РОССИЙСКАЯ АКАДЕМИЯ НАУК
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР

СООБЩЕНИЯ ПО ПРИКЛАДНОЙ МАТЕМАТИКЕ

Е.В. ДЮКОВА, А.С. ИНЯКИН

ЗАДАЧА ТАКСОНОМИИ И ТУПИКОВЫЕ ПОКРЫТИЯ ЦЕЛОЧИСЛЕННОЙ МАТРИЦЫ

ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР РАН
МОСКВА 2001

УДК 519.714

Ответственный редактор
член-корр. РАН К.В. Рудаков

В статье изучаются вопросы применения методов кластерного анализа для работы с множествами данных, на которых достаточно сложно определить функцию расстояния. Предложен подход к решению задачи кластеризации с целочисленной информацией, основанный на построении специальных наборов признаков, не содержащихся в признаковых описаниях объектов. Задача сводится к построению тупиковых покрытий целочисленных матриц.

Рецензенты: С.А. Жданов,
К.В. Воронцов

Научное издание
© Вычислительный центр РАН, 2001.

Введение[1]

В самых различных сферах деятельности и различных областях науки - медицине, астрономии, геологии, при проведении социологических и экономических исследований - зачастую необходимо разбить совокупность объектов на "однородные" группы (классы). Особенность заключается в отсутствии обучающей информации, поэтому полезность получаемого решения напрямую зависит от введенного при постановке задачи понятия "однородность" объектов, или, более точно, "мера близости" (мера подобия) объектов. Задачи такого типа называют задачами таксономии.

Пусть имеется выборка $\Omega = S_1, S_2, \ldots, S_m$, каждый элемент которой описывается набором из n признаков (свойств), $S_i=(x_{i1}, x_{i2}, \ldots, x_{in}), i=1, 2, \ldots, m$. Требуется разбить данную выборку на однородные группы (классы), число которых может быть заранее заданным, а может быть и неизвестным. Процесс деления на классы называют кластеризацией или кластерным анализом.

[1] Работа выполнена при частичной поддержке Российского Фонда Фундаментальных Исследований (коды проектов 01-01-00575, 00-15-96064) и Программы Президиума РАН «Интеллектуальные компьютерные системы», проект №1.1.

При постановке задачи необходимо описать правила, которым должны подчиняться объекты из одного класса и объекты из разных классов. В качестве такого правила можно рассматривать компактность объектов в заданном признаковом пространстве, т.е. правило, в соответствии с которым, "расстояние" между объектами, отнесенными в один класс, должно быть, например, не более заданного. Может также использоваться "хорошая" взаимная удаленность классов; например, "расстояние" между объектами, отнесенными в разные классы, должно быть не менее заданного.

В данной работе рассматриваются некоторые подходы к решению задач таксономии или кластеризации с целочисленной информацией. Отличительной особенностью предлагаемого метода от классических является то, что при решении указанных задач не нужно подбирать функцию расстояния, которая не всегда может быть достаточно простой для вычисления. В его основе лежат следующие соображения.

Рассмотрим ситуацию, когда определяется степень принадлежности объекта S к группе объектов M. Если описание объекта S содержит набор значений признаков, который не присутствует в описании ни у одного объекта из

M, то можно сказать, что объединение *S* и *M* нарушает внутреннюю структуру множества *M*. Рассматривая различные комбинации значений признаков, не содержащихся в описаниях объектов из *M*, можно оценить близость объекта *S* к множеству *M*. Таким образом, при определении степени близости объекта к множеству *M* информативными считаются такие наборы значений признаков, которые отсутствуют в описаниях всех объектов из класса *M*.

На основе приведенных выше рассуждений был построен алгоритм, который в ряде случаев позволил значительно улучшить результат кластеризации на модельных и реальных задачах по сравнению с такими известными алгоритмами, как алгоритмы, основанные на принципах «ближайший сосед», «дальний сосед» и выбор центрального элемента, в которых в качестве функции расстояния используется расстояние Хемминга.

В предложенном алгоритме кластеризации задача оценки близости объекта *S* к множеству *M* сводится к поиску так называемых тупиковых σ-покрытий целочисленной матрицы, образованной признаковыми описаниями объектов из *M*. Данное понятие является обобщением хорошо известного в дискретной математике понятия неприводимого

покрытия булевой матрицы и первоначально было введено в [1, 2] в связи с задачами построения тупиковых представительных наборов в алгоритмах типа "Кора" [3, 4] и изучением метрических свойств множества представительных наборов. Наиболее полно дискретный подход к решению задач распознавания описан в [5, 6].

1. Основные определения. Алгоритм поиска тупиковых σ-покрытий целочисленной матрицы

Введем следующие обозначения:

M_{mn}^{k}, $k \geq 2$, - множество всех матриц размера $m \times n$ с элементами из $\{0,1,\ldots,k-1\}$;

E_k^r, $k \geq 2$, - множество всех k-х наборов длины r;

$Q_p(\sigma)$, $p \in \{1, 2, \ldots, r\}$, $\sigma \in E_k^r$, $\sigma = (\sigma_1, \ldots, \sigma_r)$, - множество всех таких наборов $(\beta_1, \ldots, \beta_r)$ в E_k^r, для которых $\beta_p \neq \sigma_p$ и $\beta_j = \sigma_j$ при $j \in \{1, 2, \ldots, r\} \setminus \{p\}$.

Пусть $L \in M_{mn}^{k}$, $\sigma \in E_k^r$, $\sigma = (\sigma_1, \ldots, \sigma_r)$.

Набор H из r различных столбцов матрицы L назовем σ-покрытием, если подматрица L^H матрицы L, образованная столбцами набора H, не содержит строку $(\sigma_1, \ldots, \sigma_r)$.

Набор столбцов H, являющийся σ-покрытием матрицы L, назовем тупиковым σ-покрытием, если для любого $p \in \{1, 2, \ldots, r\}$ подматрица L^H содержит хотя бы одну строку из $Q_p(\sigma)$. В случае $k = 2$ и $\sigma = (0, ..., 0)$ понятие тупикового σ-покрытия совпадает с хорошо известным понятием неприводимого покрытия булевой матрицы [7].

Обозначим $B(L,\sigma)$ множество всех тупиковых σ-покрытий матрицы L. Пусть

$$B(L) = \bigcup_{r=1}^{n} \bigcup_{\sigma \in E_k^r} B(L,\sigma).$$

Как известно, задача поиска неприводимых покрытий булевой матрицы является трудной в вычислительном плане задачей и, как следствие, задача построения тупиковых σ-покрытий не менее сложна. В [1, 2] показано, что если число столбцов n бинарной матрицы существенно превосходит число ее строк m, то при $n \to \infty$ почти всегда (для почти всех матриц размера $m \times n$) число неприводимых покрытий асимптотически совпадает с числом единичных подматриц. Поэтому почти всегда построение набора столбцов, содержащего единичную подматрицу, приводит к построению неприводимого покрытия.

Покажем, что задачу построения тупиковых σ-покрытий целочисленной матрицы L можно свести к задаче построения неприводимых покрытий булевой матрицы, которая специальным образом строится по исходной.

Пусть матрица L имеет вид

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix},$$

где каждый элемент столбца с номером j может принимать k_j, $k_j \le k$, значений (для дальнейших рассуждений удобно считать, что элементы столбца с номером j принимают значения от 1 до k_j).

Положим

$$\delta(x_{ij}, k) = \begin{cases} 1, & x_{ij} \ne k, \\ 0, & x_{ij} = k, \end{cases} \quad i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, n.$$

Построим матрицу L^B вида

$$\left[\begin{array}{ccc|c|ccc} \delta(x_{11},1) & \cdots & \delta(x_{11},k_1) & & \delta(x_{1n},1) & \cdots & \delta(x_{1n},k_n) \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \delta(x_{m1},1) & \cdots & \delta(x_{m1},k_1) & & \delta(x_{mn},1) & \cdots & \delta(x_{mn},k_n) \end{array}\right]$$

Нетрудно заметить, что столбцу с номером j исходной матрицы соответствует группа, состоящая из k_j столбцов бинарной матрицы L^B, обозначаемая далее g_j.

Обозначим через (j,i) столбец матрицы L^B из группы g_j с номером i, $j \in \{1, 2, \ldots, n\}$, $i \in \{1, 2, \ldots, k_j\}$.

Очевидным является следующее

Утверждение. *Набор столбцов матрицы* L^B *вида* $((j_1, i_1), \ldots, (j_r, i_r))$, $r \in 1, 2, \ldots, n$, $j_l \in \{1, 2, \ldots, n\}$, $i_l \in \{1, 2, \ldots, k_{j_l}\}$ $j_{l_1} \neq j_{l_2}$ *при* $l_1 \neq l_2$, $l_1, l_2 = 1, 2, \ldots, r$, *является неприводимым покрытием тогда и только тогда, когда набор столбцов с номерами* $(j_1, \ldots, j_r)$ *является тупиковым* $(i_1, \ldots, i_r)$*-покрытием, матрицы L.*

Таким образом, для поиска тупиковых σ-покрытий можно использовать любой алгоритм построения неприводимых покрытий булевой матрицы.

При реализации алгоритма кластеризации, описанного в разд. 2, использовался алгоритм поиска неприводимых покрытий булевой матрицы, суть которого состоит в следующем.

Строится множество наборов столбцов, являющееся специальным подмножеством множества покрытий исходной матрицы.

Формально предлагаемый алгоритм поиска неприводимых покрытий может быть описан следующей рекурсивной процедурой.

Пусть D - булева матрица. Обозначим R множество номеров строк матрицы D, C - множество номеров столбцов матрицы D; $D(A,B)$ - матрица, образованная элементами, находящимися на пересечении строк с номерами из A, $A \subseteq R$ и столбцов с номерами из B, $B \subseteq C$.

Столбец вида $b = (b_1, \ldots, b_k)^T$ будем называть единичным, если $b_i = 1$ при $i = 1, 2, \ldots, k$, и нулевым, если $b_i = 0$ при $i = 1, 2, \ldots, k$. Будем говорить, что столбец $c = (c_1, \ldots, c_k)^T$ охватывает $b = (b_1, \ldots, b_k)^T$, если $c_i \geq b_i$ при $i = 1, 2, \ldots, k$.

Глубина рекурсии равна 1.

1. Пусть C_1^1 и C_1^0 соответственно - множество номеров единичных столбцов и множество номеров нулевых столбцов матрицы D. Тогда для каждого j_1 из C_1^1 столбец с номером j_1 образует неприводимое покрытие матрицы D.

2. Положить $C_1^* = \emptyset$.

3. Если $C \setminus (C_1^* \cup C_1^1 \cup C_1^0) \neq \emptyset$, то присвоить j_1 любой номер из $C \setminus (C_1^* \cup C_1^1 \cup C_1^0)$, $C_1^* = C_1^* \cup \{j_1\}$, C_1^c - множество номеров столбцов из $C \setminus (C_1^* \cup C_1^1 \cup C_1^0)$, покрывающих столбец j_1, R_1 - множество номеров строк из R, которые в пересечении со столбцом j_1 дают единицу. Выполнить рекурсию с глубиной 2, затем повторить шаг 3.

4. Если $C \setminus (C_1^* \cup C_1^1 \cup C_1^0) = \emptyset$, то выйти из процедуры.

Глубина рекурсии i, $i \geq 2$.

1. Если $C \setminus \bigcup_{k=1}^{i-1} (C_k^* \cup C_k^c \cup C_k^1 \cup C_k^0) = \emptyset$, то выйти из текущей рекурсии.

2. Пусть C_i^1 и C_i^0 соответственно - множество номеров единичных столбцов и множество номеров нулевых столбцов матрицы $D(R \setminus \bigcup_{k=1}^{i-1} R_k, C \setminus \bigcup_{k=1}^{i-1}(C_k^* \cup C_k^c \cup C_k^1 \cup C_k^0))$. Тогда для каждого j_i из C_i^1 набор столбцов с номерами $(j_1, \ldots, j_i)$ образует покрытие матрицы D, которое затем проверяется на тупиковость.

3. Положить $C_i^* = \emptyset$.

4. Если $C \setminus (\bigcup_{k=1}^{i} (C_k^* \cup C_k^1 \cup C_k^0) \cup \bigcup_{k=1}^{i-1} C_k^c) \neq \emptyset$, то присвоить j_i любой номер из $C \setminus (\bigcup_{k=1}^{i} (C_k^* \cup C_k^1 \cup C_k^0) \cup \bigcup_{k=1}^{i-1} C_k^c)$, $C_i^* = C_i^* \cup \{j_i\}$, C_i^c - множество номеров столбцов из $C \setminus (\bigcup_{k=1}^{i} (C_k^* \cup C_k^1 \cup C_k^0) \cup \bigcup_{k=1}^{i-1} C_k^c)$, покрывающих столбец j_i, R_i - множество номеров строк из $R \setminus \bigcup_{k=1}^{i-1} R_k$, которые в

пересечении со столбцом j_i дают единицу. Выполнить рекурсию с глубиной $i+1$, затем повторить шаг 4.

5. Если $C \setminus (\bigcup_{k=1}^{i} (C_k^* \cup C_k^1 \cup C_k^0) \cup \bigcup_{k=1}^{i-1} C_k^c) = \emptyset$, то выйти из текущей рекурсии.

По окончании работы процедуры будет построено множество всех неприводимых покрытий матрицы D.

2. Описание алгоритма кластеризации, основанного на построении тупиковых σ-покрытий

Пусть $\Omega = \{S_1, S_2, \ldots, S_m\}$ – множество объектов, подлежащих кластеризации, каждый из которых описывается набором из *n* целочисленных признаков.

Для простоты описания работы алгоритма кластеризации, основанного на построении тупиковых σ-покрытий, введем ряд дополнительных обозначений:

$L(\Omega)$ - матрица, образованная признаковыми описаниями объектов из множества Ω (строкам матрицы соответствуют объекты, столбцам - признаки);

$\sigma(\Omega)$ - множество σ-покрытий матрицы $L(\Omega)$;

$\sigma^T(\Omega)$ - множество тупиковых σ-покрытий матрицы $L(\Omega)$;

$\sigma^T(\Omega_f \mid \Omega_e)$ - множество тупиковых σ-покрытий матрицы $L(\Omega_f)$, которые также являются σ-покрытиями матрицы $L(\Omega_f \bigcup \Omega_e)$;

$P(\Omega_f, \Omega_e) = |\sigma^T(\Omega_f \mid \Omega_e)| \big/ |\sigma^T(\Omega_f)|$, здесь |A| - мощность множества A.

Зададим некоторое число p, $p \in (0,1)$. Будем говорить, что множество Ω_e близко к Ω_f по порогу p, если $P(\Omega_f, \Omega_e) > p$.

Формирование первого класса M_1 осуществляется следующим образом.

Шаг 1. Положить $M_1 = S$, где S - произвольный элемент из множества Ω. Если $\Omega \setminus M_1 = \emptyset$, то алгоритм заканчивает работу. В противном случае перейти к шагу 2.

Шаг 2. Если в $\Omega \setminus M_1$ существует элемент S' такой, что $\mathrm{P}(M_1, \{S'\}) = \max\limits_{S \in \Omega \setminus M_1} \mathrm{P}(M_1, \{S\})$ и

$\mathrm{P}(M_1, \{S'\}) \geq p$, то положить $M_1 = M_1 \bigcup \{S'\}$ и повторить шаг 2. В противном случае перейти к формированию следующего класса.

Пусть построены классы M_1, …, M_{i-1}, $i \geq 2$, и пусть $\widetilde{\Omega}_i = \Omega \setminus \bigcup_{j=1}^{i-1} M_j$.

Если $\widetilde{\Omega}_i = \emptyset$, то алгоритм заканчивает работу, в противном случае осуществляется формирование класса M_i. Процедура построения класса M_i полностью аналогична процедуре построения класса M_1, при условии, что за исходное множество вместо Ω берется $\widetilde{\Omega}_i$.

В данном алгоритме для определения близости объектов или групп объектов используются подописания специального вида (тупиковые σ-покрытия), что дает возможность не вводить функцию расстояния между объектами.

Число *p* обычно выбирается из интервала (0.5, 0.9). Соответственно, чем больше величина *p,* тем большее число классов образуется при кластеризации. При определении близости множеств вместо мощности каждого множества может быть использована сумма весов покрытий этого

множества. В качестве веса покрытия может быть использована величина, равная отношению суммы весов признаков, входящих в покрытие к квадрату длины покрытия (в случае, когда веса признаков не заданы, вес каждого признака может быть положен равным единице).

3. Результаты тестирования на модельных и реальных задачах

В данном разделе приводятся результаты тестирования на модельных и реальных задачах алгоритмов кластеризации, основанных на принципах «дальний сосед» (алгоритм A_1), «ближайший сосед» (алгоритм A_2), выбора центрального элемента (алгоритм A_3) и алгоритма кластеризации, основанного на построении тупиковых σ-покрытий (алгоритм A_4).

Алгоритмы A_1, A_2 и A_3 относятся к классу так называемых иерархических методов группировки, суть которой состоит в следующем. Рассматривается последовательность разделений m объектов на группы. Первый шаг последовательности - разделение на m групп таких, что каждая группа содержит в точности по одному

объекту. Следующий шаг - разделение на m-1 группу, затем на m-2 групп и так далее до m-го шага, в котором все объекты образуют одну группу. Таким образом, k-му шагу соответствует разделение на m+1-k групп, где k=1, …, m. Последовательность разделений называется иерархической группировкой, если любые два объекта S_1 и S_2, объединенные в одну группу на k-м шаге, содержатся в одной группе и на всех последующих шагах.

Пусть требуется разделить m объектов S_1, S_2, ..., S_m, на c классов. Основные этапы иерархической группировки описываются следующей процедурой.

1. Пусть $\tilde{c} = m$ и $\Omega_i = \{S_i\}$, $i = 1,\ldots,m$.
2. Если $\tilde{c} \le c$, выход из процедуры.
3. Найти такую пару групп Ω_i , Ω_j для которой справедливо $\rho(\Omega_i, \Omega_j) = \min_{k \ne l} \rho(\Omega_k, \Omega_l)$, где $\rho(\Omega_I, \Omega_j)$ - расстояние между группами Ω_i и Ω_j (см. ниже).
4. Объединить Ω_i и Ω_j , уничтожить Ω_j и уменьшить $\tilde{c}$ на единицу. Перейти к шагу 2.

Процедура заканчивает работу, когда достигнуто заданное число групп ($\tilde{c} = c$).

Заметим, что в случае, когда число групп, на которые нужно разбить множество, не задано, применяются различные условия окончания процедуры группировки, в частности если наименьшее расстояние между группами на i-м шаге более чем в p раз превосходит наименьшее расстояние на $(i-1)$ – м шаге, где $p>1$ заранее задано (обычно $p>1.5$).

Заметим также, что результат работы описанной выше процедуры сильно зависит от выбора используемой функции расстояния между группами. Для тестирования использовались перечисленные ниже функции.

1. Расстояние, измеряемое по принципу "ближайший сосед":

$$\rho(\Omega_r, \Omega_p) = \min_{S_i, S_j} \rho(S_i, S_j),\ S_i \in \Omega_r,\ S_j \in \Omega_p,\ r, p=1, \ldots, c.$$

2. Расстояние, измеряемое по принципу "дальний сосед":

$$\rho(\Omega_r, \Omega_p) = \max_{S_i, S_j} \rho(S_i, S_j),\ S_i \in \Omega_r,\ S_j \in \Omega_p,\ r, p=1, \ldots, c.$$

3. Расстояние между "центральными элементами" классов:

$\rho(\Omega_r, \Omega_p) = \rho(\overline{S}_r, \overline{S}_p)$, где $\overline{S}_r$ - центральный элемент класса Ω_r, а $\overline{S}_p$ - центральный элемент класса Ω_p. В частности, в качестве центрального элемента класса можно выбрать центр тяжести класса.

В перечисленных выше формулах расстояние $\rho(S_i, S_j)$ между объектами $S_i = (x_{i1}, \ldots, x_{in})$ и $S_j = (x_{j1}, \ldots, x_{jn})$ вычислялось по формуле $\rho(S_i, S_j) = \sum_{k=1}^{n} | x_{ik} - x_{jk} |$.

Сравнение алгоритма A_4 с алгоритмами A_1, A_2 и A_3 проводилось на модельных множествах данных, изображенных на рис. 1-3.

Рис. 1. Два отрезка

Рис. 2. Вложенные друг в друга квадраты

Рис. 3. Две скобки

Результаты работы алгоритма A_1 изображены на рис. 4 - 6. Заметим, что алгоритм A_1 дает "правильное", т.е. наиболее естественное на интуитивном уровне восприятия, разбиение множества, изображенного на рис. 1 при произвольном порядке следования объектов в исходной

выборке (вообще говоря, результаты работы описываемых алгоритмов зависят от этого порядка).

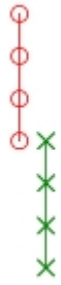

Рис. 4. Два отрезка

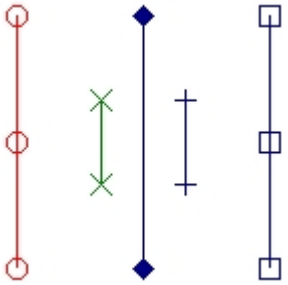

Рис. 5. Вложенные друг в друга квадраты

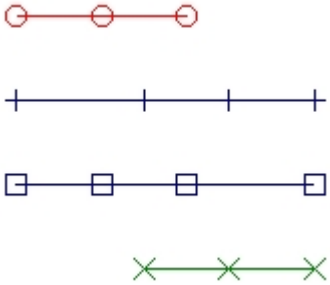

Рис. 6. Две скобки

На рис. 7 - 9 изображены результаты работы алгоритма A_2. Применение алгоритма A_2 существенно "улучшает" разбиение множества, изображенного на рис. 2, но "ухудшает" разбиение множеств изображенных на рис. 1 и 3.

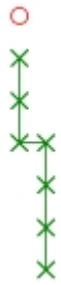

Рис. 7. Два отрезка

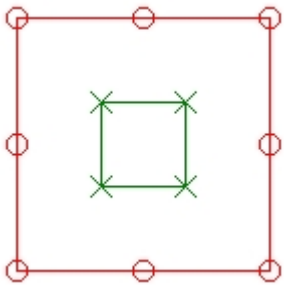

Рис. 8. Вложенные друг в друга квадраты

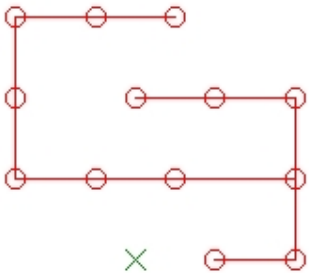

Рис. 9. Две скобки

На рис. 10 - 12 изображены результаты работы алгоритма A_3. Алгоритм, основанный на выборе центрального элемента, демонстрирует некое усреднение методов «ближайший сосед» и «дальний сосед».

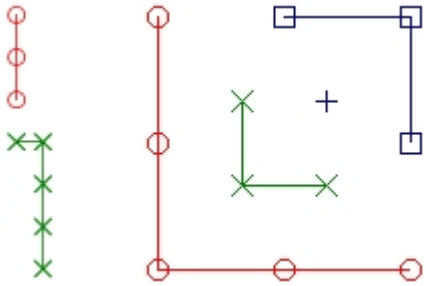

Рис. 10. Два отрезка

Рис. 11. Вложенные друг в друга квадраты

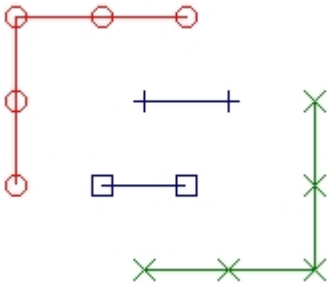

Рис. 12. Две скобки

На рис. 13 - 15 изображены результаты работы алгоритма A_4. Алгоритм, основанный на построении тупиковых σ-покрытий, в отличие от предыдущих алгоритмов показывает лучшие результаты.

Рис. 13. Два отрезка

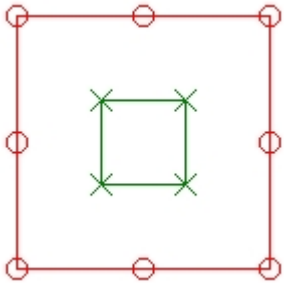

Рис. 14. Вложенные друг в друга квадраты

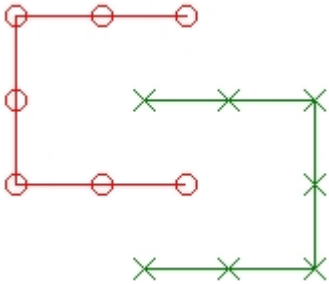

Рис. 15. Две скобки

Алгоритмы кластеризации A_3 и A_4 были также протестированы на реальной информации социологического опроса. Анкета социологического опроса состояла из 21 вопроса - 21 признака, с количеством вариантов ответов от 2 до 15. Всего в опросе приняло участие 799 респондентов. Множество респондентов было разбито на классы по ключевому вопросу анкеты «*Ваше отношение к партии А*». На этот вопрос было предложено четыре варианта ответа: симпатия, равнодушие, неприязнь и трудно сказать. Таким образом было выделено 4 класса. Для каждого эксперимента случайным образом выбиралась небольшая группа (от 4 до 20) респондентов из каждого класса. Информация о принадлежности к какому-либо классу исключалась. Затем на полученном множестве проводилась кластеризация

алгоритмом, основанным на построении тупиковых σ-покрытий (алгоритм A_4), и алгоритмом, основанным на вычислении расстояния Хемминга с выбором центрального элемента (алгоритм A_3). Всего было проведено 32 эксперимента. Разбиения, полученные в результате работы указанных алгоритмов, сравнивались с исходным разбиением на классы. Качество работы алгоритма характеризовалось отношением числа «правильно» классифицированных пар объектов к общему числу пар. При этом классификация считалась «правильной», если выполнялось одно из условий: **а)** пара объектов, принадлежавших одному классу в исходном разбиении, после кластеризации также оказывалась в одном классе; **б)** пара объектов, принадлежавших различным классам в исходном разбиении, после кластеризации также не попадала в один класс.

Точность кластеризации, показываемая алгоритмом A_3 при проведении экспериментов, лежала в пределах от 60% до 80%. Точность кластеризации, показываемая алгоритмом A_4, лежала в пределах от 70% до 90%.

Результаты сравнения на модельных и реальных задачах свидетельствуют о том, что алгоритмы, основанные на

вычислении расстояния Хемминга, отличаются высокой скоростью счета. Эти методы применяются в основном для решения задач с бинарной информацией. Их можно применять и в задачах с признаками большей значности, но в этом случае они плохо работают со специфическими множествами исходных данных, например с множествами данных, изображенных на рис. 1 - 3. Алгоритм A_4 обладает большей трудоемкостью, но не ограничен бинарными признаками и показывает в этом случае более качественные результаты.

Приведенные в разд. 1-3 результаты были доложены на Всероссийской конференции «Математические методы распознавания образов» и опубликованы в сборнике докладов этой конференции [8, 9].

Литература

1. *Дюкова Е.В.* О сложности реализации некоторых процедур распознавания // Ж. вычисл. матем. и матем. физ., 1987. Т. 27. № 1. С. 114–127.
2. *Дюкова Е.В.* Алгоритмы распознавания типа "Кора": сложность реализации и метрические свойства //

Распознавание, классификация, прогноз (матем. методы и их применение). М.: Наука, 1989. Вып. 2. С. 99–125.

3. *Вайнцвайг М.Н.* Алгоритм обучения распознаванию образов "Кора" // Алгоритмы обучения распознаванию образов. М.: Сов. радио, 1973. С. 82–91.
4. *Баскакова Л.В., Журавлев Ю.И.* Модель распознающих алгоритмов с представительными наборами и системами опорных множеств // Ж. вычисл. матем. и матем. физ., 1981. Т. 21. № 5. С. 1264–1275.
5. *Дюкова Е.В., Журавлев Ю.И.* Дискретный анализ признаковых описаний в задачах распознавания большой размерности // Ж. вычисл. матем. и матем. физ., 2000. Т. 40. № 8. С. 1264–1278.
6. *Djukova E.V., Zhuravlev Yu.l.* Diskrete methods of information analysis and algorithm synthesis // J. Pattern Recognition and Image Analysis. 1997. V. 7. № 2. P. 192–207.
7. Дискретная математика и математические вопросы кибернетики / Под ред. С.В. Яблонского, О.Б. Лупанова. М.: Наука, 1974.
8. Дюкова Е.В., Инякин А.С. Решение задачи таксономии на основе построения σ-покрытий // Докл. Всеросс. конф.

«Математические методы распознавания образов 9», М.: АЛЕВ-В, 1999, С. 45-47.

9. Дюкова Е.В., Инякин А.С. Задача таксономии и тупиковые покрытия целочисленной матрицы // Докл. Всеросс. конф. «Математические методы распознавания образов 10», М.: АЛЕВ-В, 2001, С. 44-47.

Содержание

Введение 3

1. Основные определения. Алгоритм поиска тупиковых σ-покрытий целочисленной матрицы 6

2. Описание алгоритма кластеризации, основанного на построении тупиковых σ-покрытий 13

3. Результаты тестирования на модельных и реальных задачах 16

Литература 24