

ОТЧЁТ

ПО

ЛАБОРАТОРНОЙ РАБОТЕ

**«Самоорганизующиеся карты Кохонена и методы
кластерного анализа»**

Учебная дисциплина «Нейронные сети»

Группа: БПМ-16-2

Студент: Новицкий Дмитрий

Преподаватель: доц., к.т.н. Курочкин И. И.

Отметка:

Дата защиты:

2019 г.

Оглавление

Постановка задачи	3
Основное условие	3
Результаты и визуализация.....	3
Входные/выходные данные	3
Загрузка файлов	3
Описание работы программы.....	5
Используемые параметры системы	5
Краткий алгоритм работы программы	5
Результаты работы программы	7
Результаты работы обучения карты Кохонена	7
Результат работы кластеризации.....	12

Постановка задачи

Основное условие

1. Реализовать самоорганизующуюся карту Кохонена (SOM)
2. Реализовать метод кластерного анализа с евклидовой метрикой + метрика (по вариантам – метрика Чебышева)
3. Реализовать вычисление не менее 4 различных показателей качества разделения.
4. Провести сравнительный анализ качества работы SOM и метода кластерного анализа на различных датасетах (сгенерированных самостоятельно и эталонных датасетах).
 - линейно разделимые множества (с расстоянием между группами в 10^3 раз больше, чем диаметр группы);
 - линейно разделимые множества (группы расположены близко или касаются друг друга);
 - линейно неразделимое множество (средняя площадь пересечения классов 10-20%);
 - линейно неразделимое множество (средняя площадь пересечения классов 50-70%);
 - 2 эталонных датасета с различным количеством признаков, но не менее 7.

Результаты и визуализация

1. Визуализация результатов работы SOM.
2. Для 2-хмерных данных визуализация результатов работы метода кластерного анализа.
3. Сравнительная таблица для 2 методов кластеризации с показателями качества разделения для различных датасетов.

Входные/выходные данные

1. Входные данные (датасеты) в виде текстового файла. (к примеру, экспорт таблицы из Excel в формате TXT или CSV).
2. Выходные данные (результаты) сохраняются в виде, необходимом для формирования отчета PDF или DOCX.

Загрузка файлов

1. Работающее приложение.

2. Датасеты.
3. Текстовый файл со сценариями запуска (к примеру: myapp.exe dataset1 results).
4. Подробный отчет по ЛР с приведенными результатами и пояснениями (к примеру, если Вы получили какой-то график, приведите ниже сценарий и параметры запуска вычислительного приложения).

Описание работы программы

Алгоритм работы карты Кохонена был реализован в данной лабораторной работе с минимальным использованием сторонних библиотек. В лабораторной работе были необходимы следующие библиотеки:

- `sklearn.manifold` – TSNE. Данная библиотека необходима для уменьшения размерности данных для отрисовки на плоскости
- `numpy`
- `random`
- `matplotlib.pyplot`. Данная библиотека необходима для отрисовки данных на плоскости
- `copy`
- `math`

Для реализации карты Кохонена были созданы два класса – класс для реализации карты Кохонена и класс для нейронов в карте Кохонена.

Используемые параметры системы

Изначально передаются следующие параметры в класс Карты Кохонена:

- путь к исходному датасету
- длина сетки карты Кохонена
- ширина сетки карты Кохонена
- количество эпох «обучения» карты Кохонена
- скорость обучения карты Кохонена
- тип метрики
- радиус для кластеризации данных

В классе карты Кохонена по умолчанию задаются следующие параметры:

- `sigma_0`
- `tau_1`
- `tau_2`
- `speed_0`

Все вышеперечисленные значения используются в процессе обучения карты Кохонена.

Краткий алгоритм работы программы

1. Инициализируем экземпляр класса карты Кохонена и считываем датасет с файла
2. Проверяем датасет на корректность

3. Конвертируем значения переменной датасета класса карты Кохонена в тип значений float
4. Удаляем из датасета значения классов (последний столбец)
5. Инициализируем нейроны карты Кохонена. В классе карты Кохонена создаём список и заполняем его экземплярами класса «нейрон»
6. Если количество признаков в датасете равно двум (двумерные данные), то рисуем их на плоскости
7. Применяем нормализацию к данным датасета
8. Применяем алгоритм обучения карты Кохонена с использованием заданных метрик
9. Применяем алгоритм кластеризации обученной карты Кохонена на основе заданного метода кластерного анализа

Результаты работы программы

Рассмотрим сначала результат работы обучения карты Кохонена, а затем рассмотрим процесс кластеризации.

Результаты работы обучения карты Кохонена

Результат работы программы будем рассматривать на моём датасете – «predictiong a pulsar star».

Установим следующие параметры работы карты Кохонена:

- длина сетки карты Кохонена – 8
- ширина сетки карты Кохонена – 8
- количество эпох «обучения» карты Кохонена – 10000
- скорость обучения карты Кохонена – 0.1
- тип метрики – Евклидова

Результаты работы программы будут следующими:



Рис. 1. Итерация №0.

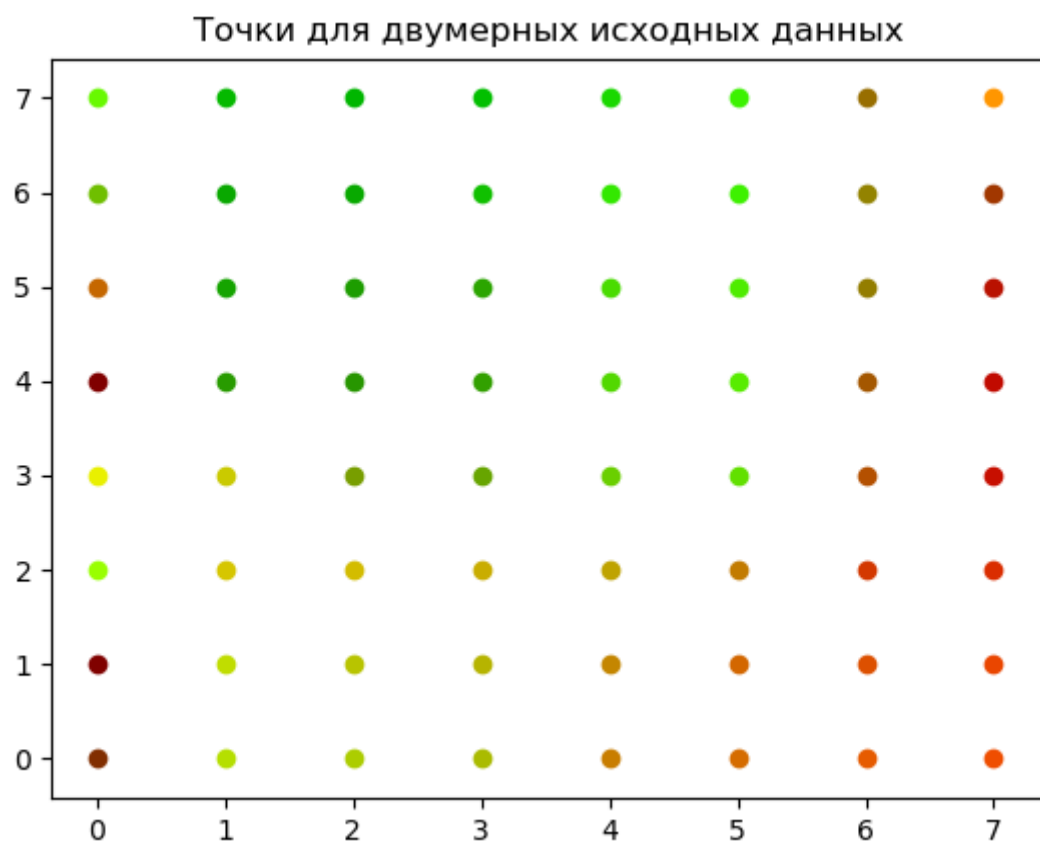


Рис. 2. Итерация № 2500.

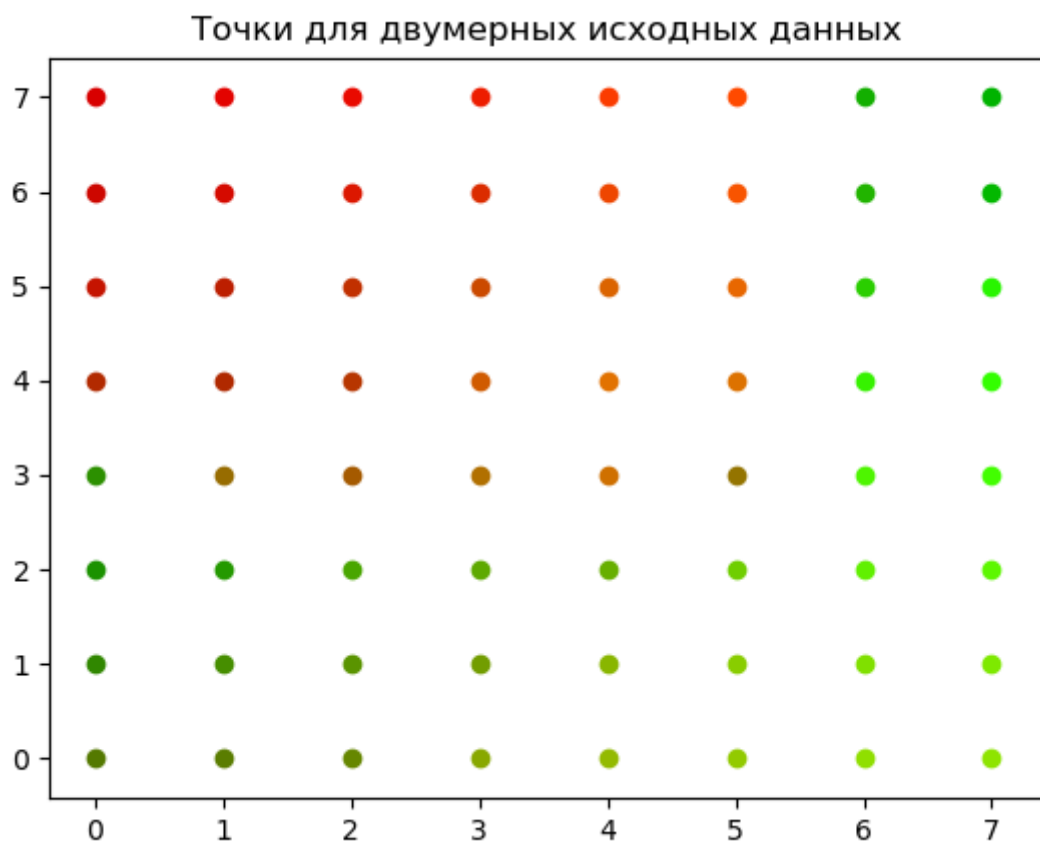


Рис. 3. Итерация № 5000.

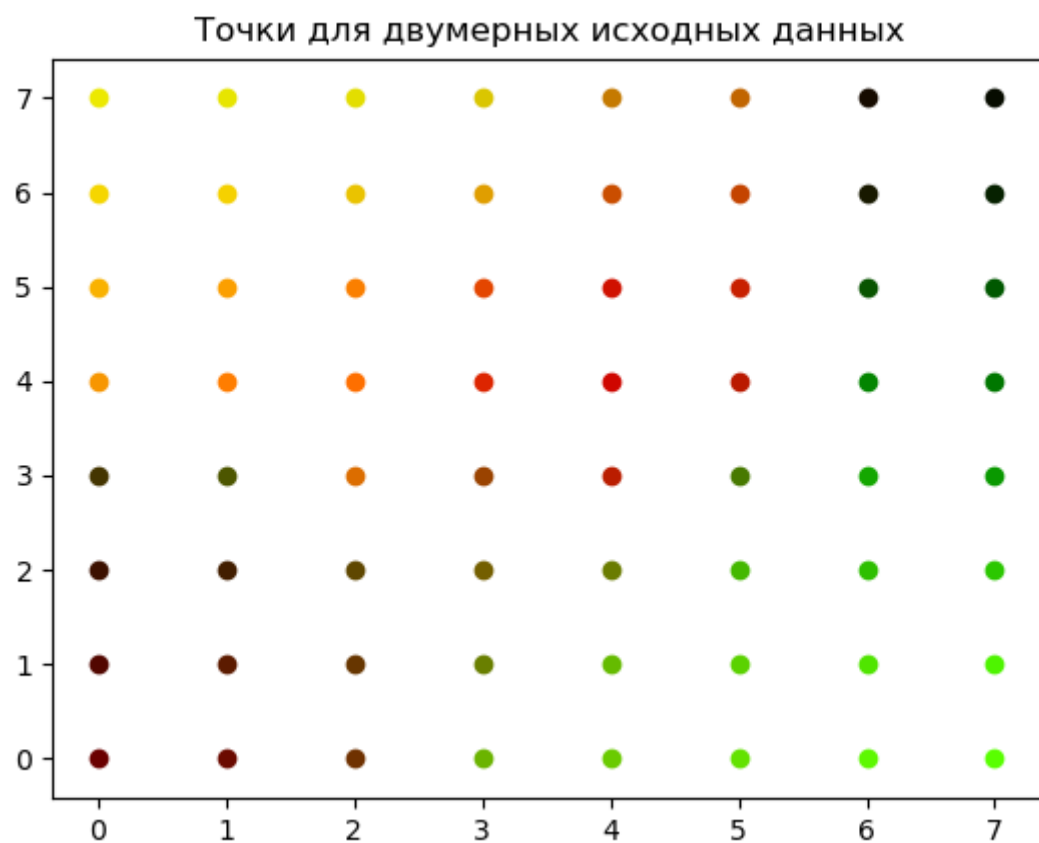


Рис. 4. Итерация № 7500.

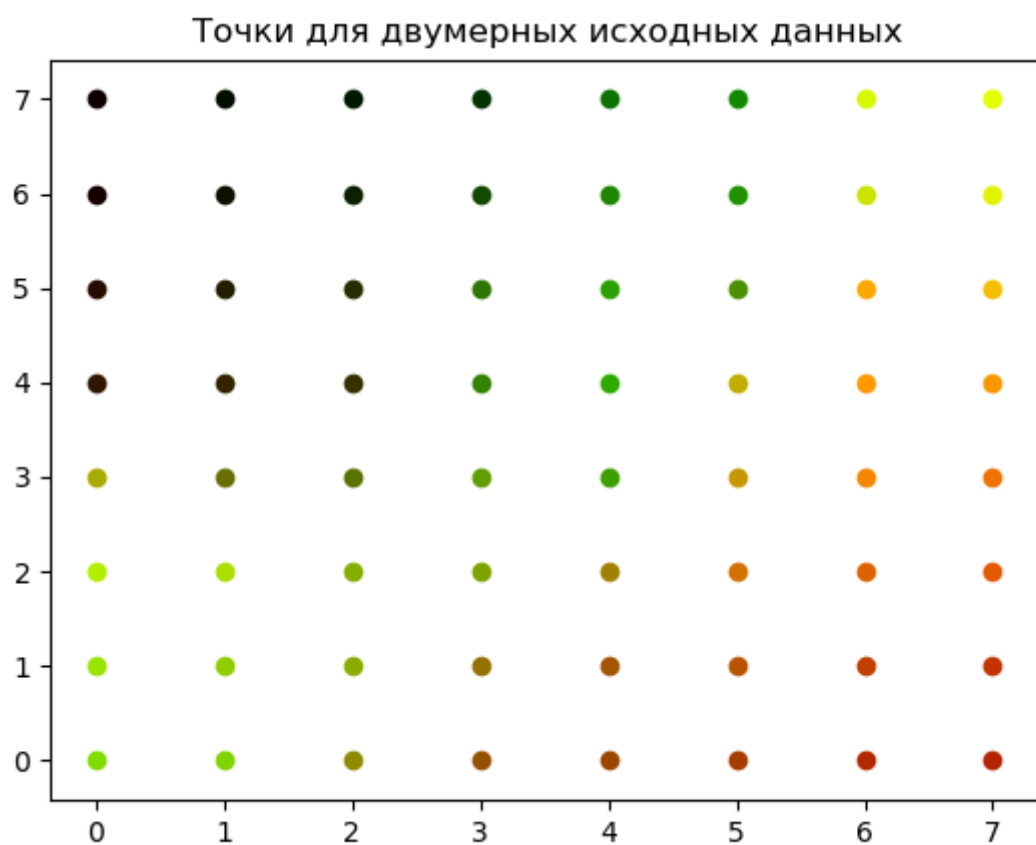


Рис. 5. Итерация № 10000.

Установим следующие параметры работы карты Кохонена:

- длина сетки карты Кохонена – 8
- ширина сетки карты Кохонена – 8
- количество эпох «обучения» карты Кохонена – 50000
- скорость обучения карты Кохонена – 0.1
- тип метрики – Чебышева

Результаты работы программы будут следующими:

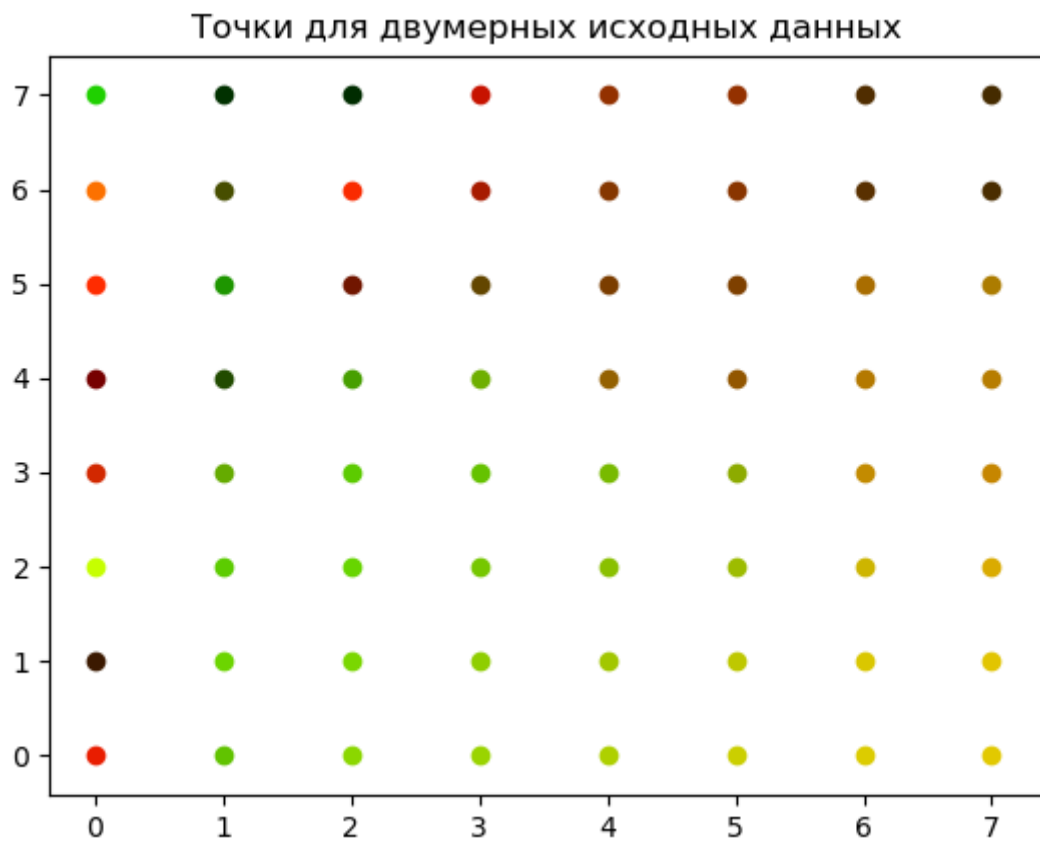


Рис. 6. Итерация № 10000.

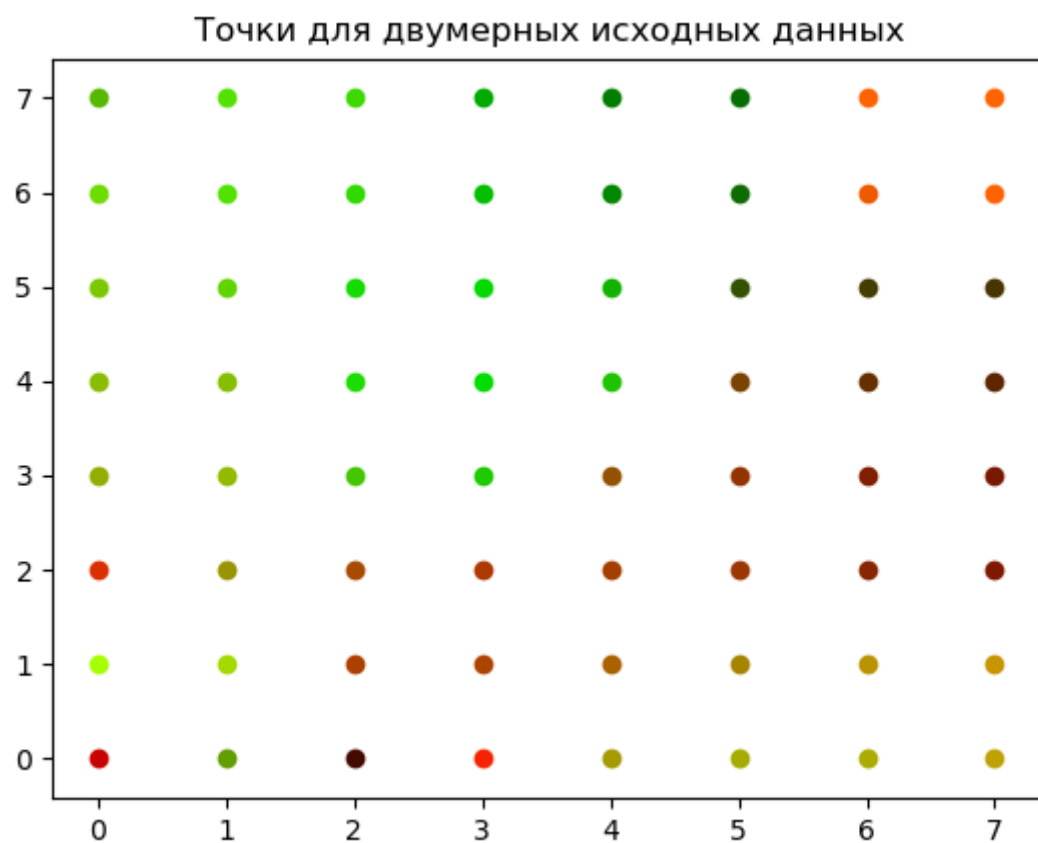


Рис. 7. Итерация № 25000.

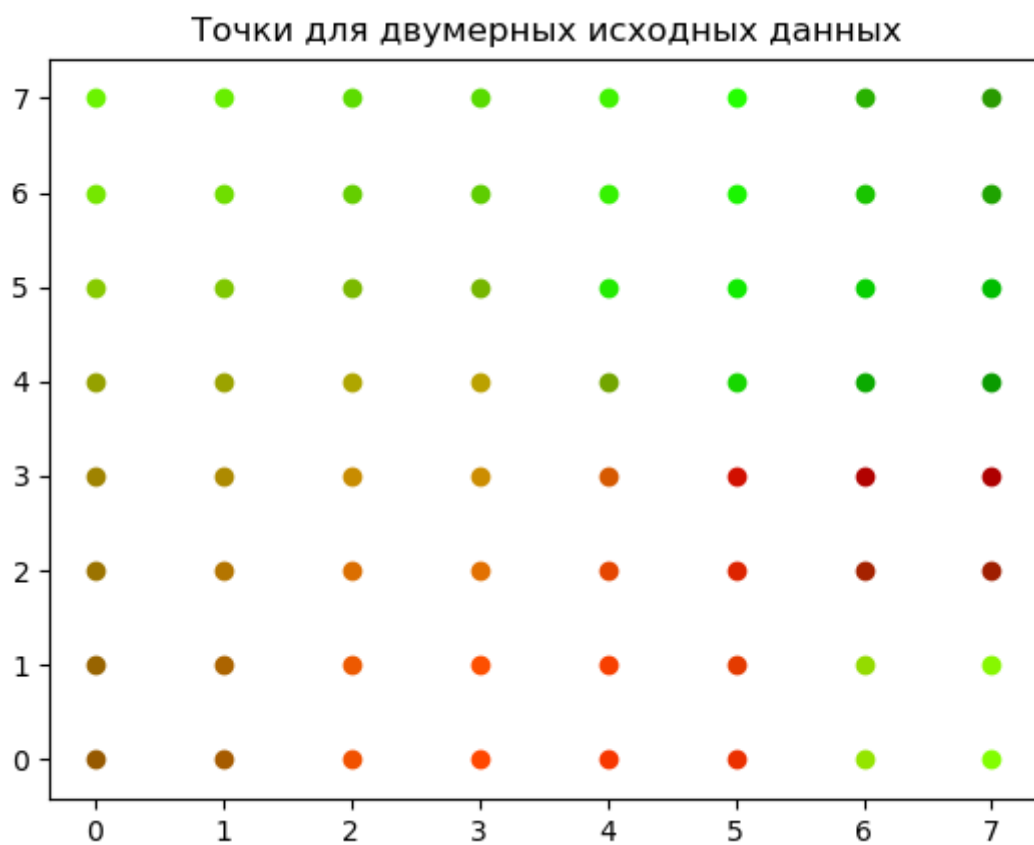


Рис. 8. Итерация № 25000.

Результат работы кластеризации

Установим следующие параметры работы карты Кохонена:

- длина сетки карты Кохонена – 8
- ширина сетки карты Кохонена – 8
- количество эпох «обучения» карты Кохонена – 1000
- скорость обучения карты Кохонена – 0.1
- тип метрики – Евклидова
- радиус для кластеризации данных – 5

Результаты работы программы будут следующими:



Рис. 9. Нейроны после кластеризации.

Как видно, все нейроны окрашены в один цвет. Это значит, что все нейроны сетки карты Кохонена были отнесены к одному классу. Это говорит о большом значении радиуса кластеризации данных.

Изменим радиус кластеризации данных с 5 до 1. Результаты работы программы будут следующими:

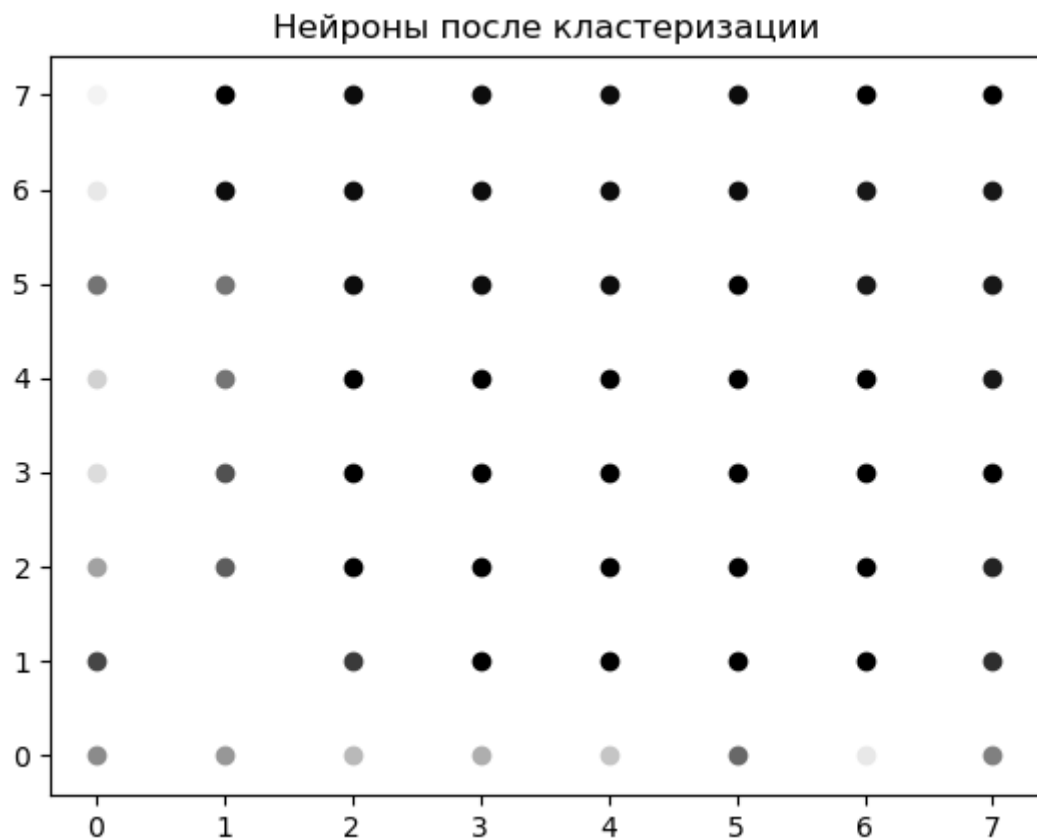


Рис. 10. Нейроны после кластеризации.

Всего получилось 23 класса (кластера), что является большим значением, что говорит о низком значении радиуса для кластеризации, что делает кластеризацию очень чувствительной к помехам.

Изменим радиус кластеризации данных до 3. Результаты работы программы будут следующими:

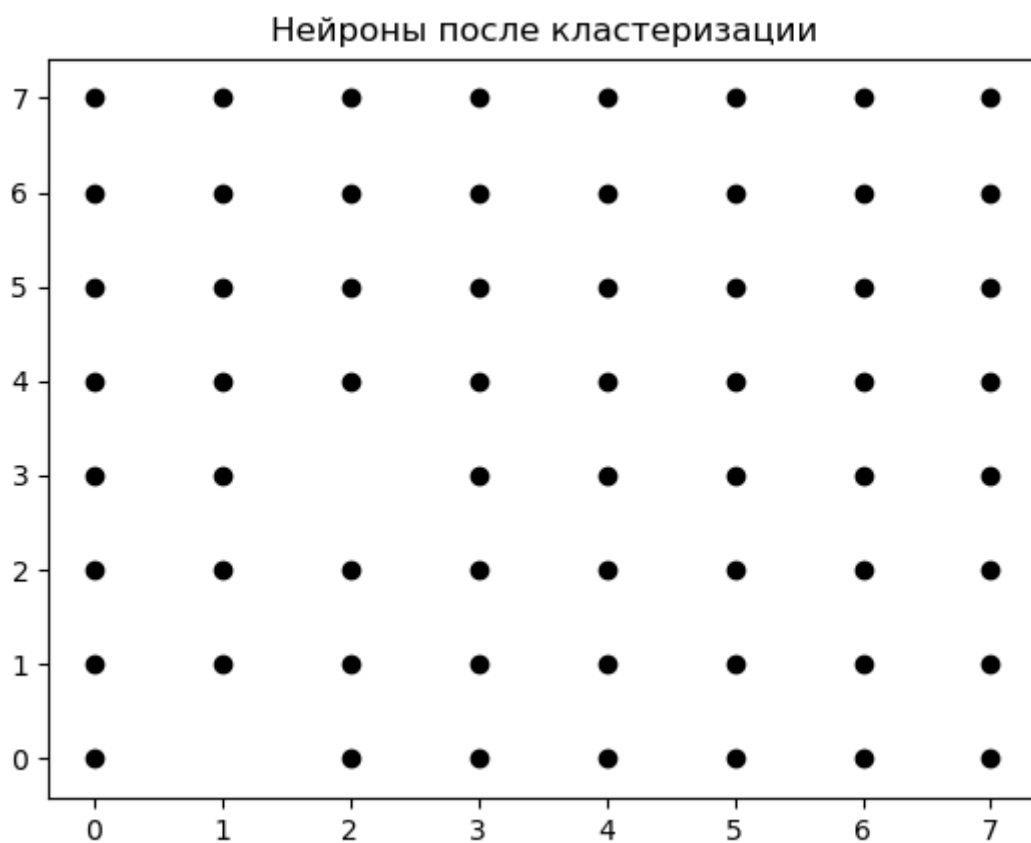


Рис. 11. Нейроны после кластеризации.

Результат работы говорит о более точной классификации.