Санкт-Петербургский
Политехнический университет Петра Великого

Направление подготовки 01.03.02
«Прикладная математика и информатика»
Биоинформатика

Лабораторная работа №3
Тема: «Кластеризация многомерных объектов
методом Kernel K-means»
Дисциплина: «Многомерный статистический анализ»

Студент: Кулянин Анатолий
Преподаватель: Павлова Л.В.

Группа: 5030102/10401

Санкт-Петербург
2025

Содержание

1 Постановка задачи

Задача:

1. Подготовка модельных данных
   (a) данные хорошо разделены
   (b) данные плохо разделены
2. Кластеризация модельных данных:
   Алгоритм Kernel k-means (или, по договорённости, обычный k-means): для различных значений гиперпараметра K провести кластеризацию модельных данных (хорошо и плохо разделенных). Оценить качество кластеризации (визуально; вычислить внешние/внутренние критерии). Построить график J(C(K)), выбрать оптимальное значение K (или найти K, минимизируя D(K)).
   Привести графическую иллюстрацию результатов кластеризации, также представить результаты в виде таблицю
3. Привести анализ полученных результатов.

2 Подготовка данных

В качестве хорошо разделённых данных берутся выборки из многомерных нормальных распределений с одинаковыми диспресиями.

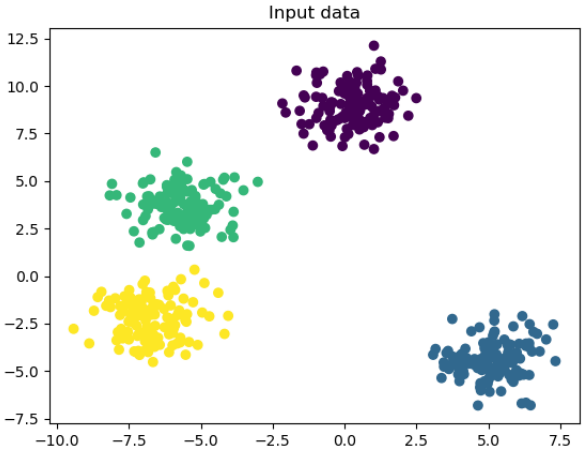

Рис. 1: Пример хорощо разделённых данных

В качестве плохо разделённых данных берутся выборки "Moons"с шумом.

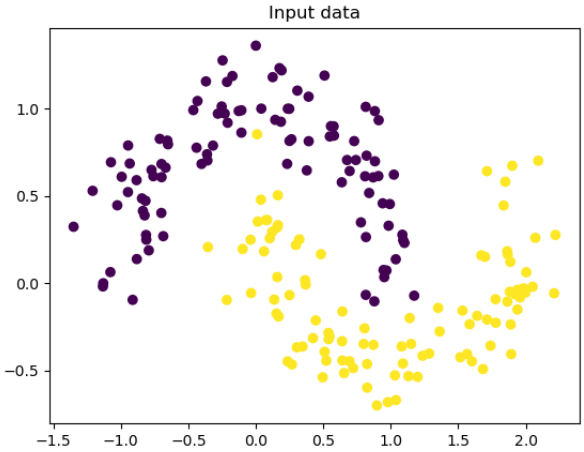

Рис. 2: Пример плохо разделённых данных

3 Кластеризация модельных данных

3.1 Хорошо разделённые

Проведём кластеризацию для $K \in [2, 7]$.

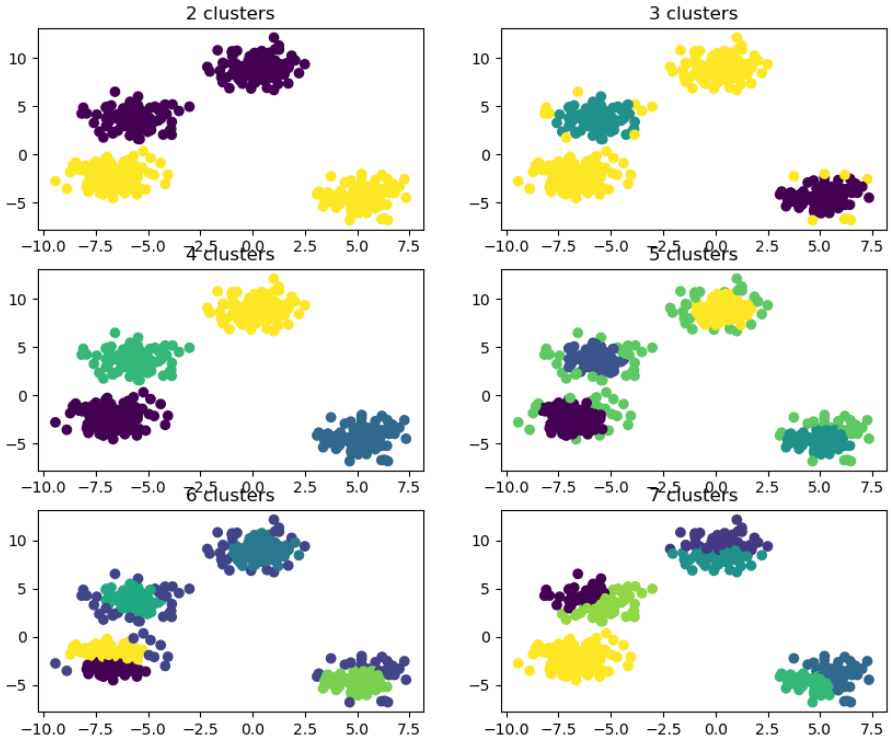

Рис. 3: Кластеризация blobs для K от 2 до 7

Значения критерия "Силуэт":

Kernel counts	Silhouette mean
2	0.286
3	0.417
4	**0.514**
5	0.408
6	0.518
7	0.499

По графикам можно заметить, что ядерный k-means правильно кластеризут данные (при $K = 4$). При попытки выделить 5 кластеров, метод выделил 4 кластера, похожих на изначальные, а в 5-ый кластер поместил "выбросы"из первых 4-х кластеров. Аналогичное поведение наблюдается и для 6-ти кластеров. Более яркий пример такого поведения можно рассмотреть на немного зашумлённых данных ткого же типа.

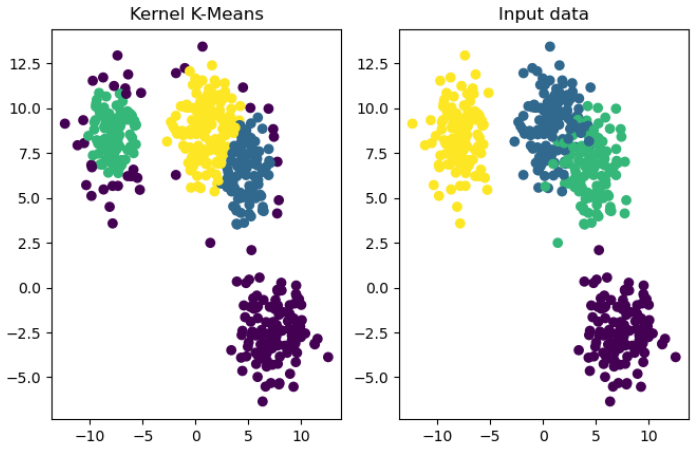

Рис. 4: Кластеризация немного зашумлённых blobs(K = 4) и исходные данные

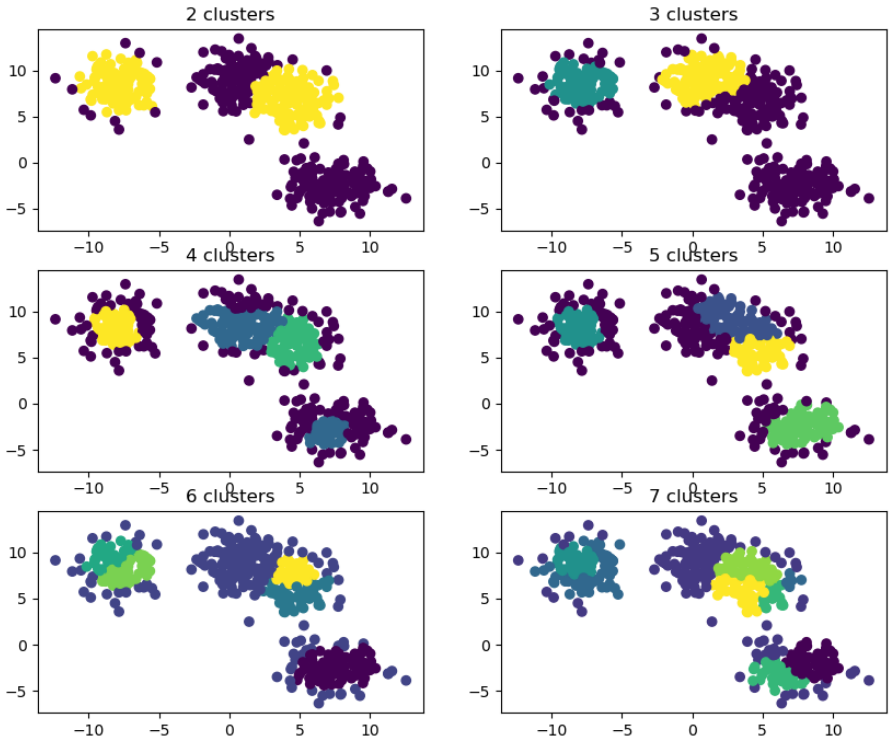

Рис. 5: Кластеризация немного зашумлённых blobs для K от 2 до 7

На графиках видно, что при любом количестве кластеров, метод объединяет все выбросы в один кластер.

3.2 Плохо разделённые

Проведём кластеризацию для $K \in [2, 7]$, используя Гауссого ядро.

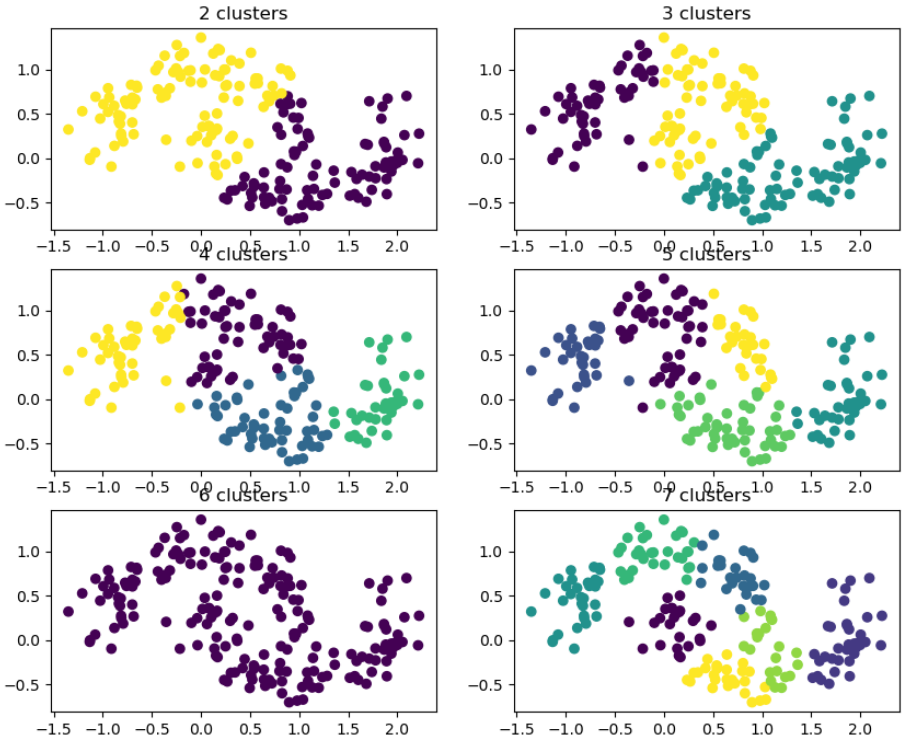

Рис. 6: Кластеризация moons для K от 2 до 7

Значения критерия "Силуэт":

Kernel counts	Silhouette mean
2	0.674
3	0.666
4	0.717
5	0.740
6	nan
7	0.761

По графику видно, что метод с RBF ядром полность теряет форму исходных данных, но хорошо рзбивает их на небольшие подкластеры. Это свойство хорошо отражается в таблице значения критерия "Силуэт".

3.3 Сравнение разных ядер

Проведём сравнения работы метода с разными ядрами для moons

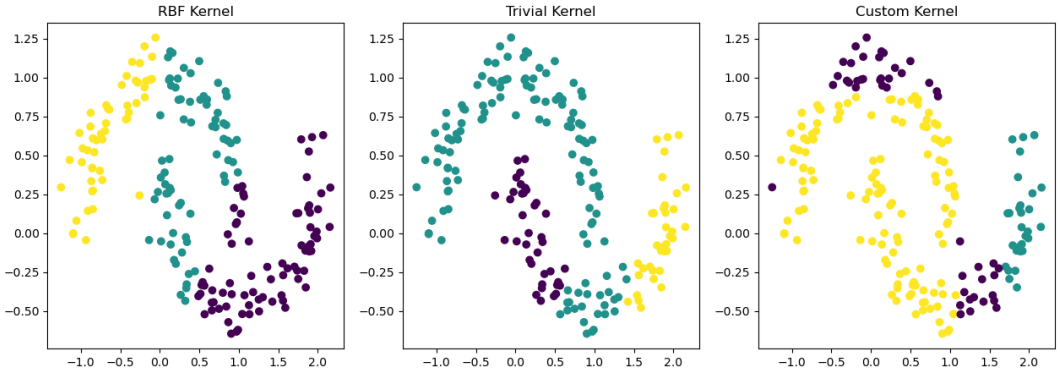

Рис. 7: Кластеризация moons с разными ядрами

В качестве ядер были взяты:

- RBF : $k(x, y) = \exp\left(-\frac{||x - y||}{2}\right)$
- Trivial : $k(x, y) = 2||x|| \cdot ||y||$
- Custom : $k(x, y) = \exp\left(-\frac{||f(x) - f(y)||}{2}\right), \quad f(t) = t^3$

Заметно, что тривиальное ядро лучше остальных сохраняет форму исходных кластеров.

Проведём кластеризацию с тривиальным ядром для различного числа кластеров.

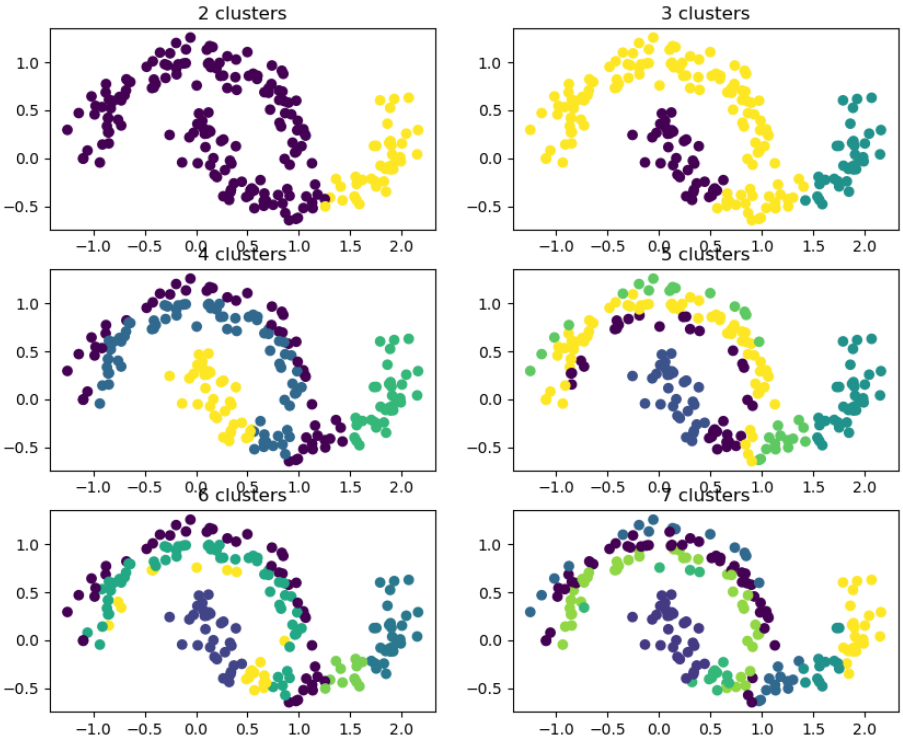

Рис. 8: Кластеризация moons с тривиальным ядром

Значения критерия "Силуэт":

Kernel counts	Silhouette mean
2	0.9
3	0.92
4	0.844
5	0.835
6	0.821
7	0.812

Из графиков и таблицы видно, что в отличие от RBF ядра, тривиальное ядро лучше сохраняет форму исходных данных, но плохо справляется с задачей разбиения над подкластеры.

Интересным примером сохранения формы тривиальным ядром является разбиения данных типа коцентрических кругов

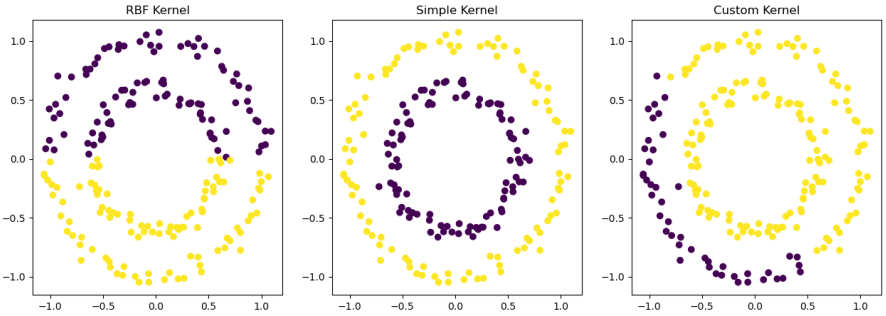

Рис. 9: Кластеризация circle с разными ядрами

А вот пользовательское ядро не дало сильных изменений в сравнение с RBF, лишь немного лучше сохраняя форму

4 Выводы

Kernel k-means является гибким инструментом в задачах кластеризации. С помощью правильного выбора ядра можно проводить кластеризацию различных типов данных.

RBF ядро хорошо подходит для клатеризации выпуклых множеств. Также подходит для разбиения исходных кластеров на более маленькие подкластеры, без потери качества (по критерию "Силуэт") и способено выделять "выбросы"в отдельный кластер.

Тривиальное хорошо подходит для данных сложной формы.