Научная статья
УДК 004.8
EDN SBDRVK
DOI 10.17150/2713-1734.2023.5(4).379-391

П.А. Головинский
Воронежский государственный технический университет, г. Воронеж, Российская Федерация

А.С. Тарасова
Воронежский государственный технический университет, г. Воронеж, Российская Федерация

Вязкий гравитационный алгоритм кластеризации многомерных данных

Аннотация. Кластеризация является одним из первых стандартных шагов при анализе больших данных. Она необходима для дальнейшего решения задач классификации и группового прогноза. Нами исследуется вязкая модификация гравитационного алгоритма кластеризации данных (VGSA), которая является развитием уже зарекомендовавшего себя ранее подхода. Отдельные записи данных рассматриваются в VGSA как точки в многомерном пространстве, между которыми действует парное центральное притяжение. Массы взаимодействующих точек приняты одинаковыми, что соответствует специфике кластеризации в отличие от задачи поиска оптимального значения целевой функции, при решении которой массы частиц увеличиваются по мере приближения к экстремуму. Обсуждается выбор вида парного взаимодействия в зависимости от предполагаемой структуры данных. Наличие большой вязкости понижает порядок динамических уравнений движения путем исключения из них ускорения. Полученные укороченные уравнения задают устойчивое движение системы, что гарантирует воспроизведение результатов при повторном запуске алгоритма. Устойчивость системы уравнений доказывается с помощью функции Ляпунова, являющейся аналогом физической потенциальной энергии. Выключение взаимодействия частиц на малых расстояний между ними обеспечивает автоматический механизм иерархической кластеризации на разных временах работы алгоритма с конечным образованием единого кластера. Прослежена связь VGSA с принципом действия самоорганизующихся карт Кохонена, который соответствует гравитационному перераспределению пробных частиц. Работа алгоритма протестирована на базе данных в сравнении с методами кластеризации K-средних, карт Кохонена и стандартного гравитационного алгоритма. Оценивалась скорость и точность кластеризации. Сделан вывод о преимуществе применения VGSA к большим данным с учетом автоматического определения числа кластеров, возможности коррекции при обновлении записей и неточного задания данных.

Ключевые слова. Кластеризация, большие данные, гравитационный алгоритм, вязкость, устойчивость по Ляпунову.

Информация о статье. Дата поступления: 15 мая 2023 г.; дата принятия к публикации: 20 ноября 2023 г.; дата онлайн-размещения: 12 декабря 2023 г.

ISSN 2713-1734

Original article

P.A. Golovinsky
Voronezh State Technical University,
Voronezh, Russian Federation
A.S. Tarasova
Voronezh State Technical University,
Voronezh, Russian Federation

Viscous Gravity Algorithm for Clustering Multidimensional Data

Abstract. Clustering is one of the first standard steps for big data analysis. It is necessary for further solving problems of classification and group forecasting. We study a viscous modification of the gravitational data clustering algorithm (VGSA), which develop already proven approach. Individual data records are considered in VGSA as points in multidimensional space, between which a paired central attraction acts. The masses of the interacting points are assumed to be the same, which corresponds to the specifics of clustering, in contrast to the problem of finding the optimal value of the objective function, in which the masses of particles increase as they approach the extremum. The choice of the type of pair interaction depending on the proposed data structure is discussed. The presence of high viscosity lowers the order of the dynamic equations of motion by excluding acceleration from them. The obtained shortened equations define the stable motion of the system, which guarantees the reproduction of the results when the algorithm is restarted. The stability of the system of equations is proved using the Lyapunov function, which is an analogue of the physical potential energy. Turning off the interaction of particles at small distances between them provides an automatic mechanism for hierarchical clustering at different stages of the algorithm with the final formation of a single cluster. The relationship between VGSA and the operating principle of Kohonen's self-organizing maps, which corresponds to the gravitational redistribution of test particles, is traced. The performance of the algorithm has been tested on the database in comparison with the methods of K-means clustering, Kohonen maps and the standard gravity algorithm. The speed and accuracy of clustering were evaluated. The conclusion is made about the advantage of applying VGSA to big data, taking into account the automatic determination of the number of clusters, the possibility of correction when updating records, and inaccurate data specification.

Keywords. Clustering, big data, gravity algorithm, viscosity, Lyapunov stability.

Article info. Received 15 May, 2023; Accepted 20 November, 2023; Available online 12 December, 2023.

Введение

Решение задачи кластеризации больших данных обычно является необходимым первым шагом к дальнейшему более детальному их анализу. Кластеры данных формируются по выбранному критерию близости абстрактных точек в многомерном пространстве, представляющем записи, путем введения той или иной метрики [1]. В отличие от обычного геометрического пространства, данные могут быть содержать разнородные величины с различной размерностью. Поэтому введение метрики непосредственно в исходном пространстве данных невозможно. Стандартным решением является приведение всех данных к безразмерному интервалу путем нормировки каждого из отдельных параметров на максимальное или среднее его значение в базе [2]. Преимущество нормировки на среднее значение состоит в исключении искажения дальнейших оценок случайными большими выбросами.

К полученным безразмерным записям можно применить ту или иную метрику, в зависимости от решаемой задачи. Если специфика решаемой задачи неизвестна или непонятна, то чаще всего используют метрику Евклида. В различных случаях применяется также метрика Манхэттен, расстояние Чебышева, расстояние Махаланобиса или степенное расстояние [3; 4]. Сам алгоритм кластеризации призван обеспечить попадание в одно множество близко расположенных данных, тогда как расположенные на большом расстоянии данные попадут в другие множества. Такое определение игнорирует распределение плотности частиц в пространстве. Отметим, что возможность оценить плотность данных возникает только при наличии больших объемов записей. Отсюда понятно отсутствие единого универсального алгоритма кластеризации данных произвольной природы и структуры.

Кластеризация является первым шагом для реализации компьютерных решений по классификации данных и прогнозированию динамики коллективного поведения типовых групп объектов. В машинном обучении кластеризация относится к задачам обучения без учителя и не требует ручной разметки данных. Меньшие требования к подготовке данных кластеризации, по сравнению с размеченными данными при обучении с учителем, приводят одновременно к меньшей точности результатов классификации. Безусловным достоинством методов кластеризации является их полностью автоматический характер. Однако количество кластеров чаще всего задается в программе извне и является гиперпараметром [5].

Существуют агломеративные алгоритмы, формирующие кластеры снизу-вверх, начиная с малых. Результатом такой кластеризации является дендрограмма, показывающая формирование вложенных кластеров разных уровней. Недостатком таких алгоритмов является неконтролируемый рост времени процедуры при кластеризации больших многомерных данных. Распространенными способами кластеризации являются алгоритмы K-средних, выделения связанных компонент, минимального покрывающего дерева, послойной кластеризации на графах и нечеткий алгоритм C-средних [6]. Одним из популярных алгоритмов кластеризации остается DBSCAN, основанный на анализе плотности данных. Его высокое качество сопровождается быстрым падением производительности с ростом размерности данных. Еще один подход связан с применением эволюционных методов, использующих генетические алгоритмы для поиска наилучшего расположения центров кластеров по заданному критерию [7].

Для практического применения, используемый метод кластеризации должен обладать быстродействием и способностью обновлять кластеры при появлении новых данных без повторения всей процедуры. Наконец, исходные данные могут быть известны с ограниченной точностью или вообще представлять собой нечеткие величины, и в этом случае стандартные методы кластеризации не дают приемлемых решений. Отмеченный нами набор требований делает задачу кластеризации достаточно сложной. Поэтому не удивительно, что инструменты кластеризации данных нуждаются в дальнейшем совершенствовании.

Альтернативный подходом является гравитационный алгоритм, основанный на аналогии с природным явлением гравитационного стягивания частиц в плотные образования [8]. В нашей работе рассматривается развитие предложенного ранее подхода [9] в приложении к кластеризации многомерных данных.

Вязкий гравитационный алгоритм

Гравитационный алгоритм кластеризации предусматривает стягивание вместе частиц-агентов соответствуют отдельным записям данных, под действием динамического закона, в котором частицы притягиваются парным центральным взаимодействием. Координаты частиц-агентов представлены векторами $\boldsymbol{x}_i = \{x_{ik}\}$, при $k \in [1; K]$, где K — число координат (полей записи). Динамическое уравнение записывается по аналогии с уравнениями движения Ньютона в виде

$$\frac{d^2\boldsymbol{x}_i}{dt^2} + v\frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{F}_i, \tag{1}$$

где $\dfrac{d^2\boldsymbol{x}_i}{dt^2}$ — ускорение i-ой частицы, $\dfrac{d\boldsymbol{x}_i}{dt}$ — скорость этой частицы, $v\dfrac{d\boldsymbol{x}_i}{dt}$ — сила вязкого трения при заданном коэффициенте трения v. А $\boldsymbol{F}_i = \{\boldsymbol{F}_{ik}\}$, при $k \in [1; K]$ — это суммарная сила, действующая на частицу i со стороны всех остальных частиц. В этом случае

$$\boldsymbol{F}_i = \sum_{j \neq i} \boldsymbol{f}_{ij}, \tag{2}$$

где $\boldsymbol{f}_{ij}$ есть сила, действующая на частицу i со стороны частицы j. [10]. В стандартном варианте гравитационного алгоритма сила трения отсутствует [11]. Это вызывает паразитный эффект проявления инерции движения в виде выброса отдельных частиц из формирующихся кластеров и разбегании частиц кластеров после их формирования. Добавление вязкого трения приводит к уравнениям в виде (1) и устраняет это негативное явление. В случае высокой вязкости ускорениями в уравнениях (1) можно пренебречь, скорость становится пропорциональной действующей силе, и мы получаем укороченное уравнение [9]

$$v \frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{F}_i, \tag{3}$$

численное интегрирование которого занимает меньше времени, чем для уравнения (1). После перехода к дискретному времени формула перерасчета координат частиц на следующем временном шаге принимает вид

$$\boldsymbol{x}_i(t+1) = \boldsymbol{x}_i(t) + \boldsymbol{F}_i(t) / v. \tag{4}$$

В уравнении (2) сила взаимодействия между частицами вовсе не обязана полностью следовать физическому закону всемирного тяготения. Достаточно, чтобы она была центральной и быстро убывала с расстоянием. Такому требованию удовлетворяет класс функций вида

$$g(\boldsymbol{r}) = C(t)\boldsymbol{r}\varphi(|\boldsymbol{r}|), \tag{5}$$

при функции $\varphi(|\boldsymbol{r}|)$ монотонно убывающей с расстоянием быстрее, чем степенная зависимость $1 / r^p$ при $p = 1$. Применительно к алгоритму VGSA для расчета силы воздействия f_{ijk} по координате k частицы-агенты j на частицу-агента i, в качестве временного коэффициента $G(t)$ выбрана гравитационная постоянная $C(t)$, являющаяся монотонно убывающей функций времени, подбираемой в зависимости от вида данных. При отсутствии дополнительной аргументации, можно положить $G(t) = 1$. Радиус вектор $\boldsymbol{r}$ определен как $x_{ijk} - x_{ik}$, в качестве функции от радиус вектора (6) выбрано отношение массы (нормированного значения фитнес функции) частицы-агенты j (8–9) к Евклидову расстоянию (7) между частицами i и j в степени p, подбираемой на основе условий решаемой задачи, для большинства тестируемых задач, оптимально $p = 2$, в то время как в законе всемирного тяготения $p = 3$.

$$\varphi(|\boldsymbol{r}|) = \frac{M_j}{R_{ij}^p}, \tag{6}$$

$$R_{ij} = \left(\sum_{k=1}^{K} (x_{jk} - x_{ik})^2 \right)^{\frac{1}{2}}, \tag{7}$$

$$M_i = \frac{max(fit) \ - fit_i}{max(fit) - min(fit)}, \tag{8}$$

$$fit_i = \sum_{k=1}^{K} \left| x_{ik} - \sum_{j=1,\ j \neq i}^{N} \frac{x_{jk}}{N} \right|. \tag{9}$$

Таким образом, сила воздействия частицы-агенты j на частицу-агента i по координате k определяется как

$$f_{ijk} = G(t)(x_{jk} - x_{ik}) \frac{M_j}{R_{ij}{}^p}. \tag{10}$$

Суперпозиция сил воздействия системы на частицу-агента i по координате k находится из уравнения

$$\boldsymbol{F}_{ik} = G(t) \sum_{j=1,\ j \neq i}^{N} \frac{(x_{jk} - x_{ik})M_j}{R_{ij}{}^p}. \tag{11}$$

Полный вектор силы действующей на частицу i определен как $\boldsymbol{F}_i = \{\boldsymbol{F}_{ik}\}$, при $k \in [1; K]$.

Для упрощения работы алгоритма, так же допустимо отключение перерасчета масс и использование функции не зависящей от них

$$\varphi(|\boldsymbol{r}|) = \frac{1}{r^2 + a^2}. \tag{12}$$

В таком случае для учета конечной точности задания данных и автоматической остановки алгоритма возможно выключение действия силы притяжения частиц-агентов на малых расстояниях $r < a$, ограничивающее процесс сжатия кластера. Полное исключение сжатия можно обеспечить, заменяя на этих расстояниях притяжение отталкиванием.

Коэффициент вязкости ν является гиперпараметром, он, как и G контролирует скорость кластеризации, обновляясь каждую временную итерацию, с той разницей, что он не равен для всех частиц-агентов итерации.

Коэффициент вязкости рассчитывается на основе параметров системы, при тестировании VGSA была выбрана минус вторая степень от суммы 1 и разности числа частиц-агентов N в итерации и позиции $pos(M_i)$ агента i в рейтинге масс по возрастанию, умноженной на коэффициент γ, равный обратной дроби к плотности частиц, умноженной на параметр $\alpha = 3$

$$v = \tau(1 + \gamma(N - pos(M_i)))^{-2},\ \gamma = \frac{L}{\alpha N}. \tag{13}$$

Здесь τ — коэффициент подбираемый из условий задачи, допустимо $\tau = 1$. L — значение длинны диапазона области определения x, а $pos(M_i) \in N$ [1; N]. Для частицы-агента i с наилучшим значением фитнес функции $pos(M_i) = N$, следовательно $\nu = \tau$, т.е. вязкость достигает своего максимального значения. Для частиц-агентов с наихудшей приспособляемостью, значение вязкости $\nu = \tau\,(1 + \gamma(N - 1))^{-2}$, т.е. достигает своего минимального значения. В результате замедления лучших частиц-агентов, снижается стягивание системы к общему центру масс, и улучшается образование отдельных центров притяжения частиц, что хорошо подходит для применения VGSA в задачах кластеризации.

Отметим определенное сходство VGSA с алгоритмом самоорганизующихся карт Кохонена (SOM) [12]. Отличие состоит в том, что в VGSA движутся все частицы, а в алгоритме Кохонена обновляется положение только пробных частиц, которые соответствуют положению центров будущих кластеров и постепенно смещаются в области максимальной локальной концентрации частиц. Это позволяет сразу сделать вывод о том, что алгоритм SOM обладает большим быстродействием, чем VGSA, но требует изначального задания числа кластеров, а результат его работы зависит от того, насколько удачно выбраны начальные положения частиц. В качестве ядра $\varphi(|\boldsymbol{r}|)$ в SOM чаще всего выступает функция Гаусса

$$\varphi(|\boldsymbol{r}|) = exp(-r^2/(2\sigma^2(t))), \tag{13}$$

в которой параметр $\sigma(t)$ определяет расстояние действия силы и уменьшается со временем. Исходя из имеющегося опыта SOM, кроме рассмотренного выше набора функций для гравитационного алгоритма, можно пробовать использовать зависимость (13).

Устойчивость алгоритма вязкой гравитационной кластеризации

Устойчивость VGSA интуитивно понятна из аналогии с физическим процессом агрегирования частиц в вязкой жидкости, но для надежности его применения требуется более серьезное обоснование, основанное на теории Ляпунова [13]. Асимптотическая устойчивость по Ляпунову означает, что любое начальное состояние системы завершается одним и тем же конечным состоянием. Для его доказательства требуется построение функции Ляпунова, ограниченной по величине снизу и монотонно убывающей по времени. Релаксации физической системы идет с понижением ее потенциальной энергии. Этот факт можно использовать для построения функции Ляпунова, подходящей для нашего случая, и прояснения устойчивости рассматриваемой нами динамической системы (3), являющейся основой VGSA. Для парного взаимодействия введем вспомогательный функционал, являющийся аналогом потенциальной энергии

$$U\left(\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|\right) = -\int_{\boldsymbol{x}_i}^{\boldsymbol{x}_j} F_j(\boldsymbol{r})d\boldsymbol{r}, \tag{14}$$

так что функция Ляпунова

$$E = \sum_{i>j} U\left(\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|\right), \tag{15}$$

тогда, производная функции Ляпунова по времени

$$\frac{dE}{dt} = \sum_i \dot{\boldsymbol{x}}_i \nabla_i E = -\frac{1}{v}\sum_i (\boldsymbol{F}_i)^2 < 0. \tag{16}$$

Таким образом, введенная нами функция Ляпунова монотонно уменьшается по времени, и динамическая система является асимптотически устойчивой. Это также означает устойчивость каждого отдельного кластера относительно вылета из него частиц. Кроме того, конечный кластер, составленный из большого количества частиц, будет иметь вид плотного шара, поскольку любое малое смещение частиц с поверхности шара будет означать рост функции Ляпунова. Проведенное нами рассмотрение дает необходимое условие устойчивости всего VGSA.

Тестирование алгоритма на данных

В качестве датасета, для проведения тестирования VGSA как алгоритма кластеризации, был выбран классический датасет Iris Species[1]. Который содержит 150 записей по 5 полей каждая, последнее поле представляет собой целевую функцию от первых четырех полей. Всего имеется три разновидности значения целевой функции.

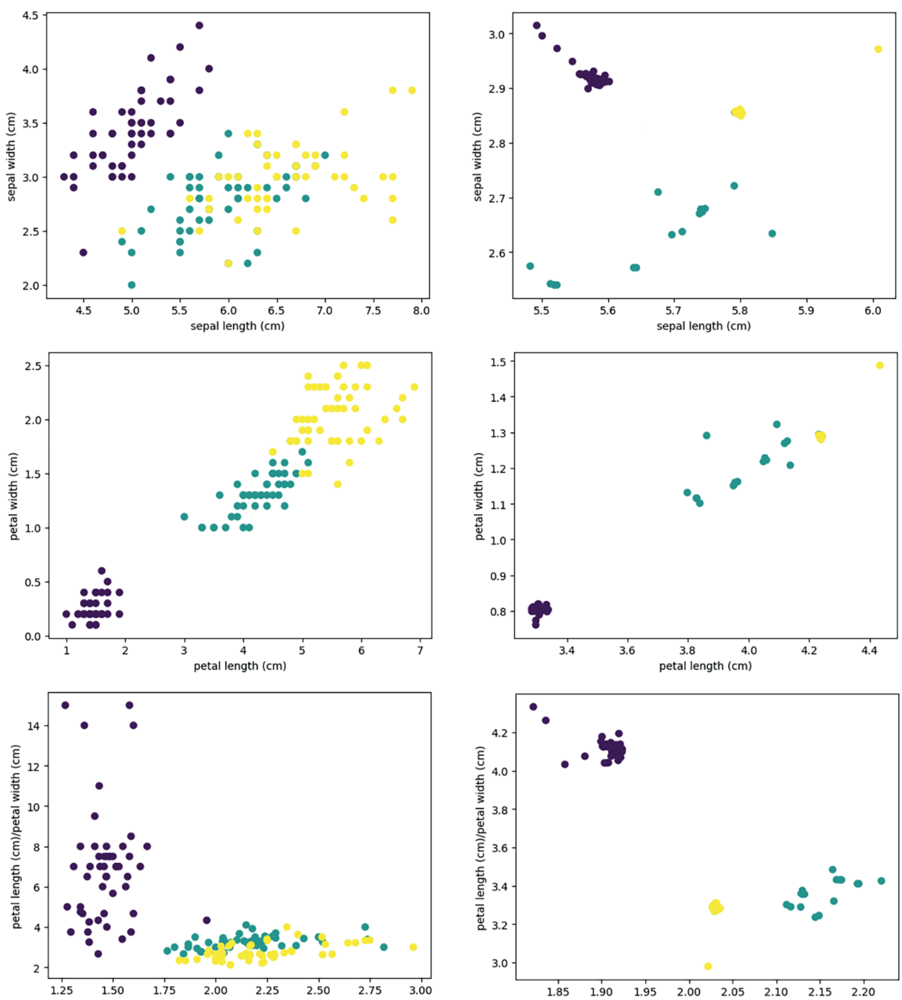
Исходя из условий задачи, создается начальный рой из 150 частиц-агентов с четырьмя координатами, количество итераций настраивается вручную, было выбрано 30 итераций. Расчет сил производился по формуле (11), для формирования случайных факторов в эволюционной модели, значения масс были умножены на случайную величину нормального распределения в интервале [0,1]. Коэффициент вязкости рассчитывался по формуле (13) при $\tau = 1$, гравитационная постоянная

$$G = G_0 e^{-\alpha \frac{1}{L}}, \tag{17}$$

[1] Iris Species // Kagle. URL: https://www.kaggle.com/datasets/uciml/iris.

где l — номер текущей итерации, L — общее количество итераций, $G_0 = 10$ — коэффициент, $G_0 = 0.2$ — коэффициент, $\alpha = 10$ — коэффициент. За серию из 50-ти тестирований наилучший результат представлен на рис. Класс «Setosa» отделился без ошибок, часть класса «Versicolor» захватил класс «Virginica». Минимальное количество ошибок 11 из 150. При удалении класса «Setosa» после его выделения, два оставшихся класса разделились с количеством 9 ошибок из 100.

Алгоритмы K-средних (с обучением и предобработкой данных) и SOM (с предобработкой данных) для датасета Iris Species, дают следующие результаты: ≈ 0,94 % [14; 15] и ≈ 0,96 % [2][16] верных результатов, в лучших попытках, соответственно, что немного выше VGSA (≈ 0,92 %).



VGSA. Кластеризация на датасете Iris Species

[2] Self Organizing Maps // RPubs. URL: https://rpubs.com/loveb/som.

Заключение

Проведенное нами расширение области применимости VGSA на многомерные данные позволило реализовать работоспособный алгоритм на языке Python. Это дает возможность интегрировать его в другие инструменты машинного обучения, имеющиеся в распространенных фреймворках [17]. Сравнение с наиболее близким по подходу методом SOM демонстрирует, что VSA показывает несколько менее высокую точность, но при этом выделяет кластеры, более устойчиво, и автоматически обеспечивает их иерархию. В то же время, VSA требует больших затрат компьютерных ресурсов. В качестве компромисса может быть предложен вариант с отслеживанием эволюции ансамбля пробных частиц, которые будут сначала притягиваться к центрам кластеров без взаимодействия между собой, а затем стягиваться к центрам кластеров [18].

В процессе работы алгоритма возможно добавление новых данных, а также выведение из расчета хорошо сформировавшихся кластеров. Положение центров кластеров обновляется без перезапуска программы, что дает возможность ее применения для динамических задач. Так же подобный подход может дать преимущество для больших данных с неизвестным числом кластеров.

Список использованной литературы

1. Suárez J.L. A Tutorial on Distance Metric Learning: Mathematical Foundations, Algorithms, Experimental Analysis, Prospects and Challenges / J.L. Suárez, S. García, F. Herrera. — DOI 10.1016/j.neucom.2020.08.017 // Neurocomputing. — 2021. — Vol. 425. — P. 300–322.

2. Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow / О. Жерон. — Санкт-Петербург : Диалектика, 2020. — 690 с.

3. Dawani J. Hands-On Mathematics for Deep Learning: Build a Solid Mathematical Foundation for Training Efficient Deep Neural Networks / J. Dawani. — Birmingham : Packt Publishing, 2020. — 364 p.

4. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects / A.E. Ezugwu, A.M. Ikotun, O.O. Oyelade [at al.]. — DOI 10.1016/j.engappai.2022.104743 // Engineering Applications of Artificial Intelligence. — 2022. — Vol. 110. — P. 104743.

5. Data Clustering. Algorithms and Applications / ed. C.C. Aggarwal, Ch.K. Reddy. — New York : CRC Press, 2014. — 652 p.

6. Воронов К.В. Лекции по алгоритмам кластеризации и многомерного шкалирования / К.В. Воронов. — Москва, 2007. — URL: https://knigogid.ru/books/1780564-lekcii-po-algoritmam-klasterizacii-i-mnogomernogo-shkalirovaniya/toread.

7. Corne D. Evolutionary Clustering / D. Corne, J. Handl, J. Knowles // Encyclopedia of Machine Learning / ed. C. Sammut, G.I. Webb. — Boston : Springer, 2011. — P. 332–337.

8. Binder P. Gravitational Clustering: A simple, robust and adaptive approach for distributed networks / P. Binder, M. Muma, A.M. Zoubir. — DOI 10.1016/j.sigpro.2018.02.034 // Signal Processing. — 2018. — Vol. 149. — P. 36–48.

9. Головинский П.А. Вязкий гравитационный алгоритм кластеризации неточных данных / П.А. Головинский. — DOI 10.17308/sait.2022.1/9203 // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. — 2022. — № 1. — С. 79–89.

10. Саймон Д. Алгоритмы эволюционной оптимизации / Д. Саймон. — Москва : ДМК Пресс, 2020. — 1002 с.

11. Hybridized Particle Swarm—Gravitational Search Algorithm for Process Optimization / R. Shankar, N. Ganesh, R. Čep [et. al.]. — DOI 10.3390/pr10030616 // Processes. — 2022. — Vol. 10, iss. 3. — P. 616.

12. Кохонен Т. Самоорганизующиеся карты / Т. Кохонен. — Москва : Бином Лаборатория знаний, 2017. — 655 с.

13. Халил Х.К. Нелинейные системы / Х.К. Халил. — Москва, 2009. — 829 с.

14. Sushkov A. Машинное обучение: от Ирисов до Телекома / A. Sushkov // Хабр. — 2017. — 23 авг. — URL: https://habr.com/ru/companies/billing/articles/334738/.

15. Khotijah S. K-Means Clustering of Iris Dataset / S. Khotijah // Kaggle. — URL: https://www.kaggle.com/code/khotijahs1/k-means-clustering-of-iris-dataset.

16. Ahangama I. U Matrix of SOM for Iris Dataset / I. Ahangama // Kaggle. — URL: https://www.kaggle.com/code/imanthaahangama/u-matrix-of-som-for-iris-dataset#5-Vector-Field-for-SOM-Generated-in-Section-4.

17. Dhawale C.A. Current Trends in Deep Learning Frameworks with Opportunities and Future Prospectus / C.A. Dhawale, K. Dhawale. — DOI 10.4018/978-1-7998-1159-6.ch003 // Neural Networks for Natural Language Processing / ed. S. Sumathi, M. Janani. — IGI Global, 2020. — P. 63–77.

18. A novel data clustering algorithm based on modified gravitational search algorithm / XiaoHong Han, Long Quan, XiaoYan Xiong [et al.]. — DOI 10.1016/j.engappai.2016.11.003 // Engineering Applications of Artificial Intelligence. — 2017. — Vol. 61. — P. 1–7.

References

1. Suárez J.L., García S., Herrera F. A Tutorial on Distance Metric Learning: Mathematical Foundations, Algorithms, Experimental Analysis, Prospects and Challenges. *Neurocomputing*, 2021, vol. 425, pp. 300–322. DOI: 10.1016/j.neucom.2020.08.017.

2. Geron A. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O`Reilly Media, 2017. 574 p. (Russ. ed.: Geron A. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. Saint Petersburg, Dialektika Publ., 2020. 690 p.).

3. Dawani J. *Hands-On Mathematics for Deep Learning: Build a Solid Mathematical Foundation for Training Efficient Deep Neural Networks*. Birmingham, Packt Publishing, 2020. 364 p.

4. Ezugwu A.E., Ikotun A.M., Oyelade O.O., Abualigah L., Agushaka J.O., Eke Ch.I., Akinyelu A.A., A Comprehensive Survey of Clustering Algorithms: State-of-the-art Machine Learning Applications, Taxonomy, Challenges, and Future Research Prospects. *Engineering Applications of Artificial Intelligence*, 2022, vol. 110, pp. 104743. DOI: 10.1016/j.engappai.2022.104743.

5. Aggarwal C.C., Reddy Ch.K. (eds). *Data Clustering. Algorithms and Applications*. New York, CRC Press, 2014. 652 p.

6. Vironov K.V. *Lectures on clustering and multidimensional scaling algorithms.* Moscow, 2007. Available at: https://knigogid.ru/books/1780564-lekcii-po-algoritmam-klasterizacii-i-mnogomernogo-shkalirovaniya/toread.

7. Corne D., Handl J., Knowles J. Evolutionary Clustering. In Sammut C., Webb G.I. (eds). *Encyclopedia of Machine Learning*. Boston, Springer, 2011, pp. 332–337.

8. Binder P., Muma M., Zoubir A.M. Gravitational Clustering: A simple, Robust and Adaptive Approach for Distributed Networks. *Signal Processing*, 2018, vol. 149, pp. 36–48. DOI: 10.1016/j.sigpro.2018.02.034.

9. Golovinski P.A. Viscous Gravitational Algorithm for Clustering Inacurate Data. *Vestnik Voronezhskogo gosudarstvennogo universiteta. Seriya: Sistemnyi analiz i informatsionnye tekhnologii = Proceedings of Voronezh State University. Series: Systems Analysis and Information Technologies*, 2022, no. 1, pp. 79–89. (In Russian). DOI: 10.17308/sait.2022.1/9203.

10. Simon D. *Evolutionary Optimization Algorithms: Biologically-Inspired and Population-Based Approaches to Computer Intelligence*. Hoboken, New Jersey, John Wiley & Sons Inc., 2013. 742 p. (Russ. ed.: Simon D. *Evolutionary Optimization Algorithms.* Moscow, DMK Press Publ., 2020. 1002 p.).

11. Shankar R., Ganesh N., Čep R., Narayanan R.C., Pal S., Kalita K. Hybridized Particle Swarm—Gravitational Search Algorithm for Process Optimization. *Processes*, 2022, vol. 10, iss. 3, pp. 616. DOI: 10.3390/pr10030616.

12. Kohonen T. *Self-Organizing Maps.* Springer Science, 2001. 501 p. (Russ. ed.: Kohonen T. *Self-Organizing Maps.* Moscow, Binom Laboratoriya znanii Publ., 2017. 655 p.).

13. Khalil H.K. *Nonlinear Systems*. Prentice Hall, 1996. 734 p. (Russ. ed.: Khalil H.K. *Nonlinear Systems.* Moscow, 2009. 829 p.).

14. Sushkov A. Machine learning: from Irises to Telecom. *Habr*, 2017, August 23. Available at: https://habr.com/ru/companies/billing/articles/334738/.

15. Khotijah S. K-Means Clustering of Iris Dataset. *Kaggle*. Available at: https://www.kaggle.com/code/khotijahs1/k-means-clustering-of-iris-dataset.

16. Ahangama I. U Matrix of SOM for Iris Dataset. *Kaggle*. Available at: https://www.kaggle.com/code/imanthaahangama/u-matrix-of-som-for-iris-dataset#5-Vector-Field-for-SOM-Generated-in-Section-4

17. Dhawale C.A., Dhawale K. Current Trends in Deep Learning Frameworks with Opportunities and Future Prospectus. In Sumathi S., Janani M. (eds). *Neural Networks for Natural Language Processing*. IGI Global, 2020, pp. 63–77. DOI: 10.4018/978-1-7998-1159-6.ch003.

18. XiaoHong Han, Long Quan, XiaoYan Xiong, Matt Almeter, Jie Xiang, Yuan Lan. A Novel Data Clustering Algorithm Based on Modified Gravitational Search Algorithm. *Engineering Applications of Artificial Intelligence*, 2017, vol. 61, pp. 1–7. DOI: 10.1016/j.engappai.2016.11.003.

Информация об авторах

Головинский Павел Абрамович — доктор физизико-математических наук, профессор кафедры инноватики и строительной физики им. И.С. Суровцева, Воронежский государственный технический университет, г. Воронеж, Российская Федерация, e-mail: golovinski@bk.ru.

Тарасова Анна Сергеевна — аспирант кафедры инноватики и строительной физики им. И.С. Суровцева, Воронежский государственный технический университет, г. Воронеж, Российская Федерация, e-mail: anna222tarasova@yandex.ru.

Information about the Authors

Pavel A. Golovinski — D.Sc. in Physical and Mathematical Sciences, Professor of the Department of Innovation and Building Physics Named after I.S. Surovtsev, Voronezh State Technical University, Voronezh, Russian Federation, e-mail: golovinski@bk.ru.

Anna S. Tarasova — PhD Student, Department of Innovation and Building Physics named after I.S. Surovtsev, Voronezh State Technical University, Voronezh, Russian Federation, e-mail: anna222tarasova@yandex.ru.

Вклад авторов

Все авторы сделали эквивалентный вклад в подготовку публикации. Авторы заявляют об отсутствии конфликта интересов.

Contribution of the Authors

The authors contributed equally to this article. The authors declare no conflicts of interests.

Для цитирования

Головинский П.А. Вязкий гравитационный алгоритм кластеризации многомерных данных / П.А. Головинский, А.С. Тарасова. — DOI 10.17150/2713-1734.2023.5(4).379-391. — EDN SBDRVK // System Analysis & Mathematical Modeling. — 2023. — Т. 5, № 4. — С. 379–391.

For Citation

Golovinsky P.A., Tarasova A.S. Viscous Gravity Algorithm for Clustering Multidimensional Data. *System Analysis & Mathematical Modeling*, 2023, vol. 5, no. 4, pp. 379–391. (In Russian). EDN: SBDRVK. DOI: 10.17150/2713-1734.2023.5(4).379-391.