АЛГОРИТМ УСРЕДНЕНИЯ ЦЕНТРОИДОВ ДЛЯ ПОСТРОЕНИЯ КЛАСТЕРНОГО АНСАМБЛЯ

В.В. Татарников[1], И.А. Пестунов[2], В.Б. Бериков[3]
[1] Институт математики СО РАН, Новосибирск, Россия,
[2] Институт вычислительных технологий СО РАН, Новосибирск, Россия,
[3]Новосибирский государственный университет, Новосибирск, Россия

Аннотация

В статье рассматривается коллективный подход к решению задачи кластерного анализа. Предложен алгоритм усреднения центроидов, позволяющий построить консенсусное разбиение выборки на кластеры, используя набор разбиений этой выборки любым центроидным алгоритмом. Приведены результаты применения алгоритма к модельным данным и для сегментации гиперспектральных изображений с шумовыми каналами. Рассмотрены некоторые детали реализации в многопоточном окружении, позволяющие увеличить производительность алгоритма.

Ключевые слова: кластерный ансамбль, K-means, центроид, анализ гиперспектральных изображений.

Цитирование: **Татарников, В.В.** Алгоритм усреднения центроидов для построения кластерного ансамбля / В.В. Татарников, И.А. Пестунов, В.Б. Бериков // Компьютерная оптика. – 2017. – Т. 41, № 5. – С. 712-718. – DOI: 10.18287/2412-6179-2017-41-5-712-718.

Введение

Задача кластерного анализа заключается в отыскании наилучшего с точки зрения заданного критерия качества разбиения множества объектов на подмножества (кластеры). Под критерием качества понимают некий функционал, в той или иной степени учитывающий как различия между объектами в пределах одного кластера, так и расстояния между кластерами [1, 2, 3]. Выбор меры различия объектов обусловлен спецификой конкретных задач и типами переменных, описывающих объекты.

Пусть X – набор вещественных векторов размерности M, а $N=|X|$ – число векторов в этом наборе. В задачах кластеризации объектов с вещественными характеристиками широкое применение нашёл итеративный алгоритм K-means с Евклидовым расстоянием $d(x_{i_1}, x_{i_2}) = \sqrt{\sum_{j=1}^{M} (x_{i_1 j} - x_{i_2 j})^2}$ для любых $x_{i_1}, x_{i_2} \in X$, в качестве меры различия между объектами. На каждой итерации алгоритм разбивает исходное множество объектов X на заданное число K непустых непересекающихся кластеров X_k, определяя для каждого $x \in X$ номер его кластера: $x \in X_k \Leftrightarrow k = \arg\min_{k'} d(x, c_{k'})$, где c_k – центроид (центр) кластера X_k: $c_k = (1/|X_k|)\sum_{x \in X_k} x$, при этом номер кластера k для $x \in X_k$ также называют меткой x. Алгоритм находит один из локальных минимумов суммы квадратов отклонений x от центров c_k, причём результат существенно зависит от выбора начальных центров кластеров c_k^0. Вычислительная сложность алгоритма составляет $O(KMNT)$, где T – число итераций, т.е. при фиксированных K, M, T алгоритм характеризуется линейной трудоёмкостью относительно числа объектов.

В последнее время для повышения устойчивости и качества результатов кластеризации активно применяется ансамблевый подход [4–10]. Подход заключается в построении множества кластеризаций на основе различных алгоритмов или одного алгоритма с различными параметрами и итоговой кластеризации на их основе. Применение ансамблевого подхода позволяет снижать зависимость результатов группировки от выбора параметров алгоритма, получать более устойчивые решения в условиях зашумленных данных и, если ансамбль составлен из различных алгоритмов, взглянуть на задачу с «разных точек зрения», т.е. решить в том числе задачу выбора модели [11]. Алгоритмы на основе кластерных ансамблей применимы для обработки больших объёмов данных [12], например, мультиспектральных изображений [13, 14].

Результат применения ансамблевого алгоритма называют консенсусным. Известны различные подходы к его построению[11]. Пусть $\mathcal{P}$ – множество всех возможных разбиений X на кластеры и определено отображение $\delta: \mathcal{P} \times \mathcal{P} \to \mathbb{R}$, характеризующее степень различия между двумя разбиениями, а $P^1, \ldots, P^L$ – варианты разбиений X на $K^1, \ldots, K^L$ кластеров. Тогда один из способов формального определения консенсусного разбиения заключается в отыскании разбиения P^*:

$$P^* = \arg\min_{P \in \mathcal{P}} \sum_{l=1}^{L} \delta(P, P^l),$$

т.е. разбиения, суммарно наименее отличного от разбиений в ансамбле. Таким образом, разнообразие способов определения и построения P^* во многом обусловлено разнообразием выбора функции δ, при этом к популярным δ относят [11]: улучшенный индекс Ранда (ARI) [15], нормализованное полное количество информации (NMI) [16], вариацию информации (VI) [17], включая нормализованный вариант (NVI) [18]. Указанные функции δ претендуют на универсальность, поскольку учитывают только метки объектов из разбиений и не учитывают специфичные для моделей кластеров характеристики, например, в случае с ансамблем K-means-разбиений, координаты центров кластеров. Семейство методов построения консенсусного разбиения, учитывающих центры кластеров, предложено в работе [6]. Центроиды группируются в консенсусные цепи (consensus chains) так, чтобы суммарное расстояние между центроидами в пределах одной цепи

было минимальным, для чего используется приближённый алгоритм. Таким образом, устанавливается соответствие между кластерами из различных разбиений. Предложено несколько модификаций метода: метод, использующий предположение, что $P^1,\ldots,P^L$ имеют одинаковое число кластеров, и метод для произвольного числа кластеров в каждом из разбиений. К основным преимуществам алгоритма авторы относят высокую степень масштабируемости, возможность организации параллельных вычислений. Кроме того, вычислительная сложность процедуры построения ансамблевого решения не зависит от числа объектов N. Предусмотрен этап фильтрации консенсусных цепей от выбросов, позволяющий снизить влияние шумов, однако нетрудно привести пример, в котором в каждой цепи кластер будет исключён как выброс, что приведёт к потере кластера в итоговом разбиении. Исходя из этих соображений, разбиение выборки на фрагменты в пользу масштабируемости алгоритма не всегда приводит к лучшим результатам.

Пусть $P^1,\ldots,P^L$ – варианты разбиений X на K кластеров $\{X_k^l\}_{k=1,..,K}^{l=1,..,L}$ с центрами $C^l=\{c_k^l\}_{k=1,...,K}$. Обозначим через $c^l(x)$ отображение $c^l(x)\colon X\to C^l$, которое для каждого объекта x определяет центроид из l-го разбиения. Зададим меру различия между парой разбиений P^i и P^j через суммарное расстояние между центрами кластеров, соответствующих каждому $x\in X$:

$$\delta(P^i,P^j)=\sum_{x\in X} d^2(c^i(x),c^j(x)).$$

Функция $\delta(P^i,P^j)$ обращается в 0, когда центры кластеров совпадают между собой для каждого $x\in X$. Консенсусным разбиением назовём разбиение

$$P^*=\arg\min_{P\in\mathcal{P}}\sum_{l=1}^{L}\delta(P,P^l).$$

Здесь разбиению $P^*=\{X_1^*,...,X_K^*\}$ соответствует набор центроидов $\{c_1^*,...,c_K^*\}$.

В параграфе 1 данной работы дан теоретический анализ построения такого разбиения. В параграфе 2 описан алгоритм и результаты его применения на реальных данных. В параграфе 3 приведены результаты применения алгоритма к модельным и реальным данным. Параграф 4 посвящён особенностям реализации с учётом возможности применения параллельных вычислений.

1. Построение консенсусного разбиения

В приведённых ранее обозначениях рассмотрим подробнее свойства разбиения P^* и связанных с ним центроидов.

Утверждение 1. Пусть P^* – консенсусное разбиение для произвольных X, K, L и набора вариантов разбиений $P^1,\ldots,P^L$. Тогда центроиды c_k^* консенсусного разбиения могут быть выражены через центроиды разбиений ансамбля следующим образом:

$$c_k^*=\frac{1}{L\,|\,X_k^*\,|}\sum_{l=1}^{L}\sum_{x\in X_k^*}c^l(x),$$

где $k=1,\ldots,K$ и X_k^* – k-й кластер разбиения P^*.

Доказательство. Рассмотрим функцию

$$f(P^*)=\sum_{l=1}^{L}\delta(P^*,P^l)=\sum_{l=1}^{L}\sum_{i=1}^{N}d^2(c^*(x_i),c^l(x_i)).$$

Для каждого разбиения P представим $c(x)$ с помощью центров c_k и индикаторных функций $I[x\in X_k]$:

$$c(x)=\sum_{k=1}^{K}I[x\in X_k]c_k,$$

где

$$I[x\in X_k]=\begin{cases}1, \text{если } x\in X_k,\\ 0, \text{иначе.}\end{cases}$$

Это позволяет выразить $f(P^*)$ через c_k^*:

$$f(c_1^*,\ldots,c_K^*)=\sum_{l=1}^{L}\sum_{i=1}^{N}d^2\Big(\sum_{q=1}^{K}I[x_i\in X_q^*]c_q^*,\ \sum_{q=1}^{K}I[x_i\in X_q^l]c_q^l\Big).$$

Поскольку центры c_k определяются набором характеристик $\{c_{kj}\}_{k=1,\ldots,K;j=1,\ldots,M}$, расстояние d между центрами можно переписать, используя формулу: $d^2(c_{k_1},c_{k_2})=\sum_{j=1}^{M}(c_{k_1 j}-c_{k_2 j})^2$, что позволяет выразить $f(P^*)$ через координаты центров кластеров:

$$f(c_{11}^*,\ldots,c_{1M}^*,\ldots,c_{K1}^*,\ldots,c_{KM}^*)=$$
$$=\sum_{l=1}^{L}\sum_{i=1}^{N}\sum_{j=1}^{M}\Big(\sum_{q=1}^{K}I[x_i\in X_q^*]c_{qj}^*-\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l\Big)^2.$$

По определению, P^* – оптимальное разбиение, тогда частные производные f по всем c_{kj}^* имеют вид:

$$\frac{\partial f}{\partial c_{kj}^*}=2\sum_{l=1}^{L}\sum_{i=1}^{N}\Big(\sum_{q=1}^{K}I[x_i\in X_q^*]c_{qj}^*-\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l\Big)I[x_i\in X_k^*]=2\sum_{l=1}^{L}\sum_{x_i\in X_k^*}\Big((c^*(x_i))_j-\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l\Big)=$$
$$=2L\,|\,X_k^*\,|\,c_{kj}^*-2\sum_{l=1}^{L}\sum_{x_i\in X_k^*}\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l,$$

и равны нулю:

$$2L\,|\,X_k^*\,|\,c_{kj}^*-2\sum_{l=1}^{L}\sum_{x_i\in X_k^*}\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l=0.$$

Полученная система уравнений имеет решение:

$$c_{kj}^*=\frac{1}{L\,|\,X_k^*\,|}\sum_{l=1}^{L}\sum_{x_i\in X_k^*}\sum_{q=1}^{K}I[x_i\in X_q^l]c_{qj}^l,$$

которое можно выразить через c_k^*:

$$c_k^*=\frac{1}{L\,|\,X_k^*\,|}\sum_{l=1}^{L}\sum_{x_i\in X_k^*}c^l(x_i).$$

Таким образом, утверждение доказано. Однако на практике вычислить $c_1^*,\ldots,c_K^*$ по найденным явным формулам не представляется возможным, поскольку

не известен состав кластеров $X_1^*,...,X_K^*$. Поэтому для построения приближённого решения предлагается подход, основанный на усреднённых значениях центров кластеров вариантов разбиений.

2. Алгоритм усреднения центроидов АС

Обозначим через $C = KMEANS_{X,K}(C_0)$ отображение, ставящее в соответствие исходному набору центроидов $C_0 = \{c_{0,1},\ldots,c_{0,K}\}$ набор центроидов $C = \{c_1,\ldots,c_K\}$, полученный в результате применения алгоритма K-means. Здесь X – некоторый набор вещественных векторов, который требуется разбить на K кластеров. Обозначим через $c(x):X\to C$ функцию, определяющую для данного объекта x центр ближайшего кластера.

Тогда предлагаемый алгоритм АС можно записать в виде следующей последовательности шагов.

Шаг 1. Построить L вариантов разбиений X на K кластеров каждый, применяя алгоритм K-means со случайно заданными исходными центроидами $C_0^l, l = 1,...,L$; вычислить $C^l = \mathrm{KMEANS}_{X,K}(C_0^l)$; определить $c^l(x):X\to C^l$.

Шаг 2. Каждому $x_i\in X$ сопоставить среднее значение центров кластеров, в которые x_i входит: $y_i = (1/L)\sum_{l=1}^{L} c^l(x_i)$, сформировав из них набор $Y = \{y_1,\ldots,y_N\}$.

Шаг 3. Построить функцию $c^Y(y)$, используя $\mathrm{KMEANS}_{Y,K}(C_0^y)$, где центроиды C_0^y также заданы случайно.

Шаг 4. Построить результирующее разбиение $c^{final}(x)$, полагая $c^{final}(x_i) = c^Y(y_i)$, $I = 1,\ldots,N$.

Другими словами, алгоритм включает этап предобработки (шаги 1–2) и этап построения окончательного разбиения (шаг 3). Далее более подробно рассмотрим этап предобработки.

Утверждение 2. Пусть $P^1,\ldots,P^n$ – варианты разбиений фиксированного множества X на K кластеров, построенные алгоритмом K-means со случайно выбранными начальными центрами $C_0^1,...,C_0^n$, где координаты центров $c_{0,1}^1,...,c_{0,K}^1,...,c_{0,1}^n,...,c_{0,K}^n$ независимы в совокупности и одинаково распределены, $C_0^l \subset \mathbb{R}^M$, $l = 1,\ldots,L$ – ограниченные множества, и для каждого $x_i\in X$ определена последовательность векторов:

$$y_i^n = \frac{1}{n}\sum_{l=1}^{n} c^l(x_i),\ i = 1,...,N\ .$$

Тогда последовательность y_i^n сходится при $n\to\infty$, $i = 1,\ldots,N$.

Доказательство. Результат алгоритма K-means $C = \mathrm{KMEANS}_{X,K}(C_0)$ можно рассматривать как детерминированную функцию, ставящую в соответствие исходным центрам полученный набор. Поскольку координаты полученных центроидов зависят только от исходных центров, то они также являются независимыми, одинаково распределёнными случайными величинами. Заметим также, что второй момент $E(c_{ij})^2 < \infty$ для любых $i = 1,\ldots,N$, $j = 1,\ldots,M$ в силу ограниченности множества X. Тогда по закону больших чисел имеет место сходимость почти наверное последовательностей компонентов векторов y_i^n при $n\to\infty$. Утверждение доказано.

Доказанное свойство говорит о статистической устойчивости предложенного алгоритма. Заметим, что на шаге 3, кроме алгоритма K-means, может выбираться любой другой алгоритм кластеризации. Общая вычислительная трудоёмкость алгоритма АС включает вычислительную трудоёмкость процедур построения кластерного ансамбля ($O(KLMNT)$), усреднения центров ($O(LMN)$) и трудоёмкость процедуры на шаге 3 ($O(KLMNT)$) в случае K-means). Таким образом, этап построения кластерного ансамбля и усреднения центров имеет линейную трудоёмкость относительно числа объектов при фиксированных K, L, M, T. Для простоты описания алгоритма число кластеров K на каждом шаге полагается одинаковым, однако возможно использование различных значений.

3. Экспериментальная проверка

Проиллюстрируем работу алгоритма на примере кластеризации набора данных Cassini из библиотеки mlbench [19]. Набор представляет собой множество точек на плоскости, визуально разделимых на три группы: два вытянутых и изогнутых кластера и один округлой формы между ними. На рис. 1*а* представлено эталонное разбиение этого набора данных. Применение алгоритма K-means с $K = 3$ (рис. 1*б*) не даёт ожидаемого результата, и это несоответствие сохраняется независимо от начальных центров, а форма получаемых при этом кластеров практически не меняется. Чтобы добиться большего разнообразия вариантов разбиения, увеличим число кластеров до $K = 7$ и построим ансамбль из $L = 3$ элементов (рис. 1*в*). На шаге 3 был выбран агломеративный иерархический алгоритм ближайшей связи. Заметим, что после шага 2 количество точек в Y с уникальными координатами существенно сократилось (с 1000 до 18), поэтому перед применением иерархического алгоритма целесообразно исключить совпадающие точки. Итоговая ансамблевая кластеризация соответствует эталону (рис. 1*а*).

Для иллюстрации работы алгоритма было также использовано гиперспектральное изображение Pavia University из репозитория [20] размером 610×340 пикселей (рис. 2*а*). Изображение содержит $M = 103$ спектральных канала и представляет собой множество точек (пикселей) $\{x_i\}\subset\mathbb{R}^M$, где $x_i = (x_{i1},\ldots,x_{iM})$, x_{ij} – значение яркости в j-м спектральном канале. Это изображение предварительно было зашумлено: к нескольким каналам исходного изображения добавлялся шум по формуле:

$$x_{ij} = x_{ij} + \alpha\cdot(\max_i(x_{ij}) - \min_i(x_{ij})),$$

где α – реализация случайной величины, равномерно распределённой на интервале $[-0{,}4; 0{,}4]$. Эксперименты показали, что при зашумлении даже одного канала изображения алгоритм K-means даёт неудовлетворительные результаты.

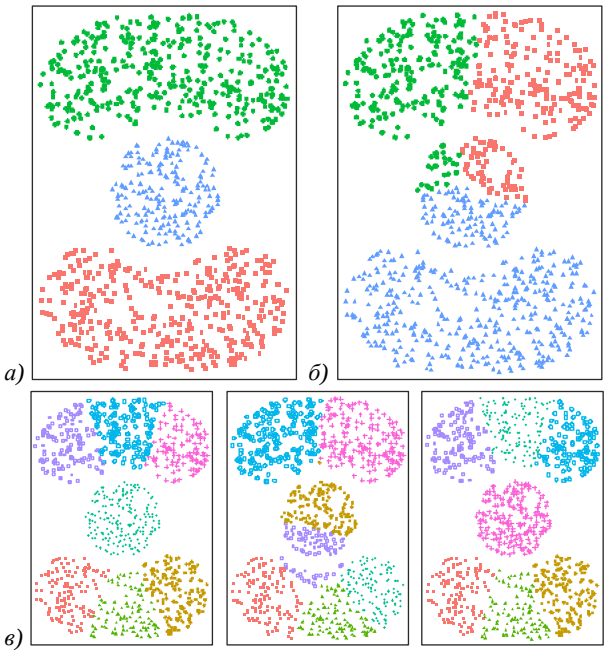

Рис. 1. а) Эталонная кластеризация набора данных Cassini; б) типичный результат применения K-means для K = 3; в) элементы ансамбля при K = 7, из которых получено итоговое разбиение

Для формирования ансамблевого решения был использован метод случайных подпространств, при котором на шаге 1 для построения каждого варианта разбиения случайным образом выбирается $D << M$ каналов изображения. При этом в алгоритме K-means для вычисления расстояний используется функция $d_W(x_i, c_k) = \sqrt{\sum_{j \in W} (x_{ij} - c_{kj})^2}$, где W – множество номеров каналов и $|W| = D$. А на шаге 2 алгоритма АС при вычислении центров кластеров и построении ансамблевого решения учитываются все M каналов.

На рис. 2*б, в* представлены результаты обработки алгоритмами K-means и АС изображения Pavia University, в котором каналы 13, 15 и 18 зашумлены в соответствии с указанной выше моделью. На рис. 2*б* хорошо видно негативное влияние шума на качество работы алгоритма K-means, которое проявляется в повышенной зернистости отдельных областей. На рис. 2*в* представлены результаты работы алгоритма АС. Из этого рисунка видно, что на нём отсутствуют негативные проявления шума.

4. Применение параллельных вычислений

Простейшая реализация алгоритма АС естественным образом вытекает из его описания: 1) последовательно выполняется L разбиений, 2) усредняются центры кластеров, 3) производится окончательное разбиение. При этом процесс построения каждого разбиения представляет собой последовательность шагов вычисления расстояний между объектами и текущими центрами кластеров $d_W(x_i, c_k)$, перераспределения объектов по кластерам и вычисления новых центров. Значительный объём вычислений в представленном алгоритме приходится на нахождение расстояний $d_W(x_i, c_k)$.

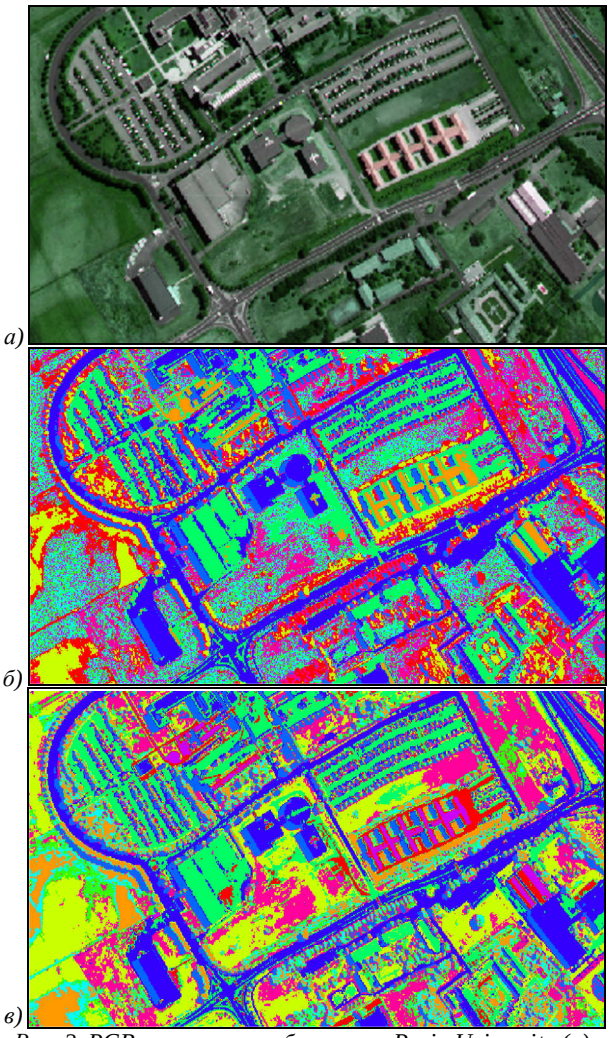

Рис. 2. RGB-композит изображения Pavia University (а); кластеризация с помощью алгоритма K-means при K = 10 (б); результат работы алгоритма АС при K = 10, L = 10, D = 10 (в)

Данная реализация является однопоточной; одним из способов повышения производительности является её модификация для выполнения в многопоточном окружении. Наиболее трудоёмким этапом алгоритма является построение L вариантов разбиений (шаг 1), поэтому это построение целесообразно распределить по разным ядрам процессора. Средние значения центров на шаге 2 также могут быть вычислены параллельно.

Последовательный и параллельный варианты алгоритма были реализованы на языке Java. Для экспериментов использовался 4-х ядерный процессор Intel i5-6400. При использовании параллельной реализации для обработки изображения Pavia University среднее время работы программы с параметрами $K = 10$, $L = 10$, $D = 10$ уменьшилось в 3,1 раза (рис. 3).

Ограничением предложенной параллельной реализации является совместное использование памяти параллельно исполняемыми потоками на шаге 1, а также использование последовательной реализации алгоритма K-means на шаге 3. Дальнейшая оптимизация данной параллельной процедуры может быть связана с применением низкоуровневых программных интерфейсов.

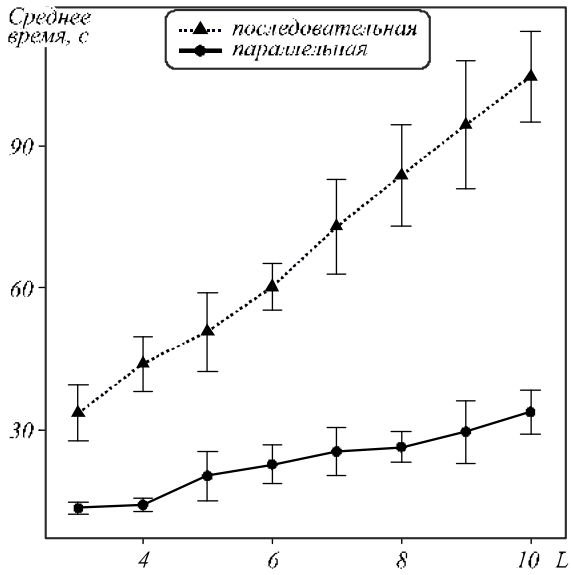

Рис. 3. Среднее время выполнения последовательной и параллельной программ в зависимости от числа элементов в ансамбле (L)

Альтернативным подходом к увеличению производительности может служить также привлечение графических процессоров. В работах [21, 22] показано, что их использование позволяет повысить производительность алгоритмов на несколько порядков. Преимущество использования графических процессоров для реализации алгоритма АС заключается в возможности вычислять расстояния между объектами и центрами кластеров одновременно для L разбиений на каждой итерации K-means. Например, в работе [23] предложена эффективная реализация вычисления матрицы расстояний для одного набора векторов на графическом процессоре, обеспечивающая ускорение в 20 раз по сравнению с однопоточной реализацией.

Заключение

В статье рассмотрен коллективный подход к решению задачи кластерного анализа. Предложен алгоритм усреднения центроидов АС, позволяющий построить консенсусное разбиение выборки на кластеры, используя набор разбиений этой выборки любым центроидным алгоритмом. Доказано, что центроиды консенсусного разбиения могут быть выражены через центроиды разбиений ансамбля. Доказана также статистическая устойчивость предложенного алгоритма. Эффективность алгоритма продемонстрирована на модельных и реальных данных. Предложена параллельная реализация алгоритма. В будущем планируется дальнейшее совершенствование алгоритма, связанное с применением графических процессоров и использованием весов при усреднении центроидов.

Литература

1. **Hastie, T.** The elements of statistical learning / T. Hastie, R. Tibshirani, J. Friedman. – 2nd ed. – New York: Springer-Verlag, 2009. – 745 p. – ISBN: 978-0-387-84857-0.
2. **Xu, Rui.** Clustering / R. Xu, D.C. Wunsch II. – Hoboken, NJ: John Wiley& Sons, Inc., 2009. – 368 p. – ISBN: 978-0-470-27680-8.
3. **Белим, С.В.** Выделение контуров на изображениях с помощью алгоритма кластеризации / С.В. Белим, П.Е. Кутлунин // Компьютерная оптика. – 2015. – Т. 39, № 1. – С. 119-124. – DOI: 10.18287/0134-2452-2015-39-1-119-124.
4. **Jain, A.K.** Data clustering: 50 years beyond K-means / A.K. Jain // Pattern Recognition Letters. – 2010. – Vol. 31, Issue 8. – P. 651-666. – DOI: 10.1016/j.patrec.2009.09.011.
5. **Ghaemi, R.** A survey: Clustering ensembles techniques / R. Ghaemi, M. Sulaiman, H. Ibrahim, N. Mustapha // Proceedings of World Academy of Science, Engineering and Technology. – 2009. – Vol. 38. – P. 644-653.
6. **Hore, P.** A scalable framework for cluster ensembles / P. Hore, L.O. Hall, D.B. Goldgof // Pattern Recognition. – 2009. – Vol. 42, Issue 5. – P. 676-688. – DOI: 10.1016/j.patcog.2008.09.027.
7. **Kashef, R.** Cooperative clustering / R. Kashef, M.S. Kamel // Pattern Recognition. – 2010. – Vol. 43, Issue 6. – P. 2315-2329. – DOI: 10.1016/j.patcog.2009.12.018.
8. **Jia, J.** Soft spectral clustering ensemble applied to image segmentation / J. Jia, B. Liu, L. Jiao // Frontiers of Computer Science in China. – 2011. – Vol. 5, Issue 1. – P. 66-78.
9. **Franek, L.** Ensemble clustering by means of clustering embedding in vector spaces / L. Franek, X. Jiang // Pattern Recognition. – 2014. – Vol. 47, Issue 2. – P. 833-842. – DOI: 10.1016/j.patcog.2013.08.019.
10. **Berikov, V.** Ensemble clustering based on weighted co-association matrices: Error bound and convergence properties / V. Berikov, I. Pestunov // Pattern Recognition. – 2017. – Vol. 63. – P. 427-436. – DOI: 10.1016/j.patcog.2016.10.017.
11. **Ghosh, J.** Cluster ensembles / J. Ghosh, A. Acharya // WIREs Data Mining Knowledge Discovery. – 2011. – Vol. 1. – P. 305-315. – DOI: 10.1002/widm.32.
12. **Пестунов, И.А.** Ансамблевый алгоритм кластеризации больших массивов данных / И.А. Пестунов, В.Б. Бериков, Е.А. Куликова, С.А. Рылов // Автометрия. – 2011. – Т. 47, № 3. – С. 49-58.
13. **Пестунов, И.А.** Иерархические алгоритмы кластеризации для сегментации мультиспектральных изображений / И.А. Пестунов, С.А. Рылов, В.Б. Бериков // Автометрия. – 2015. – Т. 51, № 4. – С. 12-22.
14. **Пестунов, И.А.** Сегментация многоспектральных изображений на основе ансамбля непараметрических алгоритмов кластеризации / И.А. Пестунов, В.Б. Бериков, Ю.Н. Синявский // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. – 2010. – Т. 5(31). – С. 56-64.
15. **Hubert, L.** Comparing partitions / L. Hubert, Ph. Arabie // Journal of Classification. – 1985. – Vol. 2. – P. 193-218.
16. **Strehl, A.** Cluster ensembles – a knowledge reuse framework for combining multiple partitions / A. Strehl, J. Ghosh // The Journal of Machine Learning Research. – 2003. – Vol. 3. – P. 583-617. – DOI: 10.1162/153244303321897735.
17. **Meilă, M.** Comparing clusterings by the variation of information / M. Meilă // Proceedings of 16th Conference on Learning Theory and 7th Kernel Workshop (COLT/Kernel 2003). – 2003. – P. 173-187.
18. **Wu, J.** External validation measures for k-means clustering: A data distribution perspective / J. Wu, J. Chen, H. Xiong, M. Xie // Expert Systems with Applications. – 2009. – Vol. 36, Issue 3, Part 2. – P. 6050-6061. – DOI: 10.1016/j.eswa.2008.06.093.
19. mlbench: Machine Learning Benchmark Problems [Электронный ресурс]. – URL: https://cran.r-project.org/web/packages/mlbench/index.html (дата обращения 02.03.17).

20. Hyperspectral Remote Sensing Scenes [Электронный ресурс]. – URL: http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes (дата обращения 02.03.17).
21. **Hossam, M.A.** Accelerated hyperspectral image recursive hierarchical segmentation using GPUs, multicore CPUs, and hybrid CPU/GPU cluster / M.A. Hossam, H.M. Ebied, M.H. Abdel-Aziz, M.F. Tolba // Journal of Real-Time Image Processing. – 2014. – P. 1-20. – DOI: 10.1007/s11554-014-0464-4.
22. **Рылов, С.А.** Использование графических процессоров NVIDIA при кластеризации мультиспектральных данных сеточным алгоритмом ССА / С.А. Рылов, И.А. Пестунов // Интерэкспо ГЕО-Сибирь. – 2015. – Т. 4, № 2. – С. 51-56.
23. **Chang, D.** Compute pairwise Euclidean distances of data points with GPUs / D. Chang, N.A. Jones, D. Li, M. Ouyang, R.K. Ragade // Proceedings of the IASTED International Symposium Computational Biology and Bioinformatics (CBB 2008). – 2008. – P. 278-283.

Сведения об авторах

Татарников Вадим Владимирович, 1990 года рождения, в 2014 году окончил магистратуру факультета информационных технологий в Новосибирском государственном университете, является аспирантом Института математики им. С. Л. Соболева СО РАН. Область научных интересов: коллективный кластерный анализ, программирование. E-mail: *vadim.tatarnikov@gmail.com* .

Пестунов Игорь Алексеевич, 1955 года рождения, в 1977 году окончил механико-математический факультет Новосибирского государственного университета, в 1998 году защитил диссертацию на соискание ученой степени кандидата наук. Работает ведущим научным сотрудником Института вычислительных технологий СО РАН. Область научных интересов: кластерный анализ, распознавание образов и анализ изображений.
E-mail: *pestunov@ict.sbras.ru* .

Бериков Владимир Борисович, 1964 года рождения, в 1986 году окончил механико-математический факультет Новосибирского государственного университета, в 1996 году защитил диссертацию на соискание ученой степени кандидата, а в 2007 году – доктора технических наук. Работает ведущим научным сотрудником Института математики СО РАН. Область научных интересов: математические методы анализа данных и их приложения в области обработки изображений. E-mail: *berikov@math.nsc.ru* .

ГРНТИ: 28.23.15.
Поступила в редакцию 26 апреля 2017 г. Окончательный вариант – 13 сентября 2017 г.

CENTROID AVERAGING ALGORITHM FOR A CLUSTERING ENSEMBLE

V.V. Tatarnikov[1], I.A. Pestunov[2], V.B. Berikov[3]
[1] Sobolev Institute of Mathematics SB RAS, Novosibirsk, Russia,
[2] Institute of Computational Technologies SB RAS, Novosibirsk, Russia,
[3]Novosibirsk State University, Novosibirsk, Russia

Abstract

A collective approach to cluster analysis is considered in the paper. An algorithm of centroid averaging is proposed. The algorithm allows constructing the consensus partition of a dataset into clusters, using a set of partitions built with any centroid-based algorithm. We discuss results of applying the proposed algorithm to modeled data and for the segmentation of hyperspectral images with noise channels. Some details of implementation in a multithreaded environment that allows increasing the algorithm performance are given.

Keywords: clustering ensemble, K-means, centroid, hyperspectral image analysis.
Citation: Tatarnikov VV, Pestunov IA, Berikov VB. Centroid averaging algorithm for a clustering ensemble. Computer Optics 2017; 41(5): 712-718. DOI: 10.18287/2412-6179-2017-41-5-712-718.

References

[1] Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. 2nd ed. New York: Springer-Verlag; 2009. ISBN: 978-0-387-84857-0.
[2] Xu R, Wunsch DC II. Clustering. Hoboken, NJ: John Wiley& Sons, Inc.; 2009. ISBN: 978-0-470-27680-8.
[3] Belim S, Kutlunin P. Boundary extraction in images using a clustering algorithm [In Russian]. Computer Optics 2015; 39(1): 119-124. DOI: 10.18287/0134-2452-2015-39-1-119-124.
[4] Jain AK. Data clustering: 50 years beyond K-means. Pattern Recognition Letters 2010; 31(8): 651-666. DOI: 10.1016/j.patrec.2009.09.011.
[5] Ghaemi R, Sulaiman M, Ibrahim H, Mustapha N. A survey: Clustering ensembles techniques. World Academy of Science, Engineering and Technology 2009; 38: 644-653.
[6] Hore P, Hall LO, Goldgof DB. A scalable framework for cluster ensembles. Pattern Recognition 2009; 42(5): 676-688. DOI: 10.1016/j.patcog.2008.09.027.
[7] Kashef R, Kamel MS. Cooperative clustering. Pattern Recognition 2010; 43(6): 2315-2329. DOI: 10.1016/j.patcog.2009.12.018.
[8] Jia J, Liu B, Jiao L. Soft spectral clustering ensemble applied to image segmentation. Frontier of Computer Science in China 2011; 5(1): 66-78.

[9] Franek L, Jiang X. Ensemble clustering by means of clustering embedding in vector spaces. Pattern Recognition 2014; 47(2): 833-842. DOI: 10.1016/j.patcog.2013.08.019.
[10] Berikov V, Pestunov I. Ensemble clustering based on weighted co-association matrices: Error bound and convergence properties. Pattern Recognition 2017; 63: 427-436. DOI: 10.1016/j.patcog.2016.10.017.
[11] Ghosh J, Acharya A. Cluster ensembles. WIREs Data Mining Knowledge Discovery 2011; 1: 305-315. DOI: 10.1002/widm.32.
[12] Pestunov I, Kulikova E, Rylov S, Berikov V. Ensemble of lustering algorithms for large datasets. Optoelectronics, Instrumentation and Data Processing 2011; 47(3): 245-252. DOI: 10.3103/S8756699011030071.
[13] Pestunov IA, Rylov SA, Berikov VB. Hierarchical clustering algorithms for segmentation of multispectral images. Optoelectronics, Instrumentation and Data Processing 2015; 51(4): 329-338. DOI: 10.3103/S8756699015040020.
[14] Pestunov IA, Berikov VB, Sinyavskiy YuN. Algorithm for multispectral image segmentation based on ensemble of nonparametric clustering algorithms [In Russian]. Vestnik SibGAU. 2010; 5(31): 56-64.
[15] Hubert L, Arabie Ph. Comparing partitions. Journal of Classification 1985; 2: 193-218.
[16] Strehl A, Ghosh J. Cluster ensembles–a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research 2003; 3: 583-617. DOI: 10.1162/153244303321897735.
[17] Meilă M. Comparing clusterings by the variation of information. Proceedings of 16th Conference on Learning Theory and 7th Kernel Workshop (COLT/Kernel 2003) 2003: 173-187.
[18] Wu J, Chen J, Xiong H, Xie M. External validation measures for k-means clustering: A data distribution perspective. Expert Systems with Applications 2009; 36(3:2): 6050-6061. DOI: 10.1016/j.eswa.2008.06.093.
[19] mlbench: Machine Learning Benchmark Problems. Source: ⟨https://cran.r-project.org/web/packages/mlbench/index.html⟩.
[20] Hyperspectral Remote Sensing Scenes. Source: ⟨http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes⟩.
[21] Hossam MA, Ebied HM, Abdel-Aziz MH, Tolba MF. Accelerated hyperspectral image recursive hierarchical segmentation using GPUs, multicore CPUs, and hybrid CPU/GPU cluster. Journal of Real-Time Image Processing 2014: 1-20. DOI: 10.1007/s11554-014-0464-4.
[22] Rylov SA, Pestunov IA. NVIDIA GPU for multispectral data clustering with grid-based algorithm CCA. Interexpo Geo-Siberia 2015; 2: 51-56.
[23] Darjen Chang, Nataniel A. Jones, Dazhuo Li, Ming Ouyang. Compute pairwise Euclidean distances of data points with GPUs. Proceedings of the IASTED International Symposium Computational Biology and Bioinformatics (CBB 2008) 2008: 278-283.

Authors' information

Vadim Vladimirovich Tatarnikov (b. 1990) graduated from Novosibirsk State University in 2014, obtained his Master's Degree in Computer Science. Currently he is Ph. D. student in Sobolev Institute of Mathematics. His research interests are collective clustering analysis and programming. E-mail: *vadim.tatarnikov@gmail.com* .

Igor Alekseevich Pestunov (b. 1955) graduated from Novosibirsk State University in 1977. He defended his PhD thesis in 1998. He is a key researcher at Institute of Computational Technologies of Siberian Branch of Russian Academy of Sciences. His current research interests include clustering, pattern recognition and image analysis. E-mail: *pestunov@ict.sbras.ru* .

Vladimir Borisovich Berikov (b. 1964) graduated from Faculty of Mechanics and Mathematics of Novosibirsk State University in 1986. He defended his PhD thesis in 1996, and in 2007, he defended a thesis for the scientific degree of the Doctor of Technical Science. Currently he works as the Leading Scientist at the Institute of mathematics SB RAS. Research interests are mathematical methods of data analysis and their application in image processing. E-mail: *berikov@math.nsc.ru* .

Received April 26, 2017. The final version – September 13, 2017.