t = \{k_{t_1}, \ldots, k_{t_j}\}$ — множество порогов признака x_t. Порог $k \in G_t$ разбивает множество $\hat{T}$ на два подмножества $\left\{T_k^{(1)}, T_k^{(2)}\right\}$, где $T_k^{(1)}$ $\left(T_k^{(2)}\right)$ состоит из объектов множества $\hat{T}$, для которых $a_{rt} \leqslant k$ $(a_{rt} > k)$, $r = 1, \ldots, m$.

Здесь применена идея корректного перекодирования вещественнозначной информации, предложенная Ю. И. Журавлевым и используемая при построении логических процедур распознавания для дискретизации исходной информации и понижения значности целочисленных данных [19]. Данный способ определения порога для признака $x_t \in X(\hat{T})$, позволяет сократить число порогов и делает эту процедуру более корректной, по сравнению со способом определения порога для признака в алгоритмах C4.5[9] и CART[10].

Для каждого найденного порога признака $x_t \in X(\hat{T})$ определяется «информативность», и в качестве оптимального порога $d(x_t)$ берется тот порог, для которого эта информативность максимальна.

Опишем различные критерии выбора признака для ветвления.

Обозначим через $f\left(K_i, \hat{T}\right)$, $i \in I$, — число объектов из множества $\hat{T}$, относящихся к классу K_i, и R_t — множество объектов из $\hat{T}$, для которых значение признака x_t не определено. Вероятность $P_t^i(\hat{T})$ того, что случайно выбранный объект из множества $\hat{T}$ будет принадлежать классу K_i, равна

$$\frac{f\left(K_i, \hat{T} \backslash R_t\right)}{\left|\hat{T} \backslash R_t\right|}.$$

В случае энтропийного критерия информативности (Gain или GainRatio) способ выбора наиболее информативной пары признак-порог $((x_t, d(x_t)))$ для ветвления заключается в следующем.

Величина, вычисляемая по формуле

$$\mathrm{Info}(\hat{T})_t = -\sum_{i=1}^{l} P_t^i(\hat{T}) \log_2 P_t^i(\hat{T})$$

называется количеством информации (энтропией) по признаку x_t, необходимое для определения класса, которому принадлежит объект из множества $\hat{T}$.

Величина, вычисляемая по формуле

$$\mathrm{Info}(x_t)_k = \frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} \mathrm{Info}\left(T_k^{(1)}\right)_t + \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|} \mathrm{Info}\left(T_k^{(2)}\right)_t$$

называется количеством информации, необходимым для определения класса, которому принадлежит объект из множества $\hat{T}$ после разбиения $\hat{T}$ по порогу k признака x_t.

Информационный выигрыш (information gain) после выбора порога k признака x_t вычисляется по формуле $\mathrm{Gain}(x_t)_k = \mathrm{Info}(\hat{T})_t - \mathrm{Info}(x_t)_k$.

При синтезе РД с использованием критерия Gain на текущем шаге построения дерева выбирается только один признак $x_{\mathrm{opt}} \in X(\hat{T})$ с оптимальным порогом $d(x_{\mathrm{opt}})$, для которого достигается наибольшее значение величины $\mathrm{Gain}(x_t)_k$.

Величина, вычисляемая по формуле

$$\mathrm{SplitInfo}(x_t)_k = -\frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} \log_2 \frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} - \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|} \log_2 \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|}$$

определяет потенциальную информацию, получаемую при разбиении множества $\hat{T}$ по порогу k признака x_t.

Оптимальным порогом в G_t для признака x_t считается порог k для которого величина

$$\text{GainRatio}(x_t)_k = \frac{\text{Gain}(x_t)_k}{\text{SplitInfo}(x_t)_k},$$

принимает свое наибольшее значение.

В случае синтеза РД с применением критерия GainRatio (нормированный информационный выигрыш) на текущем шаге построения дерева выбирается признак $x_{\text{opt}} \in X(\hat{T})$ с соответствующим оптимальным порогом $d(x_{\text{opt}})$, для которого достигается наибольшее значение величины $\text{GainRatio}(x_t)_k$.

Замечание 3. Описанная выше модификация энтропийного критерия была применена в работах [17, 24]. Отличие от аналогичных критериев, применяемых в алгоритмах ID3 и C4.5, заключается в используемой методике учета пропущенных данных в признаковых описаниях обучающих объектов при ветвлении из обычной вершины дерева. Различие методик учета пропусков будет описано ниже.

Опишем критерий Gini Index [11].

Данный критерий основан на понятии «нечистоты» в исходных данных ($\hat{T}$), который определяется следующим выражением:

$$\text{Gini}(\hat{T})_t = 1 - \sum_{i=1}^{l} \left(P_t^i(\hat{T})\right)^2$$

Оптимальным порогом в G_t для признака x_t считается порог k для которого величина

$$\text{Gini}(x_t)_k = \text{Gini}(\hat{T})_t - \left(\frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} \text{Gini}\left(T_k^{(1)}\right)_t + \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|} \text{Gini}\left(T_k^{(2)}\right)_t \right)$$

принимает свое наибольшее значение.

При синтезе РД с использованием критерия Gini Index на текущем шаге построения дерева выбирается признак $x_{\text{opt}} \in X(\hat{T})$ с соответствующим оптимальным порогом $d(x_{\text{opt}})$, для которого достигается наибольшее значение величины $\text{Gini}(x_t)_k$.

Опишем критерий Twoing [11].

Оптимальным порогом в G_t для признака x_t считается порог k для которого величина

$$\text{Twoing}(x_t)_k = 0.25 \frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|} \left(\sum_{i=1}^{l} \left| P_t^i\left(T_k^{(1)}\right) - P_t^i\left(T_k^{(2)}\right) \right| \right)^2$$

принимает свое наибольшее значение.

При синтезе РД с использованием критерия Twoing на текущем шаге построения дерева выбирается признак $x_{\text{opt}} \in X(\hat{T})$ с соответствующим оптимальным порогом $d(x_{\text{opt}})$, для которого достигается наибольшее значение величины $\text{Twoing}(x_t)_k$. В работе [10] было отмечено, что критерий Twoing предпочтительней в случае наличия большого числа классов, чем критерий Gini Index.

Опишем критерий равномерного разделения Dcrit [12].

Раньше было показано, что после перекодировки значений признака x_t по некоторому порогу k можно считать, что все значения по признаку x_t для обучающих объектов из левого подмножества $T_k^{(1)}$ равны 0 и соответственно для объектов из правого подмножества $T_k^{(2)}$ равны 1. Тем самым можно вычислить число пар объектов вида (S_{i_1}, S_{i_2}) таких, что S_{i_1} и S_{i_2} — обучающие объекты из разных классов и $S_{i_1} \in T_k^{(1)}$, $S_{i_2} \in T_k^{(2)}$. Критерий равномерного разделения пар основан на максимизации числа таких пар.

Оптимальным порогом в G_t для признака x_t считается порог k для которого величина

$$\text{Dcrit}(x_t)_k = \sum_{i=1}^{l} \prod_{j \in I \setminus \{i\}} f\left(K_i, T_k^{(1)}\right) f\left(K_j, T_k^{(2)}\right)$$

принимает свое наибольшее значение.

При синтезе РД с использованием критерия Dcrit на текущем шаге построения дерева выбирается признак $x_{\text{opt}} \in X(\hat{T})$ с соответствующим оптимальным порогом $d(x_{\text{opt}})$, для которого достигается наибольшее значение величины $\text{Dcrit}(x_t)_k$.

Опишем разработанный критерий максимизации доли объектов различных классов MDC.

После перекодировки значений признака x_t по порогу k «информативность» левого подмножества можно оценить следующим образом:

$$\text{MDC}\left(T_k^{(1)}\right) = \sum_{i=1}^{l} \sum_{j \in I \setminus \{i\}} \frac{f\left(K_i, T_k^{(1)}\right)}{f\left(K_i, \hat{T} \setminus R_t\right)} - \frac{f\left(K_j, T_k^{(1)}\right)}{f\left(K_j, \hat{T} \setminus R_t\right)}.$$

Аналогично для правого подмножества:

$$\text{MDC}\left(T_k^{(2)}\right) = \sum_{i=1}^{l} \sum_{j \in I \setminus \{i\}} \frac{f\left(K_i, T_k^{(2)}\right)}{f\left(K_i, \hat{T} \setminus R_t\right)} - \frac{f\left(K_j, T_k^{(2)}\right)}{f\left(K_j, \hat{T} \setminus R_t\right)}.$$

Оптимальным порогом в G_t для признака x_t будем считать порог k для которого величина

$$\text{MDC}(x_t)_k = \frac{\left|T_k^{(1)}\right|}{\left|\hat{T} \setminus R_t\right|} \text{MDC}\left(T_k^{(1)}\right) + \frac{\left|T_k^{(2)}\right|}{\left|\hat{T} \setminus R_t\right|} \text{MDC}\left(T_k^{(2)}\right)$$

принимает свое наибольшее значение.

При синтезе РД с использованием критерия MDC на текущем шаге построения дерева выбирается признак $x_{\text{opt}} \in X(\hat{T})$ с соответствующим оптимальным порогом $d(x_{\text{opt}})$, для которого достигается наибольшее значение величины $\text{MDC}(x_t)_k$. Отличие от критерия Gini Index и Twoing заключается в том, что мы оцениваем долю объектов класса в подмножестве как число объектов этого класса в подмножестве относительно числа объектов этого же класса в текущем множестве. В критериях Gini Index и Twoing доля объектов класса в подмножестве оценивается относительно мощности подмножества.

В описанных критериях при вычислении информативности разбиения текущего множества $\hat{T}$ по порогу k признака x_t пропущенные значения признака x_t для объектов из $\hat{T}$ не принимаются во внимание. Если в описании обучающего объекта значение признака x_t пропущено, то при ветвлении из обычной вершины, соответствующей признаку x_t, этот

объект удаляется. Применяемая методика обработки пропусков направлена на сохранение исходной информации в полном объеме.

В алгоритме C4.5 применяется другая методика: предполагается, что пропущенные значения признака x_t вероятностно распределены пропорционально частоте появления встречающихся значений. Поэтому, в алгоритме C4.5 если в описании обучающего объекта значение признака x_t пропущено, то такой объект не удаляется и при ветвлении из обычной вершины, соответствующей признаку x_t, его описание попадает и в левую и в правое ветвь с определенными весами, которые учитываются при классификации. Использование методики как в алгоритме C4.5 вносит шум в обучающие данные. Если бы на месте пропущенного значения признака x_t находилось бы какое-либо реальное число, полученное в процессе сбора данных, то оно могло бы существенно повлиять на выбор оптимального порога признака x_t, что могло бы изменить структуру дерева.

Описание других популярных методик, используемых при решении задачи классификации с пропусками, представлено в работах [25, 26]. Большинство из них основано на замене пропущенного значения одним из допустимых. Это значение может быть вычислено различными способами: как среднее по существующим значениям признака; как наиболее вероятностное значение для признака; случайно выбрано из существующих значений; получено с помощью методов k-ближайших соседей, регрессионного или кластерного анализа. Также существует методика, основанная на удалении объектов с пропущенными значениями из обучающей выборки до начала построения РД. Такой подход может применяться в случае, когда число объектов с пропущенными значениями невелико по сравнению с числом всех обучающих объектов. Недостаток данного подхода состоит в том, что теряется полезная информация, содержащаяся в удаленных объектах. Иногда применяется методика, заключающаяся в построении дополнительной ветви, выходящей из обычной вершины, соответствующей признаку x_t, в которую «попадают» все обучающие объекты, в описании которых значение признака x_t не определено [13].

В случае, если значение признака x_t для распознаваемого объекта S не определено, то применяется следующий способ классификации такого объекта. Признак x_t исключается из исходного набора признаков. Далее строится РД для объекта S, т. е. при ветвлении из обычной вершины строится только та ветвь, по которой будет осуществлен «спуск» описания объекта S. Таким образом, при построении РД для классификации объекта S, учитываются только те признаки, для которых значения в S определены. Данный способ учета пропусков в распознаваемом объекте был применен в [17, 24].

Описание алгоритма синтеза решающего дерева

В данной работе применяется простой алгоритм синтеза РД («Simple Decision Tree» (SDT)). Алгоритм SDT является рекурсивным. Пусть $T(a_{rj})$ — матрица задаваемая обучающей выборкой T. Обозначим через $\tilde{T}$ матрицу, рассматриваемую на текущем шаге алгоритма, $\tilde{X} = \{x_j \in X_T\}$ — множество всех признаков на текущем шаге рекурсии. На первом шаге $\tilde{T} = T(a_{rj})$, $X_T = \{x_1, \ldots, x_n\}$. Шаг рекурсии в алгоритме SDT представляет собой последовательность действий 1–3, описанных ниже.

1. Просматриваются все столбцы матрицы $\tilde{T}$. Если в столбце нет хотя бы двух различных значений, то этот столбец вычеркивается из $\tilde{T}$. Если после просмотра всех столбцов $\tilde{T} \neq \varnothing$, то осуществляется переход к следующему действию, иначе переходим к третьему действию.
2. Если $\tilde{X} = \varnothing$, то переходим к третьему действию рекурсии, иначе для каждого признака $x_j \in \tilde{X}$ вычисляется значение критерия ветвления $Y(x_j)_k$. Если $G_j = \varnothing$, то зна-

чение критерия ветвления полагается равным 0. Если значение критерия ветвления равно 0 для всех признаков из $\tilde{X}$, то переходим к третьему действию. Иначе выбирается признак $x_t \in \tilde{X}$ с оптимальным порогом $k = d(x_t)$ для которого значение критерия ветвления является максимальным на текущем шаге синтеза РД. Далее строится две дуги, исходящих из вершины (x_t, k). Если матрица $\tilde{T}$ состоит из одного столбца, то при построении подматриц $\tilde{T}_k^{(1)}$ и $\tilde{T}_k^{(2)}$ этот столбец не удаляется. Для левой (правой) дуги вершины (x_t, k) строится подматрица $\tilde{T}_k^{(1)}$ $(\tilde{T}_k^{(2)})$ матрицы $\tilde{T}$, полученная удалением столбца, соответствующего признаку x_t, и строк S_r, в которых $a_{rt} > k$ $(a_{rt} \leqslant k)$, $r = 1, \ldots, m$. Если подматрица $\tilde{T}_k^{(1)}$ $(\tilde{T}_k^{(2)})$ содержит объекты одного класса или состоит из одного столбца, то переходим к третьему шагу, иначе полагается $\tilde{T} = \tilde{T}_k^{(1)}$ $(\tilde{T} = \tilde{T}_k^{(2)})$, $\tilde{X} = \tilde{X} \backslash \{x_t\}$ и осуществляется рекурсивный переход к первому действию.

3. Пусть $\tilde{T}$ содержит m_v^i объектов класса K_i, $i \in I$. Строится висячая вершина v с меткой $\left(B_v, \{\omega_v^1, \ldots, \omega_v^l\}\right)$, B_v — конъюнкция соответствующая данной вершине, $\omega_v^i = (m_v^i + 1)/(m^i + l)$, где m^i — число объектов класса K_i в исходной обучающей выборке, $i \in I$.

Замечание 4. В описании алгоритма SDT не указан применяемый критерий выбора признака. В численных экспериментах (раздел 3) вместо $Y(x_j)_k$ будут рассматриваться все критерии из раздела 1.

Для того, чтобы построить РД для классификации объекта S в случае наличия пропусков в описании распознаваемого объекта $S = (b_1, \ldots, b_n)$ достаточно на первом рекурсивном шаге вместо множества $\tilde{X}$ рассматривать множество $\tilde{X}_S = \{x_j \in X_T | b_j \neq «*»\}$.

Замечание 5. Для сокращения времени классификации объекта S предлагается строить только голосующие за S листья РД. Поэтому, при спуске из обычной вершины $(x_t, d(x_t))$, если $b_t \leqslant d(x_t)$, то строится левая дуга, иначе строится правая дуга.

В численных экспериментах (раздел 3) также будет использоваться алгоритм синтеза РД без удаления просмотренного признака при спуске из обычных вершин — алгоритм SDTw. В этом случае для левой (правой) дуги вершины (x_t, k) будет строится подматрица $\tilde{T}_k^{(1)}$ $(\tilde{T}_k^{(2)})$ матрицы $\tilde{T}$, полученная удалением только строк S_r, в которых $a_{rt} > k$ $(a_{rt} \leqslant k)$, $r = 1, \ldots, m$.

Результаты численного эксперимента

Исследование разделяющих свойств критериев, указанных в разделе 1, осуществлялось на модельных данных. Каждая модель представляет собой набор значений по одному признаку x. Значения признака x в описаниях обучающих объектов отличны друг от друга, т. е. нет двух одинаковых значений по признаку x. Отличие моделей друг от друга заключается в распределении значений признака x, в количестве классов и в числе обучающих объектов каждого класса.

Замечание 6. В рассматриваемых моделях, для простоты, под порогом понимается полусумма двух соседних значений из упорядоченного множества текущих значений признака x. Поэтому число порогов на единицу меньше числа различных значений по признаку x. Пороги упорядочены по возрастанию значений.

Опишем модели и полученные результаты.

Модель 1 — имеется 50 объектов первого класса и 50 объектов второго класса. Значения признака распределены так, что последовательное применение упорядоченных порогов по признаку x приводит к последовательному «перетеканию» по одному объектов

одного класса: первый порог разделяет текущее множество объектов так, что в левом подмножестве оказывается один объект первого класса, второй порог разделяет текущее множество объектов так, что левое подмножество состоит из двух объектов первого класса, третий порог делит текущее множество так, что в левом подмножестве оказываются три объекта первого класса и т. д. Когда объекты первого класса полностью «перетекут» по некоторому порогу в левое подмножество, то, при использовании следующих порогов, начинают «перетекать» по одному объекты второго класса. Последний порог из упорядоченного множества порогов по признаку x разделяет текущее множество объектов так, что в правом подмножестве оказывается только один объект второго класса.

Модель 2 — имеется 50 объектов первого класса и 50 объектов второго класса. Значения признака распределены так, что последовательный просмотр упорядоченных порогов по признаку x приводит к поочередному «перетеканию» по одному объектов из разных классов: первый порог разделяет текущее множество объектов так, что в левом подмножестве оказывается один объект первого класса, второй порог разделяет текущее множество так, что левое подмножество состоит из одного объекта первого класса и одного объекта второго класса, третий порог разделяет объекты так, что в левое подмножество попадают два объекта первого класса и один объект второго класса и т. д. Последний порог из упорядоченного множества порогов по признаку x разделяет текущее множество объектов так, что в правом подмножестве оказывается только один объект второго класса.

Модель 3 и модель 4 аналогичны соответственно модели 1 и модели 2, но только в этих случаях рассматривается ситуация, когда в первом классе 30 объектов, а во втором 70 объектов.

Модель 5 по сути является моделью 1, но только в этой ситуации имеется 3 класса и каждый класс содержит по 50 объектов. Тем самым, когда в левое подмножество полностью «перетекут» объекты первого класса, то при применении последующих порогов начинают «перетекать» объекты второго класса. После того, как в левое подмножество «перетекут» объекты второго класса, начинают «перетекать» объекты третьего класса. Последний порог из упорядоченного множества порогов по признаку x разделяет текущее множество объектов так, что в правом подмножестве оказывается один объект третьего класса.

Модель 6 подобна модели 2, но только в этом случае имеется 3 класса и каждый класс содержит по 50 объектов. Первый порог разделяет текущее множество объектов так, что в левом подмножестве оказывается один объект первого класса. Второй порог разделяет текущее множество объектов так, что левое подмножество состоит из одного объекта первого класса и одного объекта второго класса. Третий порог разделяет объекты так, что в левое подмножество попадает один объекта первого класса, один объект второго класса и один объект третьего класса. Четвертый порог разделяет объекты так, что в левое подмножество попадает два объекта первого класса, один объект второго класса и один объект третьего класса и т. д. Последний порог из упорядоченного множества порогов по признаку x разделяет текущее множество объектов так, что в правом подмножестве оказывается один объект третьего класса.

Модель 7 и модель 8 подобны соответственно модели 1 и модели 2, но только в этой ситуации имеется 3 класса. Первый класс имеет два обучающих объекта, второй и третий класс содержит по 50 объектов.

Модель 9 подобна модели 5, но только в этом случае имеется 4 класса по 50 объектов в каждом.

Значения критериев для каждого порога и для каждой модели изображены на рис. 1, рис. 2 и рис. 3. По оси абсцисс — номер просматриваемого порога из упорядоченного множества порогов по признаку x, по оси ординат — значение критерия для соответствующего порога признака x, т. е. точка на графике с координатой (i, j) соответствует значению критерия (величина j) для порога с номером i.

Все критерии одинаково достигают максимума в первой и третьей модели при использовании порога, когда в левое подмножество попадают все объекты первого класса, а в правое подмножество все объекты второго класса.

Результаты по второй модели: критерии Gain, GainRatio, Gini Index и Twoing достигают максимума на порогах, когда в любом подмножестве оказывается только один объект; критерий MDC принимает максимальное значение на порогах, когда число объектов одного из двух классов в любом подмножестве не равно числу объектов из другого класса; критерий Dcrit достигает максимальное значение на порогах, когда в левом или в правом подмножестве находится ровно половина объектов одного из двух классов, причем в этом подмножестве также могу находиться объекты из другого класса.

Результаты по четвертой модели: критерии Gain, GainRatio, Gini Index, Twoing и MDC достигают максимума при применении порога, когда в правом подмножестве оказывается максимальное число объектов из второго класса, а все остальные объекты попадают в левое подмножество; критерий Dcrit принимает максимальное значение в двух случаях (значения на вершине гребня), когда в одном из двух подмножеств находится 49 объектов, а в другом подмножестве 51 объект.

Результаты по пятой модели: критерии Gain, GainRatio, Gini Index и Twoing достигают максимума на порогах, когда в любом подмножестве находятся все объекты одного класса и только объекты этого класса; критерии MDC и Dcrit достигают максимума на порогах, когда в одном подмножестве находятся все объекты одного класса и в этом подмножестве также могут быть объекты из других классов.

Результаты по шестой модели (аналогичны результатам по второй модели): критерии Gain, GainRatio, Gini Index и Twoing достигают максимума на порогах, когда в одном подмножестве оказывается только один объект; критерий MDC имеет максимальное значение, когда число объектов одного из трех классов не равно числу объектов из других классов; критерий Dcrit имеет максимальное значение на порогах, когда в левом или в правом подмножестве находится ровно половину объектов из двух любых классов, причем в этом подмножестве также могу находиться объекты из другого класса.

Результаты по седьмой модели: критерии Gain, Dcrit, Gini Index и Twoing достигают максимума при использовании порога, когда в правом подмножестве оказываются все объекты третьего класса и только объекты этого класса; критерий GainRatio достигает максимума на порогах, когда в одном подмножестве оказываются все объекты одного из трех классов и только этого класса; критерий MDC достигает максимума на порогах, когда в правом подмножестве оказываются все объекты из третьего класса и в этом подмножестве могут быть объекты из других классов.

Результаты по восьмой модели: критерии GainRatio, Gini Index и Twoing достигают максимума на пороге, когда в левое подмножество попадает только один объект из первого класса; критерий Gain и MDC достигает максимума на пороге, когда в левом подмножестве оказываются все объекты первого класса и по одному объекту из второго и третьего класса, причем критерий MDC достигает еще одного максимума на пороге, когда в левом подмножестве оказываются все объекты первого класса, два объекта второго класса и один объект третьего класса; критерий Dcrit максимален на пороге, когда в левом подмноже-

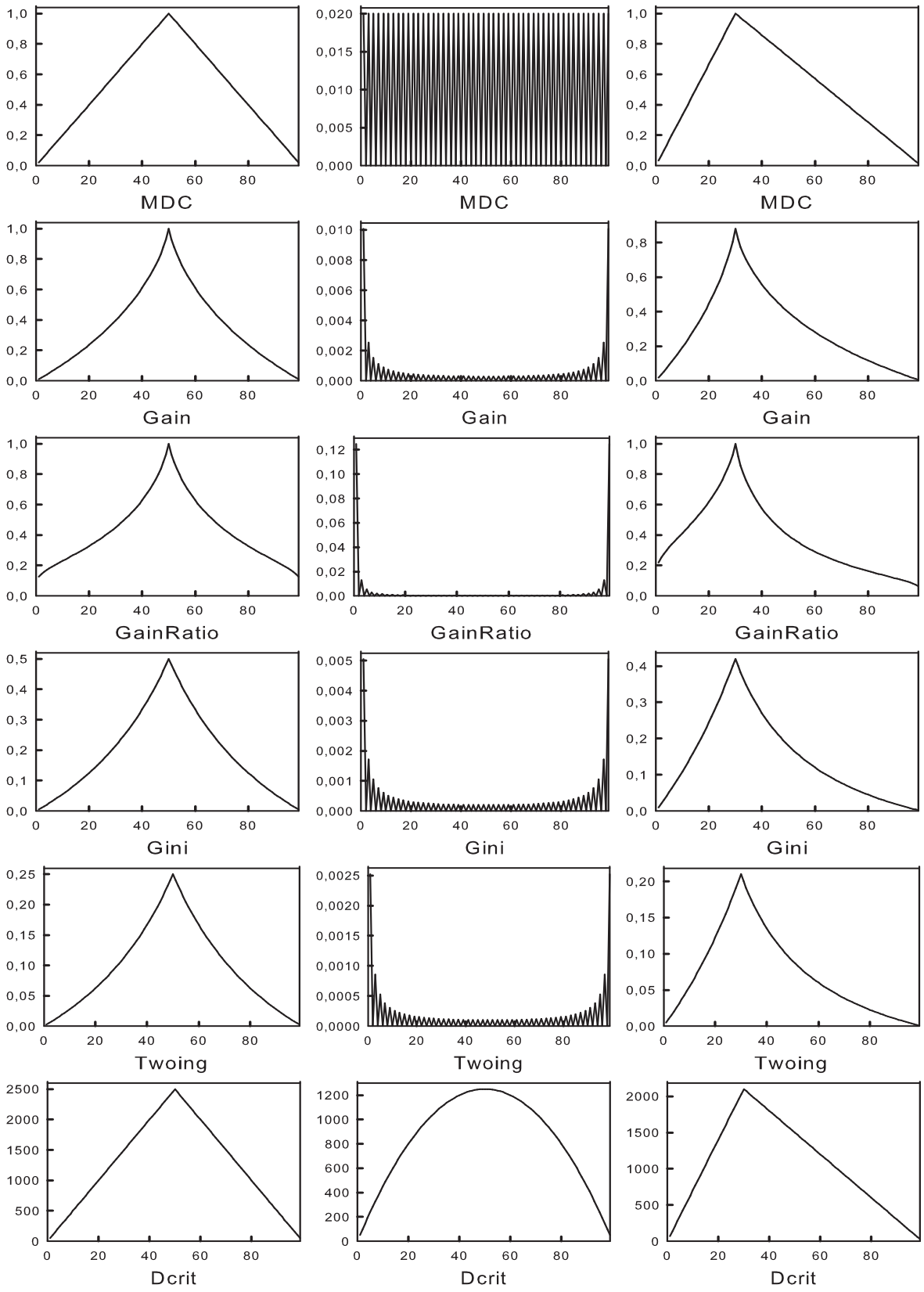


Рис. 1. Изменение значений критериев ветвления от порогов в модели 1 (слева), в модели 2 (по центру) и модели 3 (справа)

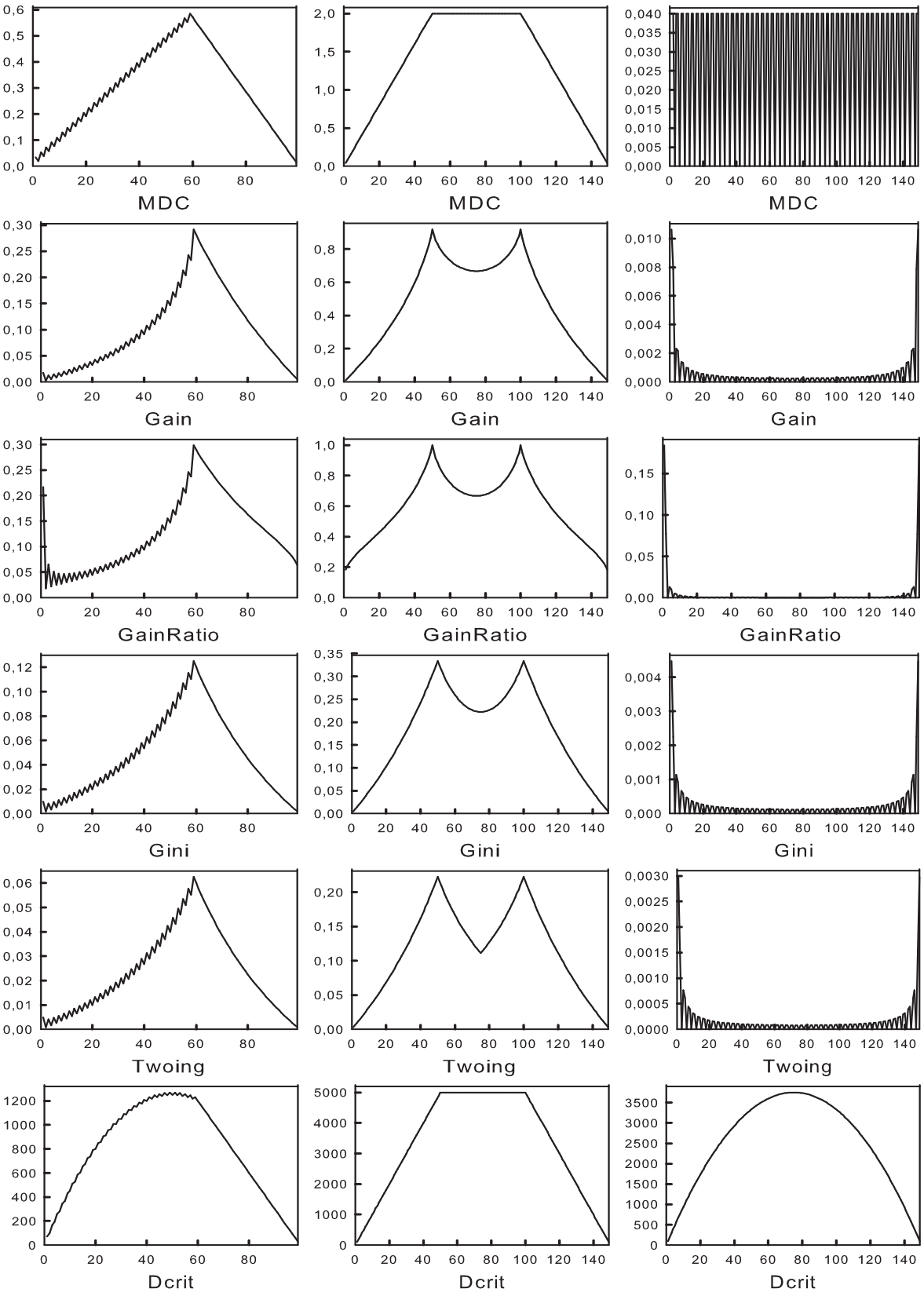


Рис. 2. Изменение значений критериев ветвления от порогов в модели 4 (слева), в модели 5 (по центру) и модели 6 (справа)

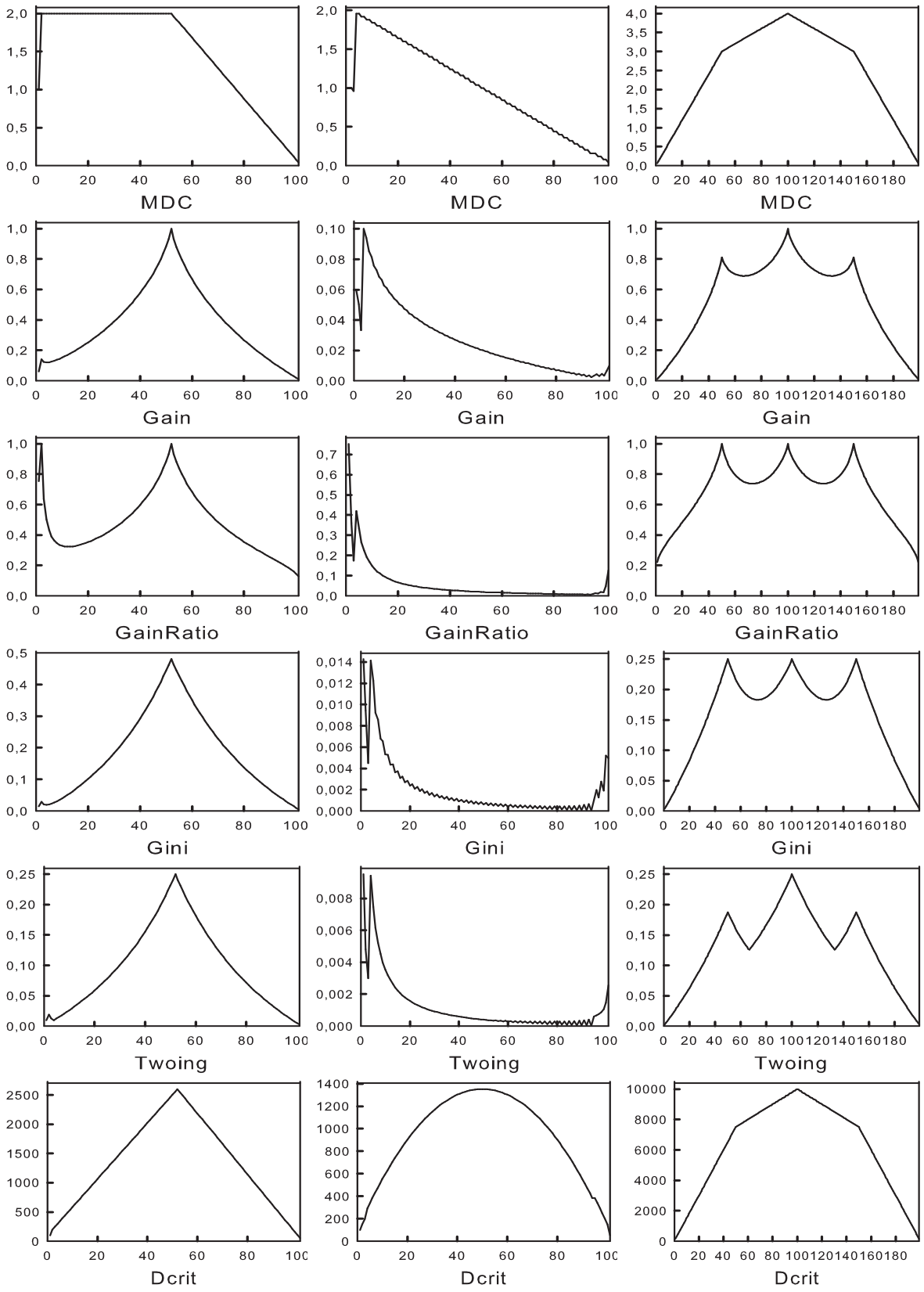


Рис. 3. Изменение значений критериев ветвления от порогов в модели 7 (слева), в модели 8 (по центру) и модели 9 (справа)

стве оказываются 24 объекта из второго или из третьего класса и все объекты первого класса.

Результаты по девятой модели: критерии MDC, Gain, Dcrit, Twoing GainRatio и Gini Index достигают максимума на пороге, когда в левое подмножество попадают все объекты первого и второго класса, а все объекты третьего и четвертого класса оказываются в правом подмножестве; критерии GainRatio и Gini index достигают также максимумов, когда в левом или в правом подмножестве оказываются все объекты только одного из четырех классов.

Таким образом, критерий Dcrit разделяет объекты так, чтобы в левом и в правом подмножестве оказалось больше объектов из разных классов, не смотря даже на то, что в одном подмножестве могут быть объекты из разных классов. Критерий MDC разделяет объекты так, чтобы максимизировать число объектов каждого класса в левом и в правом подмножестве, при этом большое влияние на разделение объектов оказывает число объектов соответствующего класса в исходном множестве. На большинстве модельных данных критерии Gain, GainRatio, Gini Index и Twoing достигают максимальное значение на одних и тех же порогах, что говорит о том, что они близки между собой по оценке наиболее «информативного» разделения исходного множества обучающих объектов. Отличие указанных четырех критериев друг от друга наиболее заметно на последних трех моделях (неравномерное распределение объектов по трем классам и ситуация наличия объектов из более, чем трех классов). Критерий Gain направлен на отделение класса с наибольшим числом объектов целиком по сравнению с классом, состоящего из небольшого числа объектов (модель 7 и 8), а также на определение более оптимального разделения исходного множества при наличии большого числа классов (модель 9). Критерий GainRatio сглаживает значение критерия Gain, оценивая мощность подмножеств относительно мощности исходного множества. Так же, как и критерий Gain, критерий GainRatio направлен на отделение класса целиком, но не обращает внимание на его мощность, что может привести к тому, что можно пропустить отделение большого класса целиком (модель 7) и отделение небольшого класса целиком с небольшим числом объектов из других классов (модель 8), а также можно пропустить более оптимальное разделение объектов (модель 9). Похожий вывод отмечен в работе [27], где отмечено, что критерий GainRation предпочитает несбалансированные разделения текущего множества, где одно из подмножеств намного меньше других. Критерий Gini Index и Twoing нацелены на отделение более мощного класса целиком (модель 7), а наличие объектов из других классов («примесей») может негативно сказаться на отделении небольшого класса целиком с небольшим числом объектов из других классов (модель 8). Критерий Gini Index направлен на выделение подмножеств равного размера и «чистоты» ([4, 11]). Критерий Twoing позволяет также определить более оптимальное разделение исходного множества при наличии большого числа классов (модель 9), что и было ранее отмечено в работе [11].

Исследование структурных свойств и качества РД, построенного с помощью алгоритма SDT (SDTw), в зависимости от применяемого критерия ветвления, указанных в разделе 1, осуществлялось на 26 реальных задачах из репозитория ВЦ РАН [28] и на 10 задачах (Breast Cancer Wisconsin, Credit Approval, Pima Indians Diabetes, Glass Identification, Johns Hopkins University Ionosphere, LED Display, Large Soybean, Tic-Tac-Toe Endgame, Lymphography Domain, Dermatology) из репозитория UCI [29]. Рассматривались следующие свойства РД: глубина РД, средняя глубина листьев РД, качество РД (с помощью метода LOO («leave-one-out») [6] и анализа распределения отступов обучающих объек-

тов [27]), число листьев РД и взвешенная глубина распределения описаний обучающих объектов в листьях РД.

Важность первого свойства заключается в том, что, во-первых, при небольшой глубине РД увеличивается интерпретируемость решающего правила, во-вторых, как показано в работах [30, 31], оценка вероятности ошибки решающего правила уменьшается с уменьшением глубины дерева, в-третьих, чем меньше глубина, тем меньше ранг конъюнкций и тем выше вероятность обнаружения неслучайной конъюнктивной закономерности [1, 3]. Следует также отметить, что РД с глубиной k_1 и РД с глубиной k_2, где $k_2 > k_1$, принадлежат к разным классам решающих функций, так как от глубины дерева зависит максимальное число листьев. Например, для рассматриваемого в данной работе РД (из каждой обычной вершины выходит ровно две дуги), максимальное число листье $mu = 2^k$, где k — глубина дерева. Следовательно, вероятность ошибки, зависящей от числа листьев в дереве, для класса РД с глубиной k_1 будет ниже, чем для класса РД с глубиной k_2 [1, 3, 32]. В-четвертых, выбор критерия ветвления, приводящего к синтезу РД с наименьшей глубиной, позволяет конструировать более качественные «сложные» модели РД ([13, 1, 17, 23, 33, 34]), т. к. также снижается сложность рассматриваемого класса решающих функций ([1, 30, 24]).

Вторая характеристика РД показывает на какой глубине больше всего концентрируются листья дерева. Следует отметить, что разность между глубиной дерева и средней глубиной листьев дерева показывает «сбалансированность» РД: чем ближе эти значения друг к другу, тем более сбалансированным является дерево [12, 4].

Шестая характеристика — глубина концентрации обучающих объектов, описания которых принадлежат интервалам истинности соответствующих листьев РД. Позволяет охарактеризовать «быстроту» разделения объектов, принадлежащих разным классам. Чем меньше значение этой характеристики, тем быстрее происходит разделение обучающих объектов из разных классов.

Совместно вторая и шестая характеристика показывают, в каком-то смысле, «оптимальность» распределения обучающих объектов по листьям РД. Чем ближе значение указанных характеристик друг к другу, тем «оптимальнее» распределены описания обучающих объектов по листьям дерева. В дальнейшем значения этих характеристик могут быть использованы для более надежного и качественного «обрезания» РД: если, например, описания практически всех обучающих объектов, принадлежащие интервалам истинности конъюнкций соответствующих листьев РД, сконцентрированы ближе к корню дерева, и лишь небольшая часть описаний попадает в интервалы истинности конъюнкций соответствующих листьев, находящихся глубже, то эти листья в дальнейшем можно обрезать, тем самым получить РД с меньшей глубиной и с более «оптимальным» распределением описаний обучающих объектов по листьям РД.

Шестая характеристика также может быть использована для оценки надежности РД [1, 3, 32, 31]. Чем меньше листьев в дереве, тем уже класс РД и тем выше надежность РД.

Опишем полученные результаты по исследованию качества и структурных свойств РД, построенного с помощью алгоритма SDT (SDTw), в зависимости от применяемого критерия ветвления

Качество алгоритма SDT (SDTw) оценивалось методом LOO. Вычислялась величина $\tilde{Q} = \sum_{i=1}^{l} q_i/l$, где q_i — процент правильно классифицированных объектов класса K_i, l — число классов в обучающей выборке.

Замечание 7. Стоит отметить, что в [17] показано, что величина $Q = \frac{m^+}{m}$, где m^+ — число всех правильно классифицированных объектов, не является корректной величиной по сравнению с величиной Q, в случае неравномерного распределения обучающих объектов по классам. Величина Q, в отличие от величины $\tilde{Q}$, учитывает распределение объектов по классам, и поэтому чаще всего $Q \geqslant \tilde{Q}$. В связи с этим в данной работе не рассматривался функционал Q.

В табл. 1 для каждой задачи и каждого критерия ветвления, применяемого при синтезе РД алгоритмом SDT (SDTw), приведены значения величины $\tilde{Q}$. Бирюзовым цветом выделены максимальные значения величины $\tilde{Q}$ для задачи при применении алгоритма SDT с различными критериями ветвления, а зеленым цветом — максимальные значения величины $\tilde{Q}$ для задачи при использовании алгоритма SDTw. Звездочкой помечены задачи с наличием пропусков в признаковых описаниях объектов.

Наилучшее качество алгоритма SDT было достигнуто при использовании критериев: GainRatio, MDC и Gain. Качество РД с критерием GainRatio оказалось лучше на 13 задачах по сравнению с другими критериями. При применении критерия Gain или MDC качество РД оказалось лучше на 10 задачах по сравнению с другими критериями ветвления. В среднем по всем задачам качество деревьев с критериями GainRatio, MDC, Gain, Twoing и Gini Index оказалось схожим. Наименьшее качество в среднем по всем задачам показал алгоритм SDT с критерием Dcrit.

Наилучшее качество алгоритма SDTw было достигнуто при применении критериев: MDC, Dcrit и GainRatio. Качество РД с критерием MDC оказалось лучше на 10 задачах по сравнению с другими критериями ветвления. При использовании критерия Dcrit качество РД оказалось лучше на 9 задачах, с критерием GainRatio на 8 задачах. В среднем по всем задачам качество деревьев с критериями GainRatio, MDC, Gain, Twoing и Gini Index оказалось схожим. Наименьшее качество в среднем по всем задачам показал алгоритм SDTw с критерием Dcrit.

При использовании критерия MDC или Twoing среднее качество алгоритма SDT на всех задачах сопоставимо с алгоритмом SDTw. Среднее качество алгоритма SDT с критерием Gain или GainRatio немного лучше по сравнению с алгоритмом SDTw, обратная ситуация наблюдается для алгоритмов на основе критерия Dcrit или Gini Index.

В данной работе для семи задач были также построены графики распределения отступов обучающих объектов [27] — рис. 4 и рис. 5.

Из графиков 4 и 5 видно, что лучше «отодвигают» обучающие объекты от границы классов критерий Gini Index и Twoing. Хуже «отодвигают» обучающие объекты от границы классов критерий MDC и Dcrit.

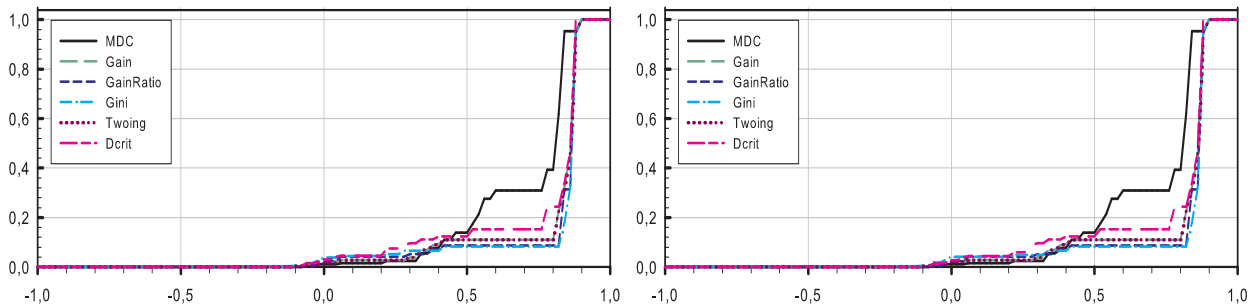
Из графиков 4 и 5 также видно, что распределение отступов улучшилось при переходе от алгоритма SDT к алгоритму SDTw для критерия GainRatio. При переходе от алгоритма SDT к алгоритму SDTw распределение отступов немного ухудшилось для критерия Dcrit и критерия MDC, а для Gain, Gini Index и Twoing немного улучшилось.

В табл. 2 представлены значения структурных характеристик РД при использовании алгоритма SDT с различными критериями. Обозначим через μ — число листьев дерева, k_i — глубина i-ого листа, m_i — число обучающих объектов, описание которых принадлежит интервалу истинности конъюнкции, приписанной листу i, $\hat{m} = \sum_{i=1}^{\mu} m_i$. В табл. 2 содержатся значения по следующим величинам: μ, $k = \max\limits_{i=1,\ldots,\mu} k_i$ — глубина РД, $\Delta_k = \sum_{i=1}^{\mu} k_i/\mu$ — средняя глубина листьев, $\Delta_o = \sum_{i=1}^{\mu} k_i m_i/\hat{m}$ — взвешенная глубина распределения описаний обучающих объектов в листьях РД, $\Delta_r = |\Delta_o - \Delta_k|$. Среди всех

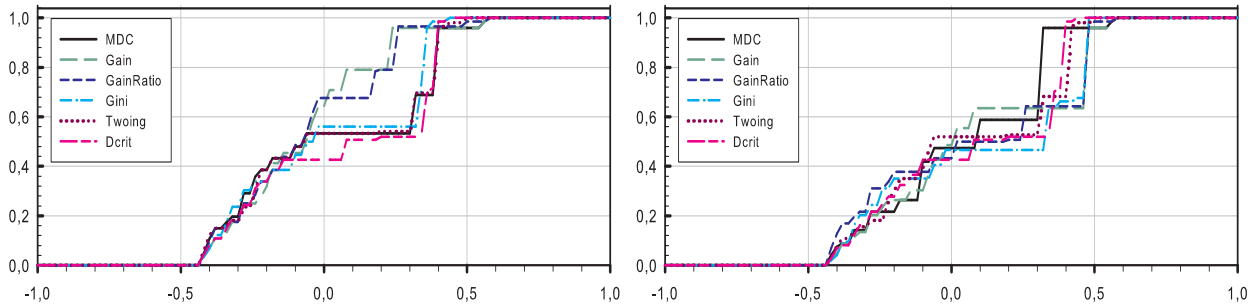
Таблица 1. Эффективность критериев ветвления

Описание задачи		MDC		Gain		GainRatio		Gini Index		Twoing		Dcrit	
No	$(\|K_1\|,\ldots,\|K_l\|,n)$	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw
1	$(48,12,69)*$	61,5	61,5	64,6	66,7	56,3	57,3	64,6	64,6	64,6	64,6	46,9	43,8
2	$(23,173,9)*$	61	60,6	60,7	61,9	57,3	61	62,8	63,8	62,8	63,8	54,8	59,7
3	$(23,173,17)*$	58,6	61,9	54,7	50,8	54	54,7	44,8	50,2	45,4	50,2	44	40,5
4	$(152,190,15)*$	79,1	79,6	78,4	81,1	81,4	82,2	78,7	79,8	78,7	79,8	74,7	82,8
5	$(16,17,12)*$	51,3	48,3	51,5	51,5	57,5	57,5	48,5	51,5	48,5	51,5	75,9	72,8
6	$(48,23,8)*$	77,5	67,8	76,4	66,7	63,7	62,6	65,7	64,6	65,7	64,6	59,3	63,4
7	$(89,42,9)*$	76,3	71,5	78,6	71,5	75,1	71,6	78,1	75,6	78,1	75,6	59,5	57,4
8	$(76,33,24,7)$	86,9	85,9	86,9	85,9	90,6	89,5	88,3	87,3	88,3	87,3	89,5	89,5
9	$(86,31,22,20,8,13)$	37,3	35	34,6	26,1	41,5	27,8	27,7	28,1	31,5	27,3	36,4	38,8
10	$(120,150,13)$	77,4	75,5	76,2	74,8	74,4	71,3	75	73,2	75	73,2	77,4	76,3
11	$(32,123,19)*$	75,2	79,1	61,3	64,9	75,5	75,8	78,2	73,9	78,2	73,9	63,7	63,7
12	$(218,126,9)*$	95,1	94	96,1	95,1	94,2	94	94,8	94,8	94,8	94,8	95,4	93,9
13	$(38,107,35)$	69,8	70,6	75,4	73,6	77,4	77	77,3	77,3	77,3	77,3	68	71,4
14	$(35,72,35)$	59,7	57,5	58,2	60,4	65,4	52,6	65,3	66	65,3	66	57,7	59
15	$(38,35,35)$	60,2	61,4	60,3	60,3	82,3	79,5	58,8	58,8	58,8	58,8	68,6	72,7
16	$(38,72,35)$	76,4	71,7	84	83,3	76,4	76,4	76,5	77,8	76,5	77,8	69,1	68,4
17	$(30,102,24)*$	63,6	64,4	55,8	54,3	59,1	51,8	55	54,7	55	53	62	62,7
18	$(51,218,24)*$	47,8	54,1	58	52,4	54,6	57,3	54	56,5	54	56,5	56,2	54,7
19	$(51,218,21)*$	52,2	52,6	61,2	55,6	52,8	56,5	55,8	58,9	56,1	58,9	57,2	54,7
20	$(60,15,39,5)*$	70,5	68,8	64,3	67,3	69,3	64,5	68,2	66,7	68,2	66,2	68,2	71,8
21	$(47,30,7)*$	83,6	87,4	89,1	84,7	89,7	90,3	89,1	89,1	88	88	83,6	87,4
22	$(40,40,18)*$	68,8	66,3	72,5	73,8	72,5	63,8	68,8	70	68,8	70	55	60
23	$(11,47,15)$	83,2	78,6	89,1	91,2	78,9	88	90,1	90,1	90,1	90,1	87,7	83,2
24	$(39,22,18)$	66,3	74,1	69,3	69,3	80	80	72,6	72,6	72,6	72,6	73,1	76,4
25	$(52,25,8)*$	55,7	52,5	48,9	51,7	44,5	50,6	54,8	54,6	56,8	53,7	64,5	61,5
26	$(59,71,48,13)$	95,6	96,6	93,4	94,9	95,6	95,7	88,4	87,6	88,1	87,9	91,2	91,6
27	$(458,241,9)*$	94,6	94,3	93,1	92	94,4	91,9	94,5	91,9	94,5	91,9	94	94,4
28	$(307,383,15)*$	80	78,6	81,3	83,5	81,4	81,5	77,9	80,7	77,9	80,7	83,3	80,3
29	$(500,268,8)$	69,5	68,3	72,4	67,7	66,2	66,8	70,6	67,2	70,6	67	68,3	64,5
30	$(70,76,17,13,9,29,9)$	65	69,6	69,3	68,3	56,6	64,9	59,4	57,7	66,9	63,8	58,8	66,4
31	$(126,225,34)$	85	85,2	85,7	88,1	93,6	93,6	86,3	87,2	86,1	87	80,1	85,3
32	$(300,330,309,315,310,269,302,304,276,285,7)$	100	100	100	100	100	100	100	100	100	100	100	100
33	$(20,20,20,88,44,20,20,92,20,20,20,44,20,91,91,15,14,16,8,35)*$	92,1	92,3	91,4	93	92,8	94,1	91,6	93,8	91,3	93,5	81,6	79
34	$(626,332,9)$	86,2	93,2	86,4	93,3	83	88,3	85	93,6	85	93,6	86,7	91
35	$(2,81,61,4,18)$	60,9	68,1	46,8	51,1	58	66,8	51,4	48,1	57	58,3	52,9	48,9
36	$(112,61,72,49,52,20,34)$	93,6	93,6	93,5	93,5	90,6	90,4	90,5	92,1	91,6	92,7	89,3	89,7
Среднее значение		72,71	72,79	72,76	72,23	73,24	72,99	71,92	72,23	72,45	72,55	70,41	71,04

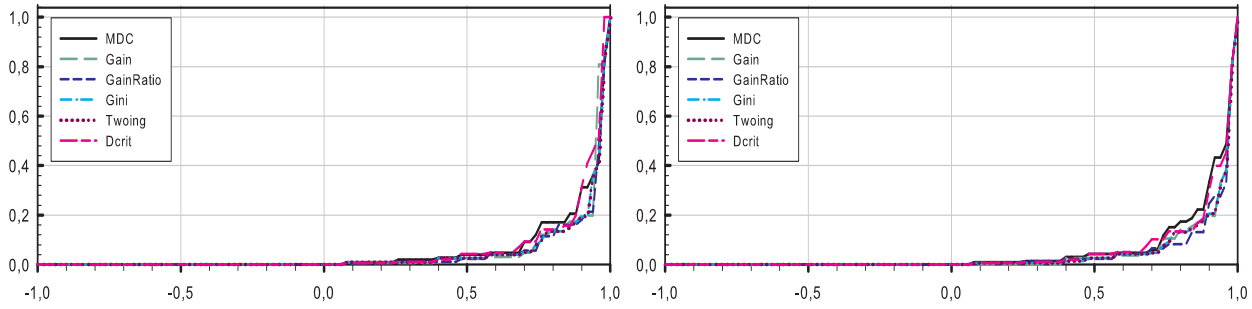
РД, зависящих от применяемого критерия ветвления, для каждой задачи желтым цветом выделены минимальные значения величины μ, коричневым — минимальные значения



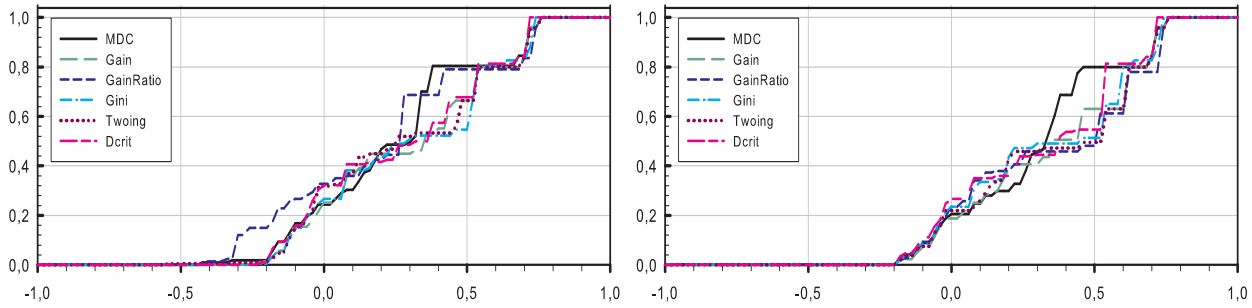
(а) Графики распределения отступов для задачи 36



(б) Графики распределения отступов для задачи 35



(в) Графики распределения отступов для задачи 31

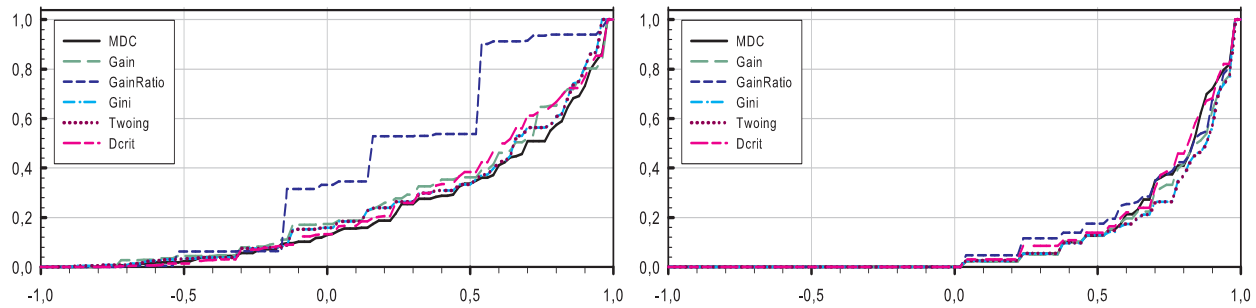


(г) Графики распределения отступов для задачи 30

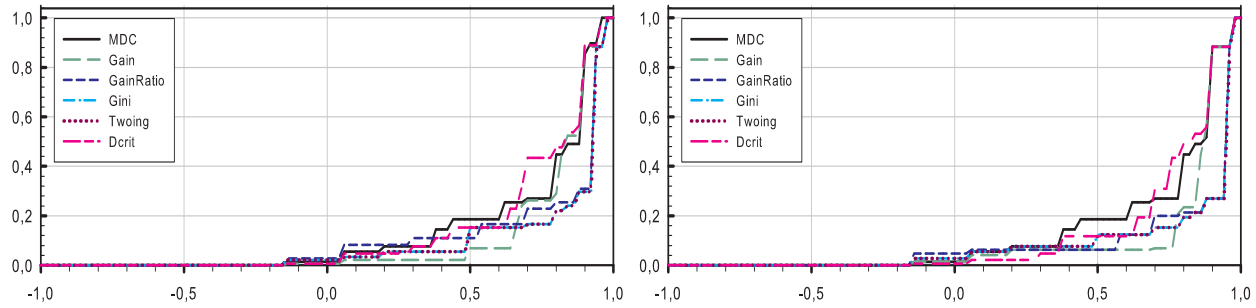
Рис. 4. Графики распределения отступов для алгоритма SDT (слева) и SDTw (справа)

величины k, бирюзовым — минимальные значения величины Δ_k, зеленым — минимальные значения величины Δ_o, фиолетовым — минимальные значения величины Δ_r. Последняя строка в табл. 2 обозначена как Δ — среднее значение по всем задачам для соответствующего критерия и соответствующей характеристики РД.

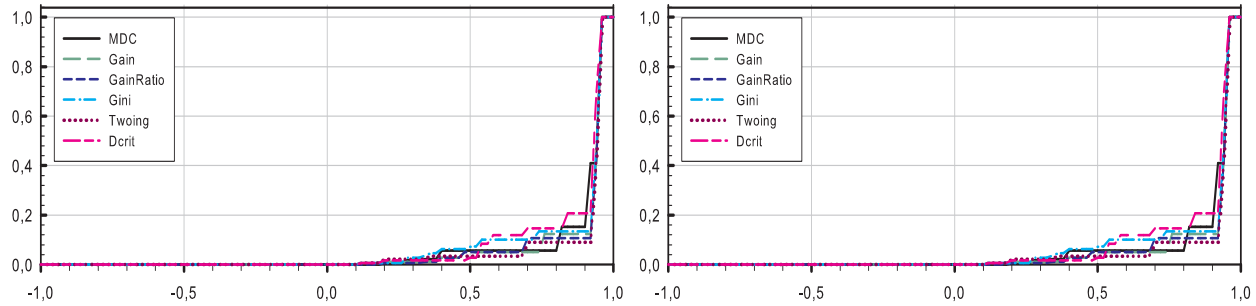
Результаты по структурным свойствам РД из табл. 2 следующие.



(а) Графики распределения отступов для задачи 29



(б) Графики распределения отступов для задачи 13



(в) Графики распределения отступов для задачи 26

Рис. 5. Графики распределения отступов для алгоритма SDT (слева) и SDTw (справа)

Наилучшие результаты среди всех критериев, используемых в алгоритме SDT, по минимальному числу листьев в РД показали критерии Gain, GainRatio и MDC. Критерий Gain оказался лучше по этому показателю на 20 задачах, GainRatio на 17 задачах, а MDC на 11 задачах. Наихудшие результаты продемонстрировал алгоритм SDT с критерием Dcrit. По среднему числу листьев в РД: критерий GainRatio сопоставим с критерием Gain, им немного уступают критерии MDC, Gini Index и Twoing, наихудшее значение было получено при использовании критерия Dcrit.

Наилучшие результаты среди всех критериев, применяемых в алгоритме SDT, по минимальной глубине РД показали критерии Dcrit, MDC и Gain. Критерий Dcrit оказался лучше по этому показателю на 27 задачах, MDC на 18 задачах, а Gain на 17 задачах. Наихудшие результаты продемонстрировал алгоритм SDT с критерием GainRatio. По средней глубине РД: критерий MDC сопоставим с критерием Dcrit, им немного уступают критерии Gain, Gini Index и Twoing, наихудшее значение было получено при применении критерия GainRatio.

Таблица 2. Структурные свойства РД, построенного алгоритмом SDT, в зависимости от критерия выбора признака для ветвления

No	MDC					Gain					GainRatio					Gini Index					Twoing					Dcrit				
-	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r
1	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	8	4	3,1	2,8	0,3
2	4	3	2,3	1,5	0,8	6	4	2,8	2,9	0,1	6	4	3	3	0	6	4	2,8	2,9	0,1	6	4	2,8	2,9	0,1	27	7	5	4,4	0,6
3	5	4	2,8	1,5	1,3	3	2	1,7	1,3	0,4	9	6	4,2	4,6	0,4	15	6	4,5	4,1	0,4	15	6	4,5	4,1	0,4	43	8	5,7	4,9	0,8
4	53	11	6,8	4,9	1,9	52	13	7,7	6,1	1,6	45	14	9,2	8,4	0,8	55	13	7,7	6,3	1,4	55	13	7,7	6,3	1,4	56	9	6,5	4,8	1,7
5	4	3	2,3	2,2	0,1	4	3	2,3	2,2	0,1	4	3	2,3	2,6	0,3	4	3	2,3	2,2	0,1	4	3	2,3	2,2	0,1	8	4	3,1	2,9	0,2
6	13	7	4,5	3,1	1,4	12	6	4,2	3,1	1,1	16	8	6,1	6,3	0,2	15	6	4,2	3,5	0,7	15	6	4,2	3,5	0,7	19	6	4,5	3,8	0,7
7	7	5	3,4	2,4	1	6	4	3	2,4	0,6	7	5	3,6	3	0,6	6	4	3	2,4	0,6	6	4	3	2,4	0,6	37	6	5,4	5	0,4
8	11	5	3,9	2,6	1,3	11	5	3,9	2,6	1,3	11	5	3,8	3	0,8	10	5	3,6	2,9	0,7	10	5	3,6	2,9	0,7	10	5	3,8	2,4	1,4
9	66	10	6,8	5,9	0,9	60	11	6,8	5,9	0,9	49	13	10,3	11	0,7	64	12	7,1	6,2	0,9	66	11	7,2	6,6	0,6	63	8	6,3	5,5	0,8
10	47	9	6,1	4,9	1,2	47	11	6,3	5	1,3	63	13	9,4	9,5	0,1	49	12	6,4	5,7	0,7	49	12	6,4	5,7	0,7	48	8	6	4,9	1,1
11	10	4	3,4	3,2	0,2	11	5	3,6	3,2	0,4	12	8	5,3	5,5	0,2	11	5	3,7	3,9	0,2	11	5	3,7	3,9	0,2	24	6	4,8	3,9	0,9
12	13	7	4,4	2,6	1,8	13	7	4,4	2,6	1,8	19	9	5,6	4	1,6	16	7	4,6	3,5	1,1	16	7	4,6	3,5	1,1	18	7	5	2,8	2,2
13	21	7	5,1	3,9	1,2	17	10	6,1	4,6	1,5	25	15	9,7	11,4	1,7	18	10	6,8	7,5	0,7	18	10	6,8	7,5	0,7	22	6	4,7	3,9	0,8
14	20	6	4,7	4,3	0,4	21	11	6,4	5,4	1	31	18	11,4	11,8	0,4	23	11	6,5	5,9	0,6	23	11	6,5	5,9	0,6	25	5	4,7	4,5	0,2
15	13	6	4,8	3,7	1,1	13	8	5,2	3,8	1,4	12	10	6,3	6,1	0,2	13	6	4,8	3,7	1,1	13	6	4,8	3,7	1,1	14	5	4,1	3,5	0,6
16	16	7	4,8	3,8	1	13	9	5,5	5,1	0,4	18	13	8,4	9,1	0,7	14	9	5,4	5,7	0,3	14	9	5,4	5,7	0,3	18	5	4,5	3,5	1
17	25	7	5,1	4,4	0,7	24	9	5,3	4,5	0,8	25	13	8,5	7,2	1,3	25	9	5,9	4,9	1	25	9	5,9	4,9	1	27	7	5	4,5	0,5
18	38	10	5,9	4,8	1,1	37	10	6,5	5,9	0,6	43	15	9,7	9,3	0,4	42	10	6,9	6,5	0,4	42	10	6,9	6,5	0,4	72	9	6,6	5,5	1,1
19	48	10	6,2	5,2	1	52	13	7,4	6,5	0,9	61	19	11,6	10,5	1,1	54	13	7,6	7,1	0,5	54	13	7,6	7,1	0,5	73	9	6,6	5,5	1,1
20	17	5	4,8	4,5	0,3	13	5	4,4	4,4	0	9	5	4,1	4,4	0,3	13	5	4,4	4,4	0	13	5	4,4	4,4	0	24	5	4,7	4,4	0,3
21	13	6	4,3	3,2	1,1	10	6	4,2	2,9	1,3	12	7	4,9	3,9	1	11	6	4,3	3,3	1	11	6	4,3	3,3	1	13	6	4,3	3,2	1,1
22	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	13	5	3,8	3,4	0,4
23	5	4	2,8	1,5	1,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	4	2,8	1,5	1,3
24	8	4	3,1	3,1	0	7	4	3,1	3,4	0,3	10	8	5,2	5,5	0,3	9	7	4,2	4,7	0,5	9	7	4,2	4,7	0,5	8	4	3,1	3,1	0
25	22	7	4,8	4,2	0,6	20	8	5,5	5	0,5	11	8	5,5	7	1,5	21	8	5,6	5	0,6	22	8	5,7	5	0,7	26	7	5,2	4,4	0,8
26	9	4	3,3	2,9	0,4	8	4	3,1	2,9	0,2	8	5	3,5	3,4	0,1	12	5	3,8	3,6	0,2	9	4	3,3	3,4	0,1	12	5	3,8	3,2	0,6
27	31	8	5,9	3,6	2,3	30	8	5,8	3,6	2,2	33	9	6,5	7	0,5	33	9	6,2	5,4	0,8	33	9	6,2	5,4	0,8	44	8	6,3	3,9	2,4
28	111	13	7,9	6,3	1,6	104	13	9	6,9	2,1	84	14	10,2	9,4	0,8	107	12	8,6	6,8	1,8	107	12	8,6	6,8	1,8	126	11	7,6	6	1,6
29	110	8	7,3	6,6	0,7	71	8	6,9	6,8	0,1	24	8	6,1	7,9	1,8	85	8	7,1	7,3	0,2	85	8	7,1	7,3	0,2	176	8	7,6	6,9	0,7
30	51	9	6,3	6,1	0,2	44	9	6,3	6,2	0,1	30	9	6,7	7,7	1	50	9	6,9	6,7	0,2	47	9	6,3	6,4	0,1	64	8	6,3	5,6	0,7
31	26	7	5,5	4,5	1	23	13	7,4	6,2	1,2	24	17	10,8	10,4	0,4	24	8	5,5	5,3	0,2	24	8	5,5	5,3	0,2	26	7	5,1	4,2	0,9
32	10	4	3,5	3,5	0	10	4	3,4	3,4	0	10	4	3,5	3,5	0	10	6	4,2	4,1	0,1	10	4	3,4	3,4	0	10	4	3,4	3,4	0
33	81	14	8,8	6,5	2,3	74	18	9,7	6,7	3	66	26	17	11,6	5,4	75	17	9,9	6,8	3,1	73	18	9,9	6,8	3,1	132	11	7,8	6,3	1,5
34	124	9	7,9	6,1	1,8	125	9	8	6,2	1,8	123	9	8	6,2	1,8	124	9	7,9	6,2	1,7	124	9	7,9	6,2	1,7	140	9	7,9	6,4	1,5
35	36	10	6,4	5,5	0,9	33	11	6,7	5,6	1,1	35	13	8,5	7,7	0,8	35	9	5,9	5,2	0,7	38	11	6,6	5,4	1,2	34	7	5,5	4,4	1,1
36	24	11	5,6	4,3	1,3	22	12	6	4,3	1,7	18	13	7,8	5,4	2,4	22	12	6,7	4,4	2,3	21	11	5,7	4,3	1,4	30	9	5,8	3,9	1,9
Δ	30,5	6,9	4,8	3,9	1	27,9	7,8	5,1	4,2	0,9	26,8	9,8	6,7	6,4	0,8	30	7,8	5,3	4,7	0,7	29,9	7,8	5,2	4,7	0,7	41,5	6, 7	5,2	4,3	0,9

Наилучшие результаты среди всех критериев, используемых в алгоритме SDT, по минимальной средней глубине листьев РД показали критерии Dcrit, MDC и Gain. Критерий Dcrit оказался лучше по этому показателю на 16 задачах, Gain и MDC на 14 задачах. Наихудшие результаты по минимальному числу листьев в дереве показал алгоритм SDT с критерием GainRatio. По среднему средней глубины листьев РД: критерий MDC оказался лучше всех, ему немного уступают критерии Gain, Dcrit, Gini Index и Twoing, наихудшее значение было получено при синтезе дерева с помощью критерия GainRatio.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDT, по минимальной взвешенной глубине распределения описаний обучающих объектов в листьях РД показали критерии MDC, Dcrit и Gain. Критерий MDC оказался лучше по этому показателю на 20 задачах, Dcrit на 17 задачах, а Gain на 12 задачах. Наихудшие результаты продемонстрировал алгоритм SDT с критерием GainRatio. По среднему взвешенной глубины распределения описаний обучающих объектов в листьях РД: критерий MDC ока-

зался лучше всех, ему немного уступают критерии Gain и Dcrit, им в свою очередь уступают критерии Gini Index и Twoing, наихудшее значение было получено при применении критерия GainRatio.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDT, по минимальной абсолютной разницы между средней глубиной листьев и взвешенной глубиной распределения описаний обучающих объектов в листьях РД показали критерии Twoing, GainRatio и Gini Index. Критерий Twoing оказался лучше по этому показателю на 18 задачах, GainRatio и Gini Index на 15 задачах. Наихудшие результаты продемонстрировал алгоритм SDT с критерием MDC. По среднему абсолютной разницы между средней глубиной листьев и взвешенной глубиной распределения описаний обучающих объектов в листьях РД: критерий Gini Index сопоставим с критерием Twoing, им немного уступают критерии GainRatio, Gain и Dcrit, наихудшее значение было получено при синтезе дерева с помощью критерия MDC.

Результаты попарного сравнения критерия MDC с каждым из остальных критериев:

1. по сравнению с критерием Gain дерево с критерием MDC получается менее глубокое, листья концентрируются немного ближе к корню дерева и немного быстрее происходит разделение обучающих объектов, при этом «оптимальность» распределения обучающих объектов по листьям является сопоставимой, однако число листьев в дереве больше;
2. по сравнению с критериями GainRatio, Gini Index и Twoing, дерево с критерием MDC получается намного менее глубокое, листья концентрируются ближе к корню дерева и быстрее происходит разделение обучающих объектов, число листьев в дереве получается сопоставимое, однако «оптимальность» распределения обучающих объектов по листьям немного хуже;
3. по сравнению с критерием Dcrit дерево с критерием MDC получается сопоставимое по глубине, листья концентрируются примерно на той же глубине, при этом скорость разделения обучающих объектов оказывается сопоставимой, однако число листьев в дереве намного меньше.

В табл. 3 представлены значения структурных характеристик РД при использовании алгоритма SDTw с различными критериями. Обозначения в табл. 3 аналогичны обозначениям из табл. 2.

Результаты по структурным свойствам РД из табл. 3 следующие.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDTw, по минимальному числу листьев в РД показали критерии Gain, MDC и Gini Index. Критерий Gain оказался лучше по этому показателю на 28 задачах, MDC на 17 задачах, а Gini Index на 11 задачах. Наихудшие результаты продемонстрировал алгоритм SDTw с критерием Dcrit. По среднему числу листьев в РД: критерий MDC сопоставим с критериями Gain, Gini Index и Twoiing, им немного уступает критерий GainRatio, наихудшее значение было получено при использовании критерия Dcrit.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDTw, по минимальной глубине РД показали критерии Dcrit, MDC и Gain. Критерий Dcrit оказался лучше по этому показателю на 23 задачах, MDC на 17 задачах, а Gain на 14 задачах. Наихудшие результаты продемонстрировал алгоритм SDTw с критерием GainRatio. По средней глубине дерева: критерий MDC сопоставим с критерием Dcrit, им немного уступают критерии Gain, Gini Index и Twoing, наихудшее значение было получено при применении критерия GainRatio.

Таблица 3. Структурные свойства РД, построенного алгоритмом SDTw, в зависимости от критерия выбора признака для ветвления

No	MDC					Gain					GainRatio					Gini Index					Twoing					Dcrit				
-	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r	μ	k	Δ_k	Δ_o	Δ_r
1	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	3	2	1,7	1,3	0,4	12	4	3,8	3,4	0,4
2	5	3	2,6	2	0,6	6	4	2,8	2,9	0,1	6	4	3	3	0	6	4	2,8	2,9	0,1	6	4	2,8	2,9	0,1	28	6	4,9	4,3	0,6
3	5	4	2,8	1,5	1,3	3	2	1,7	1,3	0,4	17	11	6,7	6,8	0,1	15	6	4,6	4,2	0,4	15	6	4,6	4,2	0,4	40	7	5,6	4,8	0,8
4	43	10	6,4	4,8	1,6	41	11	6,8	5,9	0,9	48	28	13,6	9,3	4,3	43	11	6,9	6,2	0,7	43	11	6,9	6,2	0,7	47	8	6,1	4,7	1,4
5	4	3	2,3	2,2	0,1	4	3	2,3	2,2	0,1	4	3	2,3	2,6	0,3	4	3	2,3	2,2	0,1	4	3	2,3	2,2	0,1	8	4	3,1	2,9	0,2
6	11	5	3,9	3	0,9	11	5	3,9	3	0,9	16	9	6,2	6,3	0,1	12	5	3,8	3,2	0,6	12	5	3,8	3,2	0,6	18	6	4,4	3,7	0,7
7	6	4	3	2,4	0,6	6	4	3	2,4	0,6	7	5	3,6	3	0,6	6	4	3	2,4	0,6	6	4	3	2,4	0,6	36	7	5,4	4,9	0,5
8	9	5	3,6	2,4	1,2	9	5	3,6	2,4	1,2	10	5	3,6	2,9	0,7	9	4	3,3	2,8	0,5	9	4	3,3	2,8	0,5	9	5	3,6	2,4	1,2
9	57	8	6,3	5,7	0,6	56	11	6,8	5,9	0,9	70	32	17,4	16,7	0,7	57	10	6,6	6	0,6	58	9	6,4	6	0,4	57	7	6,1	5,4	0,7
10	46	9	6	4,8	1,2	44	10	6,1	4,9	1,2	59	15	9,9	10,7	0,8	46	9	6,1	5,6	0,5	46	9	6,1	5,6	0,5	45	8	5,9	4,8	1,1
11	10	4	3,4	3,2	0,2	10	4	3,4	3,2	0,2	11	7	5	5,5	0,5	10	5	3,6	3,9	0,3	10	5	3,6	3,9	0,3	24	6	4,8	3,9	0,9
12	13	7	4,4	2,6	1,8	13	7	4,4	2,6	1,8	16	7	4,9	3,8	1,1	15	6	4,4	3,4	1	15	6	4,4	3,4	1	15	6	4,5	2,7	1,8
13	21	7	5,1	3,9	1,2	16	9	5,8	4,6	1,2	25	17	9,9	10,9	1	20	11	7,3	8,1	0,8	20	11	7,3	8,1	0,8	20	6	4,6	3,8	0,8
14	20	6	4,7	4,3	0,4	22	10	6,4	5,6	0,8	30	25	13,9	14,4	0,5	21	11	6,4	5,9	0,5	21	11	6,4	5,9	0,5	25	6	4,8	4,5	0,3
15	13	6	4,8	3,7	1,1	12	8	4,9	3,8	1,1	12	10	6,3	6,1	0,2	13	6	4,8	3,7	1,1	13	6	4,8	3,7	1,1	14	5	4,1	3,4	0,7
16	16	7	4,8	3,8	1	13	9	5,5	5,1	0,4	18	13	8,5	9,2	0,7	15	9	5,8	6,3	0,5	15	9	5,8	6,3	0,5	18	5	4,5	3,5	1
17	22	6	4,7	4,1	0,6	21	6	4,7	4,1	0,6	28	22	12,6	11,6	1	23	9	5,7	4,8	0,9	23	9	5,7	4,8	0,9	25	6	4,8	4,2	0,6
18	31	8	5,3	4,6	0,7	34	9	5,9	5,6	0,3	38	22	11,8	11,7	0,1	34	10	6,1	6,1	0	34	10	6,1	6,1	0	55	8	6,1	5,1	1
19	39	7	5,6	5	0,6	40	11	6,6	6,1	0,5	43	25	13	12,3	0,7	40	10	6,6	6,7	0,1	40	10	6,6	6,7	0,1	57	8	6,2	5,2	1
20	22	7	5,4	4,9	0,5	22	10	6	5,1	0,9	29	20	10,3	9,5	0,8	26	9	6,3	5,3	1	26	9	6,3	5,3	1	28	7	5,3	4,4	0,9
21	11	6	3,9	3,1	0,8	8	5	3,6	2,9	0,7	9	6	4,2	3,8	0,4	9	5	3,8	3,2	0,6	9	5	3,8	3,2	0,6	12	6	4,2	3,1	1,1
22	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	2	1	1	1	0	15	5	4,1	3,8	0,3
23	5	4	2,8	1,5	1,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	3	2,4	2,1	0,3	5	4	2,8	1,5	1,3
24	8	4	3,1	3,1	0	7	4	3,1	3,4	0,3	10	8	5,2	5,4	0,2	9	7	4,2	4,7	0,5	9	7	4,2	4,7	0,5	8	4	3,1	3,1	0
25	22	8	5	4,2	0,8	19	9	5,4	4,6	0,8	24	21	11,5	11,6	0,1	19	9	5,4	4,6	0,8	19	9	5,4	4,6	0,8	23	7	4,9	4,3	0,6
26	9	4	3,3	2,9	0,4	8	4	3,1	2,9	0,2	8	5	3,5	3,4	0,1	12	5	3,8	3,6	0,2	9	4	3,3	3,4	0,1	12	5	3,8	3,2	0,6
27	29	8	5,6	3,5	2,1	32	8	5,8	3,6	2,2	41	10	6,7	6,5	0,2	38	9	6,5	5,4	1,1	38	9	6,5	5,4	1,1	36	8	5,8	3,8	2
28	87	11	7,1	5,8	1,3	82	13	8,3	6,6	1,7	100	43	15,9	11,8	4,1	88	12	8	6,7	1,3	88	12	8	6,7	1,3	97	10	7	5,8	1,2
29	139	11	8	6,7	1,3	133	16	9	7,5	1,5	178	101	40,3	31,8	8,5	127	13	8,4	7,4	1	128	13	8,5	7,4	1,1	156	10	7,6	6,6	1
30	45	10	6,1	5,7	0,4	42	10	6,1	5,7	0,4	45	18	9,9	10,7	0,8	50	12	7,2	7	0,2	46	11	6,4	6,1	0,3	55	8	6,1	5,4	0,7
31	25	7	5,4	4,2	1,2	21	8	6,2	5,6	0,6	25	18	11	11,5	0,5	23	8	5,5	5,2	0,3	23	8	5,5	5,2	0,3	25	7	5,2	4	1,2
32	10	4	3,5	3,5	0	10	4	3,4	3,4	0	10	4	3,5	3,5	0	10	6	4,2	4,1	0,1	10	4	3,4	3,4	0	10	4	3,4	3,4	0
33	70	14	8,1	6,2	1,9	60	15	8,6	5,9	2,7	60	26	15,4	9,9	5,5	65	16	8,9	6,4	2,5	63	16	8,8	6,2	2,6	125	10	7,7	6,1	1,6
34	95	11	7,9	5,6	2,3	101	13	8,3	5,8	2,5	131	13	8,9	6,3	2,6	104	13	8,3	5,8	2,5	104	13	8,3	5,8	2,5	131	10	7,8	6,4	1,4
35	29	8	5,8	5,2	0,6	27	9	5,7	5,2	0,5	38	18	10,1	9,2	0,9	30	8	5,6	5,3	0,3	32	9	5,9	5,4	0,5	31	7	5,3	4,3	1
36	24	11	5,6	4,3	1,3	22	12	6	4,3	1,7	18	13	7,8	5,4	2,4	22	12	6,7	4,4	2,3	22	12	6	4,3	1,7	29	9	5,7	3,9	1,8
Δ	27,9	6,7	4,7	3,8	0,9	26,8	7,6	5	4,1	0,9	33,2	16,4	8,9	8,1	1,1	28,6	7,9	5,2	4,7	0,7	28,5	7,8	5,1	4,6	0,7	37,5	6,6	5,1	4,2	0,9

Наилучшие результаты среди всех критериев, используемых в алгоритме SDTw, по минимальной средней глубине листьев РД показали критерии Dcrit, MDC и Gain. Критерий Dcrit оказался лучше по этому показателю на 17 задачах, MDC на 15 задачах, а Gain на 14 задачах. Наихудшие результаты продемонстрировал алгоритм SDTw с критерием GainRatio. По среднему средней глубины листьев РД: критерий MDC оказался лучше всех, ему немного уступают критерии Gain, Dcrit, Gini Index и Twoing, наихудшее значение было получено при использовании критерия GainRatio.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDTw, по минимальной взвешенной глубине распределения описаний обучающих объектов в листьях РД показали критерии MDC, Dcrit и Gain. Критерий MDC оказался лучше по этому показателю на 21 задаче, Dcrit на 17 задачах, а Gain на 14 задачах. Наихудшие результаты продемонстрировал алгоритм SDTw с критерием GainRatio. По среднему взвешенной глубины распределения описаний обучающих объектов в листьях РД: критерий MDC

оказался лучше всех, ему немного уступают критерии Gain и Dcrit, им в свою очередь уступают критерии Gini Index и Twoing, наихудшее значение было получено при синтезе дерева с помощью критерия GainRatio.

Наилучшие результаты среди всех критериев, используемых в алгоритме SDTw, по минимальной абсолютной разницы между средней глубиной листьев и взвешенной глубиной распределения описаний обучающих объектов в листьях РД показали критерии Twoing, GainRatio и Gini Index. Критерий Twoing и Gini Index оказались лучшими по этому показателю на 14 задачах, а критерий GainRatio на 13 задачах. Наихудшие результаты продемонстрировал алгоритм SDTw с критерием MDC. По среднему абсолютной разницы между средней глубиной листьев и взвешенной глубиной распределения описаний обучающих объектов в листьях РД: критерий Gini Index сопоставим с критерием Twoing, им немного уступают критерии MDC, Gain и Dcrit, наихудшее значение было получено при применении критерия GainRatio.

Результаты попарного сравнения критерия MDC с каждым из остальных критериев:

1. по сравнению с критерием Gain дерево с критерием MDC получается менее глубокое, листья концентрируются ближе к корню дерева, быстрее происходит разделение обучающих объектов, при этом «оптимальность» распределения обучающих объектов по листьям оказывается сопоставимой, однако число листьев в дереве немного больше;
2. по сравнению с критериями GainRatio, Gini Index и Twoing, дерево с критерием MDC получается менее глубокое, листья концентрируются ближе к корню дерева и быстрее происходит разделение обучающих объектов, число листьев в дереве получается сопоставимое, однако «оптимальность» распределения обучающих объектов по листьям немного хуже;
3. по сравнению с критерием Dcrit дерево с критерием MDC получается сопоставимое по глубине, листья концентрируются примерно на той же глубине, скорость разделения обучающих объектов оказывается сопоставимой, однако число листьев в дереве намного меньше.

При переходе от алгоритма SDT к алгоритму SDTw структурные свойства РД с критериями Gain, Gini, Twoing, Dcrit и MDC практически не изменились, а для критерия GainRatio значения по всем структурным свойствам РД увеличились. Данный факт говорит о том, что при использовании алгоритма SDT с критерием GainRatio дальнейший рост дерева был сдержан особенностями алгоритма. Тем самым это показывает то, что критерий GainRatio склонен к построению дерева с большой глубиной, при этом построение листьев и разделение объектов происходит постепенно и не концентрируется в одной области дерева.

В табл. 4 представлены значения «сбалансированности» РД (разница между глубиной и средней глубиной листьев РД) при использовании алгоритма SDT и алгоритма SDTw с различными критериями ветвления. Последняя строка в табл. 4 обозначена как Δ — среднее значение «сбалансированности» РД по всем задачам для соответствующего критерия. Для каждой задачи зеленым цветом выделены значения наилучшей сбалансированности РД, построенного с помощью алгоритма SDT, а коричевым цветом — показатели наилучшей сбалансированности РД, построенного алгоритмом SDTw.

Наиболее сбалансированные РД, построенные с помощью алгоритма SDT, были получены при применении критериев: Dcrit и MDC. Деревья с критерием Dcrit оказались более сбалансированными на 22 задачах по сравнению с другими критериями, а с критерием MDC на 13 задачах. По среднему значению сбалансированности РД: критерий Dcrit

Таблица 4. Характеристика сбалансированности РД в зависимости от критерия выбора признака для ветвления

No	MDC		Gain		GainRatio		Gini Index		Twoing		Dcrit	
-	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw	SDT	SDTw
1	0,3	0,3	0,3	0,3	0,3	0,3	0,3	0,3	0,3	0,3	0,9	0,2
2	0,7	0,4	1,2	1,2	1	1	1,2	1,2	1,2	1,2	2	1,1
3	1,2	1,2	0,3	0,3	1,8	4,3	1,5	1,4	1,5	1,4	2,3	1,4
4	4,2	3,6	5,3	4,2	4,8	14,4	5,3	4,1	5,3	4,1	2,5	1,9
5	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,9	0,9
6	2,5	1,1	1,8	1,1	1,9	2,8	1,8	1,2	1,8	1,2	1,5	1,6
7	1,6	1	1	1	1,4	1,4	1	1	1	1	0,6	1,6
8	1,1	1,4	1,1	1,4	1,2	1,4	1,4	0,7	1,4	0,7	1,2	1,4
9	3,2	1,7	4,2	4,2	2,7	14,6	4,9	3,4	3,8	2,6	1,7	0,9
10	2,9	3	4,7	3,9	3,6	5,1	5,6	2,9	5,6	2,9	2	2,1
11	0,6	0,6	1,4	0,6	2,7	2	1,3	1,4	1,3	1,4	1,2	1,2
12	2,6	2,6	2,6	2,6	3,4	2,1	2,4	1,6	2,4	1,6	2	1,5
13	1,9	1,9	3,9	3,2	5,3	7,1	3,2	3,7	3,2	3,7	1,3	1,4
14	1,3	1,3	4,6	3,6	6,6	11,1	4,5	4,6	4,5	4,6	0,3	1,2
15	1,2	1,2	2,8	3,1	3,7	3,7	1,2	1,2	1,2	1,2	0,9	0,9
16	2,2	2,2	3,5	3,5	4,6	4,5	3,6	3,2	3,6	3,2	0,5	0,5
17	1,9	1,3	3,7	1,3	4,5	9,4	3,1	3,3	3,1	3,3	2	1,2
18	4,1	2,7	3,5	3,1	5,3	10,2	3,1	3,9	3,1	3,9	2,4	1,9
19	3,8	1,4	5,6	4,4	7,4	12	5,4	3,4	5,4	3,4	2,4	1,8
20	0,2	1,6	0,6	4	0,9	9,7	0,6	2,7	0,6	2,7	0,3	1,7
21	1,7	2,1	1,8	1,4	2,1	1,8	1,7	1,2	1,7	1,2	1,7	1,8
22	0	0	0	0	0	0	0	0	0	0	1,2	0,9
23	1,2	1,2	0,6	0,6	0,6	0,6	0,6	0,6	0,6	0,6	1,2	1,2
24	0,9	0,9	0,9	0,9	2,8	2,8	2,8	2,8	2,8	2,8	0,9	0,9
25	2,2	3	2,5	3,6	2,5	9,5	2,4	3,6	2,3	3,6	1,8	2,1
26	0,7	0,7	0,9	0,9	1,5	1,5	1,2	1,2	0,7	0,7	1,2	1,2
27	2,1	2,4	2,2	2,2	2,5	3,3	2,8	2,5	2,8	2,5	1,7	2,2
28	5,1	3,9	4	4,7	3,8	27,1	3,4	4	3,4	4	3,4	3
29	0,7	3	1,1	7	1,9	60,7	0,9	4,6	0,9	4,5	0,4	2,4
30	2,7	3,9	2,7	3,9	2,3	8,1	2,1	4,8	2,7	4,6	1,7	1,9
31	1,5	1,6	5,6	1,8	6,2	7	2,5	2,5	2,5	2,5	1,9	1,8
32	0,5	0,5	0,6	0,6	0,5	0,5	1,8	1,8	0,6	0,6	0,6	0,6
33	5,2	5,9	8,3	6,4	9	10,6	7,1	7,1	8,1	7,2	3,2	2,3
34	1,1	3,1	1	4,7	1	4,1	1,1	4,7	1,1	4,7	1,1	2,2
35	3,6	2,2	4,3	3,3	4,5	7,9	3,1	2,4	4,4	3,1	1,5	1,7
36	5,4	5,4	6	6	5,2	5,2	5,3	5,3	5,3	6	3,2	3,3
Δ	2	2	2,6	2,7	3,1	7,5	2,5	2,6	2,5	2,6	1,5	1,6

оказался лучше всех, ему уступает критерий MDC, им в свою очередь уступают критерии Gain, Twoing и Gini Index, наихудшее значение было получено при применении критерия GainRatio. Аналогичные результаты наблюдаются и при синтезе РД с помощью алгоритма SDTw. Однако для критерия GainRatio сбалансированность РД существенно ухудшилась при применении алгоритма SDTw по сравнению с алгоритмом SDT.

Заключение

В данной работе разработан новый критерий ветвления — критерий максимизации доли объектов различных классов (Maximum Differences of Classes (MDC)). Проведено ис-

следование критерия MDC в сравнении с критериями: Gini Index, Twoing, Gain, GainRatio и критерий равномерного разбиения (Dcrit).

На модельных задачах проведен анализ особенностей указанных критериев и показано, что:

1. Критерий MDC разделяет объекты так, чтобы максимизировать разницу долей объектов между классами в результирующих подмножествах, при этом учитывается число объектов соответствующего класса из исходного множества обучающих объектов.
2. Критерий Dcrit разделяет объекты так, чтобы в результирующих подмножествах оказалось наибольшее число объектов из разных классов, несмотря даже на то, что в одном подмножестве могут быть объекты из разных классов.
3. На большинстве модельных данных критерии Gain, GainRatio, Gini Index и Twoing близки между собой в оценке наиболее «информативного» разделения исходного множества. Отличие критериев наиболее заметно в ситуации неравномерного распределения обучающих объектов по трем классам и в ситуации наличия объектов из более, чем трех классов.
4. Критерий Gain нацелен на отделение целиком класса с наибольшим числом объектов и в случае большого числа классов позволяет определить более оптимальное разделение исходного множества.
5. Критерий GainRatio сглаживает значение критерия Gain, оценивая мощность результирующих подмножеств относительно мощности исходного множества объектов. Так же, как и критерий Gain, критерий GainRatio ориентирован на отделение класса целиком, но не учитывает его мощности, что приводит к тому, что можно пропустить отделение большого класса целиком и отделение небольшого класса целиком с небольшой долей объектов из других классов, а в случае большого числа классов можно пропустить более оптимальное разделение исходного множества.
6. Критерии Gini Index и Twoing направлены на отделение целиком класса с наибольшим числом объектов, а наличие в результирующем подмножестве небольшого числа объектов из других классов может сильно влиять на отделение целиком класса с небольшим числом объектов. Критерий Twoing в случае большого числа классов позволяет также определить более оптимальное разделение исходного множества.

На реальных задачах проведен анализ качества и структурных свойств РД при использовании различных критериев и показано, что:

1. по сравнению с критерием Gain дерево с критерием MDC получается более сбалансированное, менее глубокое, листья концентрируются ближе к корню дерева, быстрее происходит разделение обучающих объектов, при этом «оптимальность» распределения обучающих объектов по листьям дерева и среднее качество РД по всем задачам оказываются сопоставимыми, однако число листьев в дереве немного больше и немного хуже распределены отступы обучающих объектов;
2. по сравнению с критериями GainRatio, Gini Index и Twoing дерево с критерием MDC получается более сбалансированное, менее глубокое, листья концентрируются ближе к корню дерева, быстрее происходит разделение обучающих объектов, число листьев в дереве и среднее качество РД по всем задачам оказываются сопоставимыми, однако «оптимальность» распределения обучающих объектов по листьям дерева немного хуже и немного хуже распределены отступы обучающих объектов;
3. по сравнению с критерием Dcrit дерево с критерием MDC получается менее сбалансированное, сопоставимое по глубине, по средней глубине, по скорости разделения обу-

чающих объектов и по распределению отступов обучающих объектов, однако число листьев в дереве меньше и выше среднее качество РД по всем задачам.

В работе были также рассмотрены два способа синтеза РД: с удалением просмотренного признака (алгоритм SDT) и без удаления просмотренного признака (алгоритм SDTw). При переходе от алгоритма SDT к алгоритму SDTw произошли следующие изменения: структурные свойства РД с критериями Gain, Gini Index, Twoing, Dcrit и MDC практически не изменились, а для критерия GainRatio значения по всем характеристикам увеличились; распределение отступов обучающих объектов для всех критериев немного улучшилось; среднее качество РД по всем задачам с критериями MDC и Twoing практически не изменилось, с критериями Dcrit и Gini Index немного возросло, а с критериями Gain и GainRatio немного снизилось. Тот факт, что для критерия GainRatio значения по всем структурным свойствам увеличились показывает то, что применение критерия GainRatio ведет к построению глубокого и несбалансированного дерева, при этом описания обучающих объектов, «попадающих» в листья дерева, не концентрируются в одной области дерева.

Таким образом, применение нового критерия MDC позволяет получить сопоставимое по качеству и более оптимальное по структуре РД по сравнению с РД, построенного при использовании таких критериев, как: Gini Index, Twoing, Gain, GainRatio и критерий равномерного разбиения.

Литература

[1] *Дюличева Ю. Ю.* Модели коррекции редуцированных бинарных решающих деревьев: Дис. ... канд. физ.-мат. наук: 01.05.01. Симферополь: Таврический национальный ун-т им. В. И. Вернадского, 2004. 127 с.

[2] *Hyafil L., Rivest R. L.* Constructing optimal binary decision trees is NP-complete // *Inform. Process. Lett.* 1976. Vol. 5. No. 1. P. 15–17.

[3] *Донской В. И., Башта А. И.* Дискретные модели принятия решений при неполной информации. Симферополь: Таврия, 1992. 166 с.

[4] *Murthy S. K.* On growing better decision trees from data: Baltimore, MD: Johns Hopkins University, 1997. Ph.D. Dissertation. 198 p.

[5] *Anyanwu M., Shiva S.* Comparative analysis of serial decision tree classification algorithms // *Int. J. Comput. Sci. Security*, 2009. Vol. 3, no. 3. P. 230–240.

[6] *Kuncheva L. I.* Combining pattern classifiers methods and algorithms. Hoboken, NJ: John Wiley & Sons, Inc., 2004. 350 p.

[7] *Aha D. W., Breslow L. A.* Simplifying decision trees: A survey // *The Knowledge Engineering*, 1997. Vol. 12. no. 1. P. 1–40.

[8] *Quinlan J. R.* Induction of decision trees // *Mach. Learn.*, 1986. Vol. 1. no. 1. P. 81–106.

[9] *Quinlan J. R.* C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann. 1993. 302 p.

[10] *Breiman L., Friedman J. H., Olshen R. A., Stone C. J.* Classification and regression trees. Belmont, CA, USA: Wadsworth Publishing Co., 1984. 368 p.

[11] *Breiman L.* Technical note: Some properties of splitting criteria // *Mach. Learn.*, 1996. No. 24. P. 41–47.

[12] *Donskoy V. I.* Splitting criteria, binary decision tree synthesis, and algorithm LISTBB // *Intellectual Archive*. 2013. No. 1058. 25 p.

[13] *Kohavi R., Kunz C.* Option decision trees with majority votes // *Conference (International) on Machine Learning Proceedings*, 1997. P. 161–169.

[14] *Michie D., Spiegelhalter D. J., Taylor C. C.* Machine learning, neural and statistical classification. London: Ellis Horwood, 1994. 298 p.

[15] *Martin J. K.* An exact probability metric for decision tree splitting and stopping // *Mach. Learn.*, 1997. Vol. 28, no. 2-3. P. 257–291.

[16] *Payne H. J., Meisel W. S.* An algorithm for constructing optimal binary decision trees. // *IEEE Trans. Comput.*, 1977. Vol. C-26, no. 9. P. 905–916.

[17] *Genrikhov I. E., Djukova E. V.* Classification based on full decision trees // *Comp. Math. Math. Phys.*, 2012. Vol. 52, No. 4. P. 750–761.

[18] *Дюкова Е. В., Карпаева И. Л.* Модели распознающих алгоритмов, основанные на различных способах перекодировки исходной информации // *Математические методы в распознавании образов и дискретной оптимизации.* М.: ВЦ РАН, 1990. С. 43–56.

[19] *Дюкова Е. В., Журавлев Ю. И., Песков Н. В., Сахаров А. А.* Обработка вещественнозначной информации логическими процедурами распознавания // *Искусственный интеллект*, 2004. №. 2, С. 80–85.

[20] *Fomina M., Kulikov A., Vagin V.* The development of the generalization algorithm based on the rough set theory // *KDS 2005 — Data Mining and Knowledge Discovery, Actual Problems of Data Mining Proceedings*, 2006. P. 76–84.

[21] *Kotsiantis S., Kanellopoulos D.* Discretization techniques: A recent survey // *GESTS Int. Trans. Comput. Sci. Eng.*, 2006. Vol. 32(1). P. 47–58.

[22] *Han J., Kamber M.* Data mining: Concepts and techniques. Morgan Kaufmann, 2006. 743 p.

[23] *Genrikhov I. E.* Analysis of the generalization ability of a full decision tree // *Comp. Math. Math. Phys.*, 2014. Vol. 54. no. 6. P. 1033–1047.

[24] *Marlin B. M.* Missing data problems in machine learning. Ph.D. Thesis. Department of Computer Science, University of Toronto, 2008. 156 p.

[25] *Peng L., Lei L.* A review of missing data treatment methods // *Int. J. Intelligent Information Management Syst. Technol.*, 2005. Vol. 1, no. 3. P. 412–419.

[26] *Schapire R. E., Freund Y., Lee W. S., Bartlett P. L.* Boosting the margin: A new explanation for the effectiveness of voting methods // *Ann. Stat.*, 1998. Vol. 26. no. 5. P. 1651–1686.

[27] *Журавлев Ю. И., Рязанов В. В., Сенько О. В.* Распознавание. Математические методы. Программная система. Практические применения. — М.: ФАЗИС, 2006. 176 с.

[28] *Asuncion A., Newman D.* 2007. UCI machine learning repository. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

[29] *Mason L.* Margins and combined classifiers. Ph.D. Thesis. The Australian National University, 1999. 118 p.

[30] *Golea M., Bartlett P. L., Lee W. S., Mason L.* Generalization in decision trees and DNF: Does size matter? // *Adv. Neur. Inform. Process. Syst.*, 1998. Vol. 10. P. 259–265.

[31] *Донской В. И.* Оценки емкости классов алгоритмов эмпирического обобщения, полученные pVCD методом // *Ученые записки Таврического национального университета им. В. И. Вернадского*, 2010. Т. 23(62). №. 2. С. 56–65.

[32] *Djukova E. V., Peskov N. V.* A classification algorithm based on the complete decision tree // *Pattern Recognition Image Anal.*, 2007. Vol. 17, №. 3. Pp. 363–367.

[33] *Breiman L.* Random forests // *Mach. Learn.*. 2001. Vol. 45, No. 1. P. 5–32.

[34] *Freund Y., Schapire R. E.* A decision-theoretic generalization of on-line learning and an application to boosting // *Conference (European) on Computational Learning Theory Proceedings*, 1995. P. 23–37.

References

[1] *Dulicheva Yu. Yu.* 2004. The models of correction a reduced binary decision trees. Simferopol: Taurida National V. I. Vernadsky University. Ph.D. Dissertation. 127 p.

[2] *Hyafil L., Rivest R. L.* 1976. Constructing optimal binary decision trees is NP-complete. *Inform. Process. Lett.* 5(1):15–17.

[3] *Donskoy V. I., Bashta A. I.* 1992. Discrete models of decision-making with incomplete information. Simferopol: Tavriia. 166 p.

[4] *Murthy S. K.* 1997. On growing better decision trees from data. Baltimore, MD: Johns Hopkins University. Ph.D. Dissertation. 198 p.

[5] *Anyanwu M., Shiva S.* 2009. Comparative analysis of serial decision tree classification algorithms. *Int. J. Comput. Sci. Security.* 3(3):230–240.

[6] *Kuncheva L. I.* 2004. Combining pattern classifiers methods and algorithms. Hoboken, NJ: John Wiley & Sons, Inc. 350 p.

[7] *Aha D. W., Breslow L. A.* 1997. Simplifying decision trees: A survey. *The Knowledge Engineering* 12(1):1–40.

[8] *Quinlan J. R.* 1986. Induction of decision trees. *Mach. Learn.* 1(1):81–106.

[9] *Quinlan J. R.* 1993. C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann. 302 p.

[10] *Breiman L., Friedman J. H., Olshen R. A., Stone C. J.* 1984. Classification and regression trees. Belmont, CA, USA: Wadsworth Publishing Co. 368 p.

[11] *Breiman L.* 1996. Technical note: Some properties of splitting criteria. *Mach. Learn.* 24:41—47.

[12] *Donskoy V. I.* 2013. Splitting criteria, binary decision tree synthesis, and algorithm LISTBB. *Intellectual Archive.* 1058. 25 p.

[13] *Kohavi R., Kunz C.* 1997. Option decision trees with majority votes. *Conference (International) on Machine Learning Proceedings.* 161–169.

[14] *Michie D., Spiegelhalter D. J., Taylor C. C.* 1994. Machine learning, neural and statistical classification. London: Ellis Horwood. 298 p.

[15] *Martin J. K.* 1997. An exact probability metric for decision tree splitting and stopping. *Mach. Learn.* 28(2-3):257–291.

[16] *Payne H. J., Meisel W. S.* 1977. An algorithm for constructing optimal binary decision trees. *IEEE Trans. Comput.* C-26(9):905–916.

[17] *Genrikhov I. E., Djukova E. V.* 2012. Classification based on full decision trees. *Comp. Math. Math. Phys.* 52(4):750–761.

[18] *Djukova E. V., Carnaeva I. L.* 1990. Model recognition algorithms based on different ways transcoding source information. *Mathematical methods in pattern recognition and discrete optimization.* Moscow: CC RAS. 43–56.

[19] *Djukova E. V., Zhuravlev Yu. I., Peskov N. V., Saharov A. A.* 2004. Processing a real-valued information with logical recognition procedures. *Artificial Intelligence* 2:80–85.

[20] *Fomina M., Kulikov A., Vagin V.* 2006. The development of the generalization algorithm based on the rough set theory. *KDS 2005 — Data Mining and Knowledge Discovery, Actual Problems of Data Mining Proceedings.* 76–84.

[21] *Kotsiantis S., Kanellopoulos D.* 2006. Discretization techniques: a recent survey. *GESTS Int. Trans. Comput. Sci. Eng.* 32(1):47–58.

[22] *Han J., Kamber M.* 2006. Data mining: Concepts and techniques. Morgan Kaufmann. 743 p.

[23] *Djukova E. V., Peskov N. V.* 2007. A classification algorithm based on the complete decision tree. *Pattern Recognition Image Anal.* 17(3):363–367.

[24] *Genrikhov I.E.*, 2014 Analysis of the generalization ability of a full decision tree. *Comp. Math. Math. Phys.* 54(6):1033-–1047.

[25] *Marlin B. M.* 2008. Missing data problems in machine learning. Department of Computer Science, University of Toronto. PhD. Thesis. 156 p.

[26] *Peng L., Lei L.* 2005. A review of missing data treatment methods. *Int. J. Intelligent Information Management Syst. Technol.* 1(3):412–419.

[27] *Schapire R. E., Freund Y., Lee W. S., Bartlett P. L.* 1998. Boosting the margin: a new explanation for the effectiveness of voting methods. *Ann. Stat.* 6(5):1651–1686.

[28] *Zhuravlev Yu. I., Ryazanov V. V., Senko O. V.*, 2006. Recognition. Mathematical methods. Software system. Practical applications. Moscow: Phasis. 176 p.

[29] *Asuncion A., Newman D.* 2007. UCI machine learning repository. `http://www.ics.uci.edu/~mlearn/MLRepository.html` (accessed September 10, 2914).

[30] *Mason L.* 1999. Margins and combined classifiers. The Australian National University. Ph.D. Thesis. 118 p.

[31] *Golea M., Bartlett P. L., Lee W. S., Mason L.* 1998. Generalization in decision trees and DNF: Does size matter? *Adv. Neur. Inform. Process. Syst.* 10:259–265.

[32] *Donskoy V. I.* 2010. VC-dimension estimations of the basic algorithms of empirical generalization for pattern recognition problems obtained by the pVCD. *Proceedings Tauride National University named after V. I. Vernadsky* 23(2):56–65.

[33] *Breiman L.* 2001. Random forests. *Mach. Learn.* 45(1):5–32.

[34] *Freund Y., Schapire R. E.* 1995. A decision-theoretic generalization of on-line learning and an application to boosting. *Conference (European) on Computational Learning Theory.* 23–37.