МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ

по лабораторной работе №5

по дисциплине «Машинное обучение»

Тема: Кластеризация (k-средних, иерархическая)

Студент гр. 6307 _____ Золотухин М. А.

Преподаватель _____ Жангиров Т. Р.

Санкт-Петербург

2020

Загрузка данных

```python
import pandas as pd
import numpy as np

data = pd.read_csv('iris.data', header=None)
no_labeled_data = data.drop(4, axis=1)
```

K-means

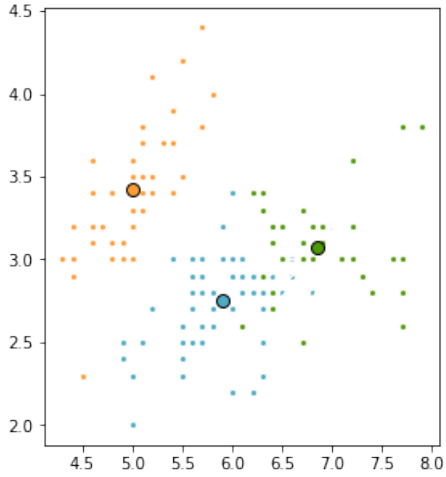
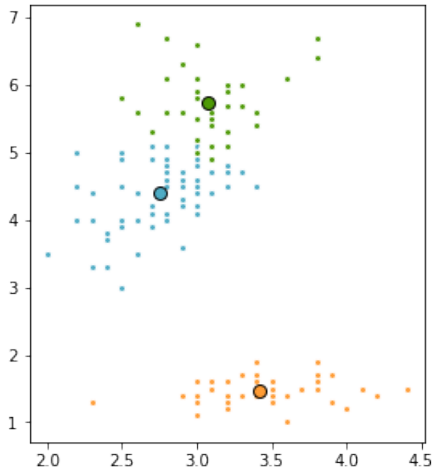
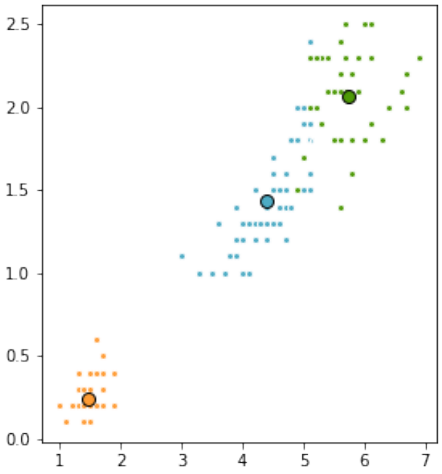
Проведем кластеризацию методов k-средних:

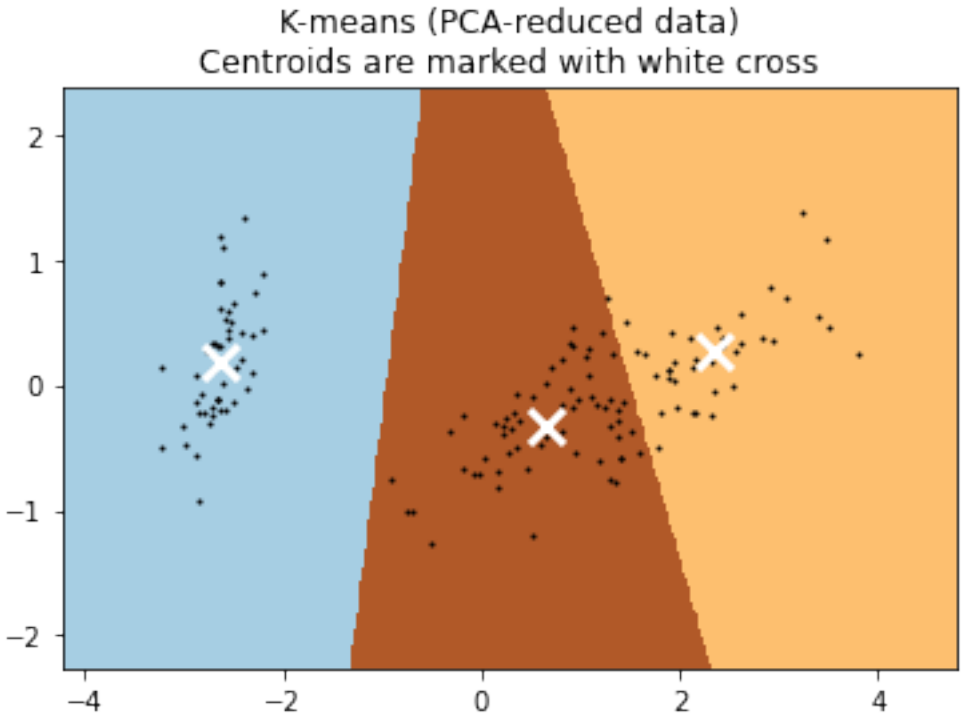
```python
from sklearn.cluster import KMeans

k_means = KMeans(init='k-means++', n_clusters=3, n_init=15)
k_means.fit(no_labeled_data)
```

Получим центры кластеров и определим какие наблюдения в какой кластер попали

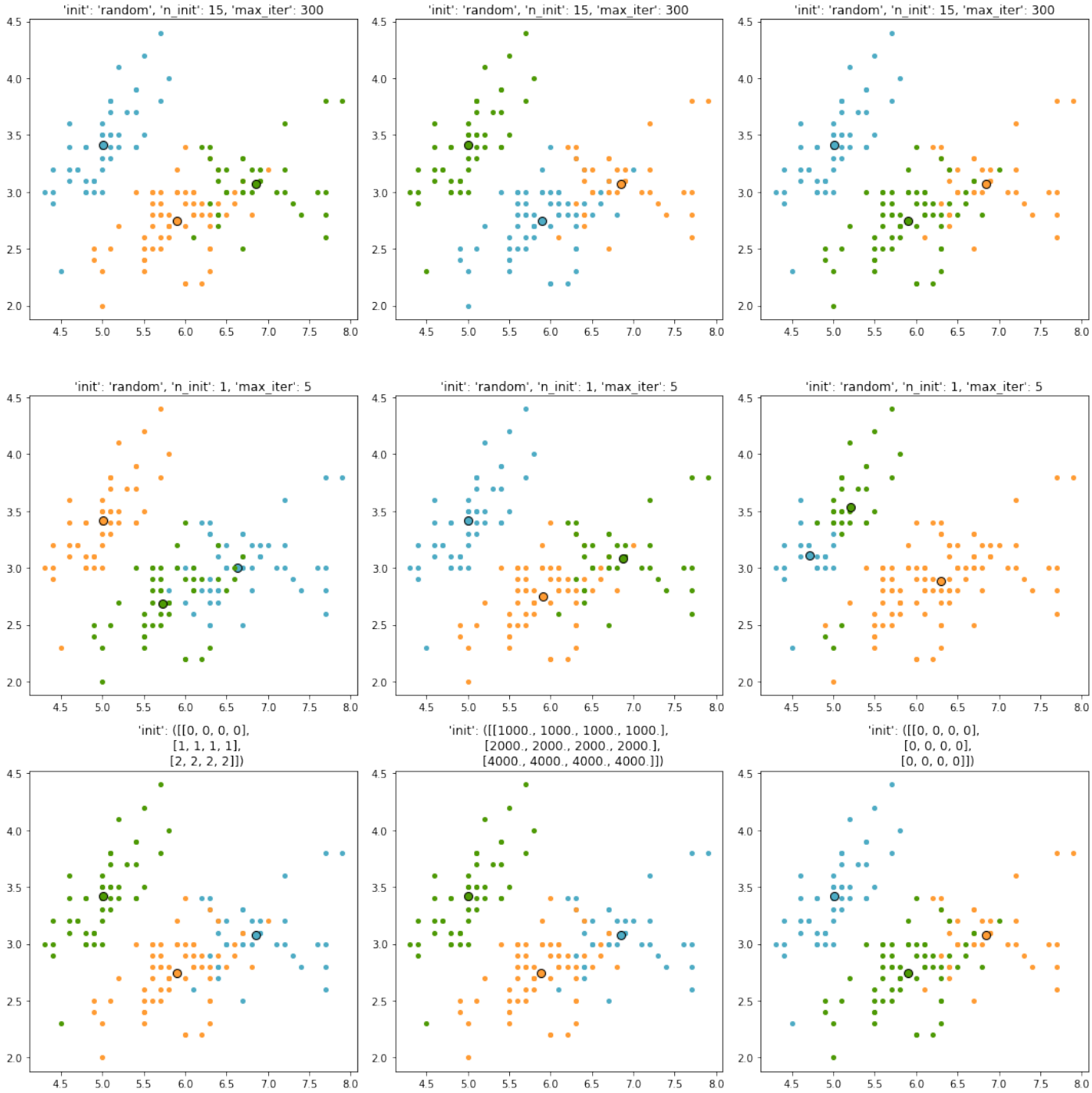
```python
from sklearn.metrics.pairwise import pairwise_distances_argmin

k_means_cluster_centers = k_means.cluster_centers_
k_means_labels = pairwise_distances_argmin(no_labeled_data,
```

k_means_cluster_centers)

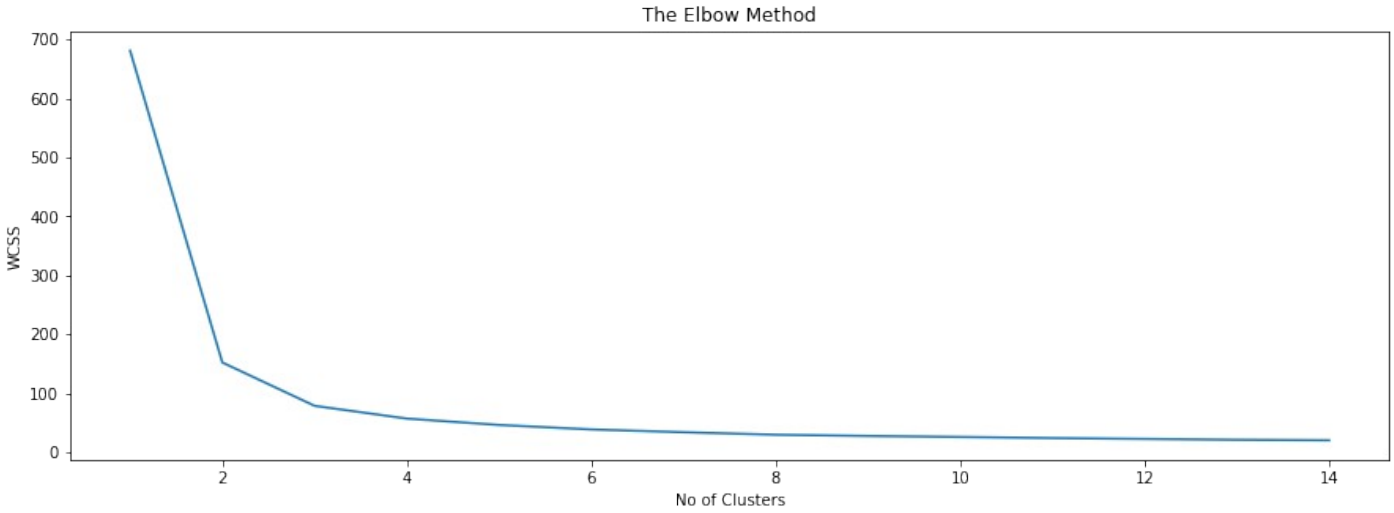
```

Построим результаты классификации для признаков попарно (1 и 2, 2 и 3, 3 и 4)










Наилучшее разделение было произведено по 1 и 2 признаку, так как для других признаков, как видно из графиков, количество кластеров лучше было взять 2, а не 3. Это субъективная оценка "наг глаз", для оценки лучше использовать более объективный метод - подсчет ошибки в зависимости от объема кластера.

`n_init` - это количество раз, когда изначальные позиции центроидов будут инициализированы с разными сидами. Чем больше, те результат (в теории) более точный (т.е. не зависисм от случайности). Особенно это важно, если в качестве метода инициализации был выбран `random`, а не `kmeans++`.

Уменьшим размерность данных до 2 используя метод главных компонент и нарисуем карту для всей области значений, на которой каждый кластер занимает определенную область со своим цветом.
```



Исследуем работу алгоритма k-средних при различных параметрах init. Сначала надо выполнить несколько раз с параметров 'random', затем для вручную выбранных точек.

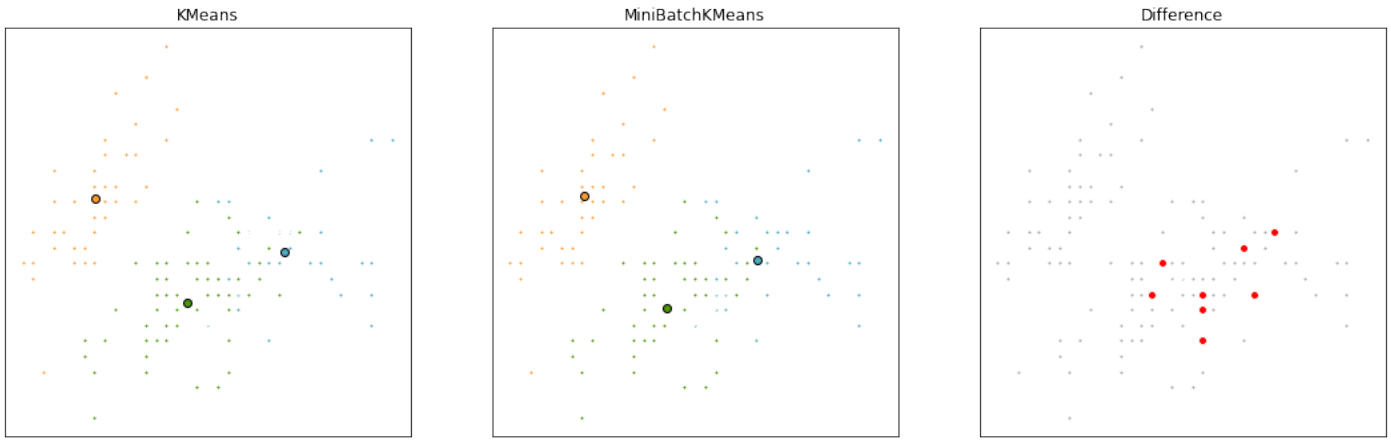


Определим наилучшее количество методом локтя.



Как видно - оптимальным значением для размера кластера является - 2.

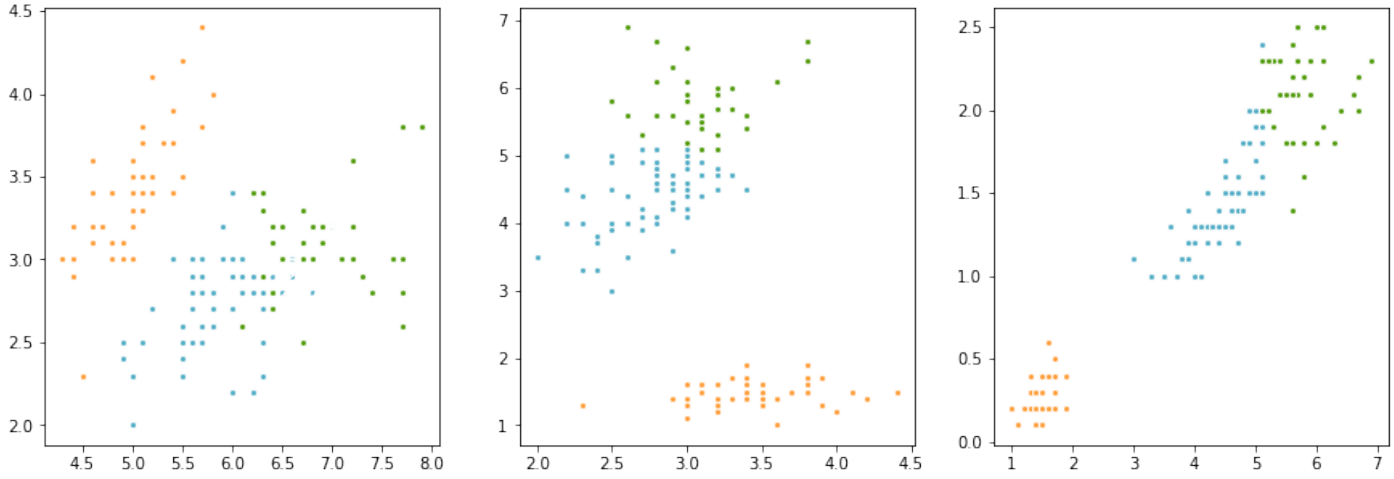
Проведём кластеризацию используя пакетную кластеризацию k-средних. Построим диаграмму рассеяния, на которой будут выделены точки, которые для разных методов попали в разные кластеры.



Отличие пакетной от обычной кластеризации в том, что первая быстрее, так как при расчетах оперирует случайной группой (batch) значений, а не каждой отдельно взятой точкой.

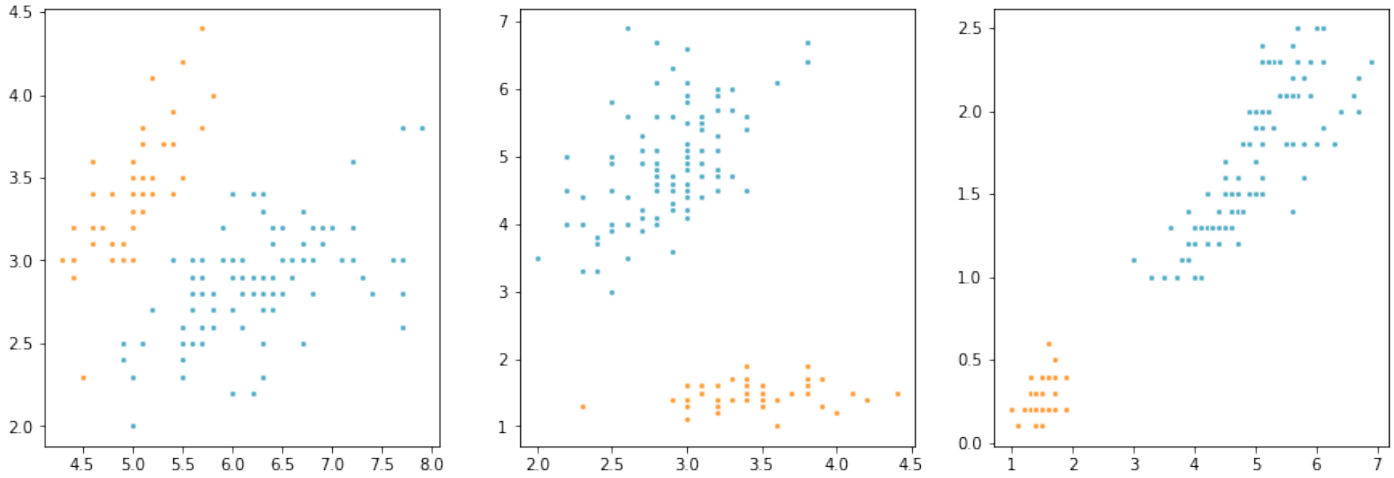
Иерархическая кластеризация

Проведем иерархическую кластеризацию на тех же данных. Отобразим результаты кластеризации

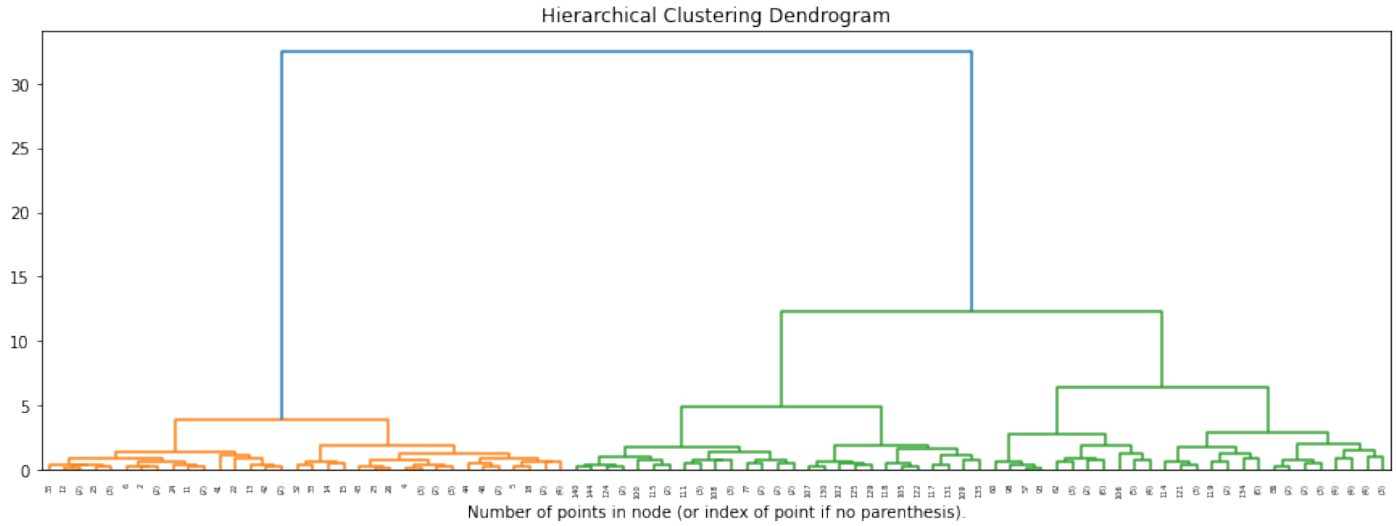


В отличие от метода k-средних иерархическая кластеризация оперирует т.н. "Деревом кластеров", где корень - это все данные, а листья - единичные измерения. Агломеративная кластеризация строит дерево слиянием более маленьких кластеров, используя заданную стратегию - ward, average и др.

Проведём исследование для различного размера кластеров (от 2 до 5).

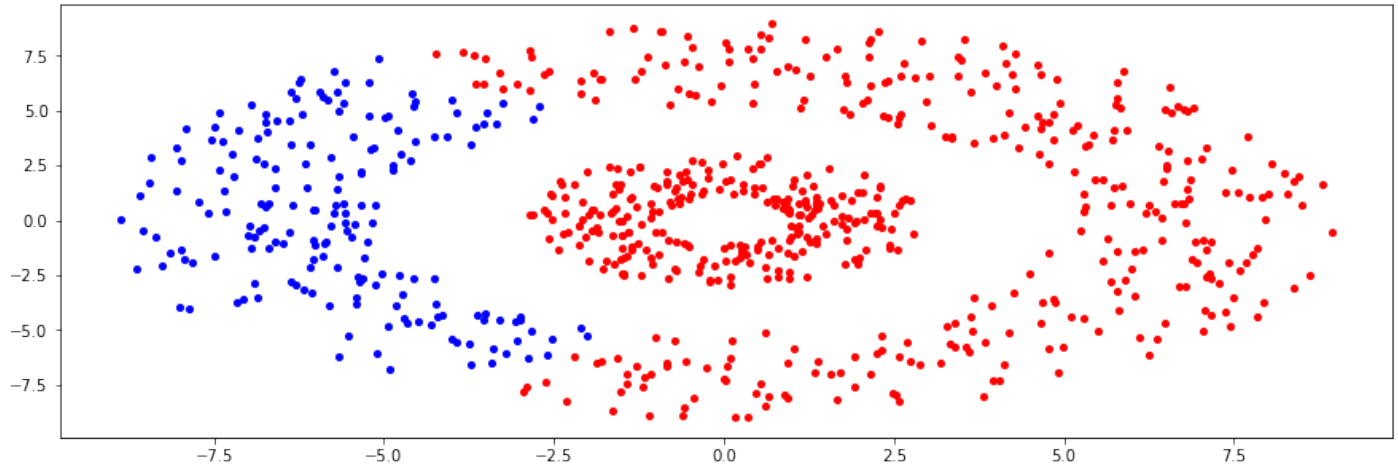


Нарисуйте дендограмму до уровня 6.



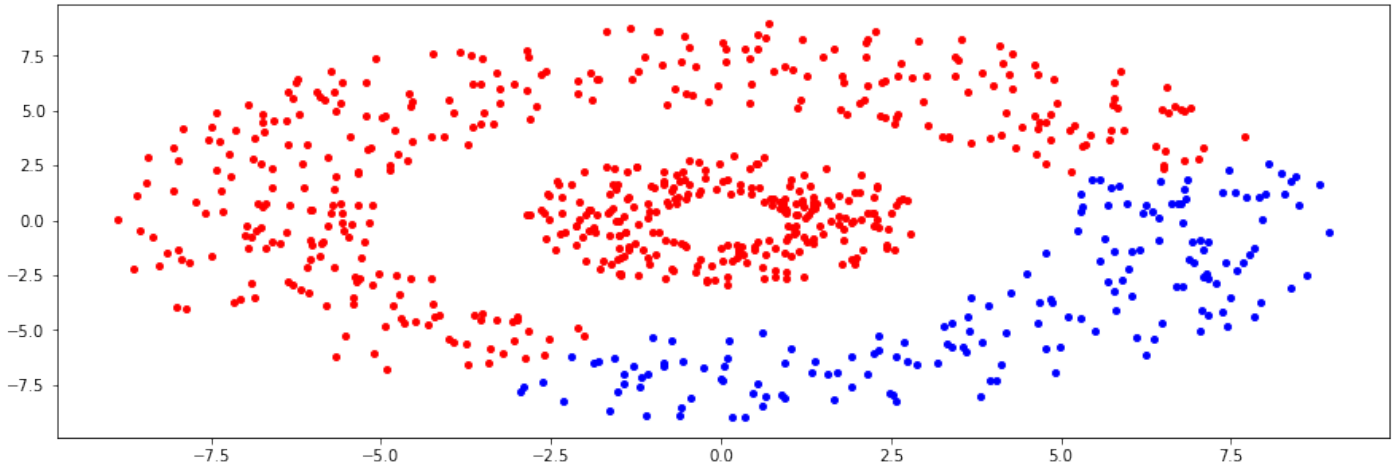
Сгенерируем случайные данные в виде двух колец. Проведём иерархическую кластеризацию. Выведите полученные результаты.

Ward:

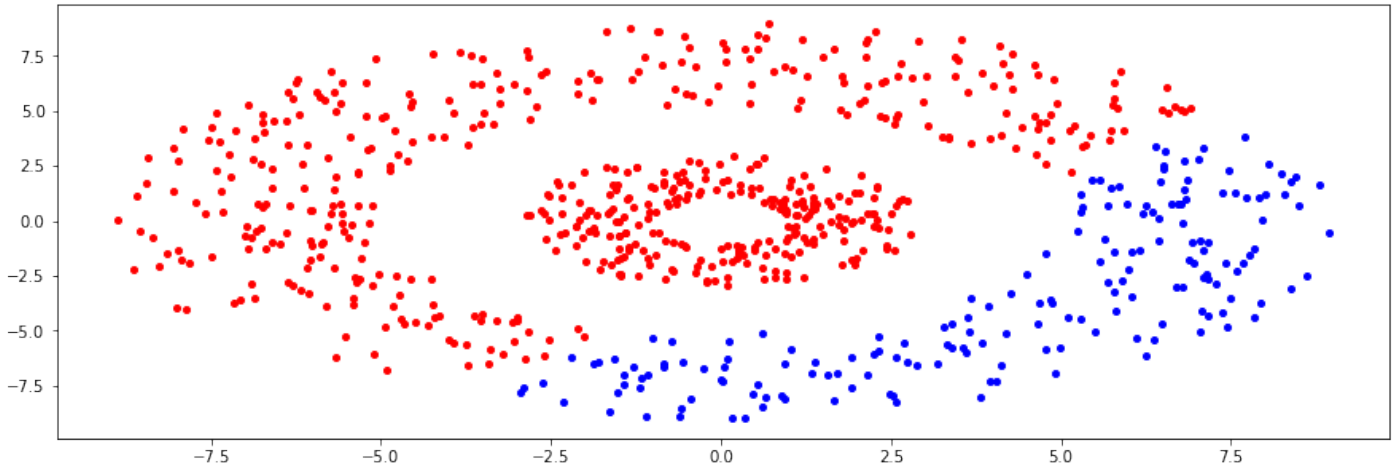


Исследуем кластеризацию при всех параметрах linkage. Отобразим и обоснуем полученные результаты.

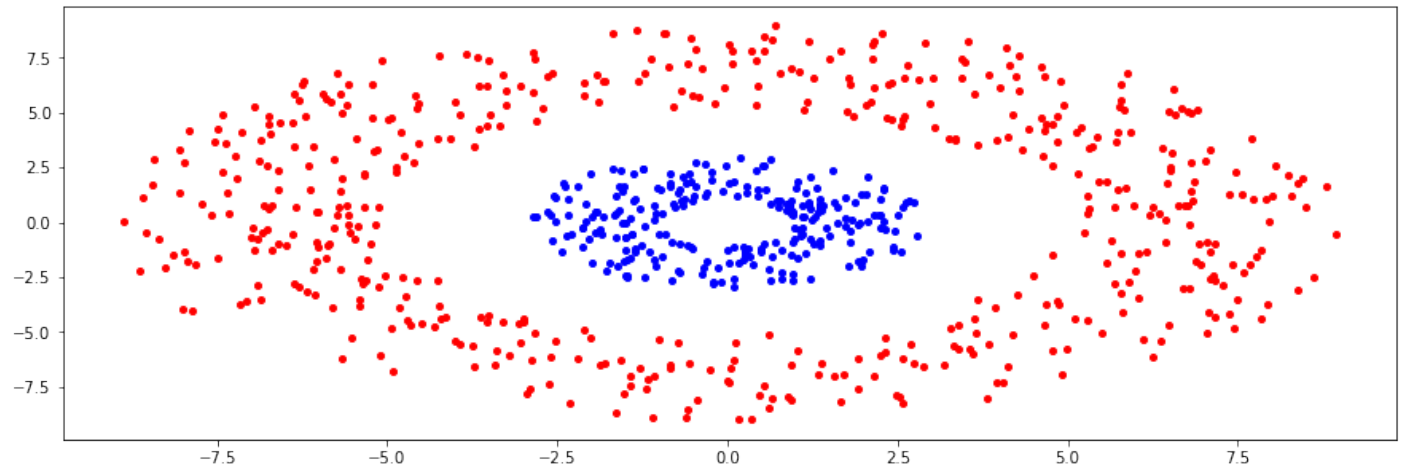
Complete:



Average:



Single:



Как видно, для двух колец больше подошел метод линковки 'single'.

- ward минимизирует дисперсию двух сливаемых кластеров
- average использует среднее расстояний каждого наблюдения в двух кластерах
- complete испольует максимум расстояний между всеми наблюдениями в двух кластерах
- single испольует минимум расстояний между всеми наблюдениями в двух кластерах

Single хорошо подходит для кластеров не шарообразной формы, а также для больших датасетов (так как работает быстро).

Ward лучше использовать для кластеров более-менее одинаковых размеров.