Вычислительная реализация рангового метода кластеризации

Г.В. Гренкин
Дальневосточный федеральный университет
glebgrenkin@gmail.com

Задача кластеризации объектов заключается в том, чтобы совокупность объектов разбить на сравнительно небольшое число однородных, в определённом смысле, кластеров.

Если набор значений признаков представить в виде точки в p-мерном признаковом пространстве, то задача кластеризации состоит в разбиении совокупности точек на сравнительно небольшое число кластеров таким образом, чтобы объекты, принадлежащие одному кластеру, находились бы на сравнительно небольших расстояниях друг от друга.

Наиболее трудным и наименее формализованным в задаче кластеризации является момент, связанный с определением однородности объектов. Для определения понятия однородности объектов вводится расстояние между двумя объектами — метрика. Близкие в смысле этой метрики объекты считаются однородными, принадлежащими одному кластеру.

В статье М. А. Гузева и Е. В. Черныш [1] был предложен ранговый метод кластеризации, в котором не используется метрика. Этот метод основан на модифицированном В. П. Масловым законе Ципфа, который имеет следующий вид:

$$\ln w \cong -\gamma \ln \left(\frac{N-r}{r} \right) + c \equiv -\gamma \ln R + c, \tag{1}$$

Здесь исходные данные $w_1, w_2, \ldots, w_n$ упорядочиваются по возрастанию, и каждому значению w ставится в соответствие порядковый номер — ранг r. В логарифмических координатах это линейная зависимость, графиком является прямая.

При разбиении данных на кластеры ранговым методом на каждом из кластеров справедлив модифицированный В. П. Масловым закон Ципфа со своими значениями параметров (γ, c), которые меняются при переходе от кластера к кластеру.

Геометрический смысл: на каждом из кластеров точки близки к прямой, при переходе от кластера к кластеру прямая меняется.

Изначально задача кластеризации эмпирических данных ранговым методом является нечётко поставленной. В связи с этим для одних и тех же данных выделять кластеры можно по-разному. Поэтому разбиение данных на кластеры зачастую может быть субъективным. Таким образом, требуется автоматизировать процесс кластеризации эмпирических данных ранговым методом.

Для вычислительной реализации рангового метода кластеризации исходную постановку задачи необходимо формализовать, требуется построить математическую модель. Нужно ответить на вопрос: какую информацию об исходных данных нужно получить?

Введём меру отклонения точки от прямой как отклонение точки от прямой по вертикали:

$$\delta(\ln R, \ln w, \gamma, c) = |\ln w - (-\gamma \ln R + c)|.$$

Зададим пороговое значение δ_0 и потребуем, чтобы для всех точек кластера $[a...b]$ значение меры отклонения не превосходило δ_0. Будем считать, что на кластере $[a...b]$

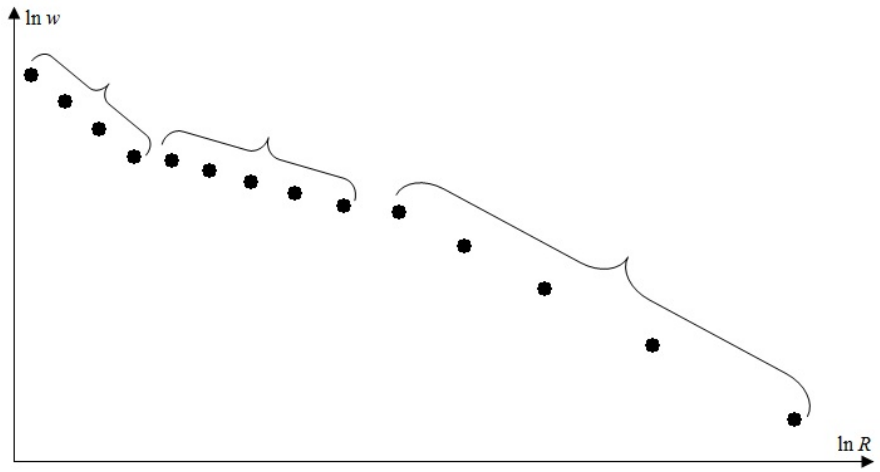

Рис. 1: Ранговый метод кластеризации

справедлив модифицированный В. П. Масловым закон Ципфа с параметрами γ, c, если наибольшее значение меры отклонения не превосходит δ_0:

$$\max_{a \le r \le b} \delta(\ln R_r, \ln w_r, \gamma, c) \le \delta_0.$$

Введём величину

$$\delta_{\min}(0) = \inf_{(\gamma,c)\in\mathbb{R}^2} \max_{a \le r \le b} \delta(\ln R_r, \ln w_r, \gamma, c).$$

Геометрический смысл: $\delta_{\min}(0)$ — это половина наименьшей высоты полосы, содержащей все точки кластера $[a...b]$.

Будем считать, что на кластере $[a...b]$ справедлив модифицированный В. П. Масловым закон Ципфа, если $\delta_{\min}(0)$ не превосходит δ_0.

Введём величину

$$\delta_{\min}(\nu_0) = \inf_{(\gamma,c)\in\mathbb{R}^2} \max_{a \le r \le b}{}_{\nu_0+1} \delta(\ln R_r, \ln w_r, \gamma, c).$$

Здесь $\max_k$ означает k-е по максимальности число. Геометрический смысл: $\delta_{\min}(\nu_0)$ — это половина наименьшей высоты полосы, содержащей все точки кластера $[a...b]$, в том случае, если ν_0 точек можно выбросить.

Будем считать, что на кластере $[a...b]$ справедлив модифицированный В. П. Масловым закон Ципфа, если $\delta_{\min}(\nu_0)$ не превосходит δ_0.

Рассмотрим множество промежутков, на которых справедлив модифицированный В. П. Масловым закон Ципфа. Те из них, которые нельзя расширить так, чтобы на расширенном промежутке также был справедлив этот закон, образуют множество максимальных промежутков.

Процесс кластеризации эмпирических данных ранговым методом мы будем рассматривать как процесс расширения кластеров. То есть если выделен некоторый кластер и он

может быть расширен (в него можно включить соседствующие с ним точки), то он расширяется. В результате такого расширения кластеры, как правило, будут пересекаться, и это есть особенность кластеризации эмпирических данных ранговым методом. Итак, чтобы найти разбиение на кластеры, найдём множество максимальных промежутков.

Таким образом, построена математическая модель, произведена формальная постановка задачи. Требуется разработать численные алгоритмы решения поставленных задач.

Требуется вычислить $\delta_{\min}(0)$ — половину наименьшей высоты полосы, содержащей все заданные точки. Это задача нахождения многочлена наилучшего равномерного приближения первой степени. Из условия альтернанса вытекает, что искомая прямая параллельна одному из рёбер выпуклой оболочки.

Выпуклой оболочкой множества точек называется наименьший выпуклый многоугольник, содержащий все точки данного множества. Для построения выпуклой оболочки в вычислительной геометрии есть алгоритм Грэхема и алгоритм Джарвиса [2].

Рассмотрим задачу вычисления $\delta_{\min}(0)$ с точки зрения проектирования точек на ось ординат во всех возможных направлениях. Введём оператор проектирования $p_\gamma(x, y)$. Проведём через точку (x, y) прямую с угловым коэффициентом $(-\gamma)$, тогда проекция точки (x, y) на ось ординат — это точка пересечения данной прямой с осью ординат. Пусть $p_\gamma(x, y)$ — ордината проекции: $p_\gamma(x, y) = x\gamma + y$. Тогда $\delta_{\min}(0)$ можно найти по следующей формуле:

$$\delta_{\min}(0) = \inf_{(\gamma,c)\in\mathbb{R}^2} \max_{a\le r\le b} |p_\gamma(x_r, y_r) - c| = \min_{\gamma\in\mathbb{R}} \min_{c\in\mathbb{R}} \max_{a\le r\le b} |p_\gamma(x_r, y_r) - c| =$$

$$= \frac{1}{2} \min_{\gamma\in\mathbb{R}} \left(\max_{a\le r\le b} p_\gamma(x_r, y_r) - \min_{a\le r\le b} p_\gamma(x_r, y_r) \right).$$

Рассмотрим следующие функции:

$$\phi_r(\gamma) = p_\gamma(x_r, y_r) = x_r\gamma + y_r,$$

$$\phi_{\max}(\gamma) = \max_{a\le r\le b} \phi_r(\gamma),\ \phi_{\min}(\gamma) = \min_{a\le r\le b} \phi_r(\gamma).$$

Требуется найти минимум функции $\phi(\gamma) = \phi_{\max}(\gamma) - \phi_{\min}(\gamma)$. Функции $\phi_{\max}(\gamma)$ и $\phi_{\min}(\gamma)$ кусочно-линейные, следовательно, $\phi(\gamma)$ — кусочно-линейная функция. Значит, минимум $\phi(\gamma)$ достигается в одной из точек излома.

Возникает задача нахождения точек излома функций $\phi_{\max}(\gamma)$ и $\phi_{\min}(\gamma)$. Графики этих функций — это ломаные.

Рассмотрим алгоритм построения ломаной — графика функции $\phi_{\max}(\gamma)$ (см. рис. 2). Начинаем строить ломаную с прямой — графика функции $\phi_b(\gamma)$ (эта прямая — текущий максимум). Находим точки пересечения этой прямой с прямыми — графиками функций $\phi_{b-1}(\gamma)$, ..., $\phi_a(\gamma)$. Выбираем точку пересечения с наименьшей абсциссой γ_1. Пусть это точка пересечения с графиком функции $\phi_{k_1}(\gamma)$. Прямая — график функции $\phi_{k_1}(\gamma)$ — становится текущим максимумом. Далее находим точки пересечения этой прямой с прямыми — графиками функций $\phi_{k_1-1}(\gamma)$, ..., $\phi_a(\gamma)$. Выбираем точку пересечения с наименьшей абсциссой γ_2. Пусть это точка пересечения с графиком функции $\phi_{k_2}(\gamma)$. Прямая — график функции $\phi_{k_2}(\gamma)$ — становится текущим максимумом. Этот процесс продолжается, пока текущим максимумом не станет прямая — график функции $\phi_a(\gamma)$.

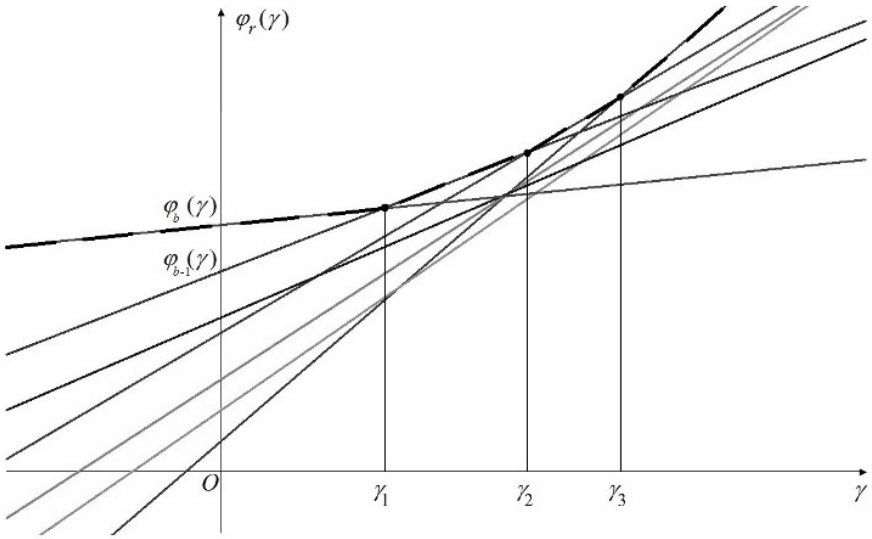

Рис. 2: Алгоритм Джарвиса (пунктиром выделен график функции $\phi_{\max}(\gamma)$)

Описанный алгоритм представляет собой алгоритм Джарвиса построения выпуклой оболочки. Прямые, образующие ломаную, соответствуют вершинам выпуклой оболочки, а абсциссы вершин ломаной соответствуют угловым коэффициентам прямых, содержащих рёбра выпуклой оболочки. Время выполнения алгоритма – $O(hM)$, где M — количество точек, h — число вершин выпуклой оболочки.

Опишем алгоритм Грэхема, который имеет оптимальное время выполнения $O(M)$ (см. рис. 3). Будем добавлять прямые по одной: сначала $\phi_b(\gamma)$, затем $\phi_{b-1}(\gamma)$, ..., $\phi_a(\gamma)$. Будем последовательно находить $\max\limits_{b-1\le r\le b}\phi_r(\gamma)$, $\max\limits_{b-2\le r\le b}\phi_r(\gamma)$, ..., $\max\limits_{a\le r\le b}\phi_r(\gamma)$.

Предположим, что ломаная — график функции $\max\limits_{q+1\le r\le b}\phi_r(\gamma)$ — уже построена. Укажем, как построить ломаную — график функции $\max\limits_{q\le r\le b}\phi_r(\gamma)$.

Можно показать, что прямая — график функции $\phi_q(\gamma)$ — пересекает ломаную — график функции $\max\limits_{q+1\le r\le b}\phi_r(\gamma)$ — ровно в одной точке с абсциссой γ'.

Ломаную — график функции $\max\limits_{q\le r\le b}\phi_r(\gamma)$ — можно получить так: на промежутке $(-\infty,\gamma']$ эта ломаная совпадает с ломаной — графиком функции $\max\limits_{q+1\le r\le b}\phi_r(\gamma)$, — а на промежутке $[\gamma',+\infty)$ — с прямой — графиком функции $\phi_q(\gamma)$.

Требуется вычислить $\delta_{\min}(\nu_0)$ — половину наименьшей высоты полосы, содержащей все заданные точки, в том случае, если ν_0 точек можно выбросить. $\delta_{\min}(\nu_0)$ можно найти по следующей формуле:

$$\delta_{\min}(\nu_0)=\inf_{(\gamma,c)\in\mathbb{R}^2}\max_{a\le r\le b}{}_{\nu_0+1}|p_\gamma(x_r,y_r)-c|=\min_{\gamma\in\mathbb{R}}\min_{c\in\mathbb{R}}\max_{a\le r\le b}{}_{\nu_0+1}|p_\gamma(x_r,y_r)-c|=$$

$$=\min_{\gamma\in\mathbb{R}}\left(\frac{1}{2}\min_{0\le\nu\le\nu_0}\left(\max_{a\le r\le b}{}_{\nu+1}p_\gamma(x_r,y_r)-\min_{a\le r\le b}{}_{\nu_0-\nu+1}p_\gamma(x_r,y_r)\right)\right)=$$

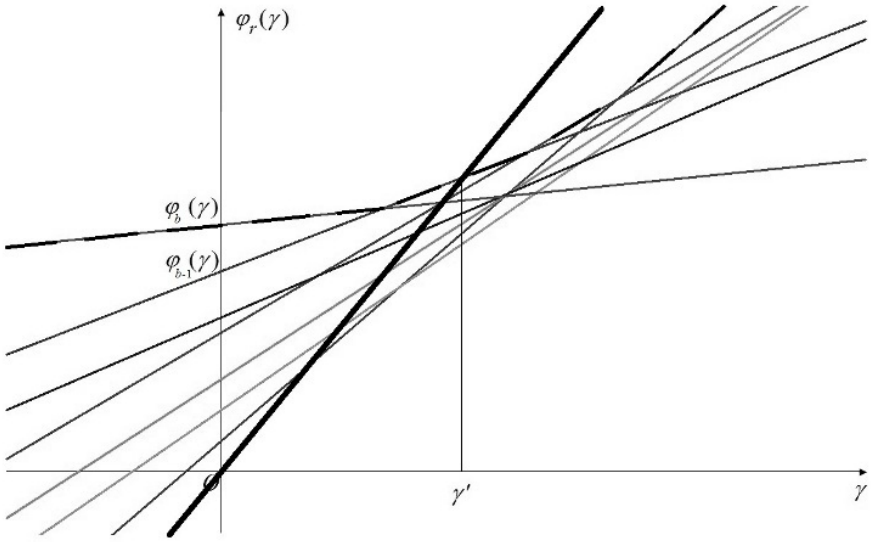

Рис. 3: Алгоритм Грэхема (пунктиром выделен график функции $\max\limits_{q+1\le r\le b}\phi_r(\gamma)$, жирная прямая — график функции $\phi_q(\gamma)$)

$$= \frac{1}{2}\min_{0\le\nu\le\nu_0}\min_{\gamma\in R}\left(\max_{a\le r\le b}{}_{\nu+1}p_\gamma(x_r,y_r) - \min_{a\le r\le b}{}_{\nu_0-\nu+1}p_\gamma(x_r,y_r)\right).$$

Рассмотрим следующие функции:

$$\phi^{(\nu)}_{\max}(\gamma) = \max_{a\le r\le b}{}_{\nu+1}\phi_r(\gamma),$$

$$\phi^{(\nu)}_{\min}(\gamma) = \min_{a\le r\le b}{}_{\nu+1}\phi_r(\gamma),$$

$$\phi_{(\nu)}(\gamma) = \phi^{(\nu)}_{\max}(\gamma) - \phi^{(\nu_0-\nu)}_{\min}(\gamma).$$

Требуется найти минимумы функций $\phi_{(\nu)}(\gamma)$, $\nu = 0,1,\ldots,\nu_0$. $\phi_{(\nu)}(\gamma)$ — кусочно-линейная функция, минимум достигается в точке излома.

Возникает задача нахождения точек излома функций $\phi^{(\nu)}_{\max}(\gamma)$ и $\phi^{(\nu)}_{\min}(\gamma)$, $\nu = 0,1,\ldots,\nu_0$.

Рассмотрим алгоритм построения ломаных — графиков функций $\phi^{(0)}_{\max}(\gamma)$, $\phi^{(1)}_{\max}(\gamma)$, $\ldots$, $\phi^{(\nu_0)}_{\max}(\gamma)$ (см. рис. 4). Будем добавлять прямые по одной: сначала $\phi_b(\gamma)$, затем $\phi_{b-1}(\gamma)$, $\ldots$, $\phi_a(\gamma)$. Будем последовательно находить $\max\limits_{q\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ $(\nu = 0,1,\ldots,\nu_0)$ для $q = b-1, b-2,\ldots,a$.

Предположим, что ломаные — графики функций $\max\limits_{q+1\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ $(\nu = 0,1,\ldots,\nu_0)$ – – уже построены. Укажем, как построить ломаные — графики функций $\max\limits_{q\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ $(\nu = 0,1,\ldots,\nu_0)$.

Можно показать, что прямая — график функции $\phi_q(\gamma)$ — пересекает каждую из ломаных — графиков функций $\max\limits_{q+1\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ $(\nu = 0,1,\ldots,\nu_0)$ — ровно в одной точке с абсциссой $\gamma'_{(\nu)}$.

Ломаные — графики функций $\max\limits_{q\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ $(\nu = 0, 1, \ldots, \nu_0)$ — можно получить следующим образом. Ломаная — график функции $\max\limits_{q\le r\le b}{}_{1}\phi_r(\gamma)$ -- на промежутке $(-\infty, \gamma'_{(0)}]$ совпадает с ломаной — графиком функции $\max\limits_{q+1\le r\le b}{}_{1}\phi_r(\gamma)$, -- а на промежутке $[\gamma'_{(0)}, +\infty)$ – – с прямой -- графиком функции $\phi_q(\gamma)$. Ломаная — график функции $\max\limits_{q\le r\le b}{}_{2}\phi_r(\gamma)$ -- на промежутке $(-\infty, \gamma'_{(1)}]$ совпадает с ломаной — графиком функции $\max\limits_{q+1\le r\le b}{}_{2}\phi_r(\gamma)$, — на промежутке $[\gamma'_{(1)}, \gamma'_{(0)}]$ — с прямой -- графиком функции $\phi_q(\gamma)$, -- а на промежутке $[\gamma'_{(0)}, +\infty)$ -- с ломаной — графиком функции $\max\limits_{q+1\le r\le b}{}_{1}\phi_r(\gamma)$. Ломаная — график функции $\max\limits_{q\le r\le b}{}_{\nu+1}\phi_r(\gamma)$ — на промежутке $(-\infty, \gamma'_{(\nu)}]$ совпадает с ломаной — графиком функции $\max\limits_{q+1\le r\le b}{}_{\nu+1}\phi_r(\gamma)$, -- на промежутке $[\gamma'_{(\nu)}, \gamma'_{(\nu-1)}]$ — с прямой — графиком функции $\phi_q(\gamma)$, -- а на промежутке $[\gamma'_{(\nu-1)}, +\infty)$ -- с ломаной — графиком функции $\max\limits_{q+1\le r\le b}{}_{\nu}\phi_r(\gamma)$.

Данный алгоритм представляет собой обобщение алгоритма Грэхема. Время выполнения алгоритма — $O(M(\nu_0 + 1))$.

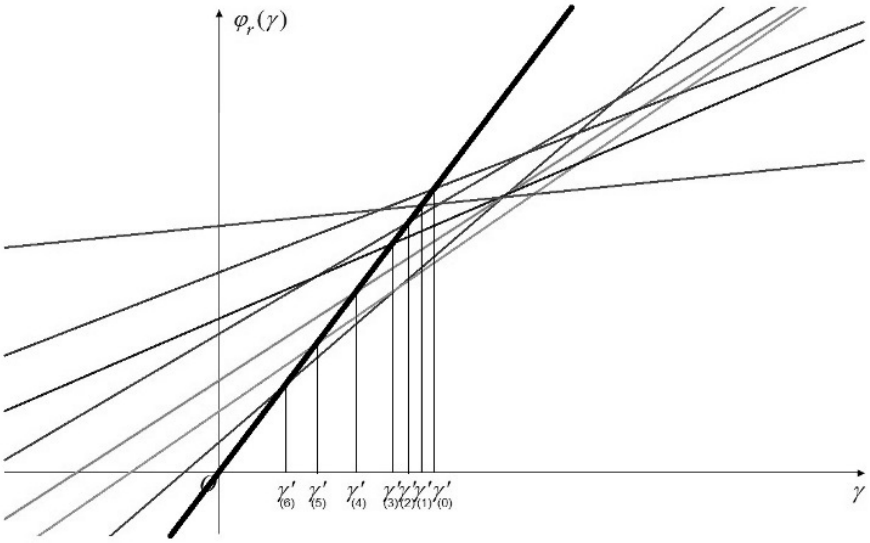

Рис. 4: Обобщённый алгоритм Грэхема

Разработана программная система [4], в которой реализованы разработанные методы. Программа написана на языке программирования C++.

Программа принимает на вход входные данные в формате CSV и входные параметры: δ_0 (пороговое значение для половины минимальной высоты полосы), ν_0 (допустимое количество аномальных точек), x (код функциональной зависимости: если $x = 1$, то используется соотношение $N = n + 1$, а если $x = 2$, то соотношение $N = 2n + 1$).

Программа выводит результаты вычислений в файл в формате HTML. Программа вычисляет множество максимальных промежутков. Также имеется возможность вычисления значения параметра γ и половины минимальной высоты полосы для интересую-

щих промежутков.

Приведём пример применения программы. На рис. 5 цветом выделено разбиение данных на кластеры, приведённое в статье [1]. Это разбиение было найдено вручную. Фигурными скобками выделено множество максимальных промежутков, выданное программой ($\delta_0 = 0,03, \nu_0 = 0$).

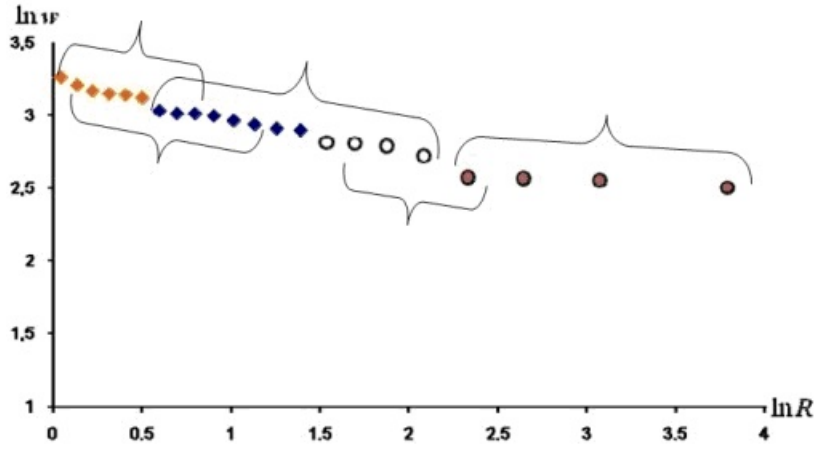

Рис. 5: Пример применения программы

Отметим, что разбиение, приведённое в статье [1], отличается от множества максимальных промежутков. Таким образом, программа позволяет избавиться от субъективности при нахождении разбиения.

Список литературы

[1] Гузев М. А., Черныш Е. В. Ранговый анализ в задачах кластеризации // Информатика и системы управления. 2009. №3(21). С. 13–19.

[2] Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. — М.: Мир, 1989.

[3] Гренкин Г. В. Методы вычислительной реализации рангового метода кластеризации // Информатика и системы управления. 2012. № 1(31). С. 71–79.

[4] Гренкин Г. В., Черныш Е. В. ClRank // Свидетельство Роспатента о государственной регистрации программы для ЭВМ. — Рег. № 2012612452 от 06.03.2012.