Инженерный вестник Дона, №7 (2025)
ivdon.ru/ru/magazine/archive/n7y2025/10194

Кластеризация данных с использованием несимметричных мер близости

В.И. Шиян[1], *В.Н. Марков*[2]

[1]*Кубанский государственный университет, Краснодар*
[2]*Кубанский государственный технологический университет, Краснодар*

Аннотация: Статья посвящена разработке алгоритмов кластеризации данных с использованием несимметричных мер близости, актуальных в задачах с направленными взаимодействиями. Предложены два алгоритма: пошаговое формирование кластеров и модификация с итеративным уточнением центров. Проведены эксперименты, включая сравнение с методом k-медоидов. Результаты показали, что алгоритм с фиксированными центрами эффективен на малых данных, а алгоритм с пересчётом центров обеспечивает более точную кластеризацию. Выбор алгоритма зависит от требований к скорости и качеству.
Ключевые слова: кластеризация, несимметричные меры близости, алгоритмы кластеризации, итеративное уточнение, k-медоиды, направленные взаимодействия, адаптивные методы.

Постановка задачи и обоснование необходимости новых алгоритмов

В задачах кластеризации традиционно используются симметричные меры близости, такие как евклидово расстояние или косинусное сходство [1, 2]. Однако в реальных задачах часто возникает необходимость в несимметричных мерах близости. Например, в социальных сетях направленные отношения между пользователями – подписки, влияния – или в биологических данных асимметричные взаимодействия по типу хищник-жертва требуют мер, где близость объекта x к y может отличаться от близости y к x [3, 4]. Симметричные меры упрощают постановку задачи и позволяют применять классические алгоритмы, такие как k-средних или алгоритм пространственной кластеризации приложений с шумом на основе плотности (Density-Based Spatial Clustering of Applications with Noise – DBSCAN) [5, 6]. Однако они игнорируют несимметричность меры, что ведёт к потере информации о структурных особенностях данных и некорректным результатам. Это подтверждает необходимость создания алгоритмов, адаптированных для работы с несимметричными мерами близости [7, 8].

Одним из ключевых аспектов кластеризации с несимметричной мерой близости является введение порогового значения R, которое определяет максимально допустимую близость между объектами внутри одного кластера. Это значение играет роль «границы», разделяющей объекты на те, которые могут находиться в одном кластере, и те, которые должны быть отнесены к разным кластерам [9].

Условие формирования кластеров на основе порога R

Для того чтобы гарантировать, что объекты внутри каждого кластера «близки» друг к другу, а объекты из разных кластеров «далеки», вводится пороговое значение R. Это значение определяет максимально допустимую меру близости между объектами внутри одного кластера. Формально, условие формирования кластеров можно описать следующим образом [10, 11].

Два объекта x и y могут находиться в одном кластере C_i, если их мера близости $W(x, y)$ не превышает порога R:

$$W(x, y) \le R \ \forall x, y \in C_i .$$

Это условие гарантирует, что все объекты внутри кластера достаточно близки друг к другу согласно мере W [12].

Два объекта x и y должны находиться в разных кластерах C_i и C_j, если их мера близости $W(x, y)$ превышает порог R:

$$W(x, y) > R \ \forall x \in C_i, y \in C_j, i \ne j .$$

Это условие гарантирует, что объекты из разных кластеров достаточно далеки друг от друга [13].

Свойства симметричной меры близости

Симметричные меры близости $W : X \times X \to$ □ обладают рядом свойств [14].

Симметричность:

$$W(x, y) = W(y, x) \ \forall x, y \in X .$$

Неотрицательность:

$$W(x, y) \geq 0 \ \forall x, y \in X .$$

Рефлексивность:

$$W(x, x) = 0 \ \forall x \in X .$$

Треугольное неравенство (для метрик):

$$W(x, z) \leq W(x, y) + W(y, z) \ \forall x, y, z \in X .$$

Постановка задачи кластеризации с несимметричной мерой близости

Пусть дан набор объектов $X = \{x_1, x_2, \ldots, x_n\}$ и несимметричная мера близости $W : X \times X \to$ □ , где $W(x_i, x_j)$ определяет близость объекта x_j к объекту x_i, и имеет свойство $W(x, y) \neq W(y, x)$ [5].

Задача кластеризации заключается в разбиении множества X на k кластеров $C = \{C_1, C_2, \ldots, C_k\}$ таким образом, чтобы объекты внутри каждого кластера были «близки» друг к другу согласно мере W, а объекты из разных кластеров были «далеки», иными словами, необходимо стремиться минимизировать суммарное расстояние внутри кластеров и максимизировать суммарное расстояние между ними [6]. Введём целевую функцию (1)

$$F(C) = \sum_{i=1}^{k} \left(\sum_{x, y \in C_i} W(x, y) \right) + \lambda k \to \min , \qquad (1)$$

где $W(x, y)$ – мера близости объекта y к объекту x, $\lambda \in [0; 1]$ – коэффициент регуляризации, который определяет, насколько сильно штрафуется увеличение числа кластеров.

Первое слагаемое $\sum_{i=1}^{k}\left(\sum_{x,\,y\in C_i} W(x,\,y)\right)$ выражает стремление к максимальной плотности кластеров через минимизацию суммарных внутрикластерных расстояний. Второе слагаемое k отражает принцип минимальной сложности модели, накладывая линейный штраф на увеличение количества кластеров.

Классические алгоритмы кластеризации, такие как k-средних, DBSCAN или иерархическая кластеризация, предполагают симметричность меры, что делает их неприменимыми для несимметричных мер. Для решения этой проблемы предлагаются два новых алгоритма, которые позволяют эффективно проводить кластеризацию.

Алгоритм 1: Пошаговое формирование кластеров

Алгоритм 1 основан на идее последовательного формирования кластеров с использованием порогового значения R. Основные шаги алгоритма:

Инициализация. Начинаем с пустого множества кластеров $C=\varnothing$ и множества некластеризованных объектов $U=X$.

Поиск минимальной пары. На каждом шаге находим пару объектов $(x,\,y)$ с минимальной мерой близости $W(x,\,y)$ среди всех пар объектов из U.

Формирование кластера. Если $W(x,\,y)\le R$, то объекты x и y добавляются в новый кластер C_k. Далее все объекты, близкие к x или y, согласно порогу R, также добавляются в этот кластер.

Перераспределение объектов. После формирования кластера C_k проверяем, можно ли перераспределить объекты из других кластеров в C_k, если они окажутся ближе к x или y, чем к своим текущим центрам.

Объединение одноэлементных кластеров. Если в процессе кластеризации образуются кластеры, состоящие из одного объекта, они объединяются с ближайшим кластером.

Процесс формирования кластера можно описать следующим образом. Пусть C_k – текущий кластер, x – его центр – первый объект, добавленный в пустой кластер. Тогда объект y добавляется в C_k, если выполняется условие: $W(x, y) \le R$.

После добавления объекта y в кластер C_k, для всех объектов $z \in U$ проверяется условие: $W(x, z) \le R$ или $W(y, z) \le R$.

Если условие выполняется, объект z также добавляется в C_k.

Вычислительная сложность алгоритма 1 – $O(n^2)$, где n – количество объектов в наборе данных. Это связано с необходимостью поиска минимальной пары объектов на каждом шаге, что требует попарного сравнения всех объектов.

Алгоритм 2: Итеративное уточнение центров кластеров

Алгоритм 2 является модификацией алгоритма 1 и включает дополнительный шаг итеративного уточнения центров кластеров. Основное отличие заключается в том, что после формирования кластера C_k, его центр – объект x обновляется по критерию минимального суммарного расстояния от центра до остальных объектов кластера. Это позволяет улучшить качество кластеризации, особенно в случаях, когда начальный выбор центра был неоптимальным.

Инициализация. Аналогично алгоритму 1, начинаем с пустого множества кластеров $C = \varnothing$ и множества некластеризованных объектов $U = X$.

Поиск минимальной пары. Находим пару объектов (x, y) с минимальной мерой близости $W(x, y)$ среди всех пар объектов из U.

Формирование кластера. Если $W(x, y) \le R$, то объекты x и y добавляются в новый кластер C_k. Далее все объекты, близкие к x или y, согласно порогу R, также добавляются в этот кластер.

Уточнение центра кластера. После формирования кластера C_k, его центр x пересчитывается как объект, минимизирующий суммарное расстояние внутри кластера (2):

$$x' = \arg\left(\min_{x \in C_k} \left(\sum_{y \in C_k} W(x, y) \right) \right). \quad (2)$$

Перераспределение объектов. Аналогично алгоритму 1, объекты из других кластеров могут быть перераспределены в C_k, если они окажутся ближе к новому центру x'.

Объединение одноэлементных кластеров. Если в процессе кластеризации образуются кластеры, состоящие из одного объекта, они объединяются с ближайшим кластером.

Вычислительная сложность алгоритма 2 – $O(n^2 \cdot k)$, где n – количество объектов, k – количество кластеров. Дополнительная сложность по сравнению с алгоритмом 1 обусловлена итеративным уточнением центров кластеров, которое выполняется для каждого кластера на каждом шаге.

Различие между алгоритмом 1 и алгоритмом 2

Основное отличие между алгоритмом 1 и алгоритмом 2 заключается в наличии шага уточнения центра кластера в алгоритме 2. В алгоритме 1 центр кластера фиксируется на этапе его формирования и не изменяется в дальнейшем. В алгоритме 2 центр кластера итеративно уточняется, что

позволяет улучшить качество кластеризации, особенно в случаях, когда начальный выбор центра был неоптимальным.

Эксперименты

Эксперименты проводились для оценки эффективности предложенных алгоритмов кластеризации на различных наборах данных с разным количеством объектов. В качестве тестовых данных использовались синтетические наборы – точки в двумерном пространстве с заданными направлениями близости, а также классический набор Iris [15]. Во всех экспериментах коэффициент λ в целевой функции устанавливался равным единице, делённой на число объектов n, что обеспечивало автоматическую адаптацию штрафа к масштабу данных.

В каждом эксперименте варьировались значения порога R, что позволило изучить его влияние на качество кластеризации. Результаты включают анализ количества объектов в кластерах, суммы расстояний внутри кластеров и времени выполнения алгоритмов. Это позволило сравнить скорость и точность предложенных алгоритмов в различных условиях, включая малые и большие наборы данных, а также данные с выраженной асимметрией.

Пример 1: синтетические данные с 25 объектами. В первом эксперименте с синтетическими данными, содержащими 25 объектов, при пороге $R = 0{,}8$ оба алгоритма показали идентичные результаты: сформировали по 7 кластеров с одинаковым распределением объектов и суммой расстояний, равной 27,91. Время выполнения алгоритма 1 составило 0,002 секунды, а алгоритма 2 – 0,001 секунды. При увеличении порога до $R = 1{,}2$ количество кластеров возросло до 8, но сумма расстояний снизилась до 23,13, что свидетельствует о более компактной группировке объектов. Оба

алгоритма снова показали одинаковые результаты, выполнившись за 0,001 секунды.

На рис. 1 представлена визуализация кластеров для синтетических данных с 25 объектами при $R = 0,8$.

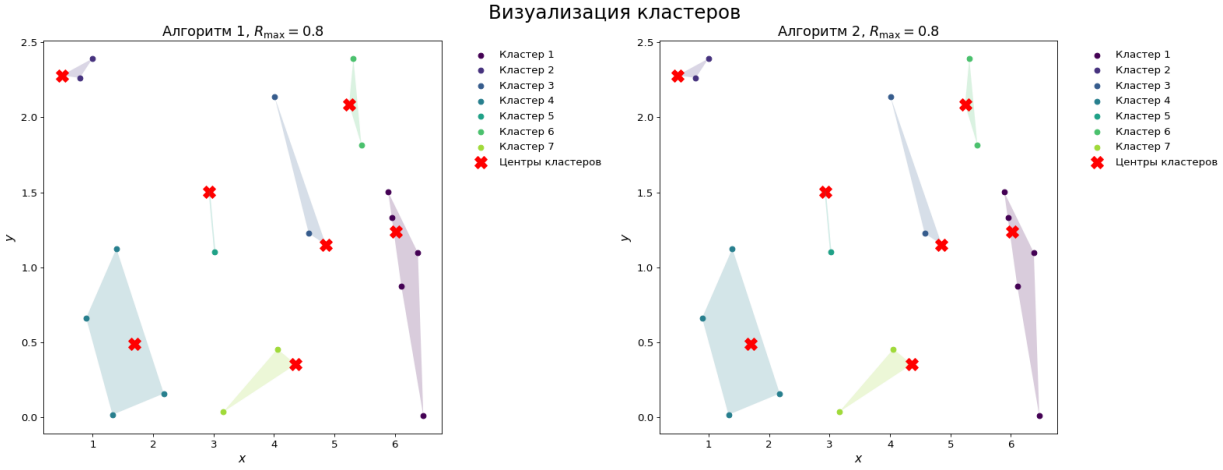

Рис. 1. – Визуализация кластеров для синтетических данных с 25 объектами при $R = 0,8$

На рис. 2 представлены результаты кластеризации при $R = 0,8$.

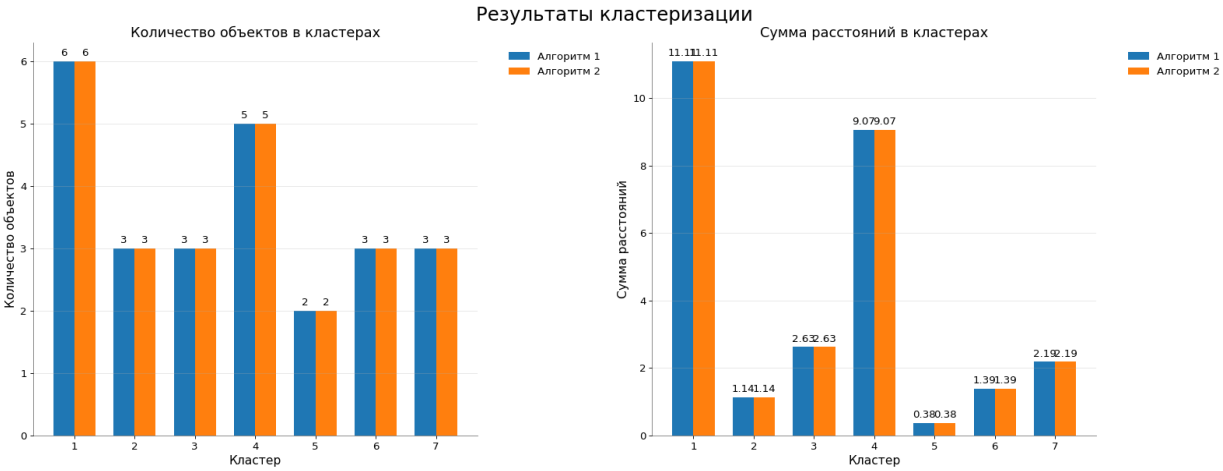

Рис. 2. – Результаты кластеризации для синтетических данных с 25 объектами при $R = 0,8$

На рис. 3 представлена визуализация кластеров для синтетических данных с 25 объектами при $R = 1,2$.

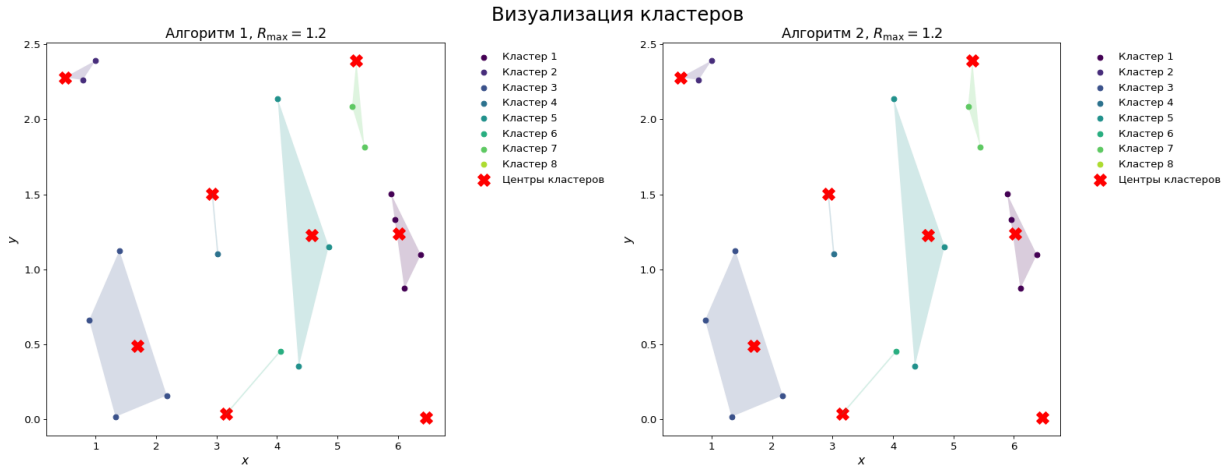

Рис. 3. – Визуализация кластеров для синтетических данных с 25 объектами при $R = 1,2$

На рис. 4 представлены результаты кластеризации при $R = 1,2$.

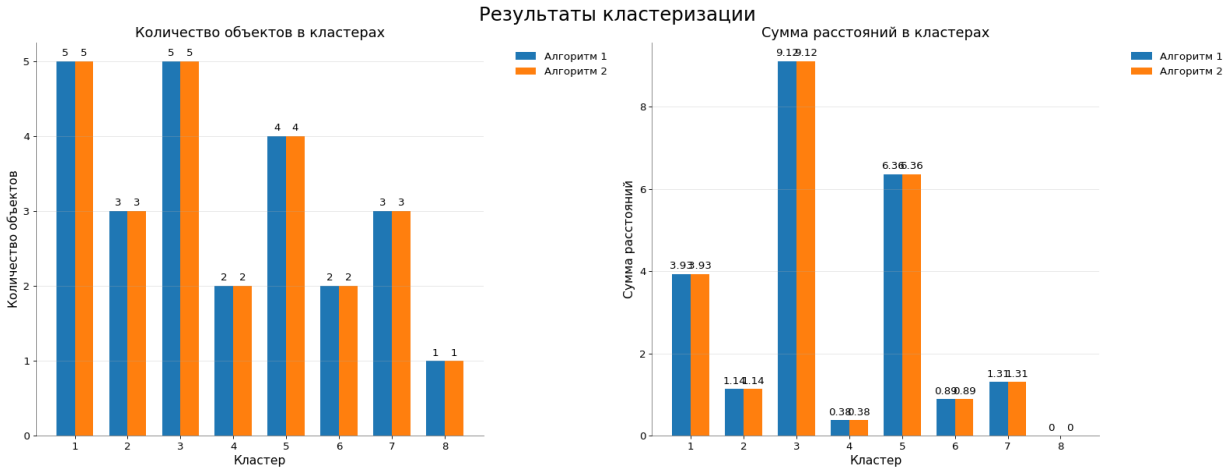

Рис. 4. – Результаты кластеризации для синтетических данных с 25 объектами при $R = 1,2$

Пример 2: синтетические данные с 125 объектами. Во втором эксперименте с 125 объектами при $R = 1,2$ алгоритм 1 сформировал 11 кластеров с суммой расстояний, равной 460,78, а алгоритм 2 – 11 кластеров с близким, но немного большей суммой расстояний, равной 466,17. Время выполнения обоих алгоритмов составило 0,036 секунд. При увеличении порога до $R = 1,7$ количество кластеров сократилось до 6, а сумма расстояний значительно выросла: до 1191,99 у алгоритма 1 и 1200,73 у

алгоритма 2, что связано с объединением более удалённых объектов. Время выполнения уменьшилось до 0.018-0.019 секунд, демонстрируя зависимость скорости работы от количества кластеров.

На рис. 5 представлена визуализация кластеров для синтетических данных с 125 объектами при $R = 1,2$.

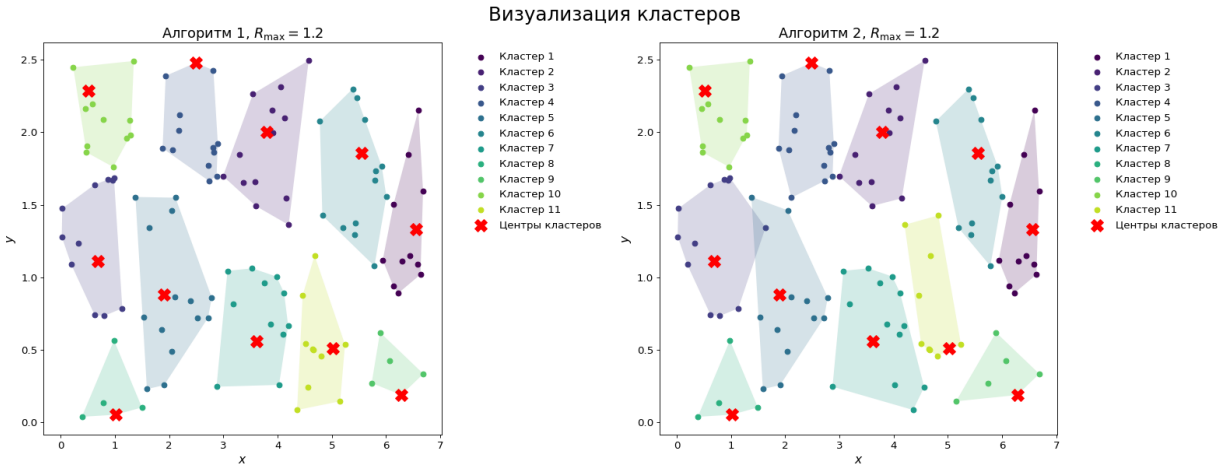

Рис. 5. – Визуализация кластеров для синтетических данных с 125 объектами при $R = 1,2$

На рис. 6 представлены результаты кластеризации для $R = 1,2$.

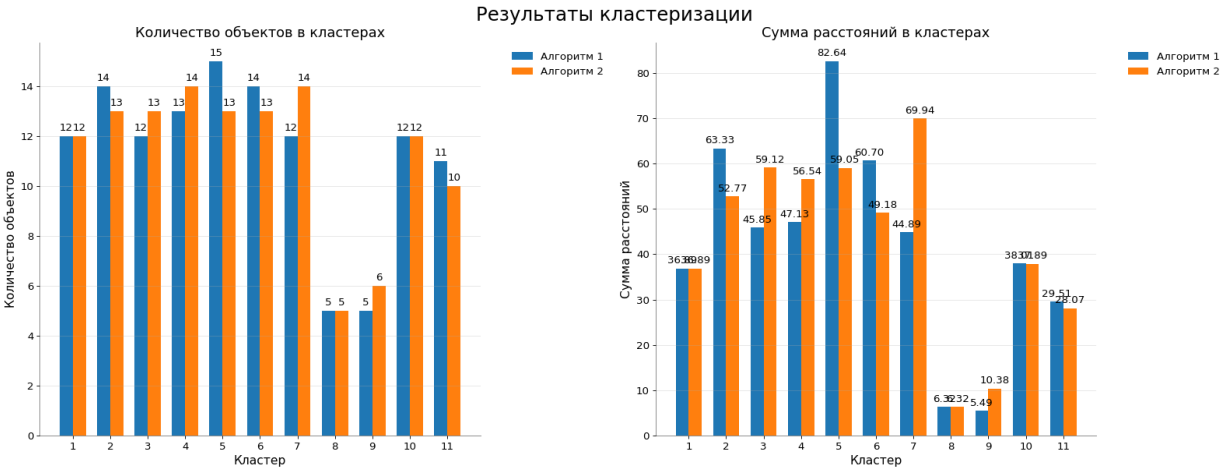

Рис. 6. – Результаты кластеризации для синтетических данных с 125 объектами при $R = 1,2$

На рис. 7 представлена визуализация кластеров для синтетических данных с 125 объектами при $R = 1,7$.

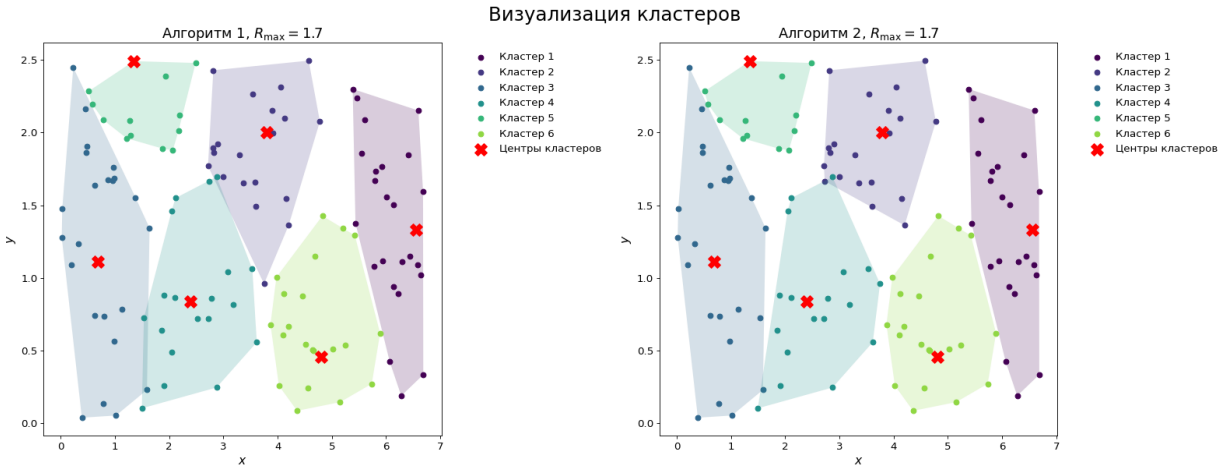

Рис. 7. – Визуализация кластеров для синтетических данных с 125 объектами при $R = 1,7$

На рис. 8 представлены результаты кластеризации при $R = 1,7$.

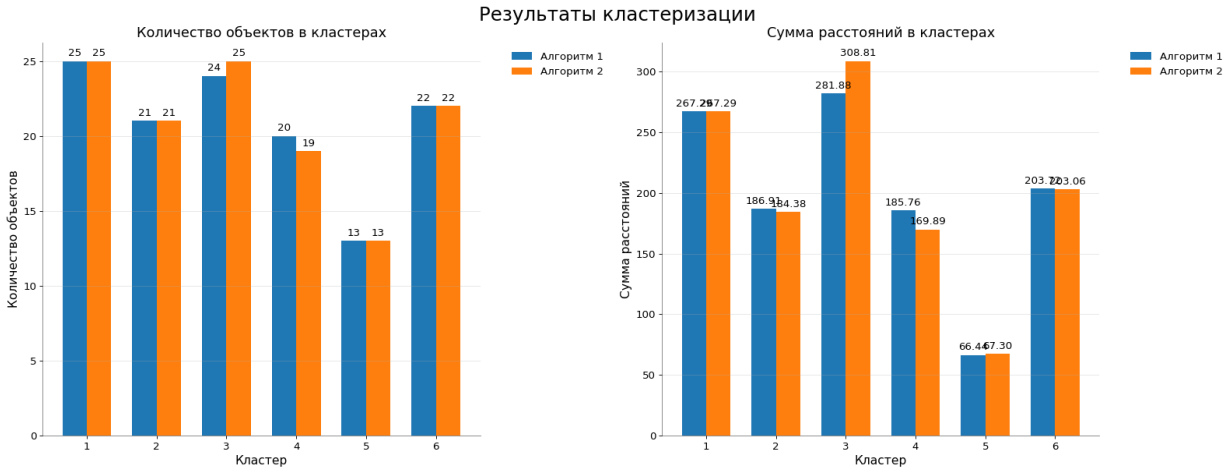

Рис. 8. – Результаты кластеризации для синтетических данных с 125 объектами при $R = 1,7$

Пример 3: Iris Data Set. В третьем эксперименте с Iris Data Set при $R = 2,1$ алгоритм 2 сформировал 4 кластера с суммой расстояний, равным 531,08, тогда как метод k-медоидов показал значительно худший результат, равный 1007,74, несмотря на одинаковое количество кластеров. При уменьшении порога до $R = 0,9$ алгоритм 2 создал 6 кластеров с суммой расстояний 216,53, в то время как k-медоиды достигли значения 255,03.

Время выполнения алгоритма 2 было выше – 0,0185 секунд против 0,003, но его качество кластеризации оказалось значительно лучше.

На рис. 9 представлена визуализация кластеров для Iris Data Set при $R = 2{,}1$.

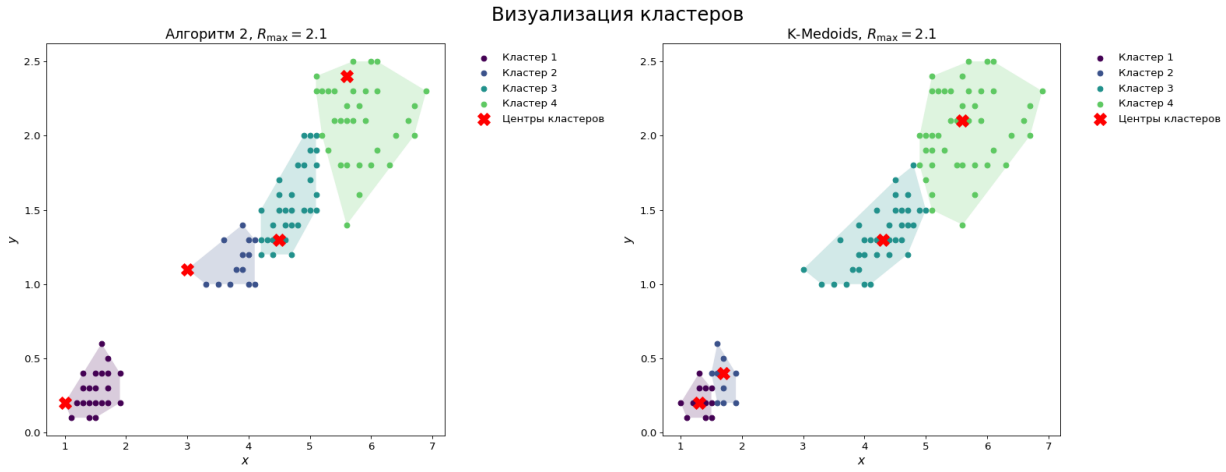

Рис. 9. – Визуализация кластеров для Iris Data Set при $R = 2{,}1$

На рис. 10 представлены результаты кластеризации при $R = 2{,}1$.

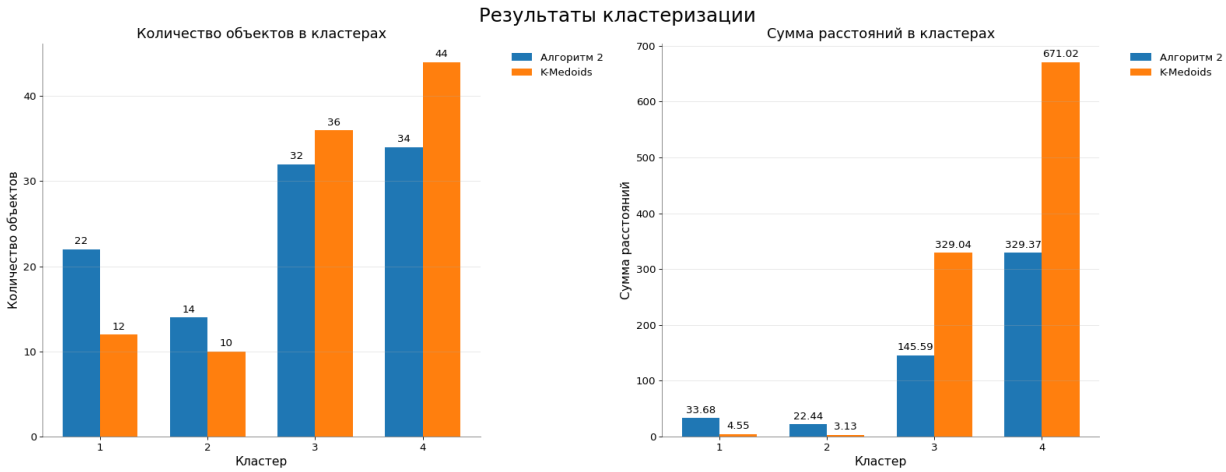

Рис. 10. – Результаты кластеризации для Iris Data Set при $R = 2{,}1$

На рис. 11 представлена визуализация кластеров для Iris Data Set при $R = 0{,}9$.

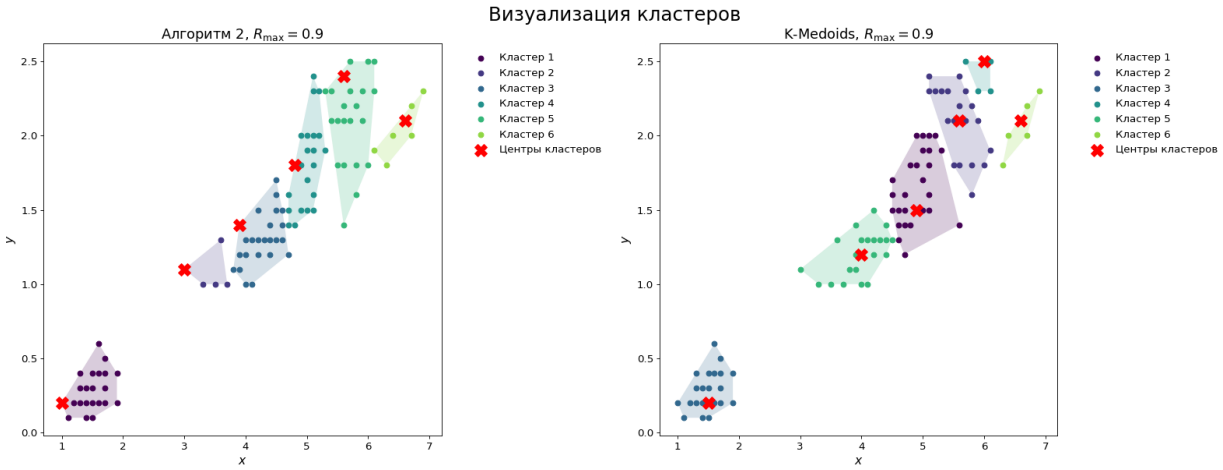

Рис. 11. – Визуализация кластеров для Iris Data Set при $R = 0{,}9$

На рис. 12 представлены результаты кластеризации при $R = 0{,}9$.

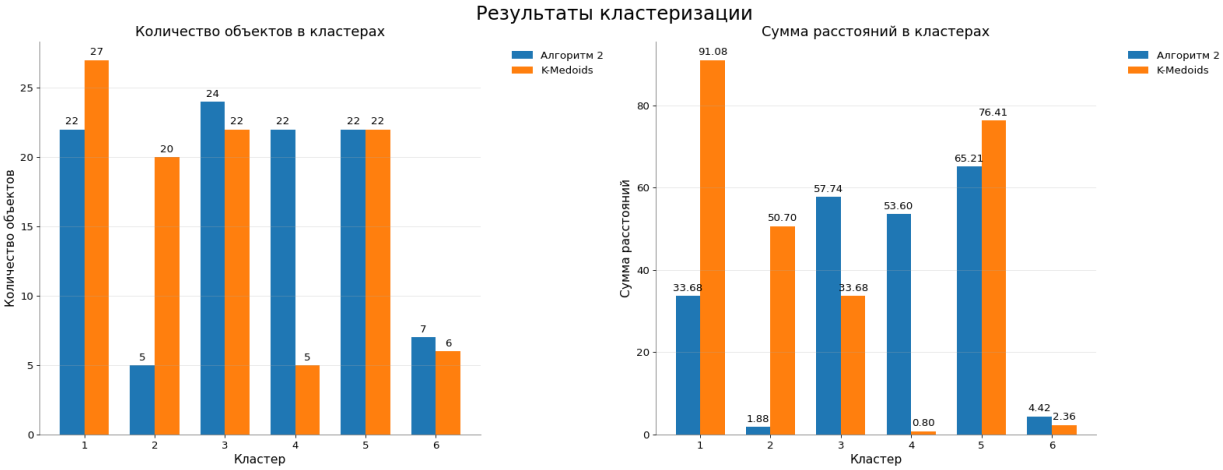

Рис. 12. – Результаты кластеризации для Iris Data Set при $R = 0{,}9$

Проведённые эксперименты показали, что алгоритм 2 особенно хорошо справляется с данными, где группы объектов расположены неравномерно и имеют сложное строение. В то время как метод k-медоидов показывает хорошие результаты на простых, равномерно распределённых данных, он хуже работает в случаях, когда объекты образуют несколько плотных скоплений с разной структурой и плотностью. Алгоритм 2 лучше адаптируется к таким сложным случаям благодаря постоянному обновлению центров кластеров, что позволяет точнее учитывать особенности расположения объектов в каждом отдельном участке данных.

В целом, оба алгоритма показали свою эффективность, но выбор между ними зависит от конкретной задачи. Если важна скорость, то предпочтение стоит отдать алгоритму 1, а если важна точность и сбалансированность кластеризации – алгоритму 2.

Литература

1. Айвазян С.А., Бухштабер В.М., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: Классификация и снижение размерности. М.: Финансы и статистика, 1989. 607 с.

2. Tryon R.C. Cluster analysis. London: Ann Arbor Edwards Bros, 1939. 139 p.

3. Мандель И.Д. Кластерный анализ. М.: Финансы и статистика, 1988. 176 с.

4. Hartigan J.A. Clustering Algorithms. New York: John Wiley & Sons Inc, 1975.

5. Хайдуков Д.С. Применение кластерного анализа в государственном управлении // Философия математики: актуальные проблемы. М.: МАКС Пресс, 2009. С. 287.

6. Jain A.K., Murty M.N., Flynn P.J. Data clustering: a review // ACM Computing Surveys. 1999. Vol. 31, №3. P. 264-323.

7. Жамбю М. Иерархический кластер-анализ и соответствия. М.: Финансы и статистика, 1988. 345 с.

8. Soni N., Ganatra A. Categorization of several clustering algorithms from different perspective: a review // Int. J. of Advanced Research in Computer Science and Software Engineering. 2012. Vol. 2, №8. P. 63-68.

9. Дюран Б., Оделл П. Кластерный анализ. М.: Статистика, 1977. 128 с.

10. Чернов А.В., Бутакова М.А., Шевчук П.С. Кластеризация данных методом растущего нейронного газа // Инженерный вестник Дона, 2020, №7 URL: ivdon.ru/ru/magazine/archive/n7y2020/6537/.

11. Гранков М.В., Аль-Габри В.М., Горлова М.Ю. Анализ и кластеризация основных факторов, влияющих на успеваемость учебных групп вуза // Инженерный вестник Дона, 2016, №4 URL: ivdon.ru/ru/magazine/archive/n4y2016/3775/.

12. Кельманов А.В., Хамидуллин С.А. Приближенный полиномиальный алгоритм для одной задачи бикластеризации последовательности // Журн. вычисл. матем. и мат. физики. 2015. Т. 55, №6. С. 1076-1085.

13. Шалымов Д.С. Алгоритмы устойчивой кластеризации на основе индексных функций и функций устойчивости // Стохастическая оптимизация в информатике. СПб.: Изд-во С.-Петербургского университета, 2008. Вып. 4. С. 236-248.

14. Madhulatha T.S. An overview on clustering methods // IOSR J. of Engineering. 2012. Vol. 2, №4. pp. 719-725.

15. UCI Machine Learning Repository: Iris Data Set. URL: archive.ics.uci.edu/ml/datasets/Iris (дата обращения: 25 мая 2025).

References

1. Ayvazyan S.A., Bukhshtaber V.M., Enyukov I.S., Meshalkin L.D. Prikladnaya statistika: Klassifikatsiya i snizhenie razmernosti [Applied statistics: Classification and dimensionality reduction]. Moscow: Finansy i statistika, 1989. 607 p.

2. Tryon R.C. Cluster analysis. London: Ann Arbor Edwards Bros, 1939. 139 p.

3. Mandel' I.D. Klasternyy analiz [Cluster analysis]. Moscow: Finansy i statistika, 1988. 176 p.

4. Hartigan J.A. Clustering Algorithms. New York: John Wiley & Sons Inc, 1975.

5. Khaydukov D.S. Filosofiya matematiki: aktual'nye problemy [Philosophy of mathematics: current issues]. Moskva: MAKS Press, 2009. P. 287.

6. Jain A.K., Murty M.N., Flynn P.J. ACM Computing Surveys. 1999. Vol. 31, №3. Pp. 264-323.

7. Zhambyu M. Ierarkhicheskiy klaster-analiz i sootvetstviya [Hierarchical cluster analysis and correspondences]. Moskva: Finansy i statistika, 1988. 345 p.

8. Soni N., Ganatra A. Int. J. of Advanced Research in Computer Science and Software Engineering. 2012. Vol. 2, №8. Pp. 63-68.

9. Dyuran B., Odell P. Klasternyy analiz [Cluster analysis]. Moscow: Statistika, 1977. 128 p.

10. Chernov A.V., Butakova M.A., Shevchuk P.S. Inzhenernyj vestnik Dona, 2020, №7. URL: ivdon.ru/ru/magazine/archive/n7y2020/6537/.

11. Grankov M.V., Al'-Gabri V.M., Gorlova M.Yu. Inzhenernyj vestnik Dona, 2016, №4. URL: ivdon.ru/ru/magazine/archive/n4y2016/3775/.

12. Kel'manov A.V., Khamidullin S.A. Zhurn. vychisl. matem. i mat. fiziki. 2015. Vol. 55, №6. Pp. 1076-1085.

13. Shalymov D.S. Stokhasticheskaya optimizatsiya v informatike. Saint Petersburg: Izd-vo S.-Peterburgskogo universiteta, 2008. Iss. 4. Pp. 236-248.

14. Madhulatha T.S. IOSR J. of Engineering. 2012. Vol. 2, №4. Pp. 719-725.

15. UCI Machine Learning Repository: Iris Data Set. URL: archive.ics.uci.edu/ml/datasets/Iris (accessed: 05/25/2025).

Дата поступления: 6.05.2025

Дата публикации: 25.06.2025