АНАЛИЗ ЭФФЕКТИВНОСТИ И ОПТИМИЗАЦИЯ АЛГОРИТМА К-СРЕДНИХ

Агурова Л.П.[1], Огнева М.В.[2]
[1] lidiya.ag@gmail.com , [2]ognevamv@gmail.com,
Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского, Саратов, Россия;

Алгоритм k-means

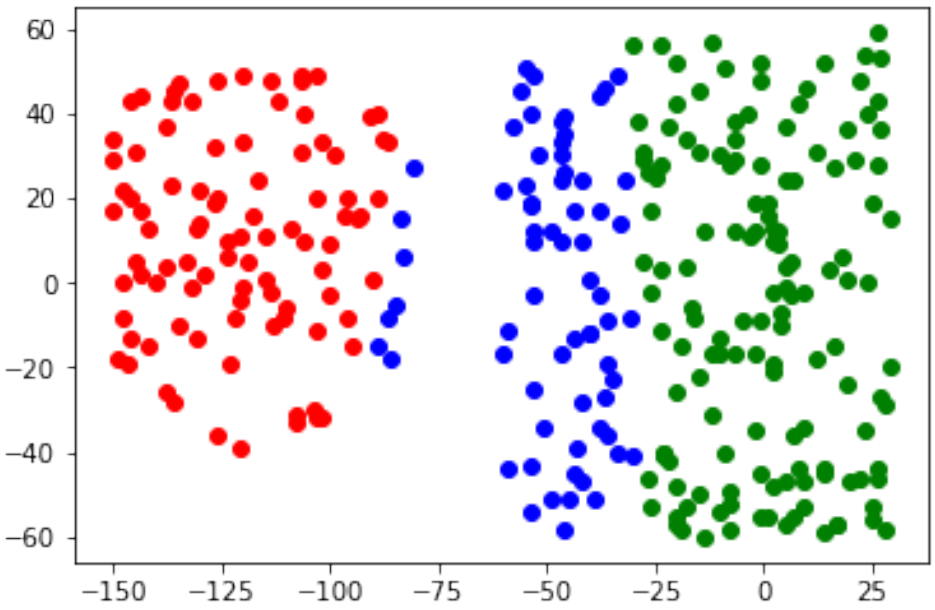
1. Случайно выбираются k точек - начальные центры кластеров.
2. Каждая точка относиться к кластеру с ближайшим центром.
3. Центры пересчитываются.
4. Если условие остановки не выполняется, вернуться к шагу 2.

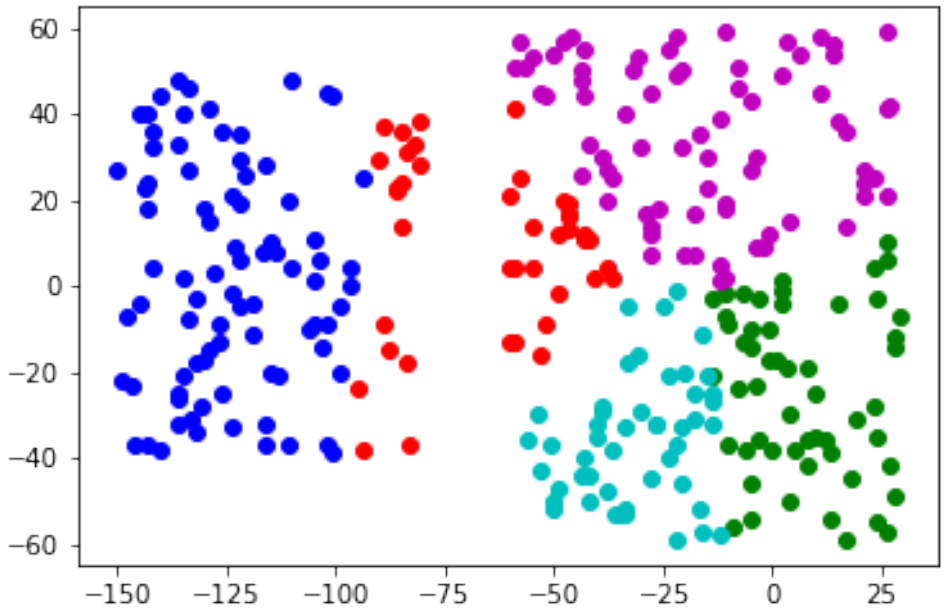
Необходимо заранее выбрать количество кластеров, на которые будут делиться объекты.

Алгоритм k-means

Результат может быть неточным и иметь погрешность.

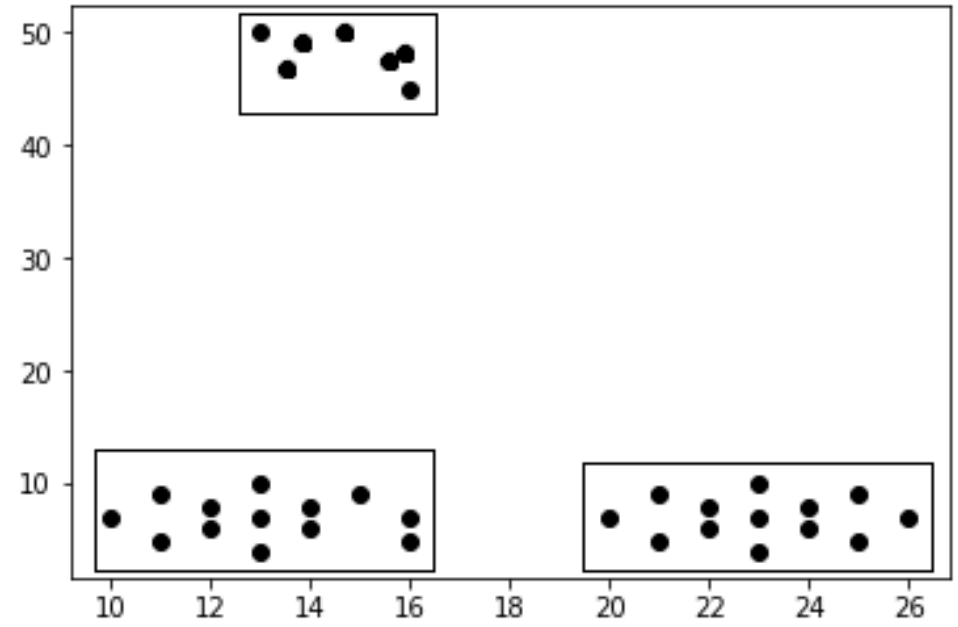
Алгоритм k-means плохо работает с кластерами, имеющими неправильную или вытянутую форму.

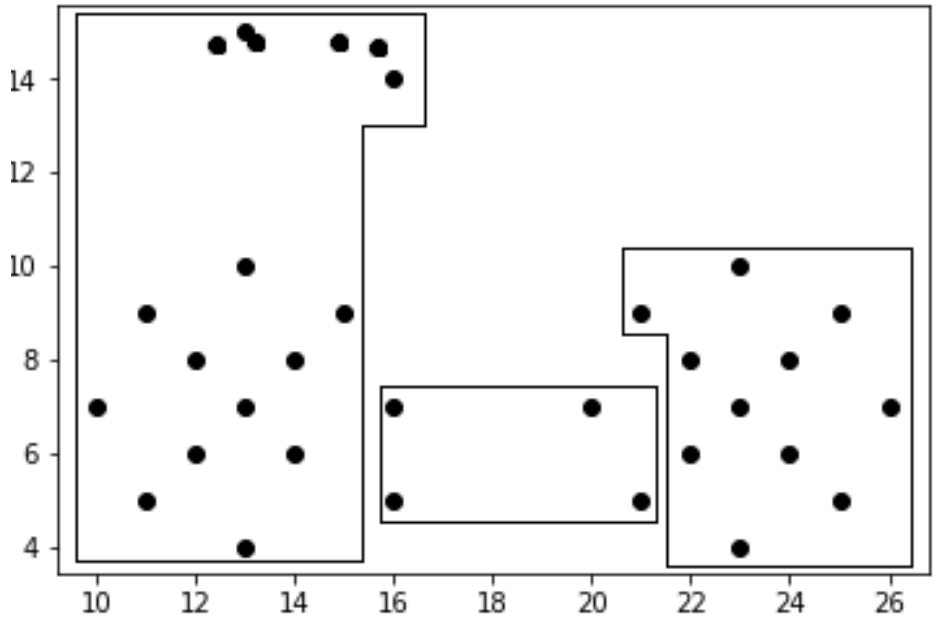




Алгоритм k-means

При значимом расстоянии между кластерами алгоритм k-means дает хорошие результаты. С уменьшением расстояния точность алгоритма снижается.





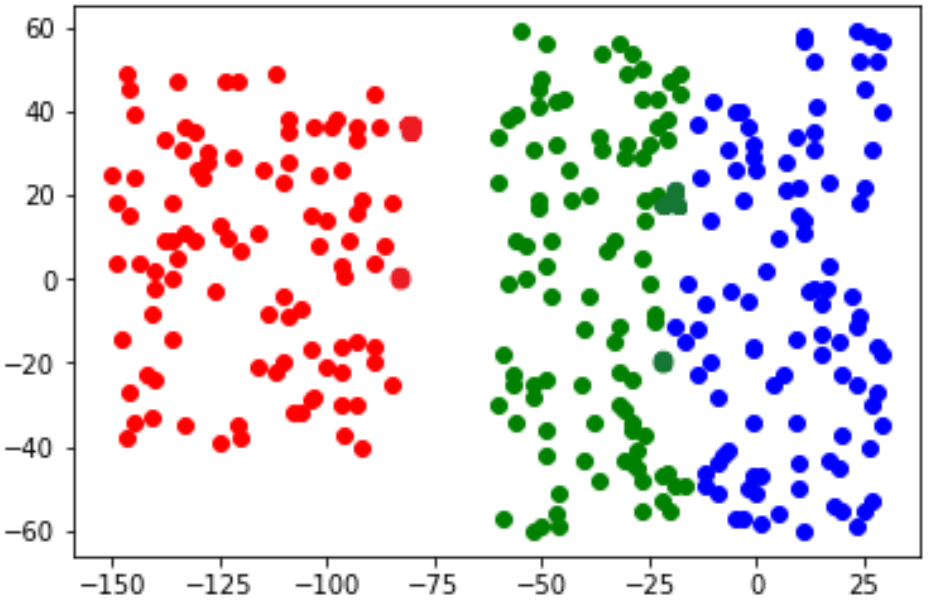
Алгоритм k-means++

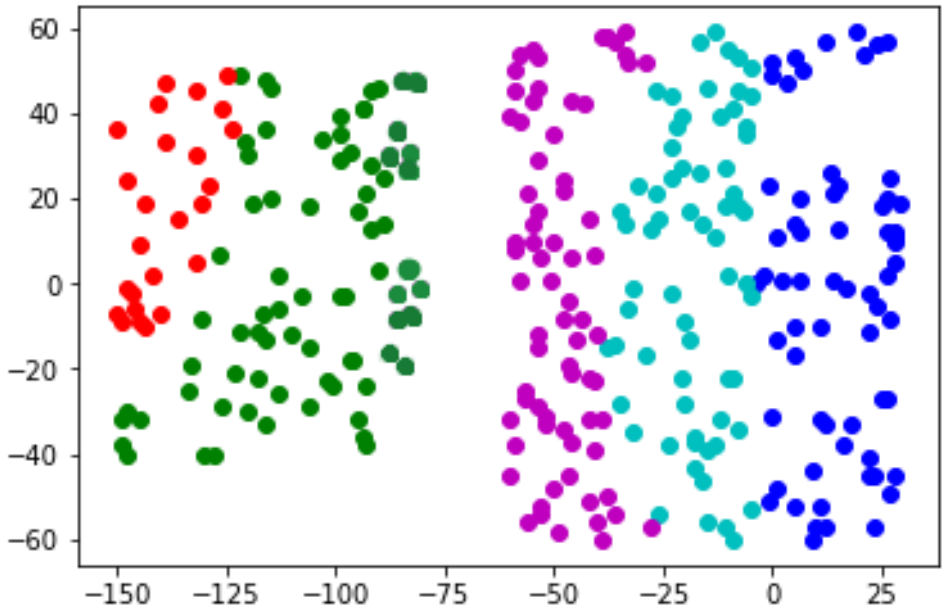
1. Случайно выбрать точку - начальный центр первого кластера.
2. Для каждой точки найти минимальное расстояние до любого из центров.
3. Следующий центр вычислить с помощью формулы вероятностного распределения.
4. Повторить шаги 2 и 3, пока количество найденных центров не будет равно k.
5. Продолжить работу с шага 2 алгоритма k-means.

Алгоритм k-means++

Вероятностное распределение позволяет выбрать начальные центры максимально далеко друг от друга.

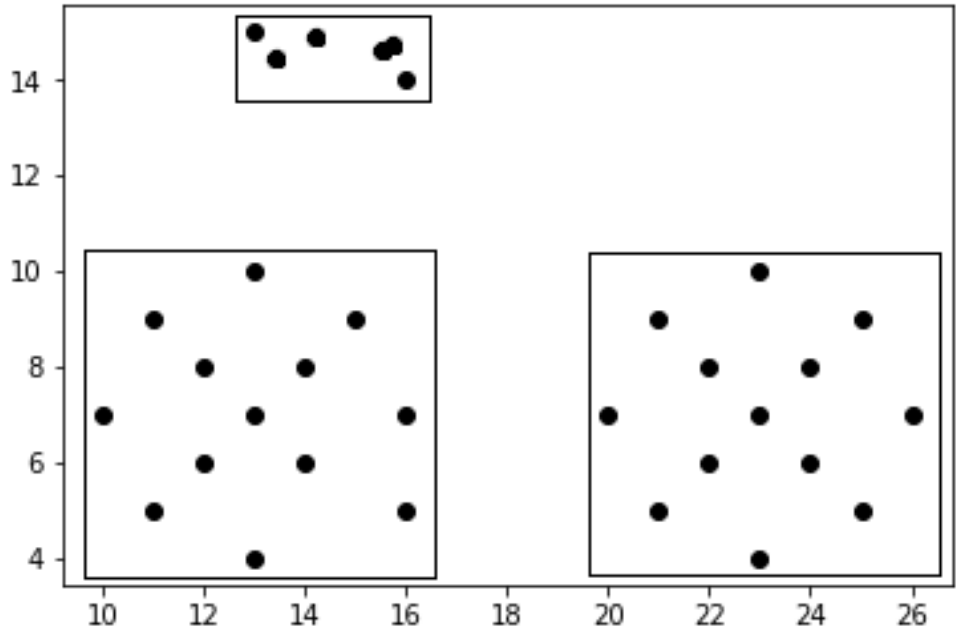
Результаты алгоритма k-means++ являются более точными, чем результаты работы алгоритма k-means.





Алгоритм k-means++

Также для кластеров правильной формы алгоритм работает точно даже с относительно небольшими расстояниями.



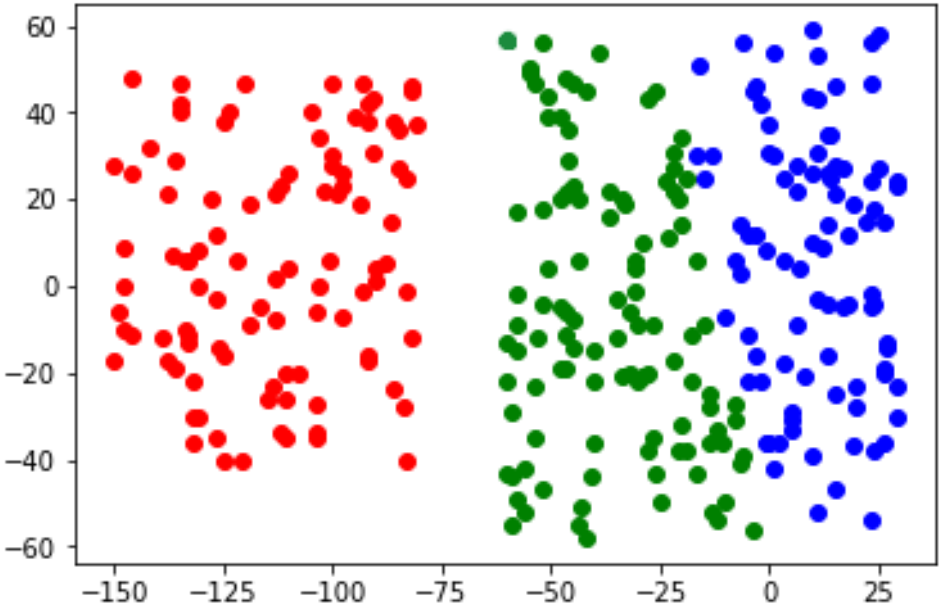
Алгоритм c-means

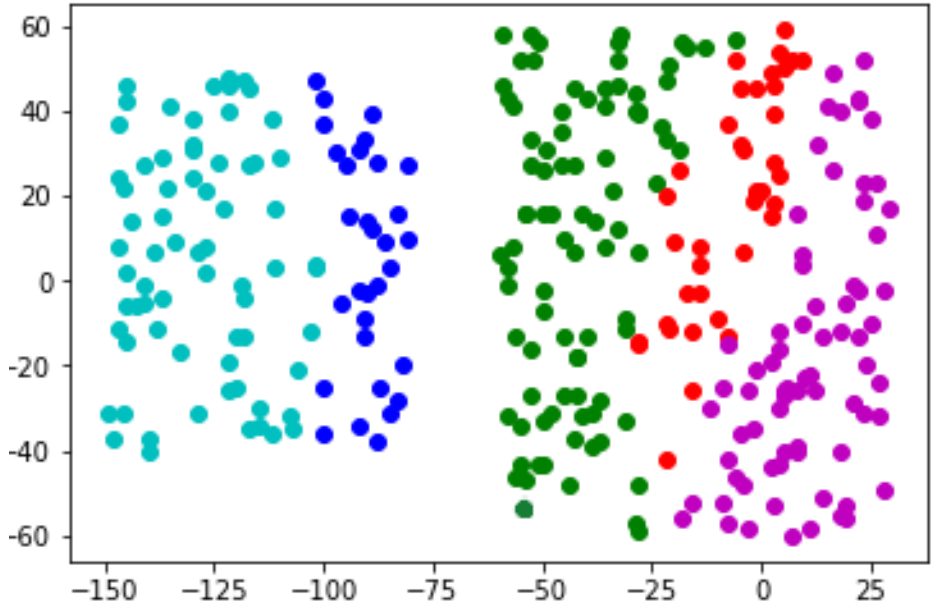
1. Для каждой точки случайным образом указать коэффициенты принадлежности.
2. Вычислить центры кластеров.
3. Обновить степени принадлежности точек кластерам.
4. Если условие остановки алгоритма не выполнено, то вернуться к шагу 2.

Определяет вероятность отношения объекта к каждому из кластеров - коэффициент принадлежности.

Алгоритм c-means

Результаты работы алгоритма c-means будут аналогичны результатам работы алгоритма k-means++.





Время выполнения

Количество точек	K-means	K-means++	C-means
1 000	13ms	14ms	33ms
5 000	40ms	77ms	1469ms
7 000	53ms	89ms	2034ms
10 000	102ms	250ms	2913ms
50 000	318ms	668ms	14800ms

Метод локтя

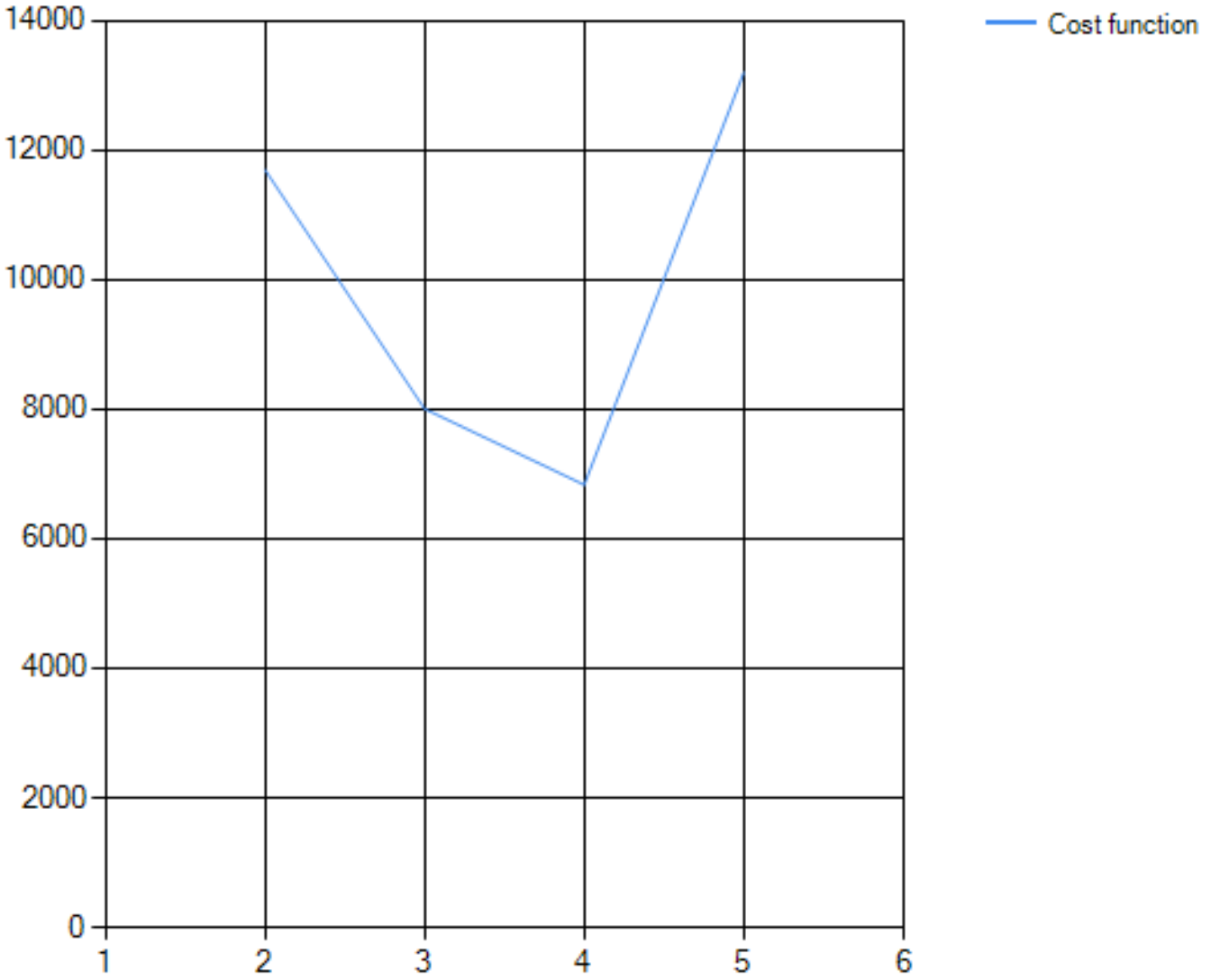
Внутрикластерная сумма квадратичных ошибок (SSE):

$$F=\sum_{i=1}^{n}\sum_{j=1}^{k}\omega^{m(i,j)}\left\|x^{(i)}-\mu^{(j)}\right\|_2^2, m\in[1,\infty],$$

где μ - центры кластеров, x - данные точки, $\omega^{(i,j)}$ є [0,1] - степень принадлежности точки i кластеру j, m - коэффициент нечеткости.

Метод локтя

k-means



Выводы

1) Алгоритм k-means дает выигрыш во времени.

2) Алгоритмы k-means++ и c-means дают точный результат для более широкого диапазона типов входных данных.

3) Метод локтя помогает избавиться от проблемы необходимости задания числа кластеров.