Programming Assignment #3

2017029916
양동해

1. 알고리즘 요약

input dataset을 DBSCAN 알고리즘으로 돌려, cluster들을 찾아내는 프로그램이다. 코드는 총 5가지 파트로 구분되며 각 파트당 알고리즘은 다음과 같다.

- Point 클래스
- DBSCAN 클래스 (및 주요 함수)
  - eps_neighborhood()
  - clustering()
- 파일 읽기
- 데이터 전처리 및 DBSCAN 실행
- 파일 쓰기

2. 주요 코드 상세

코드를 위에서 언급한 5가지 파트로 나누어, 각 파트가 무엇을 하는지 기술하였다.

[Point 클래스]

```python
# Point 클래스
class Point:
  def __init__(self, object_id, x, y):
    self.object_id = int(object_id)
    self.x = float(x)
    self.y = float(y)
    self.cluster_id = None

  def dist(self, p):
    return math.sqrt((self.x-p.x)**2 + (self.y-p.y)**2)

  def to_print(self):
    print(self.object_id, self.x, self.y, self.cluster_id)
```

Point 클래스는 input 파일에서 읽어드려 DBSCAN 클래스에서 사용될 객체(점)들의 구조를 갖는 클래스이다. 변수는 object_id, x, y, cluster_id를 가진다. object_id, x, y는 input 파일에 해당하는 데이터와 대응되며, cluster_id는 후에 클러스터링이 진행되면 몇 번째 클러스터에 해당하는지 나타내는 cluster id 이다. cluster_id가 None이면 아직 클러스터링이 진행되지 않은 것이고, cluster_id가 -1이면 outlier라는 뜻이다. 나머지 0 이상의 값은 cluster id에 해당한다. dist() 함수는 현재 점(Point 객체)과 인자로 받은 다른 점(Point 객체)과의 거리를 구하는 함수이다.

[DecisionTree 클래스]

- eps_neighborhood()

```python
  def eps_neighborhood(self, p):
    return [q for q in self.dataset if p.dist(q) <= self.eps]
```

eps_neighborhood() 함수는 인자로 받은 점(Point 객체)을 중심으로 eps 값 이하의 반경에 포함되는 점들을 구하는 함수이다. 즉, 점 p의 neighbor들을 구하는 함수이다.

- clustering()

```
def clustering(self):
  cluster_id = 0
  for p in self.dataset:
    if p.cluster_id is not None: continue # 이미 clustering 된 경우

    neighbors = self.eps_neighborhood(p)
    if len(neighbors) >= self.minPts: # p가 core point인 경우
      p.cluster_id = cluster_id
      for q in neighbors:
        if q.cluster_id is None: # clustering 안된 경우
          q.cluster_id = cluster_id
          q_neighbors = self.eps_neighborhood(q)
          if len(q_neighbors) >= self.minPts: neighbors.extend(q_neighbors)
        elif q.cluster_id == -1: # q가 border point인 경우
          q.cluster_id = cluster_id # outlier 아님
      cluster_id += 1
    else: # p가 core point가 아닌 경우
      p.cluster_id = -1 # outlier 후보로 세팅

  clusters = [[] for i in range(cluster_id)]
  for p in self.dataset:
    # p.to_print()
    if p.cluster_id != -1: clusters[p.cluster_id].append(p)

  return clusters
```

clustering() 함수는 주어진 dataset에 대해 클러스터링을 진행하는 함수이다. 우선, 클러스터링이 진행되지 않은 점(Point 객체)들에 대해서만 클러스터링을 진행하며, 그 점(Point)의 neighbor들을 모두 구한다. 그리고 neighbor들의 수가 minPts 이상이면, 그 점(Point)을 core point로 정하고 그 neighbor들의 neighbor들에 대해서도 density-reachable 한지를 파악한다. 그렇게 chain 형태로 neighbors를 늘려 density-connected 관계를 파악하고, 이 과정을 border point가 나올 때까지 반복한다. 만약 해당 점(Point)이 core point가 아니라면 outlier의 후보로 둬, 후에 그 점(Point)이 다른 클러스터로도 포함되지 않는다면 outlier가 된다. 그리고 이렇게 파악된 점들의 cluster_id별로 클러스터들을 생성해서 반환한다.

[파일 읽기]

```
# 파일 읽기
input_filename = sys.argv[1]
n = int(sys.argv[2])
eps = float(sys.argv[3])
minPts = float(sys.argv[4])

dataset_header = ["object_id", "x_coordinate", "y_coordinate"]
input_dataset = pd.read_csv(input_filename, sep="\t", names=dataset_header)
```

이 파트는 인자로 입력 받은 파일과 파라미터들을 읽는 부분이다.

[데이터 전처리 및 DBSCAN 실행]

```
# 데이터 전처리
dataset = [Point(obj[0], obj[1], obj[2]) for obj in input_dataset.values.tolist()]
# for p in dataset: p.to_print() # 출력

# DBSCAN 클래스 생성 및 clustering
dbscan = DBSCAN(dataset, eps, minPts)
clusters = dbscan.clustering()
```

이 파트는 인자로 입력 받은 데이터를 Point 객체로 만들어 dataset에 저장한 뒤, 이를 가지고 DBSCAN 알고리즘을 이용해 클러스터링을 진행하는 부분이다.

[파일 쓰기]

```
# 파일 쓰기
clusters.sort(key=len, reverse=True)
clusters = clusters[:n] # n개 만큼 선택

cluster_idx = 0
for cluster in clusters:
  with open(input_filename.split(".")[0] + "_cluster_" + str(cluster_idx) + ".txt", "w") as f:
    for p in cluster:
      f.write(str(p.object_id) + "\n")
  cluster_idx += 1
```

이 파트는 많은 data를 가진 클러스터부터 순차적으로 n개 선택하여, 파일로 저장하는 부분이다.

3. 컴파일 및 실행

- Python version: 3.8.9 + (numpy (1.22.3), pandas (1.4.2) library 사용)
- 프로그램 실행: python3 clustering.py [input data file name] [n] [Eps] [MinPts]

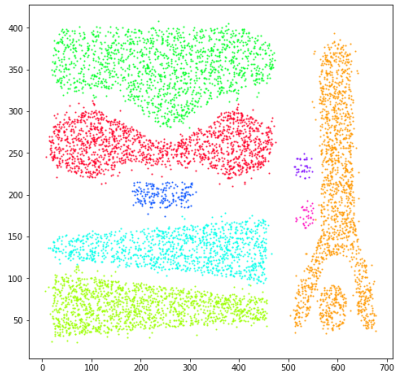
```
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % python3 --version
Python 3.8.9
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % ls
PA3.exe                         input1_cluster_3_ideal.txt      input2_cluster_0_ideal.txt      input3_cluster_0_ideal.txt
clustering.py                   input1_cluster_4_ideal.txt      input2_cluster_1_ideal.txt      input3_cluster_1_ideal.txt
input1.txt                      input1_cluster_5_ideal.txt      input2_cluster_2_ideal.txt      input3_cluster_2_ideal.txt
input1_cluster_0_ideal.txt      input1_cluster_6_ideal.txt      input2_cluster_3_ideal.txt      input3_cluster_3_ideal.txt
input1_cluster_1_ideal.txt      input1_cluster_7_ideal.txt      input2_cluster_4_ideal.txt
input1_cluster_2_ideal.txt      input2.txt                      input3.txt
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % python3 clustering.py input1.txt 8 15 22
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % ls
PA3.exe                         input1_cluster_2_ideal.txt      input1_cluster_6_ideal.txt      input2_cluster_4_ideal.txt
clustering.py                   input1_cluster_3.txt            input1_cluster_7.txt            input3.txt
input1.txt                      input1_cluster_3_ideal.txt      input1_cluster_7_ideal.txt      input3_cluster_0_ideal.txt
input1_cluster_0.txt            input1_cluster_4.txt            input2.txt                      input3_cluster_1_ideal.txt
input1_cluster_0_ideal.txt      input1_cluster_4_ideal.txt      input2_cluster_0_ideal.txt      input3_cluster_2_ideal.txt
input1_cluster_1.txt            input1_cluster_5.txt            input2_cluster_1_ideal.txt      input3_cluster_3_ideal.txt
input1_cluster_1_ideal.txt      input1_cluster_5_ideal.txt      input2_cluster_2_ideal.txt
input1_cluster_2.txt            input1_cluster_6.txt            input2_cluster_3_ideal.txt
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % mono PA3.exe input1
98.97037점%
eastsea@EastSeaui-MacBookPro Programming_Assignment3 %
```

- 각 파일별 score 측정

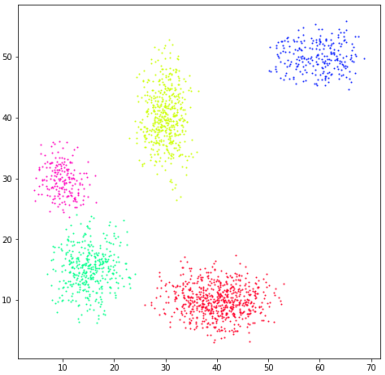
```
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % mono PA3.exe input1
98.97037점%
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % mono PA3.exe input2
94.86598점%
eastsea@EastSeaui-MacBookPro Programming_Assignment3 % mono PA3.exe input3
99.97736점%
eastsea@EastSeaui-MacBookPro Programming_Assignment3 %
```

	input1.txt	input2.txt	input3.txt
실행 시간	40~50초	10초 이내	10초 이내
Score	98.97037점	94.86598점	99.97736점

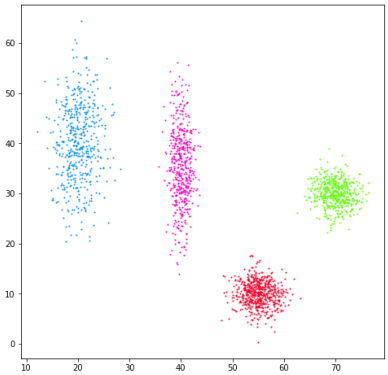
- 각 파일별 클러스터링 결과 (outlier 제외)



input1



input2



input3