Data Science Assignment3 : DBSCAN

2016025078 강덕영

- DBSCAN

data 간 모여있는 정도 혹은 밀도를 기준으로 데이터를 군집화하는 알고리즘입니다.

- Source code

assignment3 directory structure:

2022_ITE4005_2016025078/assignment3/----input-data/input1.txt,,,

----output-data/input1_cluster_0.txt,,,

----self-test/PA3.exe,,,

clustering.py

report.py

- Summary of algorithm

먼저 데이터 파일에서 각 점들의 2차원 좌표 정보를 불러옵니다. 우선적으로 모든 두 점 사이의 거리를 계산하여 저장하고, 또한 코어인 점도 미리 구해서 저장합니다. 이 두 가지 작업이 끝나면 DBSCAN 알고리즘이 시작됩니다. 어떤 점이 아직 cluster 에 속해 있지 않고 코어면, 해당 점으로부터 density-reachable 한 점들을 모두 찾아 cluster 에 포함시킵니다. 그리고 이 과정을 모든 점에 대해 반복하면 군집화가 종료됩니다.

- Functions

1. load_data(file_name): 입력 파일을 읽어 점들에 대한 정보를 리턴하는 함수입니다.

2. calculate_distance(point1, point2): 두 점간의 거리를 리턴하는 함수입니다.

3. get_distance_dictionary(data): 점 데이터를 이용하여 모든 두 점들간의 거리를 구한 후, 거리를 기준으로 오름차순 정렬합니다. 모든 점에 대한 거리 정보를 담고있는 distance_info 딕셔너리를 리턴합니다. (distance_info[2][3] 은 2번째 점과 3번째 점의 거리를 나타냅니다.)

4. identify_core(distance_info): 특정 점이 코어인지 아닌지 체크하고 해당 정보를 리턴하는 함수입니다. 앞서 get_distance_dictionary 함수를 통해 얻은 점들 간의 거리 정보들을 바탕으로 특정 점이 코어인지 아닌지 빨리 확인할 수 있습니다.

5. get_density_reachables(start_point, distance_info, is_core): 특정 점에서부터 density-reachable 한 모든 점들을 구하는 함수입니다. DFS 와 유사한 방식으로 같은 cluster 에 속할 점들을 찾습니다.

- Instruction

운영체제: macOS

코드 실행 가상환경: anaconda

코드 실행 방법:

python clustering.py input1.txt 8 15 22

python clustering.py input2.txt 5 2 7

python clustering.py input3.txt 4 5 5

위 세 가지 명령어를 통해 DBSCAN 알고리즘을 실행할 수 있으며,

결과 파일은 2022_ITE4005_2016025078/assignment3/output-file/ 경로에 저장됩니다.

(input1.txt 을 입력 파일로 사용하는 경우에는 약 1분 가량의 수행 시간이 소요됩니다.)

테스트 프로그램 실행방법:

2022_ITE4005_2016025078/assignment3/output-file/ 경로에 있는 결과 파일들을,

2022_ITE4005_2016025078/assignment3/self-test/ 경로에 복사한 후, 동일 경로에서

mono PA3 input1, mono PA3 input2, mono PA3 input3

위 세가지 명령어를 실행함으로써 군집화 결과 점수를 확인할 수 있습니다.

(결과 파일들을 self-test 폴더에 옮겨놓고 gitlab 에 push 했기 때문에, 바로 테스트 프로그램을 실행할 수 있습니다.)