Assignment #7

Content-Based Movie Recommendation using Clustering

"**movie.csv**" 파일은 movielens 데이터셋에 포함되어 있는 1,664개의 영화에 대한 장르 정보를 포함하고 있다. 첫번째 column은 영화의 제목(Title)을 의미하고, 나머지 19개의 column은 각 영화가 19개의 장르에 포함되는 지 여부를 아래 예와 같이 0/1로 나타내고 있다. 하나의 영화는 여러 장르에 포함될 수 있다.

Title	Unknown	Action	Adventure	Animation	Childrens
Toy Story	0	0	0	1	1
GoldenEye	0	1	1	0	0

1. 각 장르 별로 몇 개의 영화를 포함하고 있는지 계산해 보자. 가장 많은 영화를 포함하는 장르와 가장 적은 영화를 포함하는 장르는 무엇인가?

2. 하나의 영화는 평균적으로 몇 개의 장르에 포함되는지 계산해보자. 그리고 가장 많은 장르에 속해 있는 영화를 찾아보자. 어떤 영화가 몇 개의 장르에 속하는가?

3. 아래 순서에 따라 **Hierarchical clustering**을 사용하여 장르가 유사한 영화들의 그룹을 만들어보자.

   A. Cluster 사이의 dissimilarity 기준으로 average, complete, single linkage 각각을 사용하여 clustering을 수행하고, dendrogram을 각각 출력해보자.

   B. 세 dendrogram의 형태를 비교해보자. 어떠한 특성을 가지는가?

   C. 세 dendrogram이 가지는 height 값의 범위는 어떻게 다른가? 이러한 차이가 발생하는 이유는 무엇인가?

   D. 세 경우에 대해서, 각각 10개의 cluster를 생성해보자. 그리고 각 cluster에 속하는 영화의 수가 어떠한 분포를 가지는 지 비교해보자.

4. 아래 순서에 따라 ***k*-means clustering**을 사용하여 장르가 유사한 영화들의 그룹을 만들어보자.

   A. *k*-means 알고리즘은 시작 시에 랜덤으로 구성되는 cluster에 따라서 성능이 크게 달라질 수 있기 때문에 *k*-means를 일정 횟수만큼 반복 실행한 후 best solution을 선택한다. **kmeans**() 함수의 **nstart** 옵션을 10, 100, 1000으로 증가시켰을 때 얻어지는 solution의 목적함수 값을 비교해보자. cluster의 수 $k = 10$으로 설정한다. 그리고 **kmeans**() 실행 시마다 **set.seed**(123)으로 랜덤 시드를 설정하자.

   B. 지금부터는 **nstart** = 1000일때의 얻어지는 cluster에 대해서 분석을 진행한다. 각 cluster에 속하는 영화의 수는 어떤 분포를 가지는가? 3.D번에서 얻은 결과와 비교해보자.

   C. 각 cluster에 속한 영화들 중 각 장르에 속하는 영화의 비율을 모두 계산해보자 (아래 예와 같이 10×19 행렬의 형태로 결과 출력). 이 결과를 바탕으로 각 cluster가 주로 어떤 장르의 영화들로 구성되는지, 각 cluster 별 특성을 분석해보자.

	Action	Adventure	Animation	…
Cluster 1	0.32	0.01	0.00	
Cluster 2	0.98	0.55	0.02	

⋮

D. 어떤 사용자가 영화 "**Titanic (1997)**" 에 평점 5점을 부여하였다. 이 사용자에게 "Titanic (1997)"과 장르가 유사한 영화를 추천하고자 한다. "Titanic (1997)"은 어떤 cluster에 속해 있는가? 이 사용자에게 영화 5개를 추천해보자.

E. C-D번의 분석을 cluster의 수 $k = 5$에 대해서 동일하게 반복해보고, $k = 5$일때와 $k = 10$일때의 결과를 비교해보자.