

# COSE362 Mid-term report

2021320024 김민서

2021320303 정지원

2021320322 윤민서

2021320323 허준환

## 1. 문제 정의

이번 **Term Project**는 고려대학교 학우들의 강의 평가 데이터를 통해 학부 강의들을 어떤 유형으로 구분지을 수 있을지에 대한 문제를 해결하는 것을 목표로 한다. 대학생들이 생각하기에 가장 보편적이고 대표적인 강의 유형에는 후술할 ‘명강’과 ‘꿀강’이 있는데, 이번 프로젝트의 우선적인 목표는 이와 같이 최소 2개 이상의 강의 유형을 찾아내는 것이다. 또한, 분석 과정 및 결과에서 2가지의 강의 유형 이외에도 새로운 유형이 발견된다면, 그 유형을 새롭게 정의하고 특징을 분석하고자 한다.

‘명강’은, 성적이 잘 안 나올지라도 교수님이 전달해 주시는 내용과 학문 자체에서 얻어갈 것이 많은 강의이며, ‘꿀강’은 강의 내용이 수강자에게 알차지 않거나 이미 배운 내용이 대부분 포함되어 있음에도 불구하고 단순히 최종 성적만을 바라보는 신청을 하는 강의이다. 이외에도, 따로 이름이 붙지는 않았으나 성적 산출이나 강의 진행, 강의력 등의 이유로 들으면 좋지 않은 강의 등이 있다.

대부분의 대학생들은 강의를 구분지을 때 강의평, 그리고 이에 주어지는 난이도, 학점, 학습량 등 여러가지 요소들을 참고하고 고려하면서 강의 유형을 파악한다. 하지만 이러한 유형이 있음에도 불구하고, 이 유형들에 대한 구분 기준은 명확하게 존재하지 않고 있다. 현재 4학기를 재학 중인 시점에서 앞으로의 전공 과목을 이러한 구분 기준을 바탕으로 자신에게 적합한 강의를 더욱 신중하게 선택해야 할 필요가 있다. 따라서 이번 **Term Project**에서는 대학생들의 강의평 데이터를 바탕으로 강의 유형이 어떤 기준으로 나뉘질 수 있는지 알아보하고자 한다.

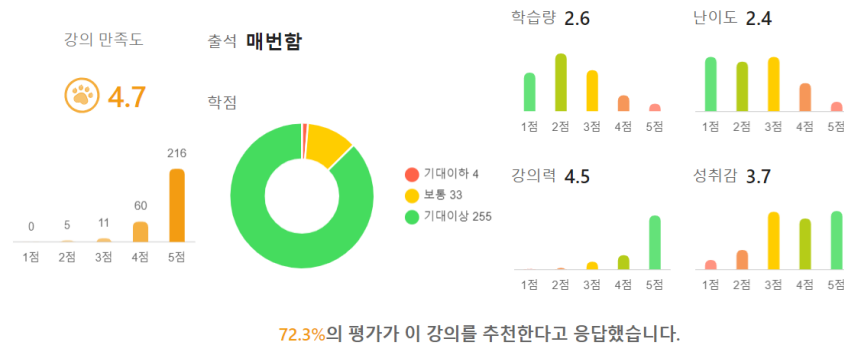
## 2. 시도해본 방법론

### 1) 데이터 설명

KLUE의 강의평가에서 평가 요소로 작용하는 것은 <Figure 1>에서 확인할 수 있는 것과 같이 수강 학점, 강의 구분, 교수, 요일 및 교시, 강의실 위치, 강의 만족도, 출석 빈도, 최종 성적의 만족도, 학습량, 난이도, 강의력, 성취감, 강의 추천 응답 비율 등이 있다. 아래는 인터페이스를 포함한 데이터의 예시이다.

강의평가

강의노트



<Figure 1: 고려대학교 전용 강의평가 페이지 KLUE에 있는 강의>

수집한 데이터의 범위는 정보대학이 출범한 2014년부터 전면 대면 수업이 진행되었던 2019년까지이다.

## 2) 데이터 수집 방법

KLUE 사이트의 경우 스크롤을 내려야 후기평들이 갱신되는 동적페이지이다. 따라서 selenium 라이브러리와 chrome의 웹 드라이버를 사용하여 강의 사이트의 리뷰 데이터들을 동적 크롤링하였다. 이후 beautiful soup 라이브러리를 통해 각 중요 평가요소를 parsing하여 데이터셋을 만들었다.

## 3) 모델 학습

강의를 구분짓는 기준에는 위에서 언급한 ‘명강’, ‘꿀강’뿐만 아니라 각각의 강의를 특징지을 수 있는 무수히 많은 기준이 포함될 수 있다고 보았다. 즉, 명확한 기준이 존재하지 않는 강의 유형의 특징 상 label이 존재하지 않을 수밖에 없다. 따라서 기계학습의 분야 중 label이 필요한 supervised learning이 아니라, label이 필요 없는 unsupervised learning이 더 적절하다고 볼 수 있다. 또한 어떤 강의 유형이 존재하고, 어떠한 기준으로 나뉘어지는지 확인하기 위해서는 Clustering을 이용하는 것이 적절하다.

이뿐만 아니라, 앞서 말했듯 KLUE에 있는 강의평들을 통해 한 강의 데이터 당 얻을 수 있는 feature의 수는 13개 이상이 존재한다. 만일 이 feature들을 모두 사용한다면, dimension 수가 매우 커져 데이터를 처리하는 데 연산이 복잡해진다. K-Means Algorithm은 Clustering 중에서도 구현이 간단하고, 계산 속도가 빠르다는 장점이 있다. 고차원의 feature들을 다루면서 생기는 연산 복잡도의 문제로 인해, Clustering 방법론 중에서도 간단하고 빠른 K-Means Algorithm을 선택하였다.

### # column에 대한 설명

1. index
2. year: 연도-학기
3. type: 학수번호

4. subject: 과목명
5. day: 요일, 교시
6. satisfy: 강의 만족도
7. writer: 강의평 작성자
8. ratings0: 학습량
9. ratings1: 난이도
10. ratings2: 학점
11. ratings3: 성취감
12. comment: 강의평
13. helpful: 강의평이 받은 추천의 개수

Clustering을 진행하기 위하여 먼저 유의미하지 않은 강의평 작성자와 강의평이 받은 추천의 개수, csv 파일을 정리하는 과정에서 생긴 index 등의 feature를 삭제하는 과정을 진행하였다. 그리고 type (학수번호), subject (과목명)과 같이 중복되는 feature를 처리하기 위해 둘 중 과목명이 포함된 subject column을 삭제하는 과정을 진행하였다. 또한 모델링 과정에서 유의미하지 않은 column을 없애는 과정도 거쳤다. 삭제한 column은 다음과 같다.

1. index
2. year
3. writer
4. comment
5. subject
6. helpful

학수번호, 교수자, 요일 및 시간은 categorical variable이고, 문자열로 이루어져 있어 수집한 데이터를 그대로 사용할 수 없다. 학수번호, 교수자의 경우 int 자료형으로 encoding하기 위해 one-hot encoding을 통하여 0 또는 1의 int 자료형으로 변환하였다. 하지만, 이와 같은 방식으로 인해 variable의 수가 100개 이상으로 급격히 늘어나게 되어 Clustering 분석이 매우 어려워졌다. 그래서 학수번호와 교수자 feature 또한 제외시켰다. 요일 및 시간은 categorical variable로 처리되어 있지만, 요일과 시간을 분리하여 저장하면 요일은 categorical variable, 시간은 numeric variable로 취급할 수 있는 feature이기 때문에 요일과 시간을 분리하였다. 요일은 월요일부터 금요일까지의 variable을 만들고 난 후, 월요일 및 수요일에 수업이 있으면 월요일과 수요일 변수만 1이고 나머지는 0으로 저장되게끔 int 자료형으로 변환하였다.


본격적으로 K-Means algorithm을 적용하기에 앞서서 MinMaxScaling을 통하여 모든 데이터의 값을 0과 1 사이에 위치하게 하였다. Scaling이 진행되지 않았을 경우에는 데이터의 값이 너무 크거나 너무 작은 경우에 model에 K-Means algorithm을 fitting하는 과정에서 0으로 수렴하게 되거나 무한대로 발산하게 되는 현상이 일어날 수 있기 때문에 전처리에서 크게 중요한 과정으로 간주하였다.

이후 Cluster 개수의 범위를 1부터 20으로 설정한 뒤 각각의 Cluster 개수에 대하여 Inertia를 계산하였다. 계산된 Inertia는 x축을 K, y축을 Inertia로 한 꺾은선 그래프로 시각화하여 Cluster의 개수를 정하는 데에 사용하였다.

Clustering의 결과를 2차원 공간에 시각화 하기 위하여 주성분 분석(Principle Component Analysis, 이하 PCA)을 통하여 component의 개수를 2와 3으로 감소시켰다. PCA의 결과로 새롭게 만들어진 두 개의 축을 각각 PCA1, PCA2, PCA3으로 이름을 정한 뒤 PCA1을 x축, PCA2를 y축, PCA3를 z축으로 한 scatter plot을 그려서 Clustering이 잘 진행되었는지 점검하는 과정을 수행하였다.

### 3. 중간 결과

앞서 설명하였던 방식대로, variable 후처리 과정을 거쳐 다음과 같은 column들이 선정되었다.

 `df.head(5)`

	satisfy	ratings0	ratings1	ratings2	ratings3	mon	tue	wed	thu	fri	period
0	3	3	3	3	3	0	1	0	1	0	2
1	3	4	4	3	3	0	1	0	1	0	2
2	1	5	5	2	1	0	1	0	1	0	2
3	4	3	4	4	4	0	1	0	1	0	2
4	4	4	4	3	4	0	1	0	1	0	2

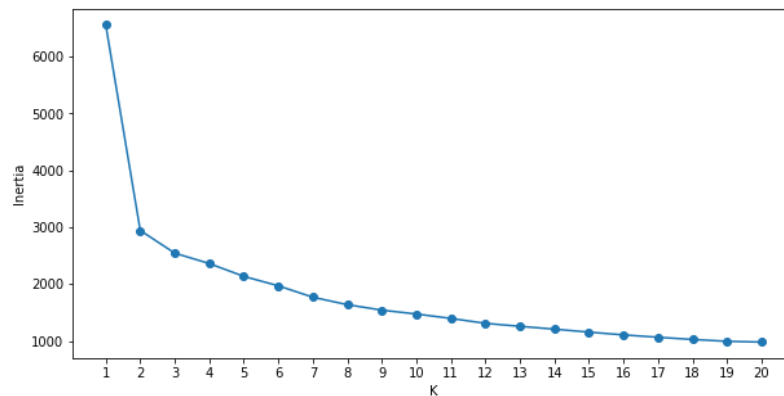
< Figure 2. variable 후처리 결과 >

# column에 대한 설명

1. satisfy: 강의 만족도
2. ratings0: 학습량
3. ratings1: 난이도
4. ratings2: 학점
5. ratings3: 성취감
6. mon: 월요일
7. tue: 화요일
8. wed: 수요일
9. thu: 목요일
10. fri: 금요일
11. period: 교시

Cluster의 개수를 정하기 위해 계산한 Inertia의 값과 시각화 결과는 다음과 같다.

K: 1	Inertia: 6556.706999130299	
K: 2	Inertia: 2942.9607246928663	Difference: 3613.7462744374325
K: 3	Inertia: 2544.793062359396	Difference: 398.16766233347016
K: 4	Inertia: 2361.6805659744987	Difference: 183.11249638489744
K: 5	Inertia: 2136.215945084027	Difference: 225.46462089047145
K: 6	Inertia: 1970.6654011302007	Difference: 165.5505439538265
K: 7	Inertia: 1770.086572245367	Difference: 200.57882888483368
K: 8	Inertia: 1638.45967931295	Difference: 131.62689293241715
K: 9	Inertia: 1542.7437965929485	Difference: 95.71588272000145
K: 10	Inertia: 1474.4276962499398	Difference: 68.31610034300866
K: 11	Inertia: 1397.1667918155736	Difference: 77.26090443436624
K: 12	Inertia: 1310.486729343309	Difference: 86.68006247226458
K: 13	Inertia: 1258.505812474927	Difference: 51.98091686838188
K: 14	Inertia: 1208.5191187797227	Difference: 49.986693695204394
K: 15	Inertia: 1155.8431260935117	Difference: 52.675992686210975
K: 16	Inertia: 1107.709070057331	Difference: 48.13405603618071
K: 17	Inertia: 1065.8985476955236	Difference: 41.81052236180744
K: 18	Inertia: 1027.5124822820617	Difference: 38.38606541346189
K: 19	Inertia: 994.9033689451003	Difference: 32.6091133369614
K: 20	Inertia: 981.1813970935816	Difference: 13.721971851518674



<Figure 3, 4: K값에 따른 Intertia의 변화>

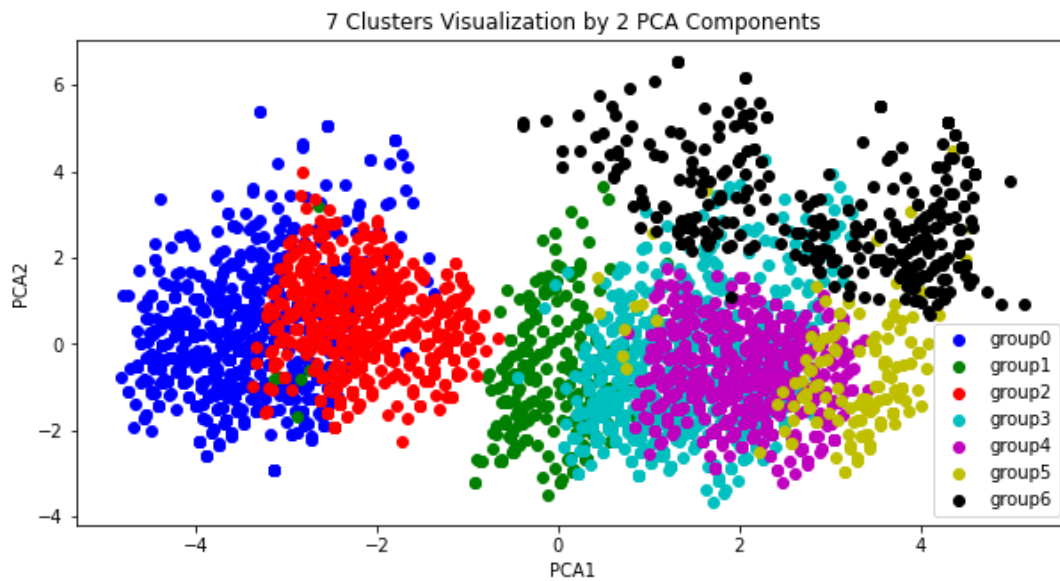
Elbow method를 통하여 K값을 결정하려고 했으나 뚜렷하게 그래프가 꺾이는 지점이 K가 2일 때로, 수많은 강의를 단 2개의 Cluster만으로 나누기엔 부족함이 있다고 생각하였다. 그렇기 때문에 현재의 K값과 이전의 K에 대한 Inertia의 차이가 마지막으로 200 이상인 지점을 기준으로 하였고 이러한 기준으로 결정된 Cluster의 개수는 7이다.

결정된 Cluster의 개수를 기반으로 하여 K-Means algorithm을 통하여 model을 fitting 하였다. fitting한 model을 바탕으로 강의평 별로 Cluster를 prediction 한 뒤에 원래의 데이터프레임에 cluster column을 추가하였다. 아래는 출력 예시이다.

	satisfy	ratings0	ratings1	ratings2	ratings3	mon	tue	wed	thu	fri	period	cluster
0	3	3	3	3	3	0	1	0	1	0	2	0
1	3	4	4	3	3	0	1	0	1	0	2	0
2	1	5	5	2	1	0	1	0	1	0	2	0
3	4	3	4	4	4	0	1	0	1	0	2	0
4	4	4	4	3	4	0	1	0	1	0	2	0

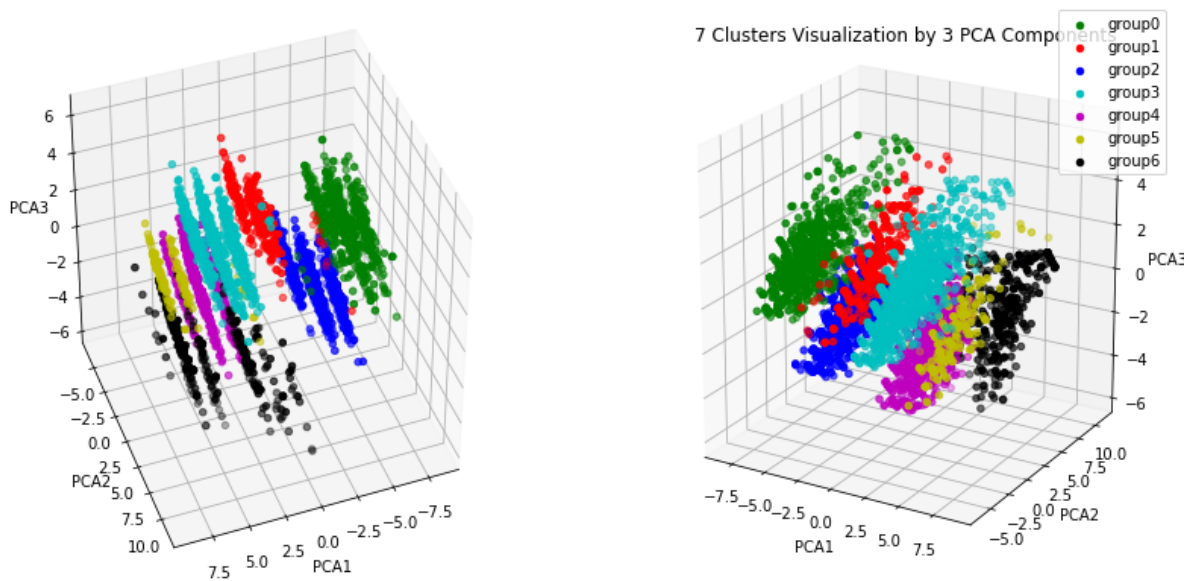
<Figure 5: Cluster가 배정된 데이터프레임>

Cluster의 번호는 0번부터 6번까지이다. 결과를 2차원 좌표평면에 표시하기 위하여 PCA를 통하여 새로운 2개의 축을 만들었고 scatter plot을 나타낸 결과는 아래와 같다. legend의 group0~group6은 각각 0번부터 6번까지의 Cluster를 나타내는 것이다.



<Figure 6: Clustering 결과를 시각화한 2차원 scatter plot>

하지만 2차원 좌표평면으로 보았을 땐, Clustering이 제대로 이루어지지 않은 것처럼 보인다. 그래서 결과를 3차원 좌표공간에도 나타내 보았다.



<Figure 7: Clustering 결과를 시각화한 3차원 scatter plot>

#### 4. 중간 결과에 대한 해석

우선 각 Cluster에 몇 개의 강의평 데이터가 속하는지 살펴보았다. 결과는 다음과 같다.

	satisfy	ratings0	ratings1	ratings2	ratings3	mon	tue	wed	thu	fri	period
cluster											
0	793	793	793	793	793	793	793	793	793	793	793
1	296	296	296	296	296	296	296	296	296	296	296
2	816	816	816	816	816	816	816	816	816	816	816
3	1056	1056	1056	1056	1056	1056	1056	1056	1056	1056	1056
4	751	751	751	751	751	751	751	751	751	751	751
5	152	152	152	152	152	152	152	152	152	152	152
6	352	352	352	352	352	352	352	352	352	352	352

<Figure 8: 각 Cluster에 해당하는 강의평 데이터 수>

K-Means algorithm은 각 Cluster의 데이터를 가장 잘 표현하는 reference vectors를 중심으로 Clustering을 반복해나간다. 즉, Cluster의 centroid를 통해서 그 Cluster의 특징을 살펴볼 수 있다. 7개의 Cluster의 reference vector는 다음과 같다.

	satisfy	ratings0	ratings1	ratings2	ratings3	mon	tue	wed	thu	fri	period
cluster											
0	3.692308	3.667087	3.658260	3.283733	3.571248	0.000000	1.000000	0.000000	1.000000	0.000000	1.974779
1	4.087838	3.243243	3.250000	4.067568	3.405405	0.165541	0.371622	0.266892	0.195946	1.000000	5.290541
2	3.851716	3.829657	3.776961	3.464461	3.838235	0.988971	0.011029	1.000000	0.000000	0.049020	1.939951
3	3.896780	3.405303	3.249053	3.681818	3.571023	0.000000	1.000000	0.000000	1.000000	0.000000	5.848485
4	4.049268	3.743009	3.648469	3.776298	3.889481	1.000000	0.000000	1.000000	0.000000	0.000000	5.888149
5	4.302632	3.151316	3.217105	4.401316	3.703947	1.000000	0.690789	0.000000	0.309211	0.000000	6.388158
6	2.454545	2.198864	2.411932	3.181818	1.775568	0.923295	0.002841	1.000000	0.000000	0.008523	4.713068

<Figure 9: 각 Cluster의 centroid>

<Figure 8>과 <Figure 9>를 살펴보면, Cluster 0은 화요일 및 목요일에 평균적으로 2교시에 시작하는 수업으로 이루어져 있음을 알 수 있다. Cluster 0의 강의 만족도는 평균적으로 3.7이고, 학습량은 3.7, 난이도는 3.7, 학점은 3.3, 성취감은 3.6 정도이다. 강의 만족도, 학습량, 난이도, 학점, 성취감은 1부터 5까지이므로, Cluster 0은 다섯가지 항목 모두 보통 정도에 해당한다고 볼 수 있다. 결론적으로, 정보대학 전공 과목 중에는 화요일 및 목요일 오전에 보통 수준의 강의로 이루어진 하나의 Cluster가 있음을 알 수 있다.

이와 마찬가지로 Cluster 1~6도 비슷하게 해석할 수 있다. Cluster 1은 금요일과 월화수목 중 한 요일에 진행되고, 평균적으로 5교시 이후에 시작된다. 평균적으로 강의 만족도는 4.0, 학습량은 3.2, 난이도는 3.3, 학점은 4.0, 성취감은 3.4이다. Cluster 5, 6을 제외한 다른 Cluster들 보다 학습량이 적지만 취득 학점과 강의 만족도가 높기에, Cluster 1은 앞에서 언급하였던 꿀강에 해당한다고 볼 수 있다. 즉, 일주일 중 금요일 오후에 한 번 수업을 진행하고, 공부량이 적으나 학점 및 강의 만족도가 높은 Cluster가 있음을 알 수 있다.

Cluster 2는 보통 월요일과 수요일 1~2교시에 진행되고, 평균적으로 강의 만족도는 3.9, 학습량은 3.8, 난이도는 3.8, 학점은 3.5, 성취감은 3.8 정도인 강의로 이루어져 있다. 그리고 Cluster 3은 화요일과 목요일 5~6 교시에 진행되고, 평균적으로 강의 만족도는 3.9, 학습량은 3.4, 난이도는 3.2, 학점은 3.6, 성취감은 3.6 정도인 강의로 이루어져 있다. Cluster

2와 3 둘 다 강의 만족도가 좋은 편이고, 나머지는 보통 수준임을 볼 수 있다. 다만 Cluster 2는 난이도가 조금 더 높은 만큼 성취도가 더 높고, Cluster 3은 난이도가 조금 더 낮은 대신 성취도가 더 낮다는 차이가 있다.

Cluster 4는 월요일과 수요일 5~6교시에 진행되고, 평균적으로 강의 만족도는 4.0, 학습량은 3.7, 난이도는 3.6, 학점은 3.8, 성취감은 3.9 정도인 강의로 이루어져 있다. 다른 Cluster에 비해 학습량이 높은 편이며 동시에 강의 만족도와 성취감이 높기 때문에, Cluster 4는 앞서 언급했던 명강에 해당한다고 볼 수 있다. Cluster 5는 월요일과, 화요일 또는 목요일 6교시 이후에 진행되고, 평균적으로 강의 만족도는 4.3, 학습량은 3.2, 난이도는 3.2, 학점은 4.4, 성취감은 3.7 정도인 강의로 이루어져 있다. 다른 Cluster들에 비해 강의 만족도와 학점 평가가 가장 높고, 강의 만족도에 비해 학습량과 난이도가 낮다. 이는 곧 Cluster 1과 비슷하게 꿀강으로 볼 수 있는데, Cluster 1보다 더 꿀강에 해당한다고 볼 수 있다.

마지막으로 Cluster 6은 보통 월요일과 수요일 4~5교시 정도에 진행되고, 평균적으로 강의 만족도는 2.5, 학습량은 2.2, 난이도는 2.4, 학점은 3.2, 성취감은 1.8 정도인 강의로 이루어져 있다. 다른 Cluster들에 비해 강의 만족도, 학습량, 난이도, 성취감이 가장 낮은 강의이다. 학점을 보통 수준으로 주긴 하나, 다른 4개의 평가가 가장 낮기에 앞서 언급했던 좋지 않은 강의에 해당함을 볼 수 있다.

7개의 Cluster들을 정리하면 다음과 같다.

Cluster	날짜	시간	분류
0	화, 목	2교시	보통
1	금, 월화수목 중 하나	5~6교시	꿀강
2	월, 수	1~2교시	보통
3	화, 목	5~6교시	보통
4	월, 수	5~6교시	명강
5	월, 화 또는 목	6교시 이후	꿀강
6	월, 수	4~5교시	좋지 않은 강의

<Figure 10: Clustering 결과의 해석>

## 5. 개선해야 할 부분과 개선 계획

### 1.1) 수집 데이터 Feature의 비일관성

2020년 2학기부터 학점 항목이 강의력 항목으로 바뀌었다. 학점과 강의력은 강의를 특징별로 구분하는 데 중요한 역할을 한다. 학점 항목은 기대 이상, 보통, 기대 이하로 후기를 남길 수 있지만 강의력에 대한 부분은 대체할 수 있는 feature가 존재하지 않는다. 따라서 강의력 항목이 생긴 전과 후를 기준으로 data의 범위가 더 넓고 개수가 더 많은 2014년부터 2019년까지의 데이터만을 사용하여 모델링을 진행했기 때문에 최신 강의의 경향이 잘 반영되지 않았다고 볼 수 있다는 것이 개선해야 할 부분이 될 수 있다고 생각한다.



## 1.2) 해결 방안 - rating prediction by imputation

2020년 1학기까지의 데이터는 강의력을 결측치로 분류한다. 그 후 각 결측치를 자연어 후기와 다른 feature를 기준으로 예측하는 모델을 만들어 강의평을 통하여 예측을 진행한 후 각각의 feature를 학습에 사용한다. kaggle의 “Trip advisor hotel review” data의 review<sup>1</sup>를 통해 rating을 예측하는 방법을 선행 연구로 하여 이와 유사한 방법으로 예측할 수 있을 것이라고 판단한다.

## 2.1) 전부 사용하지 못한 features

categorical variable인 교수명과 학수번호를 one-hot encoding 방식을 통해 1 또는 0의 int 자료형을 가지는 variables로 만들면, 변수의 수가 너무 많아진다는 문제점이 발생한다. 이로 인해 Clustering을 하여도 변수의 수가 굉장히 많아지기 때문에 해석하기 매우 어려워진다. 하지만 교수명과 학수번호(과목명)는 강의 분류에 있어 유의미한 feature일 것으로 예상되기 때문에, 이러한 문제점을 해결하여 feature로서 사용한다면 더 정확한 결과를 얻어낼 수 있을 것이다.

## 2.2) 해결 방안 - 사용하지 못한 features의 활용 방법

categorical variable인 교수명과 학수번호를 word2vec와 같은 기법을 통해서 distributed representation으로 교수명과 학수번호를 continuous vector로 나타내면, 앞서 언급했던 문제점을 해결할 수 있다. 다만 이 경우에는 word2vec에서 흔히 사용되는 CBOW나 Skipgram은 사용할 수 없어, 다른 방식을 찾아낼 필요가 보인다.

## 3.1) 강의 방식의 변화 (대면→비대면→대면)

같은 교수자가 진행하는 같은 과목의 경우에도 비대면 수업의 경우와 대면 수업의 경우에 따라 강의 방식이 크게 달라지는 경우가 많다. 초반에 언급한 꿀강으로 분류하는 기준 중 하나는 강의에 많은 시간을 투자하지 않을 수 있는 구조이거나 요구하는 과제물이 적은 것이다. 즉, 강의 방식에 따라 강의의 과제물과 이수 요건이 달라질 수 있고 이에 따라 분류가 달라질 수 있기 때문에 대면 수업 여부가 분류에서 중요한 feature가 될 수 있다. 코로나-19 방역 수칙으로 인한 수업 방식 제한이 많이 완화된 현재도 비대면 강의가 모두 사라진 것은 아니다. 또한 앞으로도 모두 대면으로 전환될 것이라는 보장이 없다.

## 3.2) 해결 방안 - 대면수업 여부 수집

대면 수업 여부를 binary feature로 추가하여 학습하는 데 feature로 사용한다면 유의미한 결과를 얻을 수 있을 것으로 예상된다.

## 4.1) K-Means algorithm

scikit-learn의 KMeans 함수에는 n\_clusters, init, n\_init, max\_iter, tol, verbose, random\_state, copy\_x, algorithm 등의 parameter가 있다.<sup>2</sup> 현재 시점에서 시도해 본 유의미한 hyperparameter는 n\_clusters뿐이다. 지속적인 hyperparameter tuning을 통하여 정의된 문제에서 해결하고자 하는 목표에 더욱 적합한 모델을 찾는 것 또한 개선해 나가야 할 점이라고 생각할 수 있다.

# parameter에 대한 설명

---

1

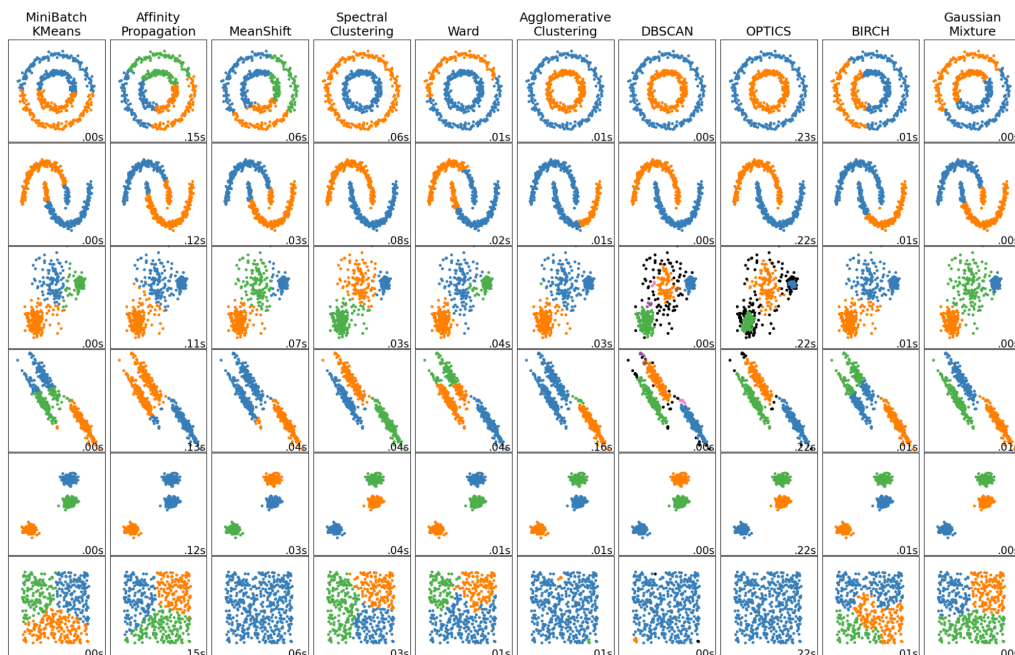
<https://medium.com/analytics-vidhya/predicting-the-ratings-of-reviews-of-a-hotel-using-machine-learning-bd756e6a9b9b>

<sup>2</sup> <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

**n\_clusters**: 형성할 cluster의 수와 생성할 centroid의 수  
**init**: k-means++, random 등의 initialization 방법  
**n\_init**: K-Means algorithm이 다른 centroid seed 실행되는 횟수  
**max\_iter**: 최대 반복 횟수  
**tol**: 두 연속된 iteration 사이의 centroid의 차이에 대한 상대적 오차가 tol보다 작을 때 수렴으로 판정  
**verbose**: 실행 과정 출력  
**random\_state**: random seed의 결정  
**copy\_x**: 원본 데이터의 보존/수정  
**algorithm**: EM-style (Expectation-Maximization), triangle inequality 등 Clustering에 사용할 algorithm을 선택

## 4.2) Other Clustering Algorithms

K-Means algorithm을 제외한 다른 Clustering algorithm도 고려해 볼 수 있다. 수업시간에 다른 Hierarchical clustering, Agglomerative clustering과 아래 그림의 다른 알고리즘들을 시도해 볼 수 있다. 최신 강의를 포함하여 데이터 전처리 과정을 다시 진행한 후, 데이터의 분포 형태를 파악하여 적절한 Clustering algorithm을 선택할 수 있다. 데이터 분포에 따라서 적용하기 용이한 algorithm이 달라지는데, scikit-learn 홈페이지의 example<sup>3</sup>에 분포 형태 별로 어떤 algorithm을 적용하면 좋을지에 대한 설명 자료를 참고할 수 있다.



<Figure 11: Clustering algorithm examples>

### ● 참고자료

[1]

<https://medium.com/analytics-vidhya/predicting-the-ratings-of-reviews-of-a-hotel-using-machine-learning-bd756e6a9b9b>

[2] <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

[3] <https://scikit-learn.org/stable/modules/clustering.html#clustering>

<sup>3</sup> <https://scikit-learn.org/stable/modules/clustering.html#clustering>