DOI: 10.24411/1993-8314-2019-100XX

Ю. О. Герман, канд. техн. наук, доцент Белорусского государственного университета информатики и радиоэлектроники, Минск,

О. В. Герман, канд. техн. наук, доцент Белорусского государственного университета информатики и радиоэлектроники, Минск,

Об одной модели кластерного анализа на неполных данных

Рассматривается задача построения кластеров на неполных данных и их использования для классификации не полностью определённых объектов. Представлен оригинальный подход, определяющий технику решения путём перехода к отысканию максимального независимого множества (максимальной клики) в нечётком графе, для которого кластер как раз и представляет максимальную клику. Не полностью определённый входной объект, подлежащий классификации (распознаванию), попадает в один из кластеров, и относительно него принимается решение, характерное для этого кластера. Подход последовательно использует модальную логическую систему формул, формализующих задачу отыскания максимального независимого множества в нечётком графе. Показывается, как эту задачу свести к задаче псевдобулевой оптимизации, и приводится простой эвристический алгоритм её решения.

Ключевые слова: нечёткий кластер, кластерная структура, отсутствующие данные, максимальное независимое множество в нечётком графе, клика, классификация (распознавание) не полностью определённых объектов

Введение

Построение кластеров при неполных данных представляет собой интересную теоретико-прикладную задачу анализа и моделирования данных [1–3]. Наличие кластерной структуры на множестве данных позволяет выявить закономерности, характерные для кластеров, и использовать их для классификации (распознавания) не полностью определённых объектов. Например, кластеры могут представлять изображения одного и того же объекта (например, лица) в разных положениях, при наличии скрывающих частей, отсутствующих фрагментов изображения и т. п. В этом случае задача классификации становится задачей распознавания образов.

Для простоты изложения мы рассмотрим таблицу 1 с нормализованными метеорологическими данными.

Таблица 1. Метеорологические данные (нормализованные)

Table 1. Meteorological data (normalized)

№	Влажность (h)	Облачность (c)	Ветер умеренный/слабый (f)	Дождь (r)
1	0,091	0	0	нет
2	0,576	0,429	0,3	нет
3	0,212	0,476	0,2	нет

№	Влажность (h)	Облачность (c)	Ветер умеренный/слабый (f)	Дождь (r)
4	0	0,095	0	нет
5	0,394	0,714	0,7	Наблюдаемые данные для прогноза
6	1	0,905	1	есть
7	0,758	1	1	есть
8	0,697	0,548	0,8	есть

В таблице 1 представлены нормализованные данные. Строка 5 – текущие данные для прогноза. Нормализация на исходных числовых данных выполняется так: для каждого столбца используем формулу $q_{iнорм.} = (q_i - q_{min})/(q_{max} - q_{min})$, где $q_{min}(q_{max})$ – минимальное (максимальное) значение в столбце; q_i – ненормализованное значение в строке i данного столбца ($q - \{h, c, f\}$). Теперь положим, что некоторые данные отсутствуют (или не измерялись) (в таблице 2 ячейки пусты), включая часть сведений о столбце «Дождь».

Таблица 2. Часть данных отсутствует

Table 2. Some data is Missing

№	Влажность (h)	Облачность (c)	Ветер умеренный/слабый (f)	Дождь (r)
1	0,091	0	0	нет
2	0,576	0,429	0,3	
3			0,2	
4	0	0,095	0	
5	0,394	0,714		?
6	1	0,905	1	есть
7	0,758	1	1	
8	0,697			

Нужно дать прогноз о дожде для строки 5 (таблицу 1 можно использовать для оценки точности восстановления кластеров).

Разумеется, пример носит иллюстративный характер. Подобная задача может встретиться в медицине, например, когда диагноз ставится при неполных данных; в психологии (оценка характера личности по фотографии); финансах и экономике и т. п. Её общая суть такова: нужно принять решение при неполных данных, опираясь на опытную таблицу также с неполными данными. Эта проблема хорошо известна в Data Mining, мы дадим некоторый разбор её во второй части работы. Поиск подходов к решению можно считать актуальной востребованной задачей.

Предлагаемый в этой статье подход использует модель нечёткого графа и связан с отысканием на нём кластеров. В кластер попадают схожие объекты, для которых применяется общее решение. Найдя, к какому кластеру относится входной объект, будем в состоянии принять решение для него.

Ребро графа будем считать *нечётким*, если определённо неизвестно, входит оно в граф или нет. Ребро считаем чётким, если определённо известно, что оно входит в граф. Граф с нечёткими рёбрами назовём нечётким.

Нас интересует отыскание максимальной клики (независимого множества) в нечётком графе, причём искомая клика должна содержать максимальное по размеру чёткое подмножество вершин графа.

Максимальную клику отождествим с одним из интересующих нас кластеров. Итак, построим граф с вершинами, соответствующими строкам таблицы 2. Какие-то вершины соединим рёбрами, какие-то нет и припишем рёбрам веса (0,5 или 1). Отсутствие ребра между парой вершин означает, что эта пара (предположительно) не принадлежит одному и тому же кластеру. Такие вершины достаточно сильно различаются хотя бы по одному признаку. Оценку различия нужно свести к какому-то количественному критерию. В этом смысле наша задача упрощается: в каждом столбце есть максимальное (a priori равное 1) и минимальное (a priori равное 0) значения признака ($q_{min}(q_{max})$). Разобьём интервал $[q_{min}, q_{max}]$ на три подинтервала (MIN, MIDDLE, MAX): $[q_{min}, q_{min} + 1/3\cdot(q_{max} - q_{min}))$, $[q_{min} + 1/3\cdot(q_{max} - q_{min}), q_{min} + 2/3\cdot(q_{max} - q_{min}))$, $[q_{min} + 2/3\cdot(q_{max} - q_{min}), q_{max})$. Если по одному и тому же (любому) признаку одна персона попадает в MIN, а вторая – в MAX, то такие персоны сильно различаются по данному признаку, и соответствующие вершины в графе не соединяем. В противном случае:

- если хотя бы один признак неизвестен у любого из объектов, то соединяем ребром соответствующую пару вершин с весом, равным 0,5;
- если по каждому признаку (все признаки известны у обоих объектов) оба объекта попадают в один и тот же полуинтервал, то соединяем соответствующие вершины ребром с весом, равным 1.

С учётом сказанного получаем граф (сходства) с матрицей M смежности, представленной на рисунке 1. Нас будет интересовать максимальная по размеру клика, содержащая максимальное число пар вершин, связанных ребром с весом, равным 1, т. е. содержащая максимально возможную чёткую часть. Такую клику мы первоначально отождествим с первым кластером. Затем можно удалить вершины первого кластера из графа (сходства) и отыскать вторую максимальную клику и т. д. Может потребоваться некоторая техника перераспределения вершин, не вошедших ни в одну клику, однако этот вопрос здесь не рассматривается.

	1	2	3	4	5	6	7	8
1	1	0,5	0,5	1				
2	0,5	1	0,5	0,5	0,5			0,5
3	0,5	0,5	1	0,5	0,5			0,5
4	1	0,5	0,5	1				
5		0,5	0,5		1	0,5	0,5	0,5
6		0,5			0,5	1	1	
7					0,5	1	1	0,5
8		0,5			0,5		0,5	1

Рис. 1. Взвешенная матрица смежности M

Fig 1. Weighted adjacency matrix M

Итак, мы получили исходную позицию для применения техники отыскания максимального независимого множества (ЗМНМ) вершин графа (и ассоциированной с ней задачей отыскания максимальной клики (ЗМК) в графе) для случая нечёткого варианта постановки этих задач. Алгоритмы решения задач ЗМНМ (ЗМК) базируются на методах комбинаторного поиска типа «ветвей и границ» или «отсечений». Достаточно полный обзор точных и эвристических методов для ЗМК дан в [4]. Лучшая сложностная оценка от числа вершин n для точных алгоритмов экспоненциальна и составляет $O(1{,}21^n)$ [4, 5]. Для нечёткого варианта можно отметить, например, [6-8]. В [7] представлено определение нечёткого графа и связанных с ним понятий. Для нечёткого варианта ЗМНМ (ЗМК) предлагается использовать аппарат нейросетей, генетических алгоритмов или нечёткого математического программирования [8]. Первые два подхода эвристические и не гарантируют отыскание требуемого оптимального множества. При использовании нечёткого математического программирования решается несколько задач чёткого математического программирования для графов, соответствующих различным градациям (уровням) нечёткой меры [9]. Поскольку ответ не единственный, то не очевидно, какой результат выбирать в качестве окончательного ответа. Интересным развитием «нечёткого» направления являются интуиционистские нечёткие графы, в частности, в [10] формулируется и решается задача о максимальной клике в интуиционистском нечётком графе. Для восстановления попущенных значений тем или иным способом используются координаты центроида или ближайших соседей в кластере или средние значения по кластеру.

Формализация задачи

Определение 1.

A. (Нечётким) независимым множеством нечёткого графа назовём любое множество

его вершин, никакие две из которых не связаны чётким ребром (таким образом, допускается всё же связь нечёткими рёбрами).

Б. Ядром нечёткого независимого множества считаем подмножество его вершин $\Psi \subseteq \Psi$, никакие две из которых не соединены
c
нечётким ребром.

Определение 2. Пусть для нечёткого графа определены два независимых множества Ψ_1, Ψ_2. Говорим, что Ψ_1 максимально предпочтительнее (*м*-предпочтительнее) Ψ_2, если размер его ядра больше размера ядра Ψ_2.

Определение 3. Нечёткое независимое множество Ψ называется *м*-максимальным, если оно имеет ядро максимального размера, а при условии равенства размеров ядер содержит наибольшее общее число вершин.

Нашей целью является отыскание *м*-максимального независимого множества нечеткого графа. Для достижения цели привлекаем аппарат многозначных логик Я. Лукасевича. Интерпретируем понятие нечёткого ребра x_{ij} как отношение (формулу) с квантором возможности $\Diamond x_{ij}$ (читается «возможно, что вершины i и j связаны ребром»). Будем использовать трёхзначную логику Лукасевича как модель для модальной логики с кванторами необходимости ($\Box$) и возможности ($\Diamond$) [9,11]. Соотношения между $\Box x$, $\Diamond x$ и формулами трёхзначной логики Я. Лукасевича доказаны А. Тарским (см. А. Ивин [8]) и далее используются непосредственно. Формула x имеет значения $val(x)$={0, 0.5, 1} в трёхзначной логике Лукасевича, где 0 означает «невозможно», 0,5 - «неопределённо» и 1 - «необходимо». Далее принимаем:

$\Box x \leftrightarrow val(x) = 1$,

$\Diamond x \leftrightarrow val(x) \geq 0{,}5$, т.к. $\neg\Diamond x \equiv \Box\neg x \leftrightarrow val(x) = 0$.

Пусть $\alpha[\mu(\alpha)]$ обозначает формулу, которая допускает только те интерпретации, в которых $val(\alpha) \geq \mu_\alpha$. Можно рассматривать μ_α как нечёткую степень истинности формулы α. Тогда:

$\Diamond x \equiv \mu(x) \geq 0{,}5$,

$\Box\neg y \equiv \mu(\neg y) = 1$.

Заменим 3-значные формулы x, $\neg x$, y, $\neg y$ двоичными векторными формулами (x_1, x_2), $(\neg x_2, \neg x_1)$, (y_1, y_2), $(\neg y_2, \neg y_1)$ как предложено в [6], т. е. $x \equiv (x_1, x_2)$, $\neg x \equiv (\neg x_2, \neg x_1)$, $y \equiv (y_1, y_2)$, $\neg y \equiv (\neg y_2, \neg y_1)$.

Для любой трехзначной формулы мы допускаем следующие векторные значения: (1,1) – «истинно», (1,0) – «неопределённо», (0,0) – «ложно».

Замечание*. Везде далее принимается недопустимость интерпретации $\alpha = (\alpha_1, \alpha_2) = (0,1)$ для любой формулы α.

Воспользуемся следующим соответствием между трёхзначными формулами и их векторным представлением [9]

$$\begin{aligned} x \vee y &\equiv (x_1 \quad y_1), \\ \neg &\equiv (\quad , \quad x), \\ x \quad y &\equiv x_1 \& y_1, \quad \& \quad), \\ \neg &\equiv (\quad , \quad y), \\ x \to y &\equiv \neg x \vee \quad = \neg_2 \quad y_1 \quad x \vee y). \end{aligned} \tag{1}$$

Допустимость указанных представлений легко проверяется на основании таблиц истинности для трёхзначных операций логики Я. Лукасевича [12].

Пусть x – трёхзначная (векторная формула) и y – двузначная пропозициональная переменная. Пропозициональная двузначная переменная y может быть представлена как трёхзначная формула, которая не допускает значение 0,5, т. е. $y = (y, y)$. Отсюда находим

$x \vee y \equiv (x_1 \vee y, x_2 \vee y)$,

$x \& y \equiv (x_1 \& y, x_2 \& y)$,

$x \to y \equiv \neg x \vee y = (\neg x_2 \vee y, \neg x_1 \vee y)$.

Аналогично для $\neg$y можно записать $\neg y = (\neg y, \neg y)$.

Рассмотрим формулу с трёхзначными переменными x, y:

$$x \quad y. \tag{2}$$

Можно последовательно переписать (2) в виде

$\mu(x) \geq 0{,}5 \vee (y_1, y_2)$,

$\mu(x) \geq 0{,}5 \equiv (1, 0) \vee (1,1) \equiv x_1$,

$\mu(x) \geq 0{,}5 \vee (y_1, y_2) = (x_1 \vee y_1, x_1 \vee y_2)$.

Ещё примеры:

$\Diamond x = \Diamond(x_1, x_2) = (1,0) \vee (1,1) \equiv x_1$.

$\Diamond\neg x = \Diamond(\neg x_2, \neg x_1) = (1,0) \vee (1,1) \equiv \neg x_2$.

$x \vee y = \Box((x_1, x_2) \vee (y_1, y_2)) = \Box(x_1 \vee y_1, x_2 \vee y_2) = (x_1 \vee y_1) \& (x_2 \vee y_2)$.

Полезно запомнить следующие эквивалентности между формулами модальной и трёхзначной логики:

$x = x_1 x_2$; $\Diamond x = x_1$, $\Box x = \Box(x_1, x_2) = x_1 x_2$ (3)

(предполагается, что x не содержит модальных кванторов).

Описание подхода к решению

Формальную постановку задачи для матрицы на рисунке 1 можно записать таким образом (обозначив строки (столбцы) матрицы через m_i):

$$\sum_{i=1,8} m_i \to \max$$

$$\begin{aligned}
&\neg m_1 \vee \neg m_5, \quad \neg m_4 \vee \neg m_6, \quad \neg m_3 \vee \neg m_6, \Diamond(\neg m_1 \vee \neg m_3),\\
&\neg m_1 \vee \neg m_6, \quad \neg m_4 \vee \neg m_7, \quad \neg m_3 \vee \neg m_7, \Diamond(\neg m_2 \vee \neg m_3),\\
&\neg m_1 \vee \neg m_7, \neg m_4 \vee \neg m_8, \quad \neg m_4 \vee \neg m_5, \Diamond(\neg m_2 \vee \neg m_4),\\
&\neg m_1 \vee \neg m_8, \neg m_6 \vee \neg m_8, \quad \neg m_2 \vee \neg m_6, \Diamond(\neg m_1 \vee \neg m_2),\\
&\Diamond(\neg m_3 \vee \neg m_4), \Diamond(\neg m_5 \vee \neg m_7), \Diamond(\neg m_3 \vee \neg m_5), \Diamond(\neg m_5 \vee \neg m_8)\\
&\Diamond(\neg m_7 \vee \neg m_8).
\end{aligned} \tag{4}$$

Используя двоичную интерпретацию модальных формул, перепишем систему в «чистых» булевских переменных. Введём представления

$$\begin{aligned}
&m_i = (x_i, y_i), \quad i = 1,8;\\
&\neg m_r \vee \neg m_s = (\neg y_r, \neg x_r) \vee (\neg y_s, \neg x_s) = (\neg y_r \vee \neg y_s, \neg x_r \vee \neg x_s);\\
&\Diamond(\neg m_r \vee \neg m_s) = \Diamond(\neg y_r \vee \neg y_s, \neg x_r \vee \neg x_s) = \neg y_r \vee \neg y_s.
\end{aligned}$$

Последняя формула получена с учётом (1). Имеем далее (5)

$$\begin{aligned}
&K \cdot y_1 + K \cdot y_2 + K \cdot y_3 + K \cdot y_4 + K \cdot y_5 + K \cdot y_6 + K \cdot y_7 + K \cdot y_8 +\\
&x + x + x + x + x + x + x + x \to \max\\
&1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8
\end{aligned}$$

$$\begin{aligned}
&\neg y_1 \vee \neg y_5, \quad \neg y_4 \vee \neg y_6, \quad \neg y_3 \vee \neg y_7, \quad \neg x_1 \vee \neg x_5,\\
&\neg y_1 \vee \neg y_6, \quad \neg y_4 \vee \neg y_7, \quad \neg y_4 \vee \neg y_5, \quad \neg x_1 \vee \neg x_6,\\
&\neg y_1 \vee \neg y_7, \quad \neg y_4 \vee \neg y_8, \quad \neg y_2 \vee \neg y_6, \quad \neg x_1 \vee \neg x_7,\\
&\neg y_1 \vee \neg y_8, \quad \neg y_6 \vee \neg y_8, \quad \neg y_3 \vee \neg y_6, \quad \neg x_1 \vee \neg x_8,
\end{aligned}$$

$$\begin{aligned}
&\neg x_4 \vee \neg x_5, \quad \neg x_4 \vee \neg x_6 \quad \neg x_4 \vee \neg x_7, \quad \neg x_4 \vee \neg x_8\\
&\neg x_3 \vee \neg x_6, \quad \neg x_2 \vee \neg x_6, \quad \neg x_6 \vee \neg x_8, \neg x_3 \vee \neg x_7,\\
&\neg y_1 \vee \neg y_3, \quad \neg y_2 \vee \neg y_3, \quad \neg y_2 \vee \neg y_4, \quad \neg y_1 \vee \neg y_2,\\
&\neg y_2 \vee \neg y_5, \quad \neg y_3 \vee \neg y_8, \quad \neg y_2 \vee \neg y_8, \quad \neg y_5 \vee \neg y_6,\\
&\neg y_3 \vee \neg y_4, \quad \neg y_5 \vee \neg y_7, \quad \neg y_3 \vee \neg y_5, \quad \neg y_5 \vee \neg y_8,\\
&\neg y_7 \vee \neg y_8,\\
&x_1 \vee \neg y_1, \quad x_2 \vee \neg y_2, \quad x_3 \vee \neg y_3, \quad x_4 \vee \neg y_4, \quad x_5 \vee \neg y_5,\\
&x \vee \neg y, \quad x \vee \neg y, \quad x \vee \neg y.\\
&6 \quad 6 \quad 7 \quad 7 \quad 8 \quad 8
\end{aligned}$$

Формулы

$$\begin{aligned}
&\neg x_1 \vee y_1, \quad \neg x_2 \vee y_2, \neg x_3 \vee y_3, \neg x_4 \vee y_4,\\
&\neg x_5 \vee y_5, \neg x_6 \vee y_6, \neg x_7 \vee y_7
\end{aligned}$$

записаны в силу Замечания*. K – большое положительное число, превосходящее, например, удвоенное число вершин (K = 17). Целевая функция в (5) допускает нечёткие рёбра, но обеспечивает максимальный размер ядра, а при равенстве размеров ядер – наибольший общий размер м-независимого множества. Задача сведена к известной задаче линейного булевского программирования и может быть решена, например, методом Балаша [13]. При больших размерах задачи поиск точных решений сопряжён со значительными вычислительными затратами. В литературе описаны подходы к решению на основе представления (не сведения) задачи (5) задачей об упаковке в рюкзак (KnapSack problem) с одним ограничением [14]. В принципе мы получаем некоторое предварительное разбиение на кластеры с последующей задачей коррекции кластеров с неполными данными об объектах [1-3]. Можно предложить следующую простую технику с учётом формы представления рассматриваемой задачи. Пусть z_i – число вхождений булевской переменной y_i в (5) в систему ограничений. Тогда последовательно включаем в решение ту переменную, для которой максимально отношение C_i / z_i, где C_i – коэффициент при данной переменной yi в целевой функции. Так, в рассматриваемом примере сначала включаем в решение переменную y_1 (имеет наибольшее

число вхождений, равное 6). После этого редуцируем систему к следующему виду (сразу установив $y_3 = 0$, $y_2 = 0$, $y_5 = 0$, $y_6 = 0$, $y_7 = 0$, $y_8 = 0$, $x_1=1$, $x_6= 0$, $x_7 = 0$, $x_8 = 0$):

$\neg x_4 \vee \neg x_5$,

$x_4 \vee \neg y_4$.

и повторяем по аналогии. Следующая переменная, попадающая в решение: $y_4 = 1$. Если остаются только переменные x_i, то используем отношение C_i / z_i, где z_i представляет x_i. Кроме того, если переменная была удалена из системы, не получив значения, то приписываем ей единичное значение (такие переменные $x_2 = 1$, $x_3 = 1$). У нас всё проще: окончательно (опуская детали) имеем: $y_1 = 1$, $y_4 = 1$, $y_3 = 0$, $y_2 = 0$, $y_5 = 0$, $y_6 = 0$, $y_7 = 0$, $y_8 = 0$, $x_1 = 1$, $x_2 = 1$, $x_3 = 1$, $x_4 = 1$, $x_5 = 0$, $x_6 = 0$, $x_7 = 0$, $x_8 = 0$. Итак, наш искомый первый кластер содержит вершины m_1, m_4 и, возможно, m_2 и m_3 (m_5, m_6, m_7, m_8 с необходимостью не принадлежат данному кластеру). Прогнозируем значение m_5. Имеется два варианта: дождь для m_5 необходим или возможен. В рамках рассмотренной модели нельзя установить, будет ли дождь с необходимостью (этого нельзя сделать даже при полных данных). Получаем ответ: рациональным решением следует считать «дождь будет». Заметим, что этот ответ получен в рамках использованной интерпретации модальных формул формулами трёхзначной логики Лукасевича. Повышение степени значности дало бы ответ и о степени возможности дождя.

Для сравнения приводим результат кластеризации в Python при неполных данных.

Реализация на языке Python

В Python неверные или пропущенные данные заменяются, как правило, термом NaN. Этот терм можно обрабатывать по-разному. Простейший вариант – заполнение пропущенных данных средними значениями в столбцах. Используют также значение медианы, значение, взвешенное по частоте встречаемости. В более сложном варианте пропущенное значение вычисляют как функцию регрессии от известных значений в строке. Проблема в том, что следует запастись значительным количеством таких функций, близким к 2^n, где n – число столбцов, чтобы предусмотреть все случаи. Ниже мы привели пример стандартного варианта реализации замены пропущенных данных в Python на основе класса Imputer пакета Sklearn. Программный скрипт на языке Python приведен в листинге 1.

Данные в программе заданы из рассмотренного примера в коллекции Data. Imputer выполняет замену пропущенных данных (NaN) с помощью стратегии средних значений в столбце (Листинг 2).

На консольном окне приведены координаты центроидов найденных кластеров: [0,37671429, 0,3809444, 0,22222] и [0,879, 0,9525, 1]. Выведены полные значения строк исходной восстановленной таблицы (рис. 2).

Построенные кластеры представлены в «плоском» варианте (рис. 3).

Интересующий нас прогнозируемый объект получил такие значения: <0,394, 0,714, 0,4166>. По евклидовой метрике он ближе к первому кластеру – самая верхняя точка, обведённая красным кружком. В первом кластере сгруппированы точки, где дождя нет, – в статье получен противоположный результат. Точка в красном квадратике также тяготеет к первому кластеру (хотя в исходной таблице к нему не относится). Итак, результаты Python-скрипта менее удачны, чем результаты статьи. В целом это служит определённым основанием считать описанный в статье подход технически целесообразным.

Заключение

Приведённый подход может подлежать критике со следующей позиции: разбиение интервала изменения каждого признака на три равновеликие части выглядит достаточно грубым вариантом в сравнении с возможным разбиением на большее число ин-

Листинг 1. Восстановление пропущенных данных и кластеризация
Listing 1. Imputation of missing data and clusterization

```
from pandas import DataFrame
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import numpy as np
import pandas as pd

Data = {'x': [0,091,0,576,np.nan, 0, 0,394, 1, 0,758, 0,697],
'y': [0,0,429, np.nan, 0,095, 0,714, 0,905, 1, np.nan],
'w': [0, 0,3, 0,2, 0, np.nan,1,1,np.nan]
   }

df = DataFrame(Data,columns=['x','y','w'])
trainingData = df.iloc[:, :].values
dataset = df.iloc[:, :].values

from sklearn.preprocessing import Imputer
imputer = Imputer(missing_values=»NaN», strategy=»mean», axis = 0)
imputer = imputer.fit(trainingData[:, :])
dataset[:, :] = imputer.transform(dataset[:, :])

kmeans = KMeans(n_clusters=2).fit(dataset)
centroids = kmeans.cluster_centers_
print(centroids)
print(df)
fig = plt.figure(figsize=(5, 5))
plt.scatter(df['x'], df['y'], df['w'])
plt.scatter(centroids[:, 0], centroids[:, 1], c='red')
plt.show()
```

Листинг 2. Заполнение отсутствующих данных средними значениями
Listing 2. Imputation of missing data with average values

```
imputer = Imputer(missing_values=»NaN», strategy=»mean», axis = 0)
imputer = imputer.fit(trainingData[:, :])
dataset[:, :] = imputer.transform(dataset[:, :])
# для построения кластеров используется метод K-средних
kmeans = KMeans(n_clusters=2).fit(dataset)
#определение центров кластеров
centroids = kmeans.cluster_centers_
```

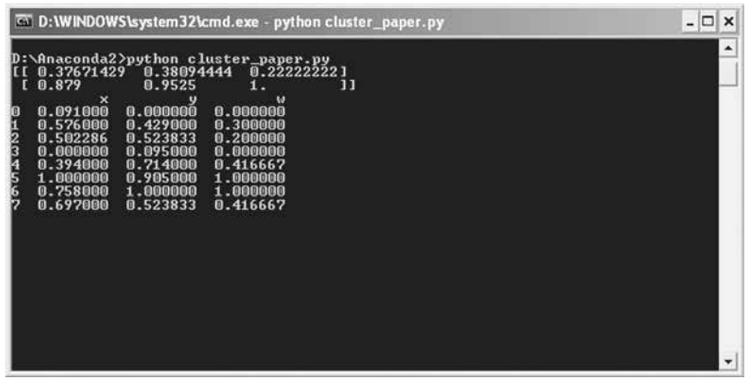

Рис. 2. Консольное окно с результатами
Fig. 2. Console window with output results

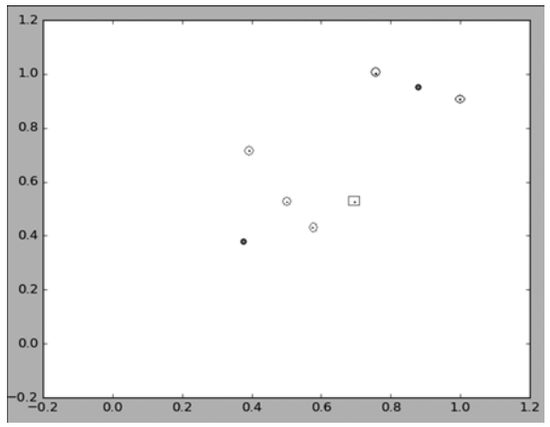

Рис. 3. Два кластера с центрами в сплошных кружках
Fig. 3. Two clusters with centers in solid circles

тервалов. Во-вторых, если даже значения признаков попадают в крайние интервалы, то это не всегда означает принадлежность к разным кластерам. Эти замечания в действительности задают точки роста, а не «нивелируют» суть подхода. Первое замечание связано с обобщением на случай, когда степени (не)чёткости отличны от 0,5. В этой ситуации следует использовать многозначную логику Лукасевича, позволяющую аппроксимировать нечёткие значения чёткими с требуемой точностью. Второе замечание требует в общем случае строить не два кластера, а больше, полагая при этом, что близкие значения всё же попадут в один общий кластер. Оценка близости значений в кластере связана с критериями однородности, силуэта (silueth), используемыми в Python и других языках.

Список литературы

1. *Hoppner F., Klawonn F., Kruse R., Runkler T.* Fuzzy cluster analysis. Methods for classification, data analysis and image recognition // John Willey & Sons. 2000. – 290 p.
2. *Timm H., Doring C., Kruse R.* Different approaches to fuzzy clustering of incomplete datasets // International Journal of Approximate Reasoning. 2004. № 35. P. 239–249.
3. *Himmelspach L., Carvalho P., Conrad S.* On Cluster Validity for Fuzzy Clustering of Incomplete Data // Proc. of the 6th International Conference on Scalable Uncertainty Management, SUM 2012, Marburg, Germany, September 17–19, 2012. Lecture Notes in Computer Science. Springer. 2012. Vol. 7520. P. 612–618.
4. *Panos M. Pardalos, Jue Xue.* The maximum clique problem // Journal of global optimization. 1994. Vol. 4(3). P. 301–328.
5. *Лифшиц Ю.* Точные алгоритмы и открытые проблемы // Современные задачи теоретической информатики. URL: http://download.yandex.ru/class/lifshits/lecture02.pdf (дата обращения 27.01.2020)
6. *Nair P. S.* Cliques and fuzzy cliques in fuzzy graphs // IFSA World Congress and 20th NAFIPS International Conference. Vancouver, Canada, 25–28 July. 2001. Vol. 4.
7. *Sunitha M. S.* Studies on Fuzzy graphs // Depart. of mathematics. Cochin university of Science and Technology, Cochin, India. 2001.
8. *Anastasiou A.* Fuzzy Mathematics: Approximation Theory. Springer. 2010. – 456 p.
9. *Герман О. В.* Неклассические логические исчисления. Минск: Белорусский государственный университет информатики и радиоэлектроники, 2012. – 124 с.
10. *Venkatesh S., Sujatha S.* Mining maximal cliques through an intuitionistic fuzzy graph //Applied Mathematics & Information Sciences, 2017, vol. 11, no. 4, pp. 1193 – 1198.
11. *Ивин А. А.* Модальные теории Я. Лукасевича. М.: Институт философии РАН, 2001. – 176 с.
12. *Карпенко А. С.* Логики Лукасевича и простые числа. М.: Наука, 2000. - 317 с.
13. *Балаш Э.* Аддитивный алгоритм для решения задач линейного программирования // Кибернетический сборник. Новая серия, вып. 6. М., 1969. – 264 с.
14. Irmeilyana, Bahtera P., Izzah H. Solution of multiple constraints knapsack problem (MCKP) by using branch and bound method and greedy Algorithm // Journal of Modeling and Optimization. 2017. Vol. 9. № 2. Pp. 112-119.

References

1. Hoppner F., Klawonn F., Kruse R., Runkler T. Fuzzy cluster analysis. Methods for classification, data analysis and image recognition. John Willey &Sons, 2000, 290 p.
2. Timm H., Doring C., Kruse R. Different approaches to fuzzy clustering of incomplete datasets. International Journal of Approximate Reasoning, no. 35, 2004, pp. 239–249
3. Himmelspach L., Carvalho P., Conrad S. On Cluster Validity for Fuzzy Clustering of Incomplete Data. In Proceedings of the 6th International Conference on Scalable Uncertainty Management, SUM 2012, Marburg, Germany, September 17-19, 2012, Lecture Notes in Computer Science, vol. 7520, Springer, pp. 612 – 618, 2012.
4. Panos M. Pardalos, Jue Xue. The maximum clique problem. Journal of global optimization, vol. 4(3), 1994, pp. 301–328.
5. Lifshiz Yu. *Tochniye algoritmy i otkrytie problemy* [Exact algorithms and open problems]. Contemporary problems in theoretical informatics. Available at: http://download.yandex.ru/class/lifshits/lecture02.pdf (accessed 27.01.2020).
6. Nair P. Cliques and fuzzy cliques in fuzzy graphs. IFSA World Congress and 20th NAFIPS International Conference (vol.4). Vancouver, Canada, 25–28 July, 2001.
7. Sunitha M.S. Studies on Fuzzy graphs. Depart. of mathematics. Cochin university of Science and Technology, Cochin, India, 2001.
8. Anastasiou A. Fuzzy Mathematics: Approximation Theory. Springer, 2010, 456p.
9. German O.V. Neclassicheskiye logicheskiye ischisleniya [Non-classical logical calculi]. Belarussian State university of informatics and radio-electronics, Minsk, 2012, 124 p.
10. Venkatesh S., Sujatha S. Mining maximal cliques through an intuitionistic fuzzy graph. Applied Mathematics & Information Sciences. vol. 11, no. 4, 2017, pp. 1193–1198.
11. Ivin A. A. Modalnie teorii Y. Lukasiewicha [Modal theories of I. Lukasiewicz]. Institute of Philosophy of the Russian Academy of Science, Moscow, 2001, 176 p.
12. Karpenko A. S. *Logiki Lukasewicha i prostye chisla* [Lukasiewicz's logics and prime numbers]. Moscow, Nauka Publ., 2000, 317 p.
13. Balas E. *Additivnyi algoritm dlja reshenija zadach lineinogo programmirovania* [Additive algorithms for solving linear programming problems]. *Kiberneticheskij sbornik. Novaja seria, vypusk 6* [Cybernetic collection. New Series. Issue 6]. Moscow, 1969, 264 p.
14. Irmeilyana, Bahtera P., Izzah H. Solution of multiple constraints knapsack problem (MCKP) by using branch and bound method and greedy Algorithm. Journal of Modeling and Optimization, vol. 9, no. 2, 2017, pp. 112-119.

DOI: 10.24411/1993-8314-2020-100XX
J. German, Belarussian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus,
O. German, Belarussian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

About one cluster analysis model on incomplete data

The clusterization problem on incomplete data is considered with its application to classification of the partly defined objects. The problem may arise in different practical areas including diagnosis making, forecasting, face recognition and so on. An original approach is outlined which involves a solution technique based on maximum independent set (maximum clique) definition in fuzzy graph (with evident interpretation of a cluster as some maximum clique in a fuzzy graph). An input partly defined object then gets to one of the cliques (clusters) with a decision undertaken specific to that clique. The entire approach subsequently uses a modal logical system, providing necessary formalization to find the maximum independent set (maximum clique) in fuzzy graph. This formalization is based on the transition from modal logic to Lukasewich multi-valued logics accordingly to Tarski theoretical results. The next step consists in transformation of the Lukasewich multi-valued logic to a classical boolean-valued system accordingly to suggested scheme. It is then shown how to formulate a pseudo-boolean optimization (pbo) problem and solve it by means of the suggested simple heuristic method which delivers a solution to a multipple kanpsack problem which we use instead of pbo. It is also noticed that there is a possibility to use different multi-valued Lukasewich logics to interpret a modal system in order to increase an accuracy of the solution. We give also a Python code realizing a standard imputation of the missing data by means of using average values and show that this technique gives incorrect results while the suggested method provides a right solution. Together with acceptable computational complexity of the suggested approach this gives good reasons to recommend the entire rechnique to practical usage.

Keywords: cluster, missing data, fuzzy graph, maximum independent set, clique, classification of partly defined objects

About authors:

J. German, *PhD in Technique, Associate Professor*

O. German, *PhD in Technique, Associate Professor*

For citation:

German J., German O. About one cluster analysis model on incomplete data. *Prikladnaya informatika* – Journal of Applied Informatics, 2020, vol.15, no. 1(85), pp. xx - xx (in Russian). DOI: 10.24411/1993-8314-2020-100XX