Групповая задача 3 - Кластеризация с объяснением.

1. Постановка задачи:

Имеются данные о смертях от COVID-19 и по другим причинам для всех округов США за период 2.01.2020-8.01.2020, сохраненные в csv-файле. Требуется оценить уровень угрозы COVID-19 в каждом штате.

2. Используемые данные:

Датасет указанных свойств из свободного доступа (сайт kaggle.com) в формате csv. Имеющиеся для каждого округа данные: штат, в который он входит; количество смертей от COVID-19 за указанный период; количество смертей в целом за указанный период.

Датасет с населением США по штату за 2018 год.

3. Важнейшие спользуемые методы:
   - Метод dendrogram библиотеки scipy.cluster.hierarchy для визуального определения оптимального количества кластеров
   - Метод k-means библиотеки sklearn.cluster для нахождения центроидов кластеров
4. Ход решения:

Импорт используемых библиотек:

```
import csv
import statistics
import math
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage
```

Проход по таблице со сбором данных о смертях в каждом округе, создание списка штатов для будущего использования:

```
Covid_death = []
Death = []
State = []

with open('AH_County-level_Provisional_COVID-19_Deaths_Counts.csv') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        Death.append(int(row['Total Deaths']))
        Covid_death.append(int(row['COVID-19 Deaths']))
        State.append(row['State'])
additional_Statenames = []
additional_Population = []
LandAreas = []

with open('State Populations.csv') as csvfile:
    reader = csv.DictReader(csvfile)


    for row in reader:
        additional_Statenames.append(row['State'])
        additional_Population.append(row['2018 Population'])
States = list(set(State))
```

```
state_to_num = {States[i]: i for i in range(len(States))}
Covid_deaths = [0]*len(States)
Deaths = [0]*len(States)
for i in range(len(State)):
    j = state_to_num[State[i]]  # index of seen State in actual states
    Covid_deaths[j] += Covid_death[i]
    Deaths[j] += Death[i]
```

Сортировка массивов с данными для возможности использования общего индекса:

```
Deaths = [k for _, k in sorted(zip(States, Deaths), key=lambda pair: pair[0])]
Covid_deaths = [k for _, k in sorted(zip(States, Covid_deaths), key=lambda pair:
pair[0])]
additional_Population = [k for _, k in sorted(zip(additional_Statenames,
additional_Population), key=lambda pair: pair[0])]
States.sort()
```

Обобщение полученных данных для каждого штата:

```
total = []
for i in range(len(Deaths)):
    population = float(additional_Population[i]) // 10000
    print(States[i], Deaths[i], Covid_deaths[i], population)
    total.append([Deaths[i]/population, Covid_deaths[i]/population])
```

Процесс кластеризации по методу k-means для 3 кластеров:

```
k = 1
first = []
second = []
third = []
while k != 0:
    first = []
    second = []
    third = []
    k = 0
    kmeans = KMeans(n_clusters=3)
    kmeans.fit(total)
    labels = kmeans.predict(total)
    centroids = kmeans.cluster_centers_
    for point in total:
        min_dist = float('inf')
        dists = []
        for centroid in centroids:
            dist = math.sqrt(((point[0] - centroid[0])**2) + ((point[1] -
centroid[1])**2))
            dists.append(dist)
            if dist < min_dist:
                min_dist = dist
        p = 0
        for dist in dists:
            if dist == min_dist:
                p += 1
                if p > 1:
                    print(point, '- точка касания')
```

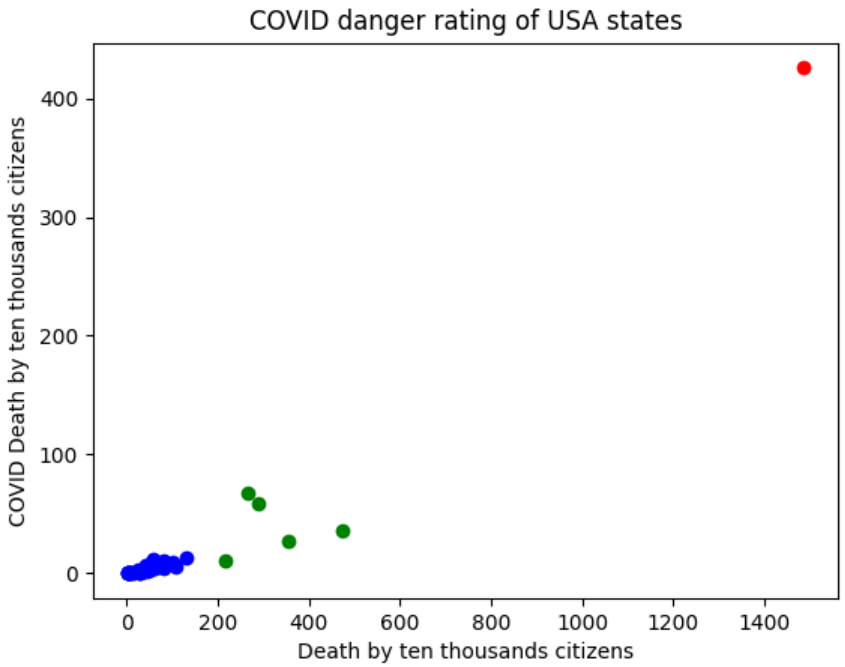
```
                k = 1
        if dists[0] == min_dist:
            first.append(point)
        elif dists[1] == min_dist:
            second.append(point)
        else:
            third.append(point)
```

Визуализация полученных кластеров:

```
classes = [first, second, third]
i = 0
color = ['blue', 'red', 'green']
for Class in classes:
    for point in Class:
        scater = plt.scatter(point[0], point[1], c=color[i])
    i += 1
plt.title('COVID danger rating of USA states')
plt.xlabel('Death by ten thousands citizens')
plt.ylabel('COVID Death by ten thousands citizens')
plt.show()
```

5. Анализ полученных результатов:

Рисунок 1: распределение штатов США по уровню опасности.




Получено распределение штатов по трём группам опасности:

1. “Синяя” - штаты с низким уровнем смертности на душу населения вообще и низким уровнем ковидной смертности в частности. Угроза мала.
2. “Синяя” - штаты с более высоким уровнем обоих видов смертности. Уровень угрозы выше, но незначительно.
3. “Красная” - Нью-Йорк. Самый высокий уровень угрозы.

Можно заметить, что смертность от COVID на душу населения для большинства штатов практически линейно зависит от смертности обычной, что, скорее всего, объясняется одними и теми же влияющими факторами (например, уровень качества медицинского обслуживания населения).