데이터 사이언스 기초

12주차 과제

2016126040 유영준

2019125071 최지운

2019125007 김규리

PRESENTATION START

PPT Contents

01

데이터 탐색 및 전처리

데이터 설명, 전처리,
변수 선택, 극단값 처리, 정규화

데이터 소개

02

계층적 군집화

덴드로그램 형성,
시각적 기준 군집 나누기

군집화

03

K-mean 군집화 적용

적절한k군집의 수,
군집 최적화, 성능평가,
각 군집에 대한 설명

군집화

04

군집화 비교

k-mean과 다른 군집화를 비교

결론 및 비교

PPT Contents

01

A

데이터 탐색 및 전처리

데이터 설명, 전처리,
변수 선택, 극단값 처리, 정규화

데이터 소개

02

계층적 군집화

덴드로그램 형성,
시각적 기준 군집 나누기

군집화

03

K-mean 군집화 적용

적절한k군집의 수,
군집 최적화, 성능평가,
각 군집에 대한 설명

군집화

04

군집화 비교

k-mean과 다른 군집화를 비교

결론 및 비교

1. 데이터 탐색 및 전처리 - 데이터 설명 및 변수 선택

데이터 설명

- 2021년 분기별 서울시 골목상권 영역 내 점포의 추정 매출 정보를 연령대별 매출로 정리하여 수집
- 총 14만개의 데이터 포인트가 존재 → 축소할 필요가 있음
- 총 80개의 변수가 존재 → 분석에 필요한 변수 선택 필요
  - 분기, 상권, 업종, 총 추정 매출, 각 인구통계학적 특성에 따른 추정 매출, 시간대별 추정 매출 등으로 구성

data	140830 obs. of 80 variables

- 데이터 분석 목표 ➔ 2021년 1분기 기준, 주요 상권들 중 20대에게 인기 있는 상권을 군집화를 통해 추출하기

결측지 확인

존재하지 않는다.

데이터 타입 변환

```
data$상권_코드_명 <- as.factor(data$상권_코드_명)
data$서비스_업종_코드_명 <- as.factor(data$서비스_업종_코드_명)
```

1. 데이터 탐색 및 전처리 - 변수 선택

군집화의 목적에 따른 변수 선택

🕹 우리 팀이 설정한 군집 분석의 목적은 서울 시에서 20대에게 인기있는 상권을 확인해보는 것
- 이를 위하여 상권명, 분기당 매출 금액, 연령대 20대 매출 금액을 변수로 설정
- 코로나 19 거리두기 정책과 확진자 수가 상권에 영향을 주었을 것으로 예상하여
  2021년 분기별 확진자 수를 인터넷에서 비교해본 결과 일별 발생 확진자의 숫자가 적었던 2021년 1분기의 데이터를 선택

최종 선택 변수

상권_코드_명: 점포가 위치한 상권의 위치
서비스_업종_코드_명: 점포의 업종 종류
분기당_매출_금액: 점포의 해당 분기 매출금액
연령대_20_매출_금액: 해당 분기 점포의 20대 손님 매출금액

1. 데이터 탐색 및 전처리 – 전처리 및 정규화

극단값 처리

극단값에 대한 처리는 하지 않음
현재 데이터 셋은 상권에서 얼마의 매출이 발생하는 가를 의미
➡
따라서 극단값을 없앨 경우 해당 상권이 사라짐
즉, 매출이 높은 상권이 사라질 가능성이 있기 때문에
극단값을 제거하지 않음

데이터 정규화

```
datav2[2:3] <- scale(datav2[2:3], center = FALSE,
apply(datav2[2:3], MARGIN = 2, FUN = max))
```

```
분기당_매출_금액     연령대_20_매출_금액
Min.   :0.000000   Min.   :0.000000
1st Qu.:0.001477   1st Qu.:0.001510
Median :0.003877   Median :0.004431
Mean   :0.012589   Mean   :0.017692
3rd Qu.:0.009691   3rd Qu.:0.011501
Max.   :1.000000   Max.   :1.000000
```

데이터 포인트 줄이기

1) 1분기의 데이터로 한정

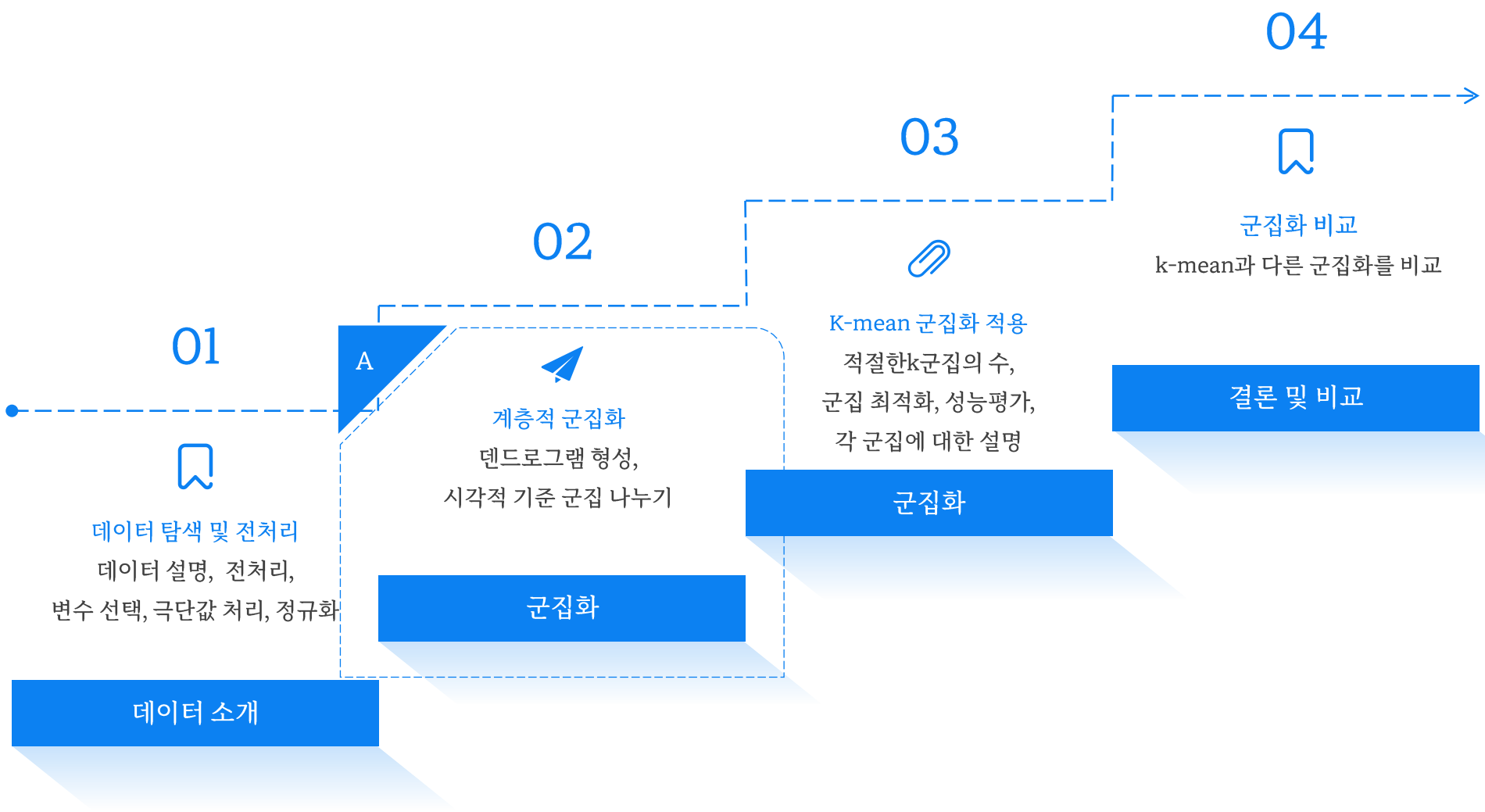
```
datav1 <- subset(data, data$기준_분기_코드==1)
datav1 <- datav1[,c('상권_코드_명','서비스_업종_코드_명', '분기당_매출_금액', '연령대_20_매출_금액')]
```

2) 상권별로 통합 – 총 1651개의 상권

```
datav2 <- as.data.frame(matrix(nrow = length(levels(datav1$상권_코드_명)), ncol = 3))
names(datav2) <- c('상권_코드_명', '분기당_매출_금액', '연령대_20_매출_금액')
datav2$상권_코드_명 <- levels(datav1$상권_코드_명)
```

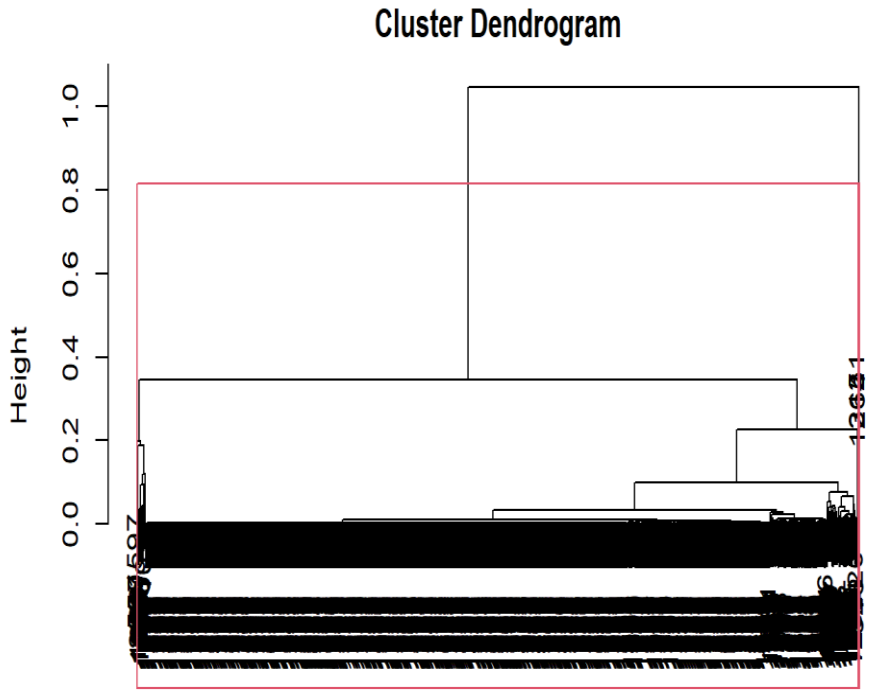
```
'data.frame':   1651 obs. of  3 variables:
 $ 상권_코드_명       : chr  "4.19민주묘지역 2번" "63빌딩" "DMC(디지털미디어시티)" "GS강동자이아파트" ...
 $ 분기당_매출_금액    : num  2.58e+09 2.30e+09 3.98e+10 3.17e+09 4.16e+09 ...
 $ 연령대_20_매출_금액: num  3.10e+08 2.17e+08 4.95e+09 2.34e+08 1.11e+09 ...
```

PPT Contents



2. 계층적 군집화 – 덴드로그램 형성, 시각적 기준 군집 나누기

Dendrogram 생성



시각적으로 확인했을 때 edge의 길이가
확연하게 길다고 판단되는 경우에 나눔
따라서 k=2로 군집 설정

거리 기반 다르게하여 계층적 군집 분석

유클리디언
Cluster method : centroid
Distance : euclidean
Number of objects: 1651

맨하탄 거리 기반 측정
Cluster method : centroid
Distance : manhattan
Number of objects: 1651

민노스키 거리 기반 측정
Cluster method : centroid
Distance : minkowski
Number of objects: 1651

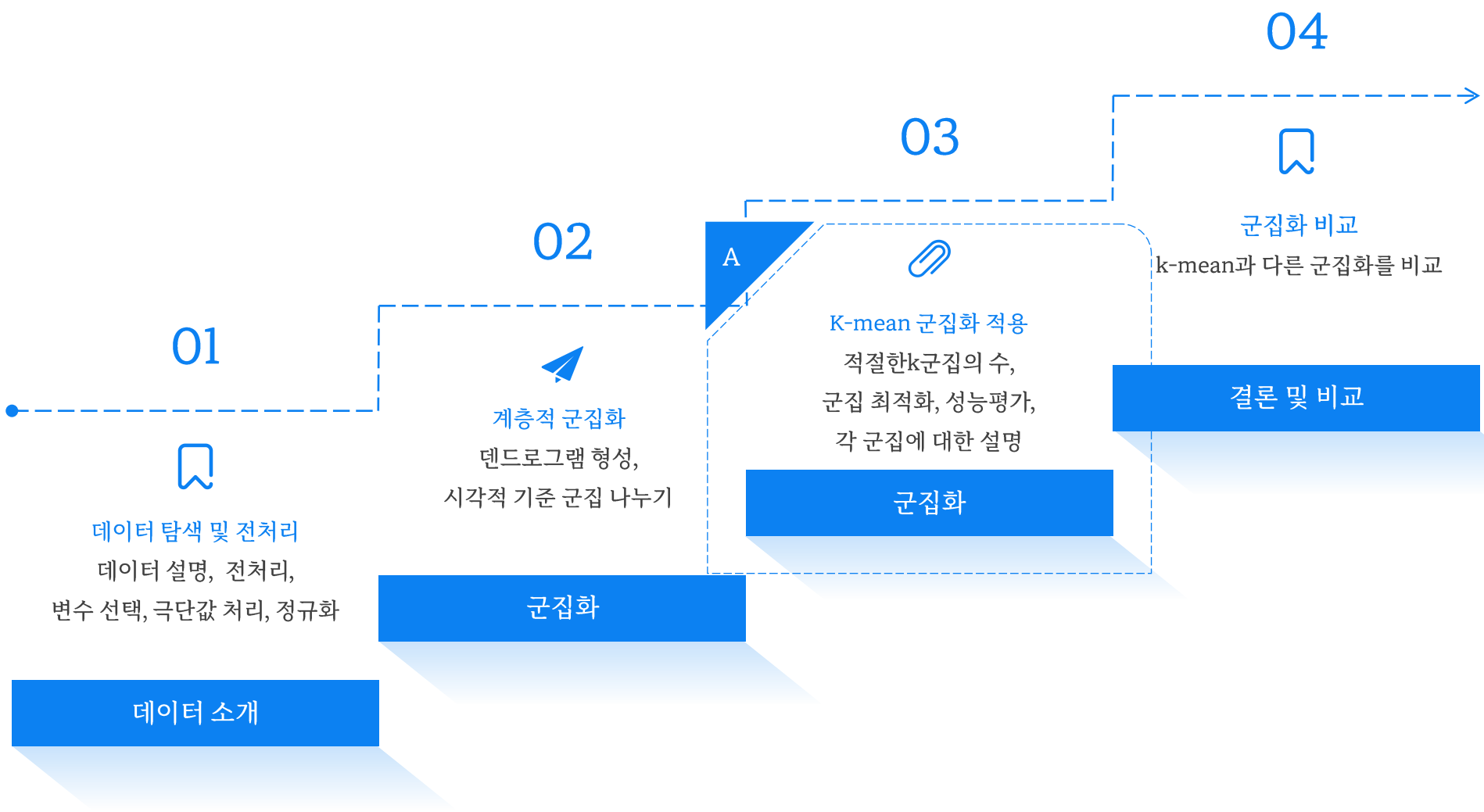
거리에 따른 분류 결과 결국 같은 분류 결과가 나오는 것을
확인 할 수 있음

```
cl_eu.k
   1    2
1604   47

cl_man.k
   1    2
1604   47

cl_min.k
   1    2
1604   47
```

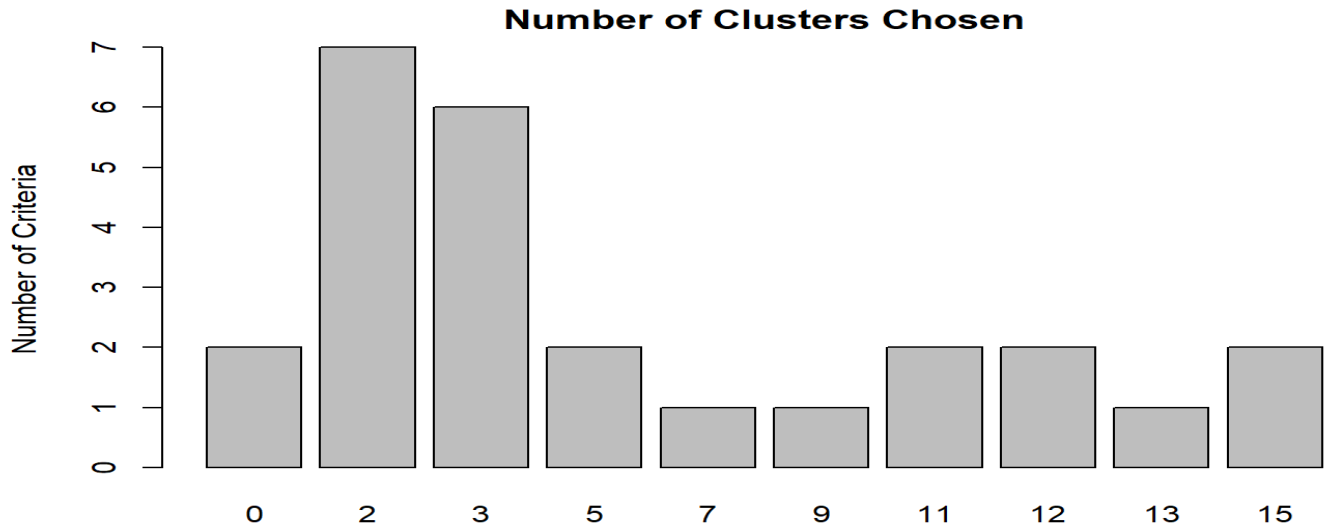
PPT Contents

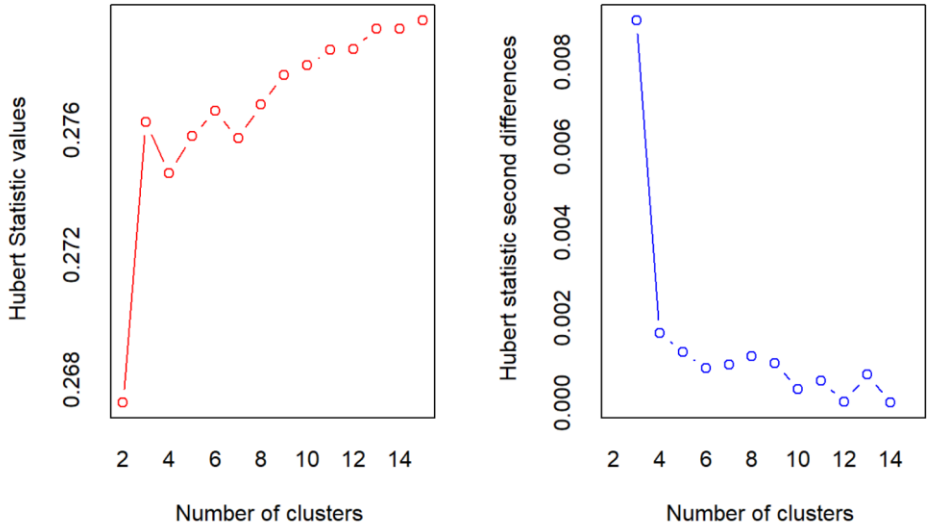


2. k-mean 군집화 적용– k-means clustering & 제곱합(ss)을 통한 적절한 군집의 수 예측

NbClust 통한 적절한 군집의 수 예측

```
# Determine K
nc <- NbClust(datav2[2:3], min.nc=2, max.nc=15, method="kmeans")
par(mfrow=c(1,1))
barplot(table(nc$Best.n[1,]),
        xlab="Numer of Clusters", ylab="Number of Criteria",
        main="Number of Clusters Chosen")
```

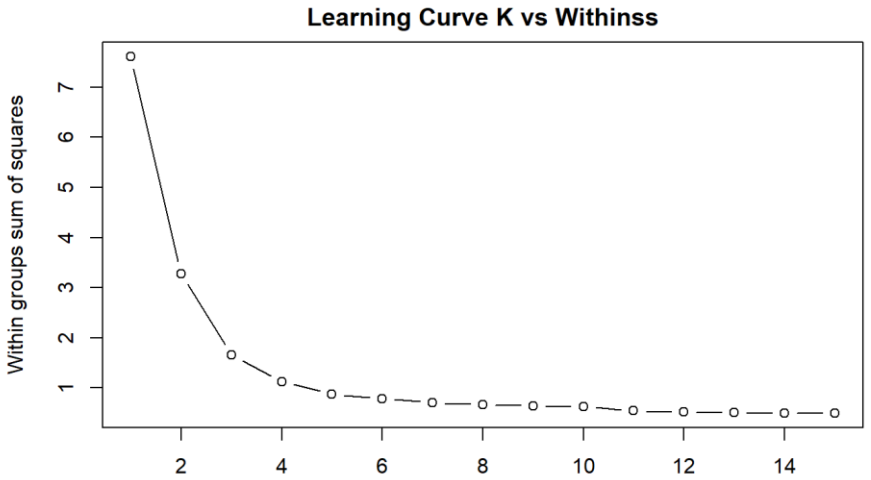




결과적으로 그래프에서 기울기가
급격하게 변하는 지점의 k값을 선택,
따라서 k=2로 설정

2. k-mean 군집화 적용– k-means clustering & 제곱합(ss)을 통한 적절한 군집의 수 예측

제곱합(ss) 통한 적절한 군집의 수 예측



학습 곡선 확인 결과 k=2를 넘는 경우,
즉 k=3 이상부터는
변화가 크지 않은 것을 확인

Result

```
분기당_매출_금액 연령대_20_매출_금액
       0.6526899            0.8031921
       0.1283586            0.2629680
```

성능 평가 (k=2일 때)

```
> datav2.kmeans[["betweenss"]]
[1] 4.334281
> datav2.kmeans[["withinss"]]
[1] 0.9602168 2.3185022
```

Betweenss와 withinss 측정하여 적절하게
군집화 되었음 확인할 수 있었음

2.k-mean 군집화 적용 – 최적화

최초의 중심 위치를 변경하여 여러 번 시도하여 최적화
: 시드 1,2,3,4 인 경우를 나누어서 확인

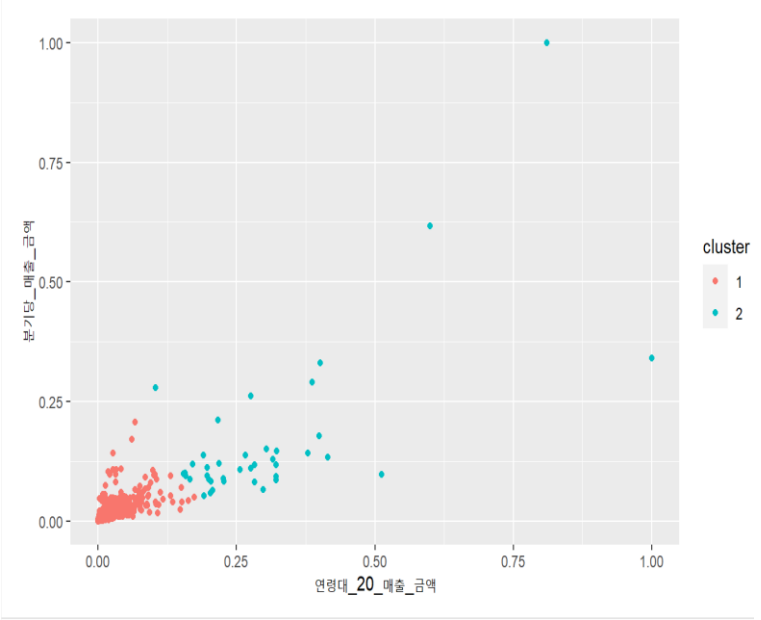
1. K=2인 경우

```
set.seed(1)
datav2.kmeans <- kmeans(datav2[2:3], centers = 2, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster <- as.factor(datav2.kmeans$cluster)

set.seed(2)
datav2.kmeans <- kmeans(datav2[2:3], centers = 2, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster <- as.factor(datav2.kmeans$cluster)

set.seed(3)
datav2.kmeans <- kmeans(datav2[2:3], centers = 2, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster <- as.factor(datav2.kmeans$cluster)

set.seed(4)
datav2.kmeans <- kmeans(datav2[2:3], centers = 2, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster <- as.factor(datav2.kmeans$cluster)
```



K=2인 경우 seed에 따른 결과가 크게 다르지 않음 확인

2.k-mean 군집화 적용 – 최적화

군집이 3개일 경우 시드에 따른 clustering 결과 차이를 파악해봄
K=2일 경우 차이가 크지 않아 단순 비교를 위하여 진행

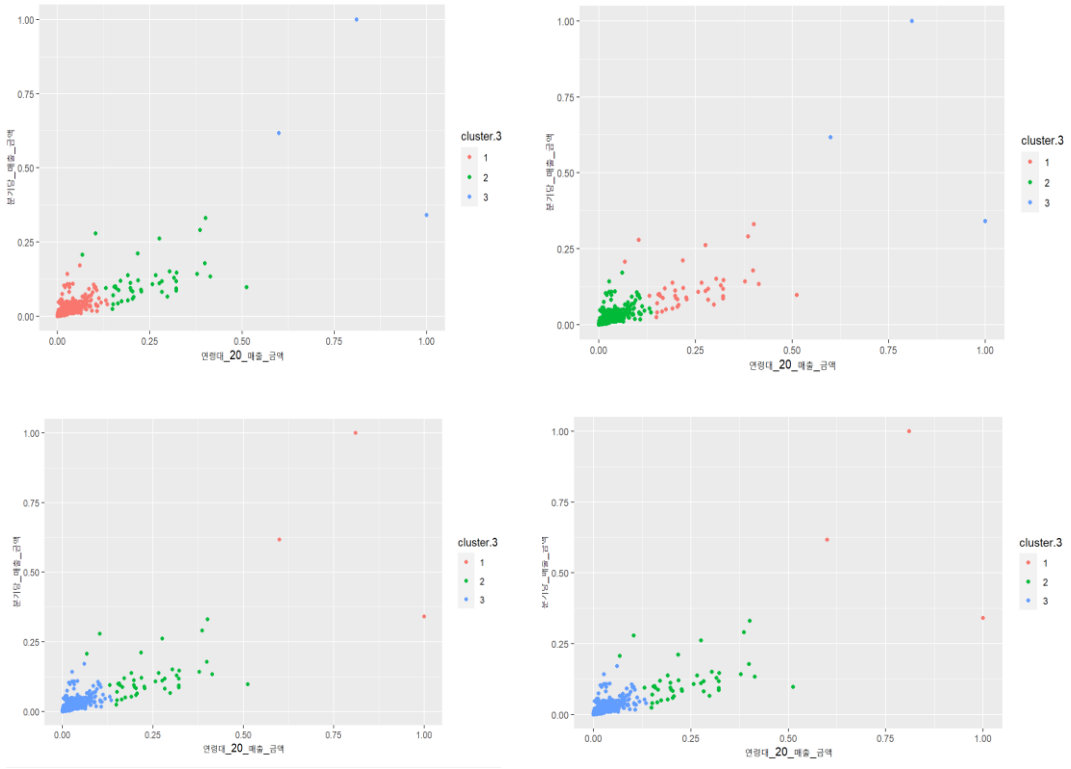
2. K=3인 경우

set.seed(1)
```
datav2.kmeans <- kmeans(datav2[2:3], centers = 3, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster.3 <- as.factor(datav2.kmeans$cluster)
```

set.seed(2)
```
datav2.kmeans <- kmeans(datav2[2:3], centers = 3, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster.3 <- as.factor(datav2.kmeans$cluster)
```

set.seed(3)
```
datav2.kmeans <- kmeans(datav2[2:3], centers = 3, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster.3 <- as.factor(datav2.kmeans$cluster)
```

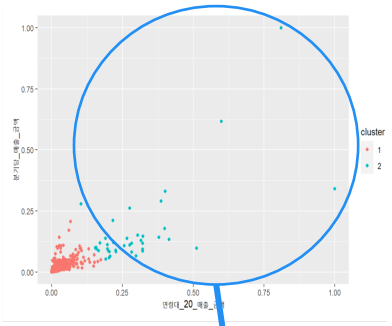
set.seed(4)
```
datav2.kmeans <- kmeans(datav2[2:3], centers = 3, iter.max = 10000)
datav2.kmeans$centers
datav2$cluster.3 <- as.factor(datav2.kmeans$cluster)
```



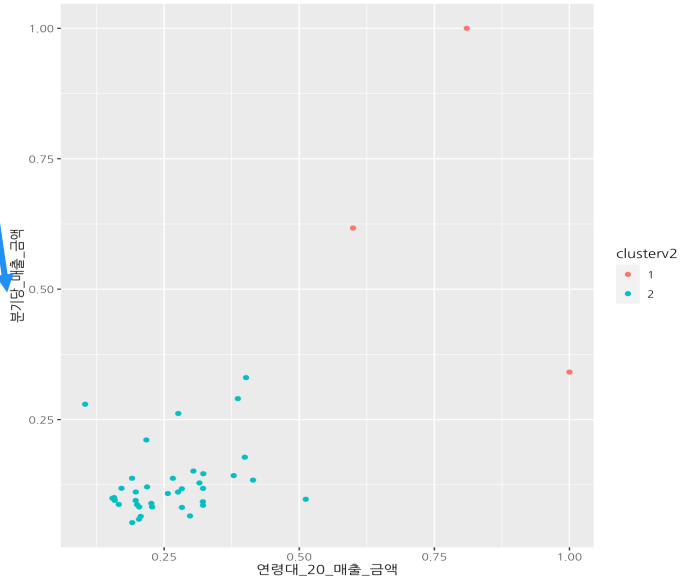
3.k-mean 군집화 적용 – 2차 분석

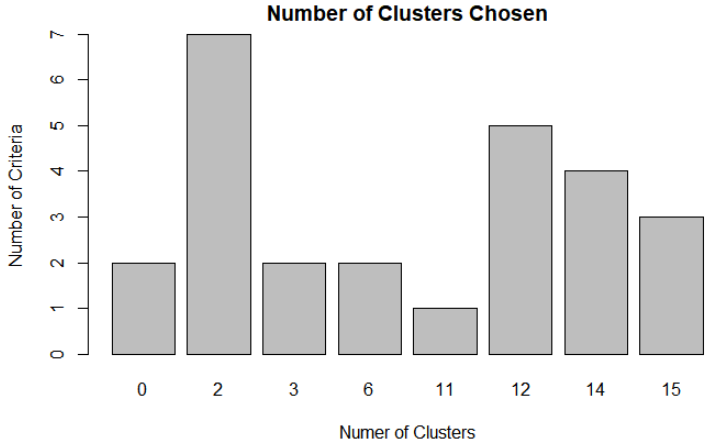
앞서 확인한 결과를 바탕으로 군집이 2개인 경우에 대해서 subset 만들고 다시 분석 진행

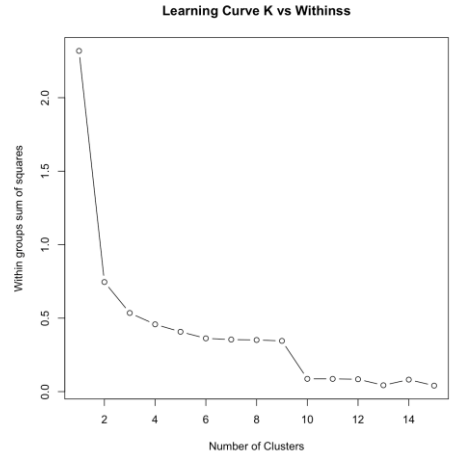
1차 분석



2차 분석



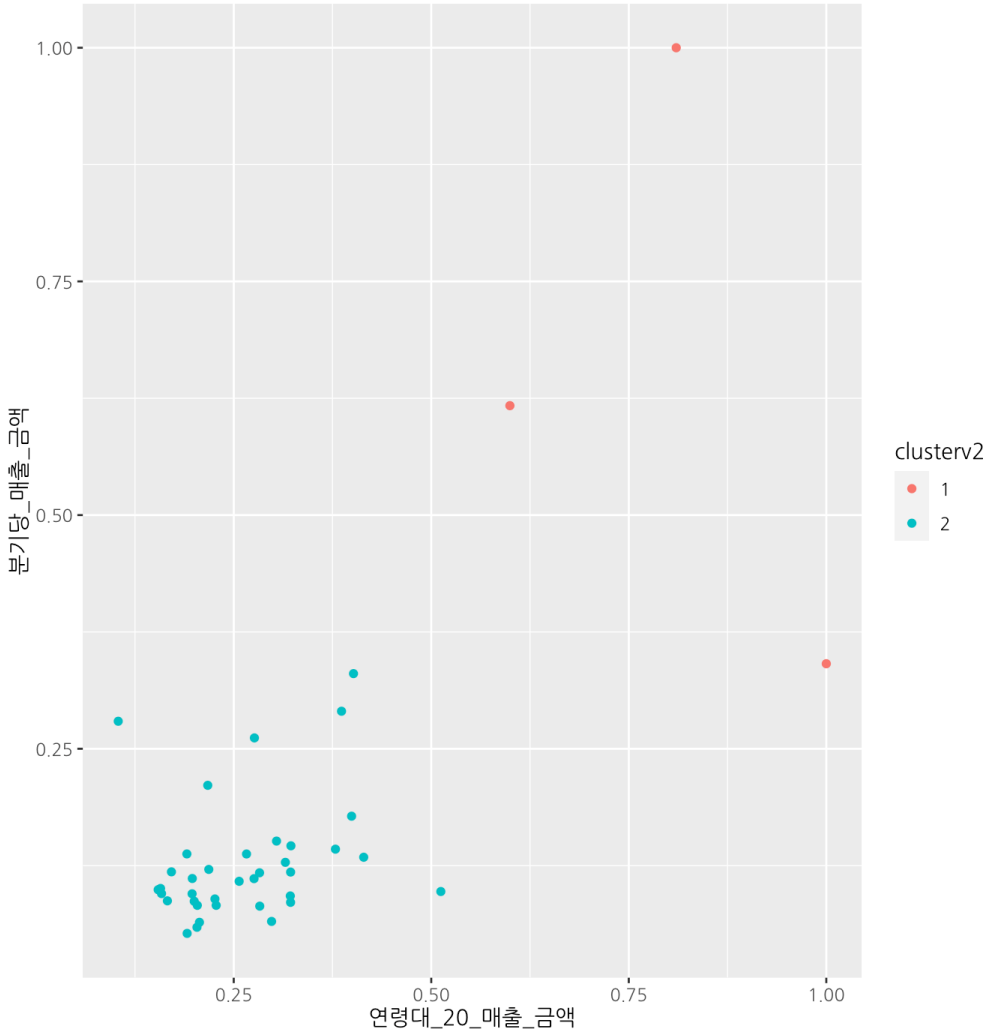




- 2차 분석에 사용하는 subset 또한 2개의 군집이 최적의 수로 결정
- 이를 기반으로 군집화를 다시 진행

3.k-mean 군집화 적용 – labeling 설명 시도

Labeling 설명



	상권_코드_명	분기당_매출_금액	연령대_20_매출_금액	cluster	cluster.3	clusterv2
34	강남역	0.3409895	1.0000000	2	2	1
295	노량진역(노량진)	0.6170802	0.5995334	2	2	1
1247	용산전자상가(용산역)	1.0000000	0.8100430	2	2	1

	상권_코드_명	분기당_매출_금액	연령대_20_매출_금액	cluster	cluster.3	clusterv2
3	가락시장	0.27944895	0.1038074	2	3	2
5	가로수길	0.08568233	0.3219370	2	3	2
10	가산디지털단지	0.33039944	0.4015714	2	3	2
22	강남 마이스 관광특구	0.14616983	0.3224014	2	3	2
81	건대입구역(건대)	0.06528695	0.2979027	2	3	2

☑ 이 plot에서 군집 2에 속하는 데이터포인트들의 경우 상권의 규모가 군집 1에 속하는 상권에 비하여 큰편은 아니지만, 20대의 매출이 전체 규모에 비해서 높다고 판단할 수 있다.

☑ 군집 1에 속하는 상권(강남역, 노량진, 용산전자상가(용산역)) 상권의 경우 상권의 규모도 크지만, 상권의 매출에서 20대의 매출이 차지하는 비율이 매우 높은 상권이라고 할 수 있다.

PPT Contents

01

데이터 탐색 및 전처리

데이터 설명, 전처리,
변수 선택, 극단값 처리, 정규화

데이터 소개

02

계층적 군집화

덴드로그램 형성,
시각적 기준 군집 나누기

군집화

03

K-mean 군집화 적용

적절한k군집의 수,
군집 최적화, 성능평가,
각 군집에 대한 설명

군집화

04

A

군집화 비교

k-mean과 다른 군집화를 비교

결론 및 비교

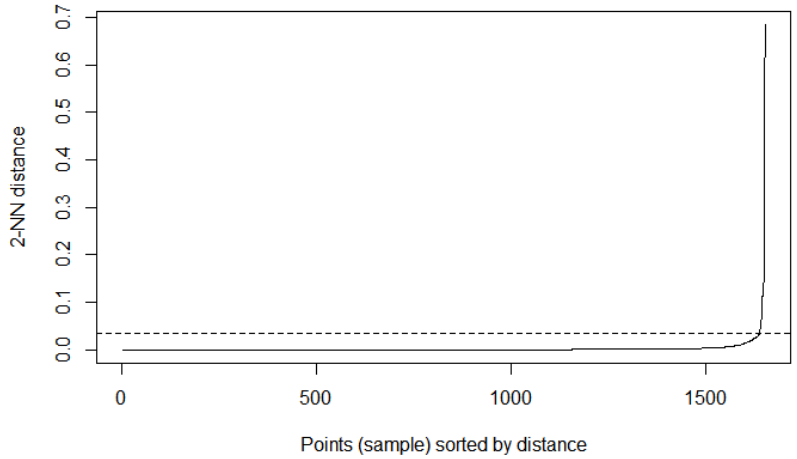
4. DBSCAN 적용 및 k-mean과 비교

최적 eps 찾기

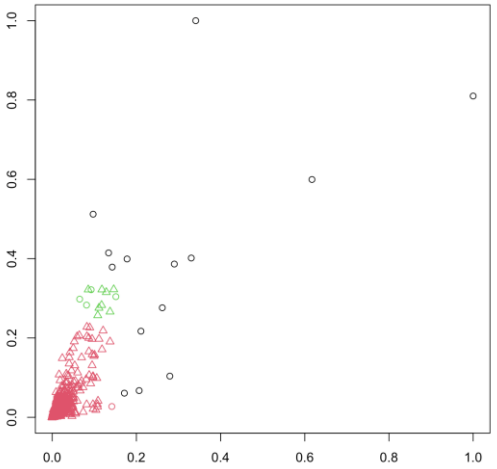
앞선 군집 분석에서 2개의 군집으로 나누는 것이 좋다고 결론
k=2일 때 기울기가 급격히 변하는 eps를 사용

아래 그래프에서 기울기가 급격하게 변하는 지점의 eps = 0.035

Minpoint는 2차원 평면에서는 4개로 설정하는 것이 좋음



군집화 결과

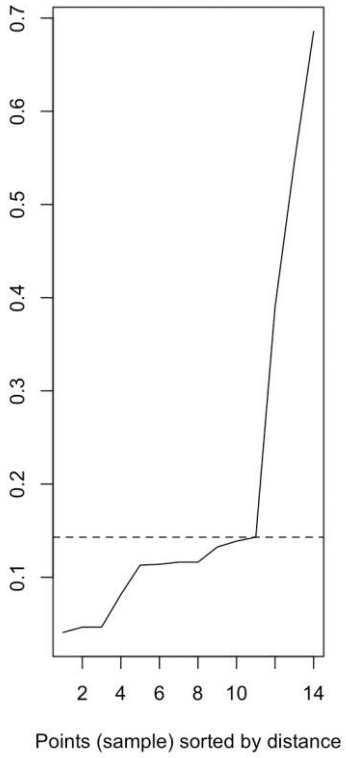



```
dbscan Pts=1651 MinPts=4 eps=0.035
         0    1  2
border  14    1  4
seed     0 1624  8
total   14 1625 12
```

- 빨간 세모의 경우(1번 군집) 상권의 규모가 작고, 20대에게도 유명하지 않은 상권임을 알 수 있음
- 검은 동그라미의(0번 군집)의 경우 섞여 있어 제대로된 군집화가 되었다고 보기 어려움 →다시 뽑아서 군집화 실시

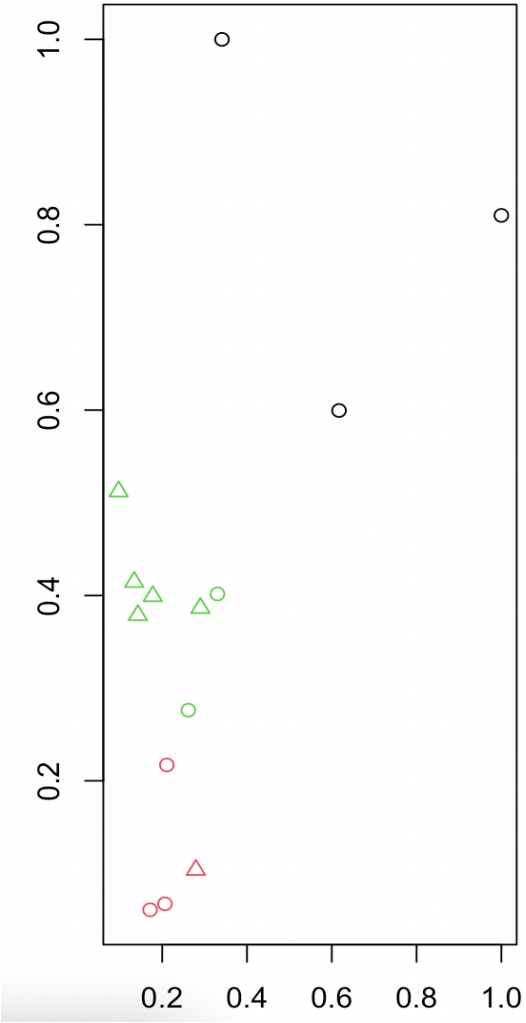
4. DBSCAN 적용 및 k-mean과 비교 - 2차분석

사용할 eps 다시 최적화



기울기가 급격히 변하는 점
K=2 → eps =0.143

결과



0번 군집(검은 동그라미) -> 규모가 큰 상권이면서 동시에 20대에게도 유명한 상권

1번 군집(빨간색) -> 규모도 비교적 적고, 20대에게 인지도가 적은 상권

2번 군집(초록색) -> 규모가 크지만 20대에게는 0번에 비하여 인지도가 없는 상권

	상권_코드_명	분기당_매출_금액	연령대_20_매출_금액	cluster	cluster.3	dbcluster
34	강남역	0.3409895	1.0000000	2	2	0
295	노량진역(노량진)	0.6170802	0.5995334	2	2	0
1247	용산전자상가(용산역)	1.0000000	0.8100430	2	2	0

	상권_코드_명	분기당_매출_금액	연령대_20_매출_금액	cluster	cluster.3	dbcluster
3	가락시장	0.2794490	0.10380741	2	3	1
404	독산동 우시장	0.2065990	0.06726503	1	3	1
498	마포농수산물시장	0.1714760	0.06081709	1	1	1
1392	종로·청계 관광특구	0.2108944	0.21713078	2	3	1

	상권_코드_명	분기당_매출_금액	연령대_20_매출_금액	cluster	cluster.3	dbcluster
10	가산디지털단지	0.33039944	0.4015714	2	3	2
431	동대문패션타운 관광특구	0.17787788	0.3991772	2	3	2
533	영등 남대문 북창동 다동 무교동 관광특구	0.29016452	0.3864306	2	3	2
1183	영등포역(영등포)	0.13395209	0.4145078	2	3	2
1331	잠실 관광특구	0.14260891	0.3787373	2	3	2
1393	종로3가역	0.26160996	0.2761664	2	3	2
1580	홍대입구역(홍대)	0.09727499	0.5121286	2	3	2

4. DBSCAN 적용 및 k-mean과 비교

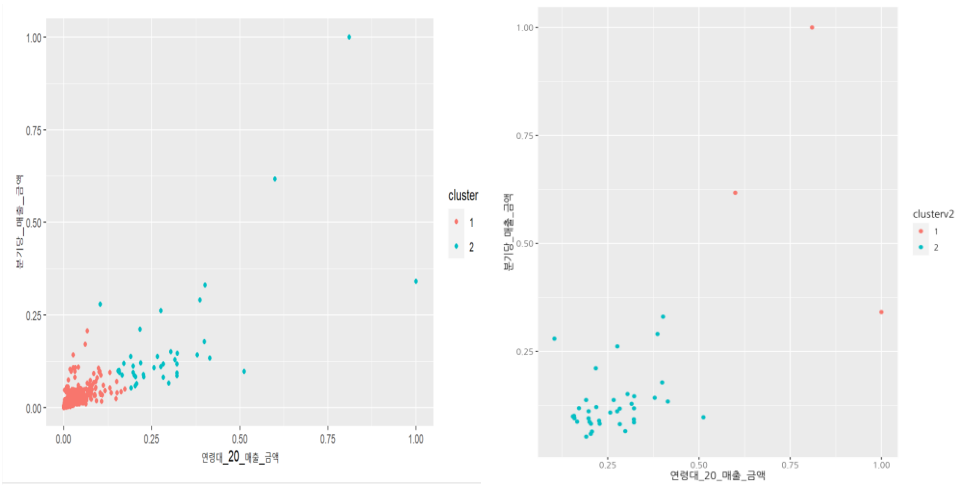
K-mean과 DBSCAN의 비교

- K-mean의 경우 초기 데이터 포인트에 따라 분석이 매번 다르게 나올 수 있음
  - 하지만 k의 수가 적으면 그 변화가 크지 않은 것을 확인

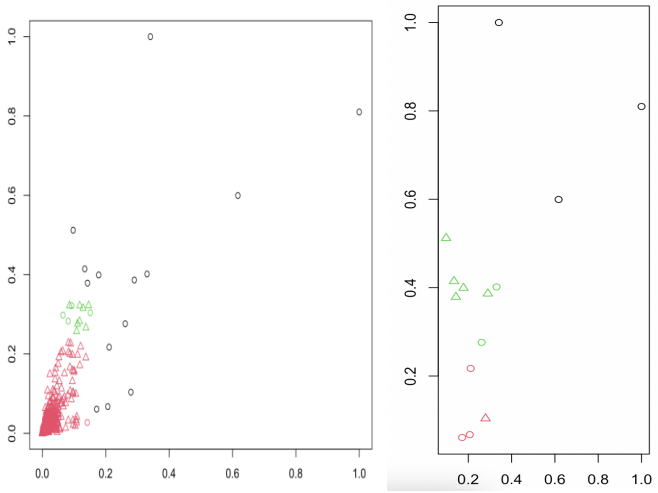
- DBSCAN의 경우 ε- 거리를 최적화 하는 방식이 명확하지 않음
  - ε 값을 얼마로 설정하냐에 따라 군집의 개수가 정해짐 (K-mean의 경우 군집의 수를 정하고 시작)

- 현재 데이터의 특성 상 특정 지역에 데이터 포인트가 몰려 있음
  때문에 밀도 기반 분석인 DBSCAN은 처음 진행한 분석에서 군집화 성능이 좋지 않다고 판단됨

K-mean



DBSCAN



데이터 사이언스 기초

12주차 과제

이상으로 발표를 마치겠습니다.

감사합니다.

PRESENTATION END