Лабораторная работа №5

Кластеризация (k-средних, иерархическая)

Цель:

Ознакомиться с методами кластеризации модуля Sklearn

Выполнение:

Загрузка данных:

1. Загрузить датасет по ссылке: https://archive.ics.uci.edu/ml/datasets/iris .
   Данные представлены в виде data файла. Данные представляют собой информацию о трех классах цветов
2. Создать Python скрипт. Загрузить данные в датафрейм

```
import pandas as pd
import numpy as np

data = pd.read_csv('iris.data',header=None)
```

K-means

1. Проведем кластеризацию методов k-средних

```
from sklearn.cluster import KMeans

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

2. Получим центры кластеров и определим какие наблюдения в какой кластер попали

```
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)
```

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

```
import matplotlib.pyplot as plt

f, ax = plt.subplots(1, 3)


colors = ['#4EACC5', '#FF9C34', '#4E9A06']
```

```
print(ax)

for i in range(3):
    my_members = k_means_labels == i
    cluster_center = k_means_cluster_centers[i]
    for j in range(3):
        ax[j].plot(no_labeled_data[my_members, j],
no_labeled_data[my_members, j+1], 'w',
                markerfacecolor=colors[i], marker='o', markersize=4)
        ax[j].plot(cluster_center[j], cluster_center[j+1], 'o',
markerfacecolor=colors[i],
                markeredgecolor='k', markersize=8)

plt.show()
```

Опишите полученные результаты. По каким из признаков произошло наилучшее разделение. Как влияет значение параметра n_init

4. Уменьшите размерность данных до 2 используя метод главных компонент и нарисуйте карту для всей области значений, на которой каждый кластер занимает определенную область со своим цветом (как это делать)
5. Исследуйте работу алгоритма k-средних при различных параметрах init. Сначала надо выполнить несколько раз с параметров 'random', затем для вручную выбранных точек
6. Определите наилучшее количество методом локтя
7. Проведите кластеризацию используя пакетную кластеризацию k-средних . В чем отличие от обычного метода k-средних. Постройте диаграмму рассеяния, на которой будут выделено точки, которые для разных методов попали в разные кластеры

Иерархическая кластеризация

1. Проведем иерархическую кластеризацию на тех же данных

```
from sklearn.cluster import AgglomerativeClustering

hier = AgglomerativeClustering(n_clusters=3, linkage='average')
hier = hier.fit(no_labeled_data)
hier_labels = hier.labels_
```

2. Отобразим результаты кластеризации

```
f, ax = plt.subplots(1, 3)

colors = ['#4EACC5', '#FF9C34', '#4E9A06']


for i in range(3):
    my_members = hier_labels == i
    for j in range(3):
        ax[j].plot(no_labeled_data[my_members, j],
no_labeled_data[my_members, j+1], 'w',
                markerfacecolor=colors[i], marker='o', markersize=4)

plt.show()
```

В чем отличия от метода k-средних

3. Проведите исследование для различного размера кластеров (от 2 до 5). Приведите полученные результаты
4. Нарисуйте дендограмму до уровня 6
5. Сгенерируйте случайные данные в виде двух колец

```
import random
import math
data1 = np.zeros([250,2])
for i in range(250):
    r = random.uniform(1, 3)
    a = random.uniform(0, 2 * math.pi)
    data1[i,0] = r * math.sin(a)
    data1[i,1] = r * math.cos(a)

data2 = np.zeros([500,2])
for i in range(500):
    r = random.uniform(5, 9)
    a = random.uniform(0, 2 * math.pi)
    data2[i,0] = r * math.sin(a)
    data2[i,1] = r * math.cos(a)

data = np.vstack((data1, data2))
```

6. Проведите иерархическую кластеризацию

```
hier = AgglomerativeClustering(n_clusters=2, linkage='ward')
hier = hier.fit(data)
hier_labels = hier.labels_
```

7. Выведите полученные результаты

```
my_members = hier_labels == 0
plt.plot(data[my_members, 0], data[my_members, 1], 'w', marker='o',
markersize=4,
         color='red',linestyle='None')
my_members = hier_labels == 1
plt.plot(data[my_members, 0], data[my_members, 1], 'w', marker='o',
markersize=4,
         color='blue',linestyle='None')
plt.show()
```

8. Исследуйте кластеризацию при всех параметрах linkage. Отобразите и обоснуйте полученные результаты. Для каких случаев, какой тип связи работает лучше всего.

#####