Кластеризация точек на плоскости

Баталов Семен

25.02.2021

1. Постановка задачи

Рассматривается плоскость, на которую случайным образом наносятся точки. Нужно решить задачу кластеризации методом «**K-Means++**». Используется язык «**Python**», подробнее о программе можно узнать в папке «**source**» проекта.

1.1. K-Means и K-Means++

Алгоритм «**K-Means**» разбивает множество элементов векторного пространства на заранее известное число кластеров «k». В нашем случае размерность пространства равна 2, а элементами являются точки, которые описываются только двумя параметрами: абсциссой и ординатой.

Основная идея заключается в том, что на каждой итерации перевычисляется центр масс для каждого кластера, полученного на предыдущем шаге, затем векторы разбиваются на кластеры вновь в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике.

Алгоритм завершается, когда на какой-то итерации не происходит изменения внутрикластерного расстояния. Это происходит за конечное число итераций, так как количество возможных разбиений конечного множества конечно, а на каждом шаге суммарное квадратичное отклонение уменьшается, поэтому зацикливание невозможно.

Важным этапом в алгоритме является первоначальная инициализация центров классов. В «**K-Means**» центры выбираются случайно.

Алгоритм «**K-Means++**» ничем, кроме способа начальной инициализации центров, не отличается от «**K-Means**». В «**K-Means++**» центры выбираются (как правило) удаленными друг от друга, что с большей вероятностью приводит к лучшим результатам, чем случайная инициализация.

2. Инструменты

Для работы была выбрана библиотека «**sklearn**», из нее были взяты алгоритмы «**KMeans**» и «**kmeans_plusplus**». Первый из них осуществляет метод «**K-Means**», второй генерирует центры кластеров в соответствии с алгоритмом «**K-Means++**».

Для генерации случайных точек на плоскости использовались алгоритмы библиотеки «**sklearn**», а именно «**make_blobs**». Этот алгоритм создает многоклассовые

наборы данных, выделяя каждому классу один или несколько нормально распределенных кластеров точек. «**make_blobs**» обеспечивает контроль относительно центров и стандартных отклонений каждого кластера и используется для тестирования алгоритмов кластеризации.

3. Результаты

В экспериментах варьировалось количество точек на плоскости, количество кластеров, коэффициенты, отвечающие за распределение точек по плоскости.

Основной задачей было показать конечный результат работы кластеризатора и в случае «**K-Means++**» оценить расположение классов, используя начальную инициализацию центров, сравнить результаты с оригинальным разбиением на классы.

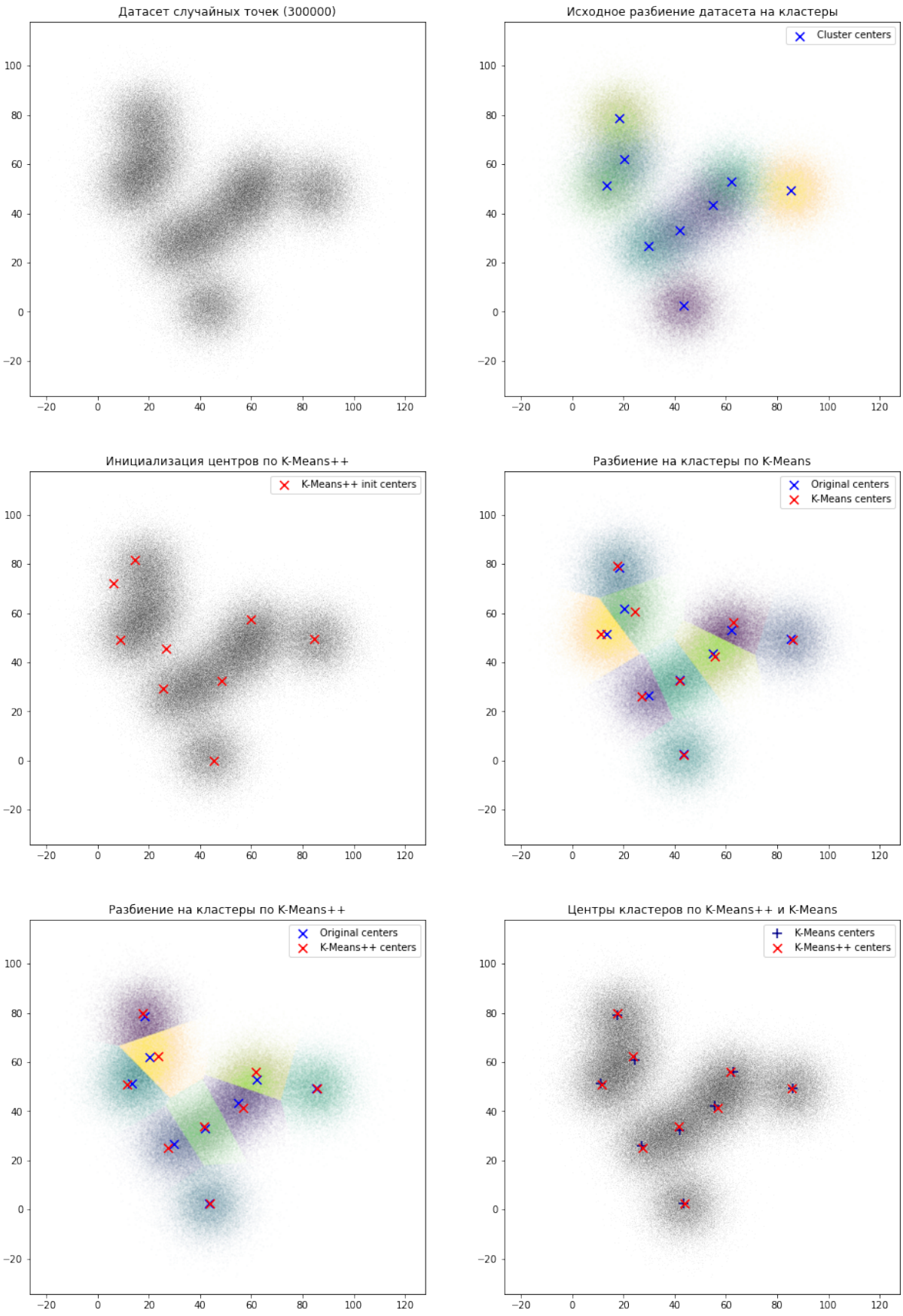

Рис. 1. 9 кластеров, 300000 точек

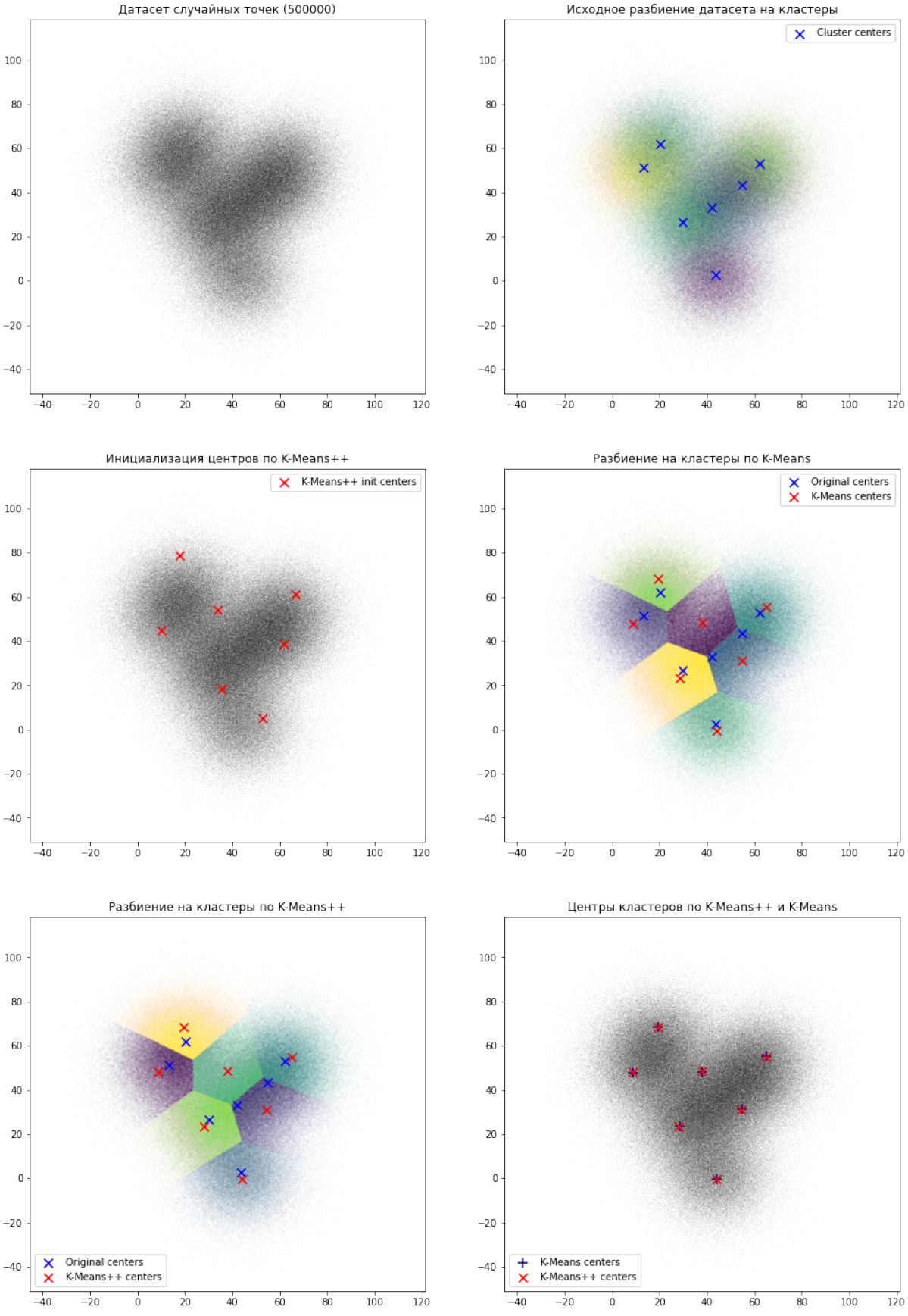

Рис. 2. 7 кластеров, 500000 точек

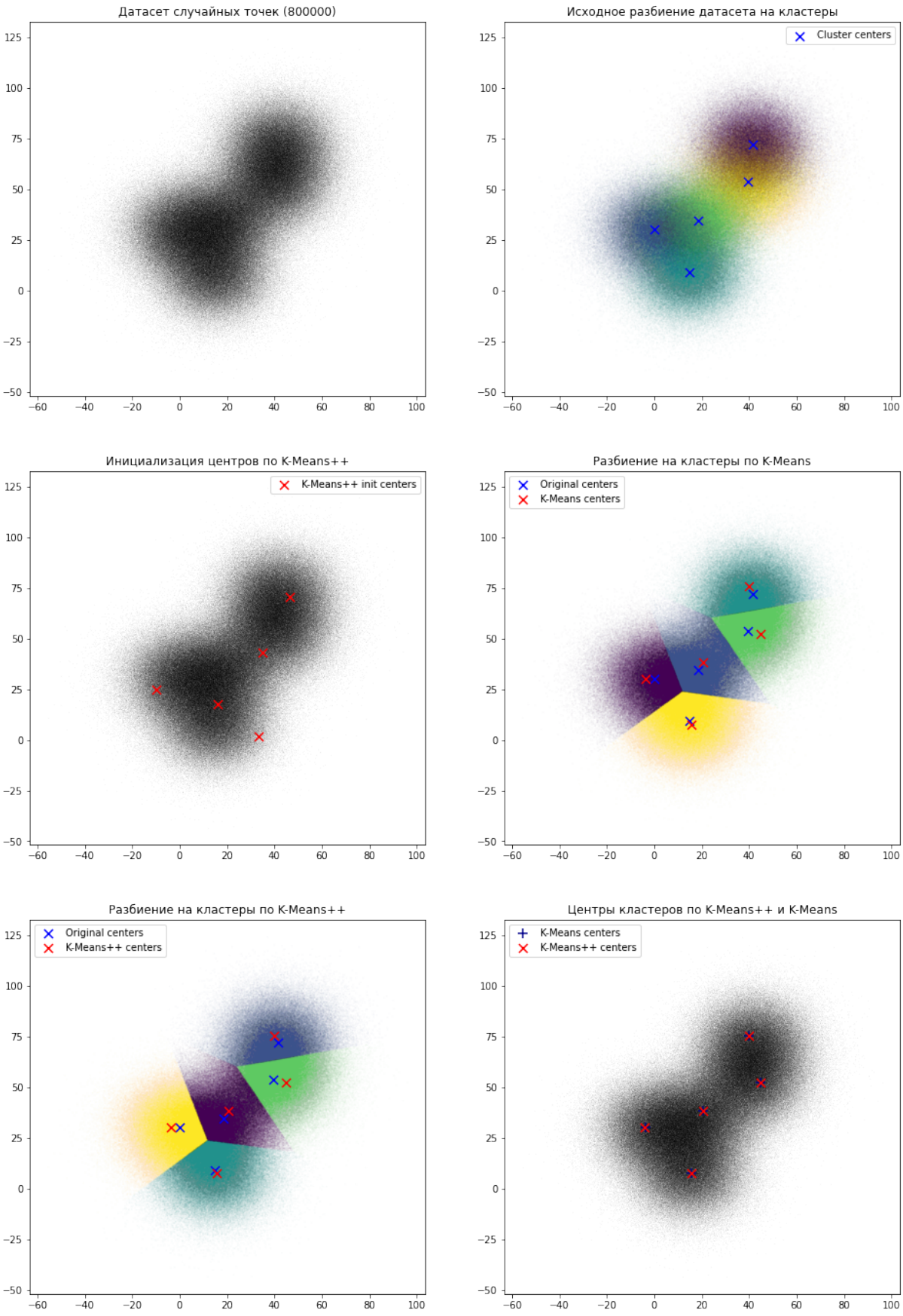

Рис. 3. 5 кластеров, 800000 точек

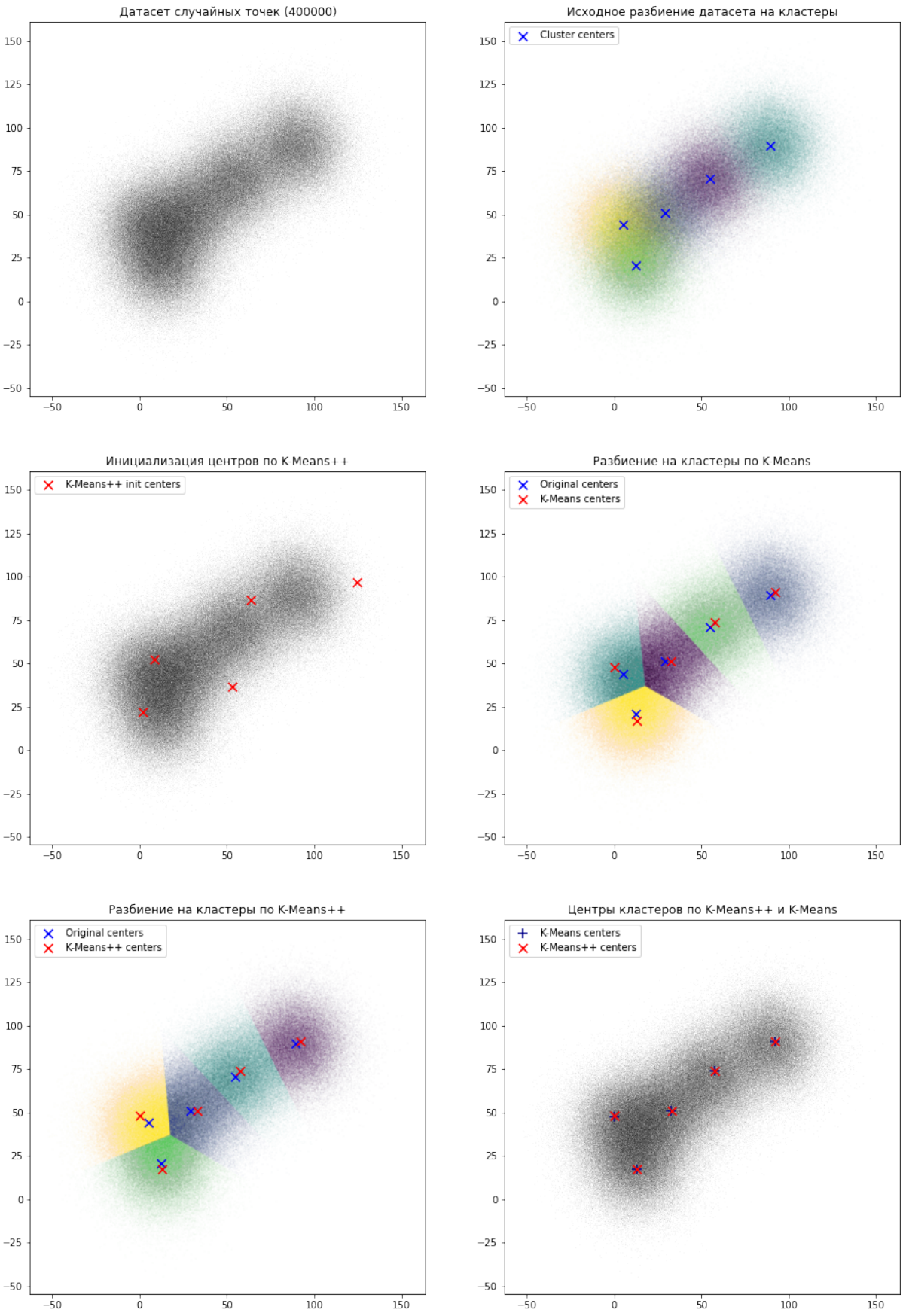

Рис. 4. 5 кластеров, 400000 точек

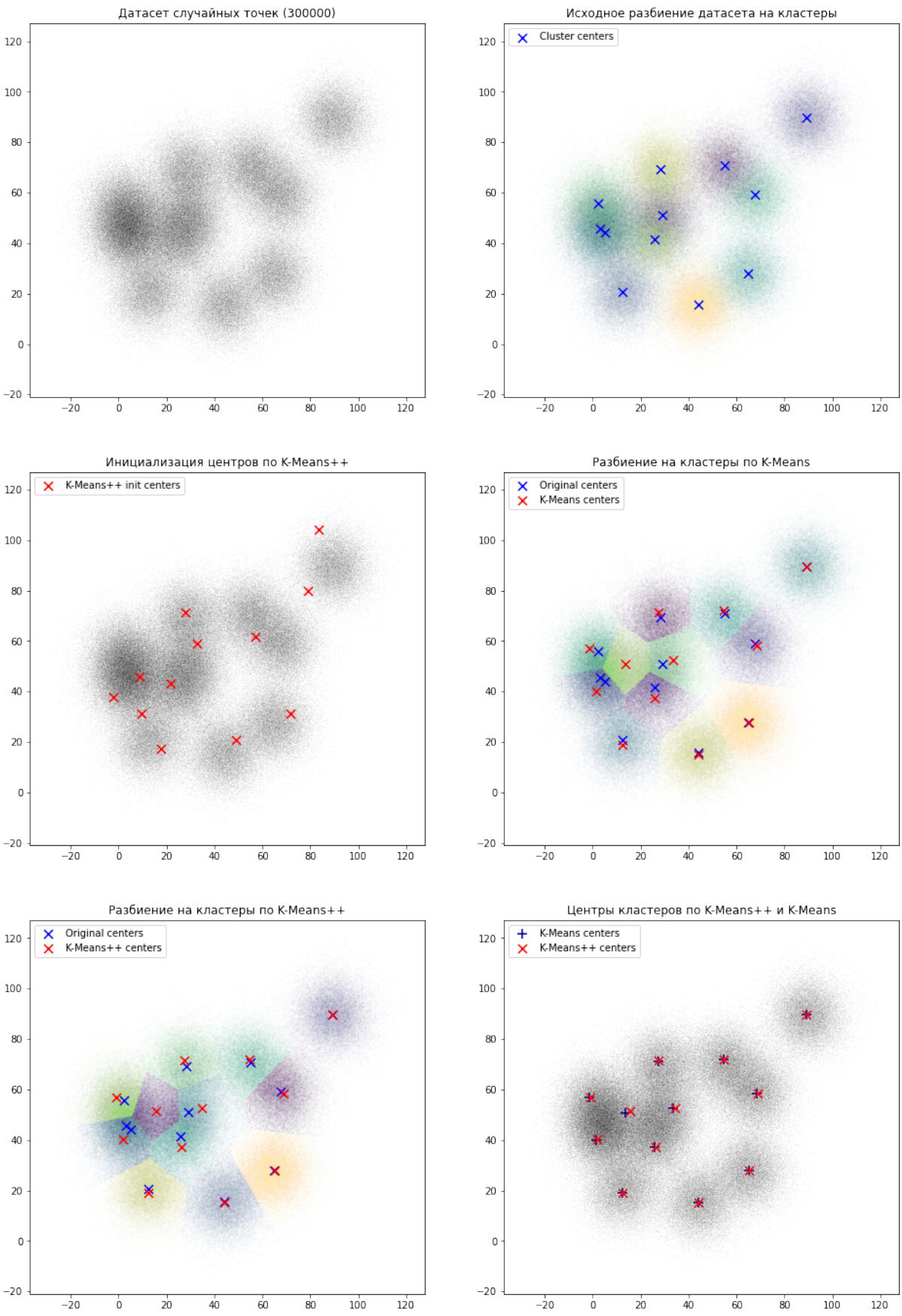

Рис. 5. 12 кластеров, 400000 точек