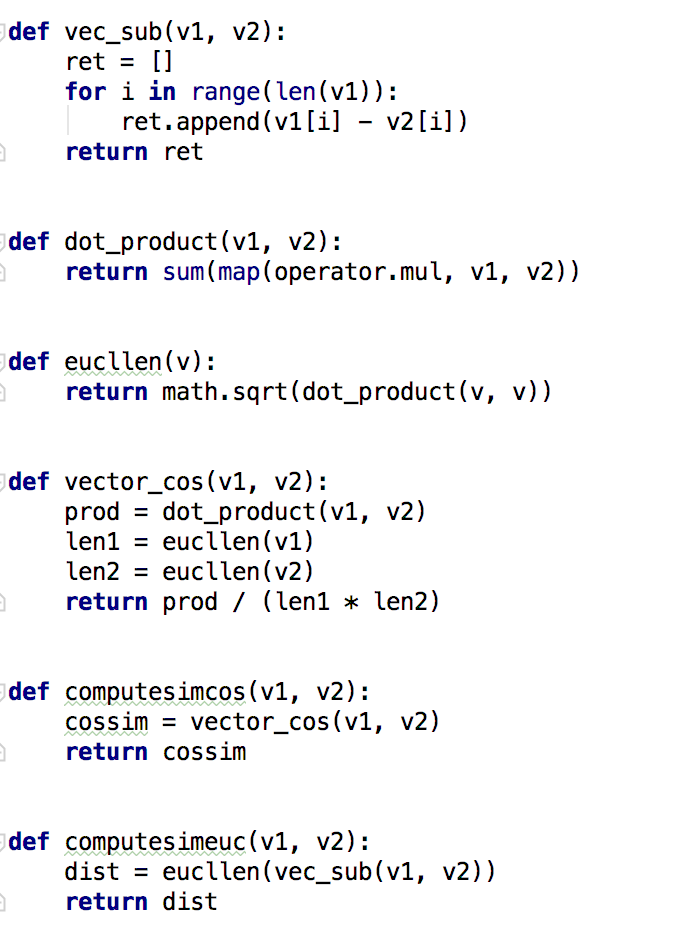
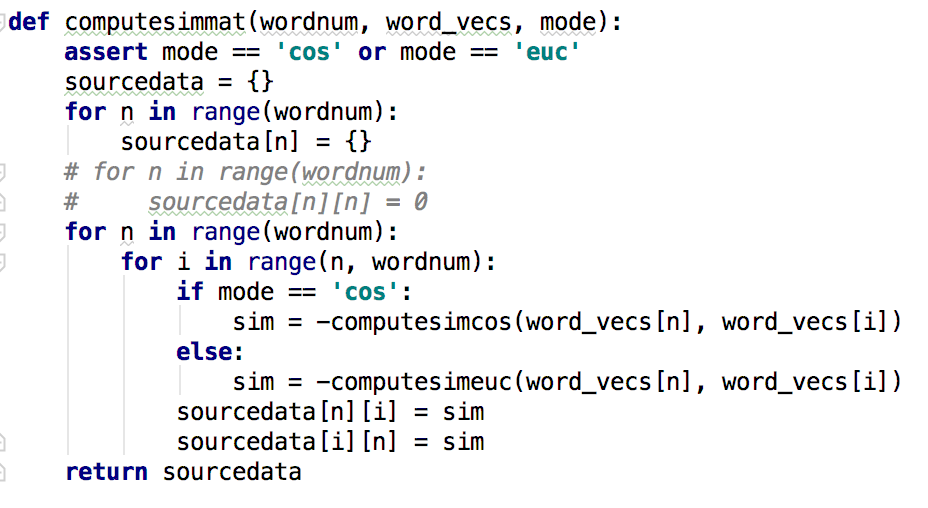
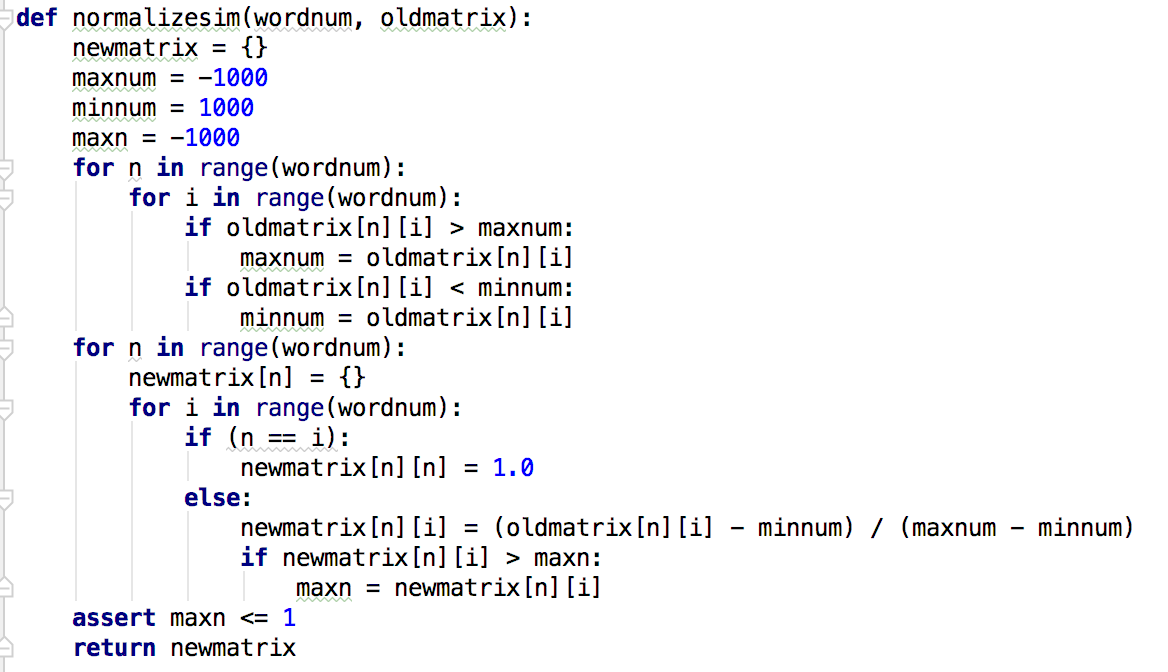
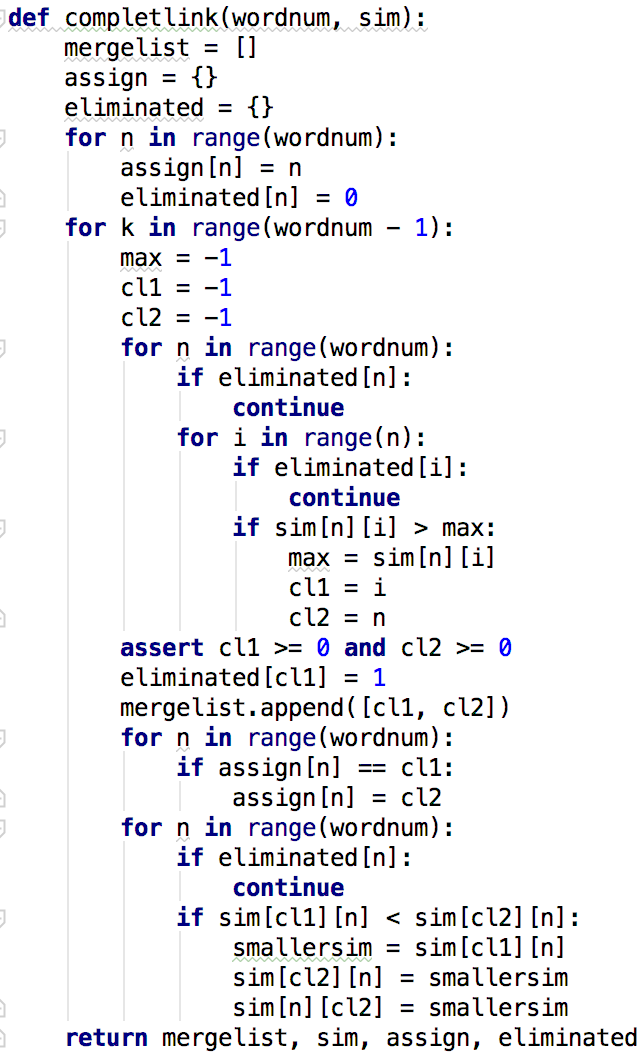
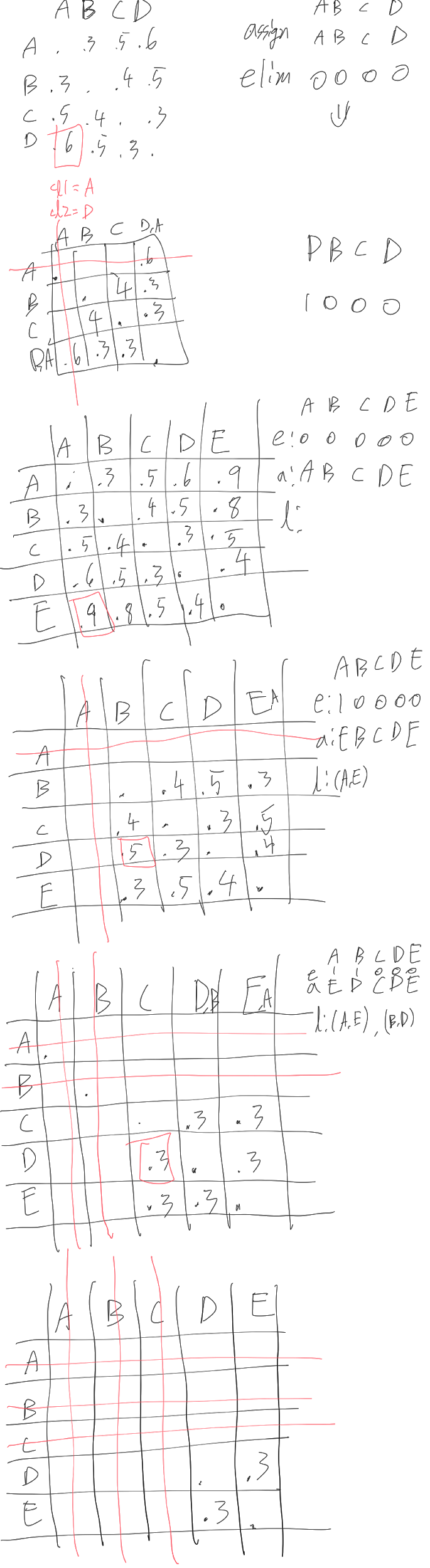
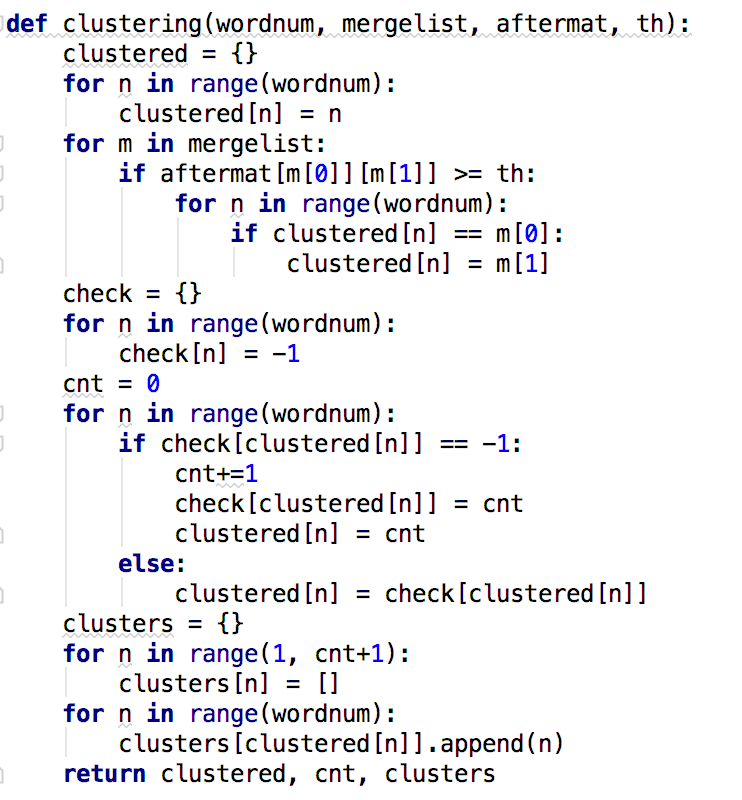
**AI Assignment #2 \_ Search Algorithms**

2014004975 이윤성

1. 컴파일 방법 및 사용 버전
   1. 컴파일 방법
      1. 
         1. sys.argv[1] 에는 similarity 의 설정값을 입력한다
         2. sys.argv[2] 에는 threshold의 설정값을 입력한다
   2. 사용 버전
      1. OS
         1. macOS High Sierra 10.13.4
      2. python3
         1. Python 3.6.2
2. 코드 설명
   1. 벡터 연산
      1. 
         1. vec\_sub은 두 벡터의 차벡터를, dot\_product는 두 벡터의 내적을, eucllen은 한 벡터의 유클리드 길이를, vector\_cos는 두 벡터간의 cos값을 구한다. argv를 통해 주어지는 similarity 모드에 따라 similarity matrix를 설정하는 것이 달라진다.
   2. similarity matrix & normalization
      1. 
         1. mode값에 따라 cos으로 할지 euclidean distance로 할 지 정해진다.
         2. 벡터간의 cos이나 distance는 dissimilarity를 구하는 것이므로 음수를 취해준다. 이는 나중에 normalization을 통해 [0,1]범위에 들어올 수 있도록 해준다.
      2. 
         1. 주어진 similarity matrix를 normalization해 [0,1]범위로 만들어준다
         2. cos의 경우 자기자신끼리의 cos값 외에도 similarity가 1.0이 되는 벡터쌍이 있다. 즉, 방향이 같은 벡터쌍이 존재한다는 뜻이다.
         3. euc의 경우 similarity가 자기자신 끼리를 제외하면 1에 근접하는 경우가 거의 없다. 즉, 모든 벡터쌍들의 euclidean 거리가 어느정도 차이난다는 것이다.
   3. complete link cluster
      1. 
         1. 수업 lecture 슬라이드에 설명된 방식처럼 complete link clustering 을 시행한다.
         2. assign 에는 0번과 8번이 합쳐지면 0번이 8번에 assign되는 형식, 즉 인덱스가 작은쪽이 큰쪽에 assign되는 것을 기록한다.
         3. eliminated는 clustering이 이루어지면서 매트릭스에서 삭제되는 행, 열의 번호를 기록한다.
         4. mergelist에는 merge되는 쌍의 번호들을, sim에는 모든 clustering이 진행된 후의 similarity matrix를 리턴한다.
         5. 위 코드의 진행과정을 행렬 그림을 통해 설명한 것이다
      2. 
         1. 클러스터링된 결과를 threshold를 초과하는 것들만 서로 cluster하도록 만든다.
         2. clustered에는 각각의 문자가 어떤 번호의 클러스터에 클러스터되었는지를 저장한다. cluster에는 각각의 클러스터에 어떤 문자들이 클러스터되었는지 저장한다.
         3. 예외적으로 최종 cluster들은 인덱스가 1부터 시작한다. 사용자 입장에서 출력의 깔끔함을 위해 그렇게 했으나, 0번부터 하는 것이 코딩에 용이했을 것 같다.
   4. entropy
      1. 
         1. 엔트로피와 information gain을 계산하는 부분이다.
         2. 로그의 윗수로 0이 들어가지 않도록 해준다.
3. 실험 결과
   1. 비교 표

|  |  |  |
| --- | --- | --- |
|  | cos | euc |
| 0.2 |  |  |
| 0.4 |  |  |
| 0.6 |  |  |
| 0.8 |  |  |

* 1. 결과
     1. 주어진 토픽과 비슷한 수의 클러스터 수가 나오고, infogain도 높은 편인 것은 mode = ‘cos’, th = 0.4인 경우이다.
     2. euc로하면 cluster수가 더 많게 나오고, infogain이 더 높다. 즉 similarity가 높기 더 힘들다.
     3. threshold를 높게 잡을 수록 더 많은 수의 cluster들이 만들어진다. similarity에 대한 장벽이 높기 때문이다.

1. 소감

클러스터링은 대표적인 비지도학습이다. 적절하게 여러 차원들에 배치한 벡터들의 정보만으로 비슷한 자료들이 누구인지 찾아주는 멋진 알고리즘이다. 이번 과제를 진행하며 여러 사람들의 정보를 수치화해 벡터로 만든뒤 normalization해 클러스터링해보고 싶어졌다. 즉 사람의 키, 몸무게, 쌍커풀의 짙은정도, 보조개의 짙은정도, 흡연량, 음주량 등등 많은 정보들을 수치화해 클러스터링 해보고 싶다. 이렇게 여러 이성들의 정보를 클러스터링해보면, 내 이상형에 근접한 사람들의 클러스터를 알 수 있지 않을까 싶다.