Лабораторная работа №5 Кластеризация (k-средних, иерархическая)

Цель:

Ознакомиться с методами кластеризации модуля Sklearn

Теоретические сведения

- Кластерный анализ | Вводный курс
- Кластеризуем лучше, чем «метод локтя» / Хабр
- Метод локтя (Elbow method)
- Mini Batch K-Means

Выполнение:

1. Загрузка данных:

1.1. Загрузить датасет по ссылке: https://archive.ics.uci.edu/dataset/109/wine.

- Данные представлены в виде data файла. Данные представляют собой информацию о **трех** классах вина.

1.2. Создать Python скрипт. Загрузить данные в датафрейм

1.3. При необходимости, произвести стандартизацию данных с использованием preprocessing.StandardScaler() из `sklearn`

1.4. Подготовить данные для дальнейшего анализа:

- Понизить размерность пространства данных до размерности `n`, при которой компоненты объясняют не менее 85% дисперсии данных. Понижение размерности пространства осуществляется с помощью метода главных компонент (PCA.fit_transform)
- Восстановить данные для полученного количества компонент (PCA.inverse_transform).

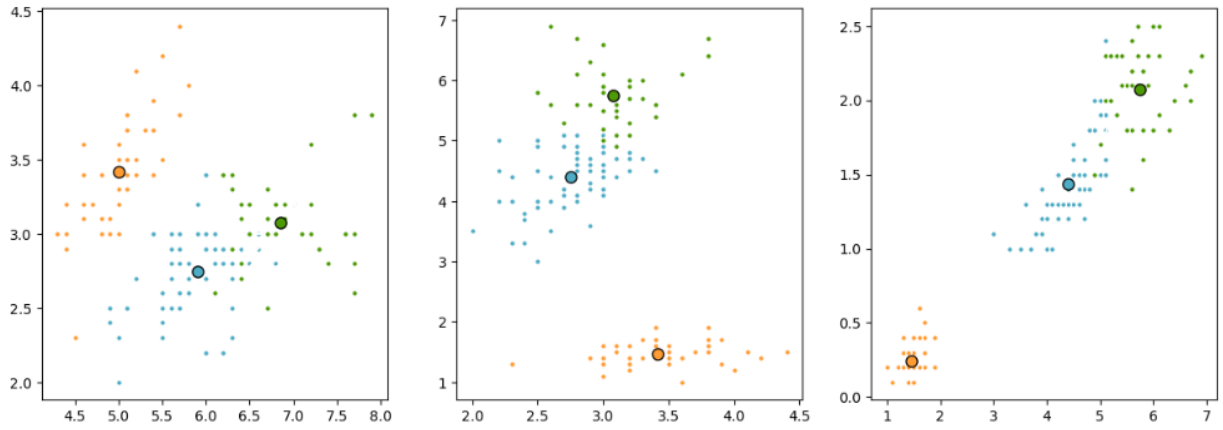
2. K-Mean

2.1. Провести кластеризацию методом k-средних. Вывести на экран значения центров кластеров и количество наблюдений, которое попало в каждый кластер

- Использовать: `KMeans` из `sklearn.cluster`
- Справочные ссылки:
  - KMeans()
  - KMeans.fit()
  - KMean.cluster_centers
  - pairwise_distances_argmin()

2.2. Построить (графически) результаты классификации для признаков попарно (1 и 2, 2 и 3, ..., `n`-1 и `n`), отобразить центры кластеров

- Пример: Comparison of the K-Means and MiniBatchKMeans clustering algorithms — scikit-learn 1.5.2 documentation

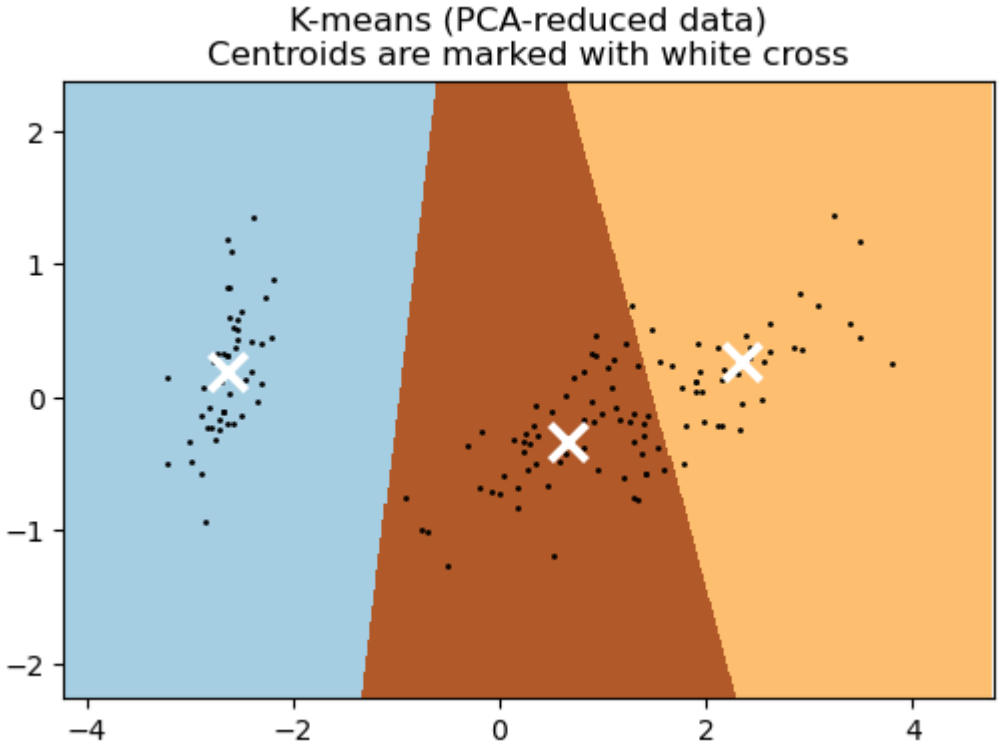


- Требуется: `matplotlib.pyplot`
- Дать пояснения:
  1. На что влияет параметр `n_init`?
  2. Чему равно оптимальное значение для `n_init`? Почему это оптимальное значение?
  3. Что происходит при увеличении и уменьшении параметра `n_init` от оптимального?
  4. По каким признакам произошло наилучшее разделение?
  5. Как изменятся результаты, если в качестве метода инициализации выбрать `random`?

2.3. Уменьшить размерность данных до `n=2` используя метод главных компонент и нарисовать карту для всей области значений, на которой

каждый кластер занимает определенную область со своим цветом

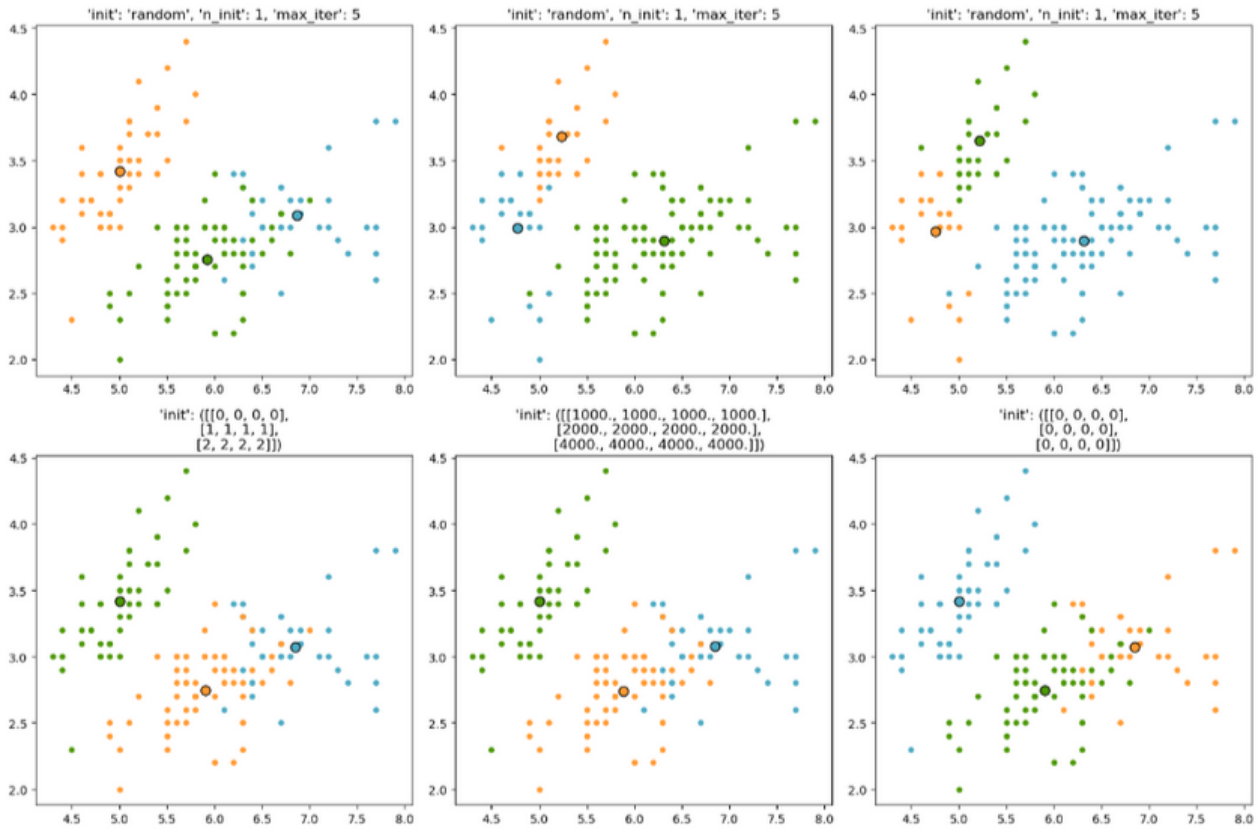
- Пример: [A demo of K-Means clustering on the handwritten digits data — scikit-learn 1.5.2 documentation]



2.4. Исследуйте работу алгоритма k-средних при различных параметрах init.

> Сначала надо выполнить два раза с параметром `random`, затем выполнить для вручную выбранных точек

- Пример результата:



- Может понадобиться:
  - random_state
  - параметр `init` в KMeans
  - параметр `max_iter` в KMeans
- Дать пояснения:
  1. Как повлиял выбор параметра `random` на результат кластеризации?
  2. Какой из вариантов оказался самым удачным и почему?
  3. Влияет ли параметр `max_iter` на результат кластеризации?

2.5. Определите наилучшее количество кластеров методом локтя

- Дать пояснения:
  1. Что означает результат WCSS?
  2. Почему найденное количество кластеров является наилучшим?
- Пример простейшего использования "Метода локтя"

Метод "Локтя"

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

for k in range(1, 11):
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(data)
    sse.append(kmeans.inertia_)
```

```
plt.figure(figsize=(10, 6))
plt.plot(range(1, 11), sse, marker='o')
plt.title('Метод "Локтя"')
plt.xlabel('Количество кластеров')
plt.ylabel('Сумма квадратов расстояний')
plt.show()
```

2.6. Проведите кластеризацию используя пакетную кластеризацию k-средних.

> Постройте диаграмму рассеяния, на которой будут выделено точки, которые для разных методов попали в разные кластеры

- Дать пояснения:
  1. В чем отличие результата пакетной кластеризации k-средних от обычного метода k-средних?
  2. Чем отличаются построенное графическое представление?

3. Иерархическая кластеризация

3.1. Провести и отобразить иерархическую кластеризацию на тех же данных (см. п.1.3. и п.2.2) с параметром `average`

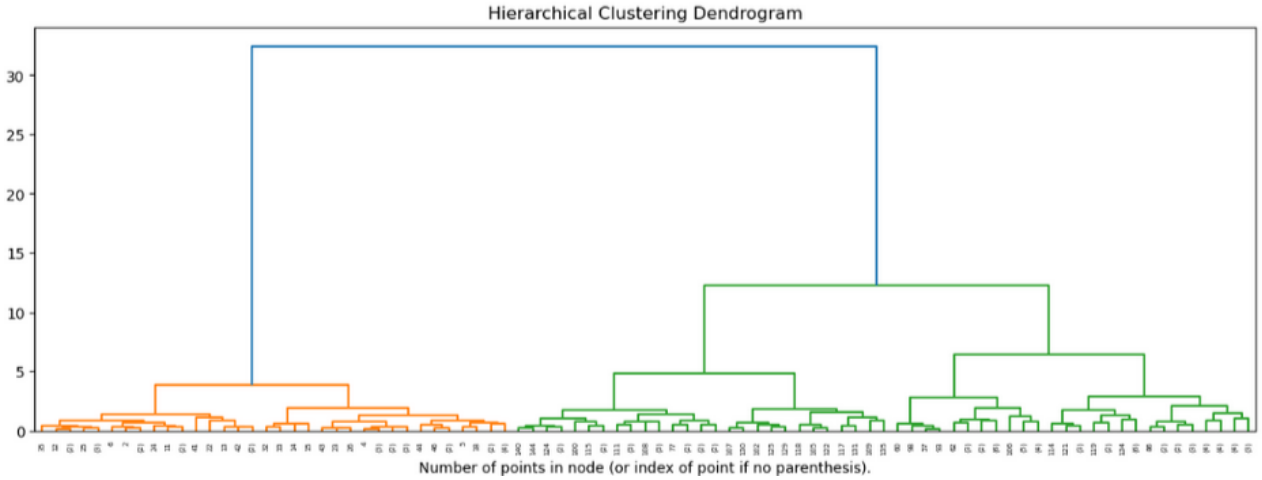
- Использовать: `AgglomerativeClustering` из `sklearn.cluster`
- Справочные ссылки:
  - AgglomerativeClustering
  - AgglomerativeClustering.fit()
  - Comparing different hierarchical linkage methods on toy datasets
- Дать пояснения:
  1. Чем отличаются результаты (графическое изображение), полученные по методу `KMeans` и `AgglomerativeClustering`?
  2. Кокай из методов дал более точные результаты для заданных исходных данных и почему?

3.2. Проведите исследование для различного размера кластеров (от 2 до 5). Приведите полученные результаты

- Использовать параметр `n_clusters` из AgglomerativeClustering
- Дать пояснения:
  1. Какой из значений параметра `n_clusters` дал наилучшие результаты и почему?

3.3. Постройте дендограмму до уровня 6

- Пример:



3.4. Сгенерируйте случайные данные (x,y) в виде двух квадратных контуров

- Замечание
  - Общее количество точек 750: внешний квадратный контур - 500; внутренний квадратный контур - 250.
  - Проведите их иерархическую кластеризацию со всеми возможными параметрами `linkage`.
  - Отобразите полученные результаты.
- Дать пояснения:
  1. Какой тип связи работает лучше всего и почему?
