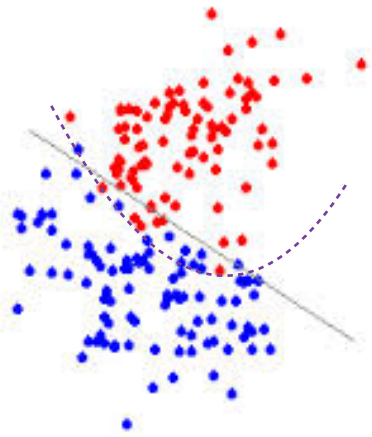
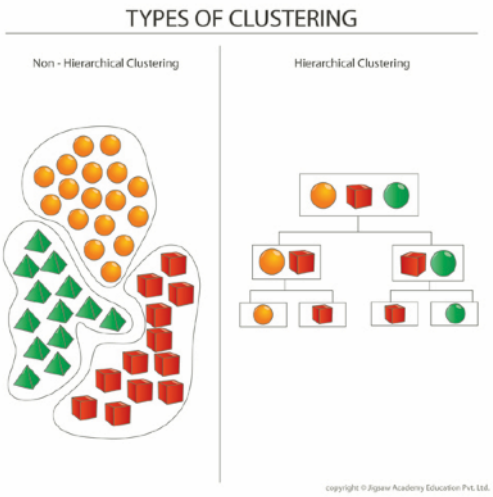
1. 개요

군집분석에서는 개체의 그룹에 대한 정보 없이 사전에는 구분이 없이 동일하나 분석 후 나누어짐(●, ■, ▲) 유사성이 가까운 개체들끼리 계층적으로 묶어 가거나 군집의 개수를 정하여 군집의 중심점을 이용하여 개체를 군집화 하는 방법이다. 판별분석은 자료 수집 시 이미 그룹이 나누어져 있어(●, ●) 이를 가장 잘 판별하는 판별규칙을 도출하여 새로운 개체의 군집을 판별하는 방법이다. 곡선의 판별규칙을 도출하는 것은 쉽지 않다.

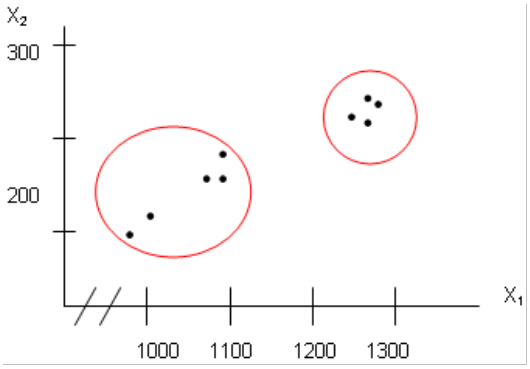




마케팅 담당자가 자사 고객들을 분류하기 위하여 나이, 학력, 소득, 결혼 상태, 자녀 수, 직업 등에 대한 정보를 수집하였다. 이를 이용하여 고객들을 집단(cluster)으로 분류하고자 할 때 사용되는 다변량 분석 방법이 군집 분석 (Clustering Analysis)이다. 판별 분석과 군집 분석의 다른 점은 판별 분석은 조사된 데이터에 개체의 집단 변수가 이미 포함되어 있으나 군집 분석은 개체들에 대해 측정된 변수에 의해 집단을 분류하게 되므로 집단의 개수와 집단의 종류(이름)는 분류 후 정해지게 된다. 즉 군집 분석은 개체들이 분석 전에는 어떤 그룹에 속하는지 알려져 있지 않다. 군집 분석은 grouping 혹은 classification이라 불리기도 한다.

개체를 분류한다? 아래 산점도와 같이 측정 변수가 2개라면 거리가 가까운 개체들끼리 묶으면 될 것이다. 12개의 개체가 아마 2개 군집(집단)으로 분류될 것이다. 두 개체간의 Euclidean 거리를 계산(측정)하고 거리가 가까운(유사성이 높음) 개체끼리 묶으면 된다.

이렇게 개체를 분류하는데 도움을 주는 그림은 1)측정 변수가 2개인 경우 산점도(scatter plot) 2)측정 변수가 3개인 경우는 Bubble Plot 3)측정 변수가 4개 이상이 경우는 주성분 변수를 이용하여 2-3개의 주성분에 대한 산점도가 있다. 그러나 이런 그래프들을 이용하여 시각적 분류는 가능하지만 실제 유사성에 의해 개체를 분류하려면 유사성이 값으로 계산되어야 한다.



판별분석과 비교

	판별분석	군집분석
분석초기	 개체는 이미 집단 분류되어 있음	 개체의 집단 구별이 없음
목적	새로운 개체를 분류 ▶개체를 잘 판별할 수 있는 판별 변수 선택이 관건이다.	위의 개체들을 분류 ▶개체들의 특성을 나타내는 변수들을 선택하는 것이 관건
분석절차	(1)판별 분석 방법 선택 →오분류가 적은 방법 사용 • Fisher method (판별 변수 선택 방법 이용, 유의 수준을 다소 높게 설정) • K Nearest Discriminant Analysis • Logistic Regression 판별 분석(변수 선택 방법 사용, 유의수준 다소 높게 설정) (2)개체 분류 경향을 파악하기 위하여 판별변수들에 산점도(by 그룹)를 그린다. 판별변수가 2개 이상이면 주성분 분석을 이용하여 산점도를 그리면 된다. (3)최종적으로 구해진 판별식에 의해 새로운 개체를 (□△○) 중 하나로 분류한다.	(1)개체 분류 방법을 선택한다. • Nearest neighbor • Furthest neighbor • Centroid neighbor • Average neighbor • Ward's minimum variance (2)군집의 개수를 정한다. • CCC • Pseudo Hotel ling's • Tree Diagram (3)개체 분류가 잘 되었는지 알아보기 위하여 산점도를 그린다. 변수가 3개 이상인 경우는 주성분 분석을 이용하여 산점도 그린다. 군집 결과는 개체 분류 방법과 군집 개수에 의해 결정된다. (4)각 군집에 적절한 이름을 붙인다.

장단점

장점

- 탐색적인 기법: 주어진 자료의 내부구조에 대한 사전정보 없이 의미 있는 자료구조를 찾아낼 수 있음
- 다양한 형태의 데이터에 적용가능: 유사성(거리)만 정의되면 모든 종류(텍스트 데이터)의 자료에 적용할 수 있음.
- 분석방법 적용 용이성: 자료의 사전정보를 필요로 하지 않아서 누구나 쉽게 분석할 수 있음

단점

- 가중치와 거리 정의: 가중치와 거리를 어떻게 정의하는 가에 따라 군집분석의 결과가 아주 민감하게 반응함
- 초기 군집 수의 결정이나, 군집 개수 결정이 쉽지 않음
- 결과의 해석이 어려움: 찾아진 군집이 무엇을 의미 하는지 데이터만을 이용해서는 알 수가 없는 경우가 많음 - 주성분 분석을 이용하여 개체 집단 표현, 인구학적 특성에 의해 개체 특성 파악

활용

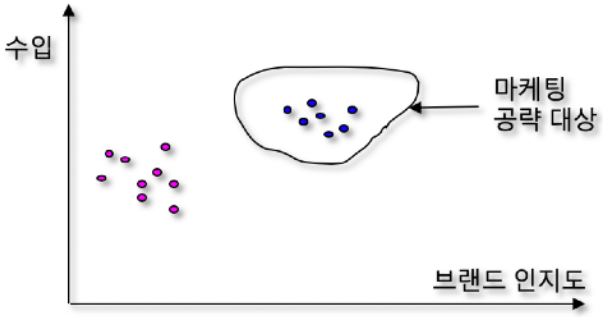
시장세분화 (마케팅)

변량	전체	추종자	EB	기본 기능
나이	28.6	27.2	19.3	53.0
교체의사	.429	.333	.833	.000
고등학생	.238	.0833	.667	.000
영업직	.238	.333	.000	.333
성별	.619	.750	.333	.667
대학생	.143	.167	.167	.000
전문직	.286	.333	.167	.333
거주지	1.33	1.50	1.17	1.00
생산직	.0476	.0833	.000	.000

- 구매태도, 구매성향, 매체사용 습관 등과 같은 특성에 있어서 공통적인 특성을 공유하는 사람, 시장,조직 등의 대상들의 군집을 발견하여 시장 세분화에 활용
- 소비자를 분류함으로써 소비자 행동 이해 가능 : 소비자들을 특성(측정 변량)에 따라 분류하여 내재된 특성 파악

- 잠재적인 신제품 기회 발견 : 전체 시장에서의 경쟁상황과 경쟁관계에 있는 상표나 기업을 속성에 따른 군집 도출.
- 개체 데이터를 일반적이고 다루기 쉽게 축약 : 표본추출을 위해 조사대상 지역이나 소비자들을 구분해야 하는 경우: 군집분석에 의해 서로 유사한 특성을 가진 대상끼리 묶어 계층이 구성되면 표본추출의 작업이 용이하고 정확성도 유지할 수 있음

고객 세분화



- 고객이 기업의 수익에 기여하는 정도를 통한 고객세분화
- 우수고객의 인구통계적 요인, 생활패턴 파악: 개별고객에 대한 맞춤관리
- 고객의 구매패턴에 따른 고객세분화
- 신상품 판촉, 교차 판매를 위한 표적집단 구성

유사성과 비 유사성

기호

군집변수 p개, $\underline{x} = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_p \end{pmatrix}$, i-번째 개체 관측치 $\underline{x}_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \cdots \\ x_{pi} \end{pmatrix}$

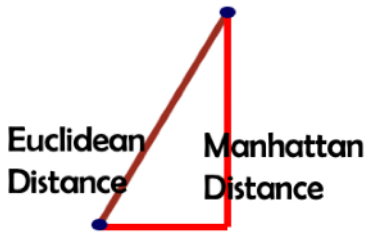
Euclidean 거리

두 개체 사이의 유사 정도를 거리로 표현할 수 있다. 거리가 멀면 유사성(similarity)이 떨어진다. 다음 식은 i-번째 개체와 j-번째 개체의 Euclidean 거리이다. 가장 많이 사용

$$d_{ij} = (\underline{x}_i - \underline{x}_j)'(\underline{x}_i - \underline{x}_j)^{1/2}$$

Manhattan 거리

직선 이동 거리, 이상치 비중 약해짐 $d_{ij} = \sum_{k}^{p} |x_{ki} - x_{kj}|$



표준화 Euclidean 거리

개체들에 대해 측정된 변수들이 단위가 다르거나 분산이 다를 경우 변수를 표준화한 후 거리를 구하는 것이 더 적절하다. 다음 식은 i-번째 개체와 j-번째 개체의 표준화 Euclidean 거리이다.

$$d_{ij} = (\underline{z}_i - \underline{z}_j)'(\underline{z}_i - \underline{z}_j)^{1/2}, \ \underline{x}_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \cdots \\ x_{pi} \end{pmatrix} - \begin{pmatrix} EX_1 \\ EX_2 \\ \cdots \\ EX_p \end{pmatrix}$$

Mahalanobis (Pearson) 거리

다음 식이 번째 개체와 번째 개체의 Mahalanobis 거리이고 Σ는 within 군집 분산-공분산 행렬 추정치이다. 거리를 이와 같이 정의하는데 문제점은 개체를 다 분류하기 전에는 Σ의 추정치를 구할 수 없음

$$d_{ij} = (\underline{x}_i - \underline{x}_j)'\Sigma^{-1}(\underline{x}_i - \underline{x}_j)^{1/2}$$

군집분석 - 계층적 방법

유사성이 가까운 순서대로 개체들을 묶어(군집화) 가는 방법으로 single-linkage clustering 방법이 이 방법 중 가장 효율적이다. Neighbor Method은 single-linkage clustering 방법 중 하나로 다음 순서에 의해 개체를 분류한다.

(1) 처음에는 개체의 수(n)만큼의 군집이 있다. 예를 들어 개체 6개가 있고 다음은 각 개체 간 Euclidean 거리(유사성)를 계산한 표이다. 처음에는 군집은 6개이다.

	1	2	3	4
1		0.1	0.7	0.2
2			0.4	0.6
3				0.3
4				

개체 (1, 2) 간 거리(유사성)은 0.1, (2, 3) 개체 유사성은 0.4이다. (2, 3) 개체 간 거리는 (3, 2) 개체 거리와 동일하므로 대각 행렬 형태이다.

(2) 유사성이 가장 가까운(거리가 가장 가까운) 개체를 군집으로 묶는다. 예제에서는 (1, 2)가 묶인다.

	(1, 2)	3	4
(1, 2)		?	?
3			0.3
4			

이제 (1, 2)와 3, 4간 거리를 어떻게 정의할 것인가?

(3) 개체가 군집으로 묶이면 개체와 새로 만들어진 군집과의 유사성을 계산한다. 군집과 군집(혹은 개체)의 유사성(거리)을 측정하는 방법은 다음 5가지가 있다.

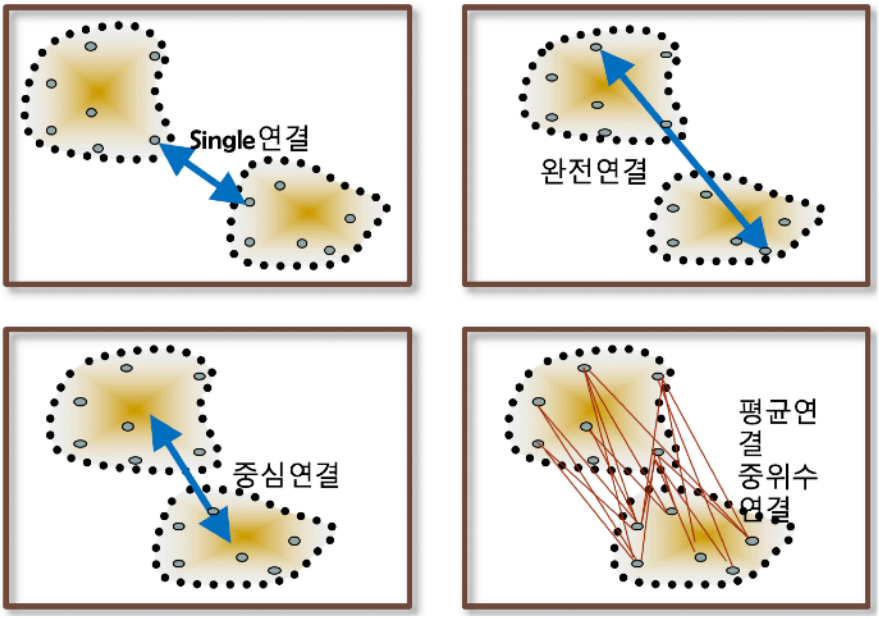
① Nearest neighbor: 두 군집의 각 개체 중 가장 가까이 있는 개체의 거리(유사성)

② Furthest neighbor: 두 군집의 각 개체 중 가장 멀리 있는 개체의 거리

③ Centroid neighbor: 군집의 평균 간의 거리

④ Average neighbor: 한 군집의 개체와 다른 군집 개체들의 각 거리 평균

⑤ Ward's minimum variance: 군집의 평균간 거리를 각 군집의 개체 개수의 역의 합으로 나눈 제곱근을 구한 거리이다.



Nearest, Furthest, Centroid neighbor, Average neighbor, Ward's minimum variance 중 어떤 방법을 사용하는 것이 좋은가? Nearest 방법은 개체간의 거리가 가까워 개체를 묶는 경향이 있어 군집의 수가 줄어들고 Furthest는 군집간 거리를 최소화 하는 경향이 있어 개체 수가 적은 군집을 얻게 한다. 그러므로 각 방법의 장단점이 있으므로 2-3개 방법을 사용하여 개체의 군집화가 보다 잘되는 방법을 선택하는 것이 좋다. 가장 많이 사용하는 방법은 Average neighbor 방법이다.

(4) 다음은 Nearest neighbor 방법에 의해 개체를 군집화 하는 과정이다.

1과 3의 거리는 0.7, 2와 3의 거리는 0.4이므로 (1, 2)와 3의 거리는 0.4가 된다. 1과 4의 거리는 0.2이고 2와 4의 거리는 0.6이므로 작은 거리 0.2가 (1, 2)와 4의 거리이다.

	(1, 2)	3	4
(1, 2)		0.4	0.2
3			0.3
4			

(1, 2)의 거리는 0.4이고 3과 4의 거리는 0.3이므로 (1, 2, 4)와 3의 거리는 0.3이 된다.

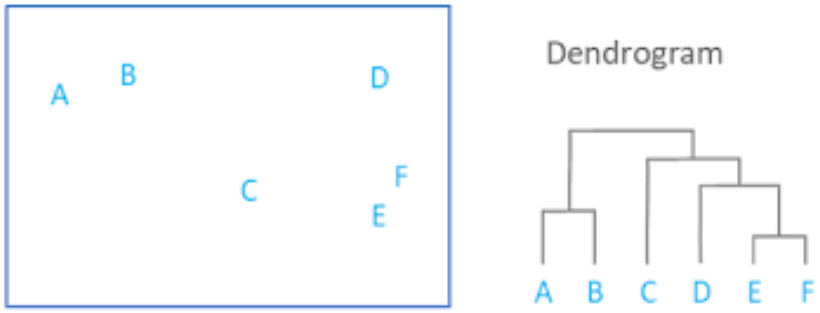
	(1, 2, 4)	3
(1, 2, 4)		0.3
3		

군집 개수

군집의 개수를 몇 개로 하면 좋은가? 그래프적 방법으로는 Tree diagram이 있고 검정 통계량을 이용하는 방법으로는 Hotel ling's 검정이나 Cubic Clustering Criterion 방법을 이용하면 된다.

계층적 나무 다이어그램 (Dendrogram)

개체의 유사성이 가장 가까운 개체부터 군집화 되는 과정을 보여주는 그림이다. 가지의 길이(높이)는 개체 간 거리의 상대적 크기이다. (E, F) 개체 -> (A, B) -> ((E, F), D) -> (((E, F), D),C) -> 모든 개체가 하나로 묶인다. 만약 군집 개수를 2개로 한다면 (A, B), (C, D, E, F) 이렇게 군집화 된다.



Pseudo Hotelling's 검정

Hotel ling's T^2검정 통계량은 두 집단 다변량 평균의 차이를 보는 통계량이다. 이를 군집 분석에 이용하는데 이와 유사 개념의 검정 통계량을 이용하여 개체의 군집간 평균의 차이가 유의하지 않으면 두 군집을 합치고 유의하면 군집 그대로 유지하는 방법이다.

Cubic Clustering Criterion

Searle(1983)이 제안한 방법으로 군집의 개수와 CCC(Cubic Clustering Criterion)의 산점도를 그려 CCC의 값이 3 이상이고 최대 값인 경우 그 때의 군집의 개수가 적당하다.

군집분석-계층적방법 (분석예제)

예제 데이터 LPGA2008.csv

2008년 미국 LPGA 골프선수들의 능력 측정 변수 (비거리, 페어웨이 안착율, 샌드(벙커) 세이버, 그린 적중률, 버팅 수), 상금, 출전 라운드 수를 조사한 데이터이다. 상금랭킹 40위 이내 선수들을 골프 관련 능력에 의해 군집화 하시오.

군집변수 6개 : 평균 비거리(+), 페어웨이 안착율(+), 그린 적중률(+), 평균 버팅 수(-), 샌드회수(-), 샌드(벙커) 세이버(+)

+ 의미 : 변수 값이 클수록(높을수록) 선수들의 능력은 높다.

- 의미 : 반대 개념임

데이터 불러오기

```
lpga<-read.csv('http://203.247.53.31/Stat_Notes/example_data/
lpga2008.csv',fileEncoding='utf-8')
lpga.subset<-subset(lpga,rank(-lpga$상금)<=40)
names(lpga.subset)
dim(lpga.subset)
```

`subset(lpga,rank(-lpga$상금)<=40)` rank(-lpga$상금) 함수는 상금 크기 역순(- 옵션 사용)으로 정렬한 후 순위를 부여한다. subset(순위<=40)는 순위 40위인 선수 (행) 데이터만 사용한다.

```
> names(lpga.subset)
[1] "골퍼"              "평균_비거리"      "페어웨이_안착율" "그린_적중률"
[5] "평균_퍼팅수"      "샌드_회수"        "샌드_세이브"     "상금"
[9] "참가_라운드수"
> dim(lpga.subset)
[1] 40  9
```

군집변수 활용 개체 (유사성) 행렬 계산 dist() 함수

개체간 유사성(거리)을 계산할 때 변수의 단위 영향이 크므로 표준화 하는 것이 먼저이다.

`scale(데이터, center = TRUE, scale = TRUE)` 평균 0, 표준편차 =1

True 옵션은 default 의미하므로 사용하지 않으면 자동 처리된다.

`dist(x, method = "euclidean")` 개체의 거리 행렬

manhattan, minkowski 방법 적용도 가능하다.

```
lpga.d<-dist(scale(lpga.subset[2:7]),method='euclidean')
head(lpga.d,3)
```

40 개체(선수)의 쌍 유사성(거리) `> choose(40,2)` `[1] 780` 780개 계산되어 lpga.d에 저장되어 있음. (1, 2) 선수의 유사성은 2.93, (1, 3)은 2.49, (1, 4) 선수의 유사성은 3.51이다.

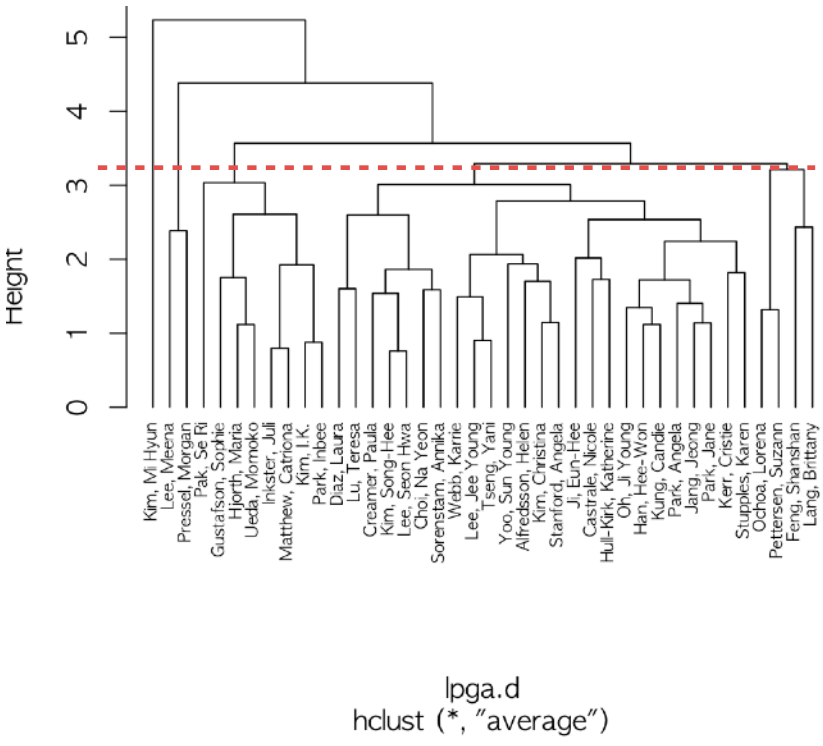
```
> head(lpga.d,3)
[1] 2.934475 2.482789 3.514831
```

덴드로그램 그리기

개체 유사성 연결방법(linkage) average 사용, hang=-1 맨 아래 행에 선수(개체) 이름이 출력된다.

```
lpga.hclust<-hclust(lpga.d,method='average')
plot(lpga.hclust,labels=lpga.subset$골퍼,hang=-1,cex = 0.6)
```

`"ward.D"`, `"ward.D2"`, `"single"`, `"complete"`, `"average"`, `"mcquitty"`, `"median"` `"centroid"`

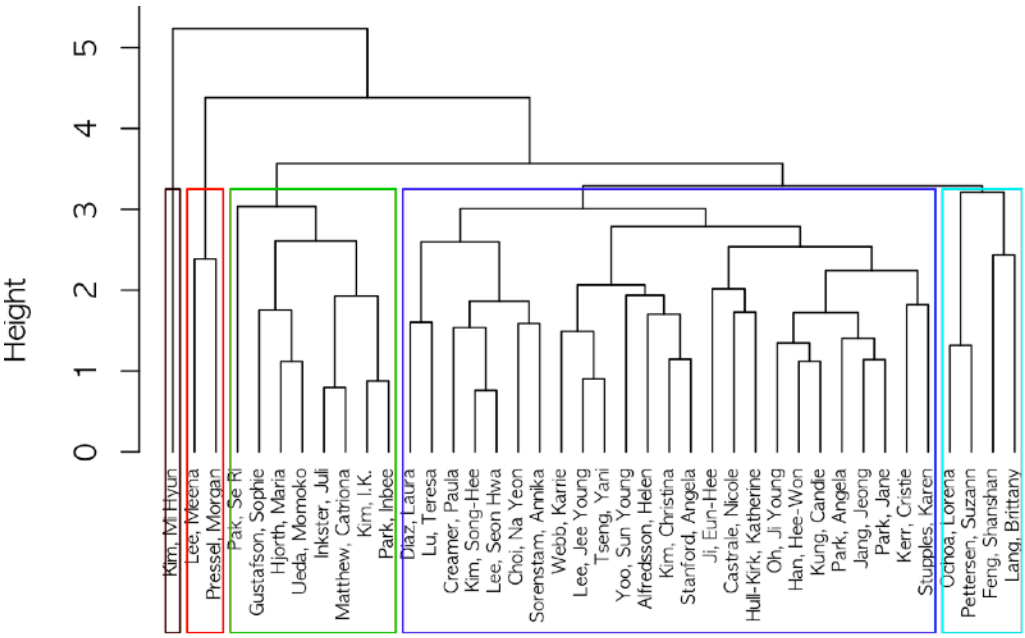


lpga.d
hclust (*, "average")

군집의 개수를 5개로 설정하자.

군집개수 결정 및 개체 군집 번호 부여

```
lpga.hclust<-hclust(lpga.d,method='average')
plot(lpga.hclust,labels=lpga.subset$골퍼,hang=-1,cex = 0.6)
rect.hclust(lpga.hclust,k=5,border=1:5) #border 컬러 색 지정
```




```
cluster<-cutree(lpga.hclust,k=5)
head(cluster,10)
```

```
> head(cluster,10)
 2 14 18 21 27 35 48 50 56 59
 1  1  1  1  1  2  3  1  3  1
```

군집 특성 파악

군집 평균, 표준편차

```
lpga.ca<-cbind(lpga.subset, cluster)
table(lpga.ca$cluster)
library(doBy)
```

```
> table(lpga.ca$cluster)

 1  2  3  4  5
25  4  8  1  2
```

=> 군집 결과 1군집만 25개 몰려 편중된 결과를 얻었음

평균 비거리 능력 : 2군집 > 3군집, 페어웨이 안착율 : 5군집 > 2군집, 그린적중률 : 2군집 > 1군집

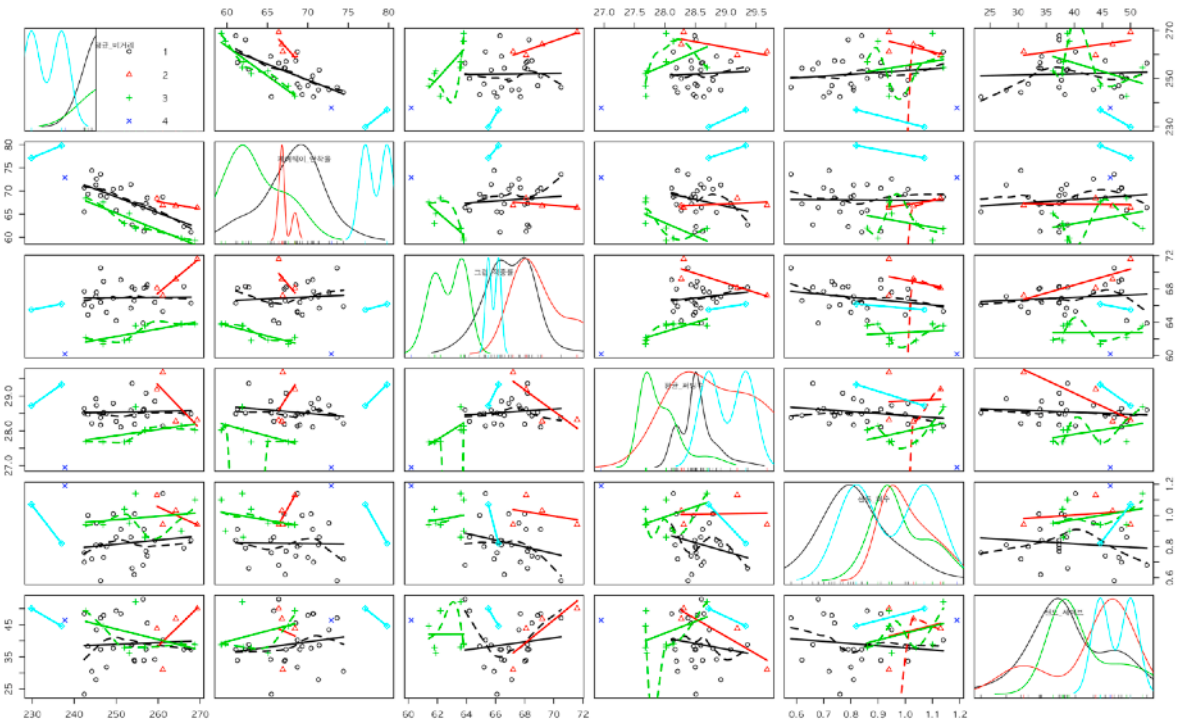
평균 퍼팅 수(-) : 4군집 > 3군집, 샌드회수(-) : 1군집 > 5군집, 샌드 세이브 : 5군집 > 4군집

```
  cluster 평균_비거리.mean 페어웨이_안착율.mean 그린_적중률.mean 평균_퍼팅수.mean
1       1         251.9280               68.0520          66.9600          28.5412
2       2         263.5750               67.1250          69.0250          28.8625
3       3         255.0625               63.3875          62.7625          27.9525
4       4         237.8000               72.9000          60.2000          26.9500
5       5         233.4000               78.4500          65.8500          29.0250
  샌드_회수.mean 샌드_세이브.mean 평균_비거리.sd 페어웨이_안착율.sd 그린_적중률.sd
1        0.82000           39.004       7.152361          3.6459018       1.6227549
2        1.01000           42.900       4.254703          0.8770215       1.9015345
3        0.98375           42.000       8.733668          3.3736108       1.0280877
4        1.19000           46.400             NA                 NA              NA
5        0.94500           47.300       5.091169          1.9091883       0.4949747
  평균_퍼팅수.sd 샌드_회수.sd 샌드_세이브.sd
```

군집변수가 많아 군집이름 부여가 쉽지 않다. 그러므로 시각화, 주성분 분석 활용한 차원 축소가 필요하다.

시각적 표현 - 산점도 행렬 : still not clear

```
library(car)
scatterplotMatrix(~평균_비거리+페어웨이_안착율+그린_적중률+평균_퍼팅수+샌드_회수
+샌드_세이브|cluster,data=lpga.ca,col=1:5)
```



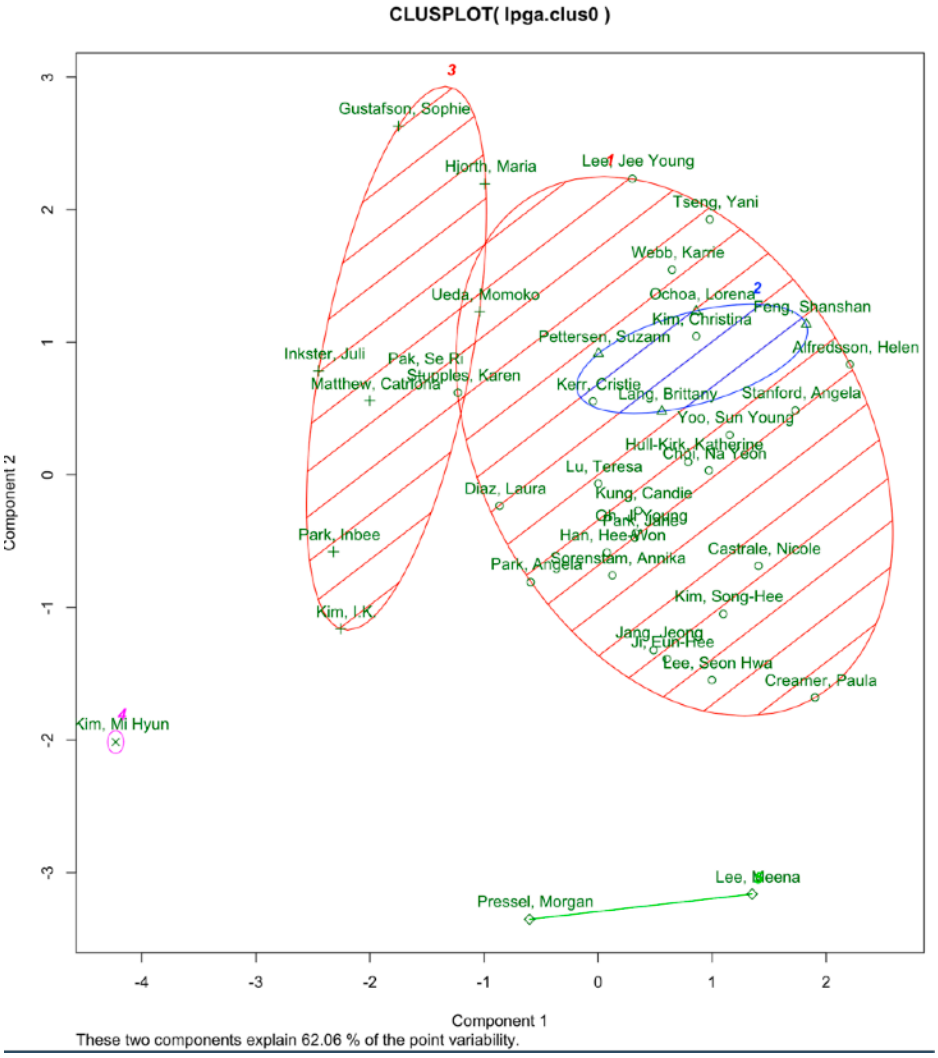
주성분분석 활용 - 함수 이용

다소 복잡한 과정을 거쳐 산점도에 선수 이름을 붙인다. 골퍼 이름변수와 군집변수 표준화 값을 열 결합하여 데이터프레임으로 만들고 rownames() 함수를 사용하여 골퍼 이름을 행 이름으로 대체한다.

```
library(cluster)
lpga.clus<-data.frame(lpga.subset$골퍼,scale(lpga.subset[2:7]))
rownames(lpga.clus)=lpga.subset$골퍼
lpga.clus0<-lpga.clus[2:7]
clusplot(lpga.clus0,cluster,shade=TRUE,lines=0,color=TRUE,labels=2)
```

4번 군집에는 김미현 혼자임, 5번 군집에는 프리셀과 이미나 두 선수 군집, Component 1, Component 2는 주성분을 의미함. 그러므로 주성분 분석을 실시하여 주성분의 이름 부여가 있어야 군집 이름 부여 가능

실제 군집2는 군집1 안에 내포되어 있어 두 군집은 동일하다고 할 수 있음



주성분 분석

```
lpga.pca<-prcomp(lpga.subset[2:7],scale=T)
lpga.pca$sdev^2 #고유값 출력
lpga.pca$sdev^2/sum(lpga.pca$sdev^2) #고유값 비율
lpga.pca$rotation #부하 출력
```

첫 2개 고유값 비율의 합은 62.06%로 위의 산점도 아래 변동 기여율 62.06%와 동일하다. 이는 위의 산점도 축이 주성분 1, 2로 만들어졌기 때문이다.

```
> lpga.pca$sdev^2 #고유값 출력
[1] 1.9208889 1.8024626 1.0146511 0.7110294 0.4095372 0.1414307
> lpga.pca$sdev^2/sum(lpga.pca$sdev^2) #고유값 비율
[1] 0.32014815 0.30041044 0.16910852 0.11850491 0.06825620 0.023
```

제1 주성분 : 그린적중율, 평균퍼팅수(-), 샌드회수 -> 정확성 성분, 정확한 타자는 그린 적중율이 높아 샌드에 빠지는 회수가 적을 것이고(-) 그러나 홀 가까이 붙이기 보다는 그린 온의 확률이 높은 것으로 판단 퍼팅 수는 높을 것임. 장타 선수는 홀 가까이 붙일 가능성이 높음. -> 정확성 (+)

제2 주성분 : 평균비거리, 페어웨이 안착율(-) -> 장타력 성분, 비거리가 많이 나간다는 것은 페어웨이 안착율이 낮을 가능성이 있으므로 반대 개념이 맞음 -> 장타력 성분 (+)

```
> lpga.pca$rotation #부하 출력
                          PC1          PC2         P
평균_비거리        0.08296889  0.680432765 0.28604147
페어웨이_안착율   0.12486882 -0.687649134 0.04128619
그린_적중률        0.60527444  0.035528088 0.35745398
평균_퍼팅수        0.58639791  0.002464521 0.07333627
샌드_회수         -0.48252383  0.078279357 0.27347729
샌드_세이브       -0.18567454 -0.238230864 0.84174482
```

군집 이름 부여

군집 3 : 장타 선수군

군집 1, 2 : 장타와 정확성 겸비한 선수군

군집 4 : 장타, 정확성 모두 낮은 선수군

군집 5 : 정확성 선수군

계층적 군집분석 사례 2 - 미국 도시 사회경제기후 환경 지표

http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/SMSA_USA.csv

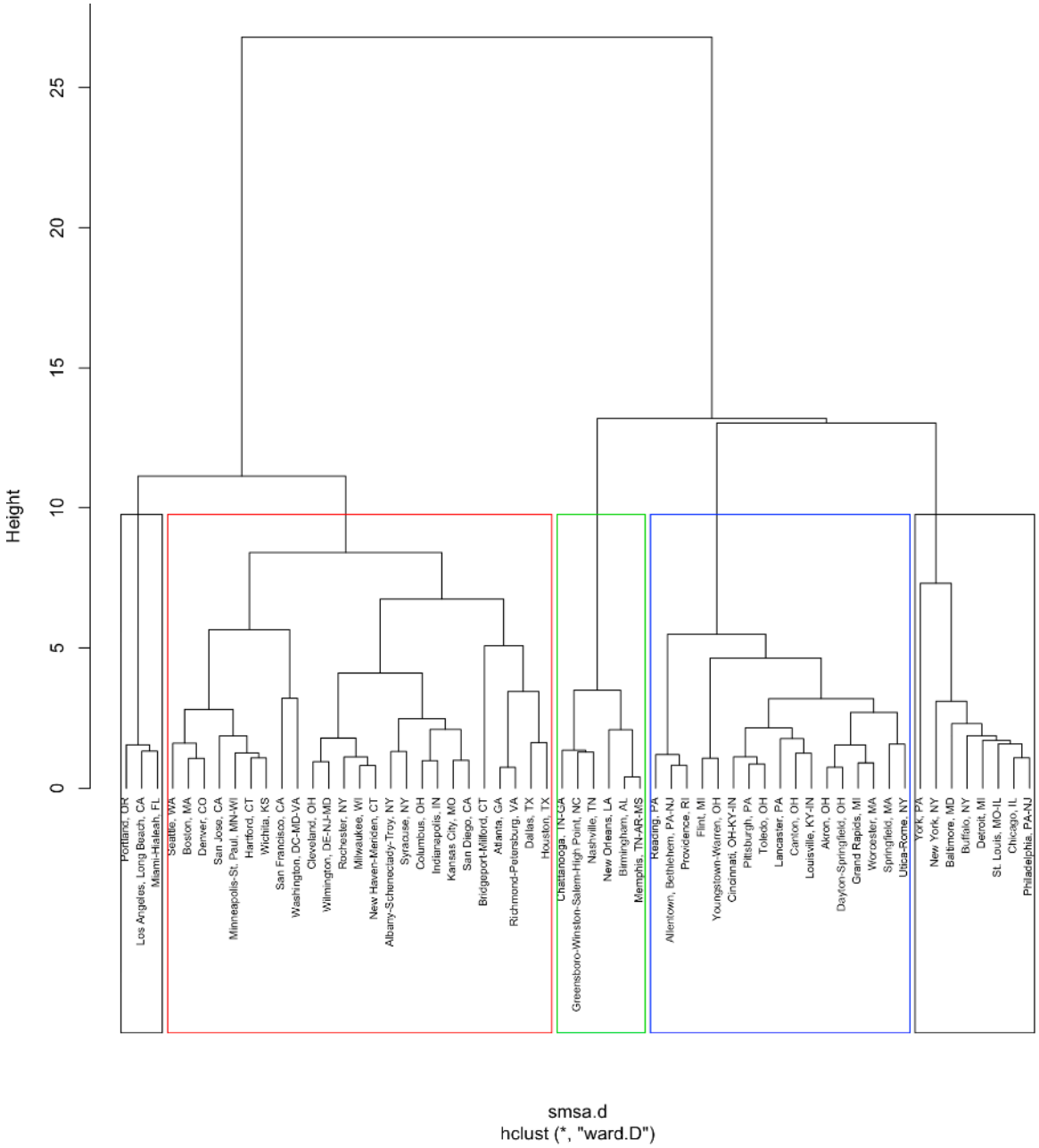
변수	변수이름	변수내용
종속변수	Mortality	사망률
기후	JanTemp	1월기온
	JulyTemp	7월기온
	RelHum	상대습도
	Rain	강우량
사회경제	Education	교육수준
	PopDensity	인구밀도
	NonWhite	비백인비율
	WC	화이트칼라 비율
	pop/house	가구당 가족수
	income	소득
환경	HCPot	오염물질1
	NOxPot	오염물질2
	S02Pot	오염물질3

```
smsa<-read.csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/
SMSA_USA.csv')
names(smsa); dim(smsa)
smsa.d<-dist(scale(smsa[c(7:10,12,13)]),method='euclidean') #유사성 거리
smsa.hclust<-hclust(smsa.d,method='ward.D') #linkage - Ward 방법
plot(smsa.hclust,labels=smsa$city_name,hang=-1,cex = 0.6) #덴드로그램
rect.hclust(smsa.hclust,k=5,border=1:4) #군집 개수 결정 및 박스 그리기
cluster<-cutree(smsa.hclust,k=5) #군집 번호 부여
smsa.ca<-cbind(smsa,cluster) #원 데이터와 군집 번호 함치기

library(doBy) #군집별 군집변수 평균 표준편차
summaryBy(education+pop_density+non_white_ratio+white_color_ratio+person_
houshold+household_income~cluster,data=smsa.ca,FUN=c(mean,sd),na.rm=T
RUE)
library(car) #군집변수 쌍체 산점도
scatterplotMatrix(~education+pop_density+non_white_ratio+white_color_ratio+p
erson_houshold+household_income|cluster,data=smsa.ca,col=1:5)


library(cluster) #주성분 군집 산점도
smsa.clus<-data.frame(smsa$city_name,scale(smsa[c(7:10,12,13)]))
names(smsa.clus)
rownames(smsa.clus)=smsa$city_name
smsa.clus0<-smsa.clus[2:7]
clusplot(smsa.clus0,cluster,shade=TRUE,lines=0,color=TRUE,labels=2)
#주성분 분석
smsa.pca<-prcomp(smsa[c(7:10,12,13)],scale=T)
smsa.pca$sdev^2 #고유값 출력
smsa.pca$sdev^2/sum(smsa.pca$sdev^2) #고유값 비율
smsa.pca$rotation #부하 출력
```

데이터 정보 : 59개 도시, 사회경제기후 환경 지표

```
> names(smsa); dim(smsa)
 [1] "city_name"        "jan_temp"         "july_temp"
 [4] "humidity"         "rainfall"         "mortality_index"
 [7] "education"        "pop_density"      "non_white_ratio"
[10] "white_color_ratio" "population"       "person_houshold"
[13] "household_income"  "HCPot"            "NOxPot"
[16] "SO2Pot"            "southern"
[1] 59 17
```

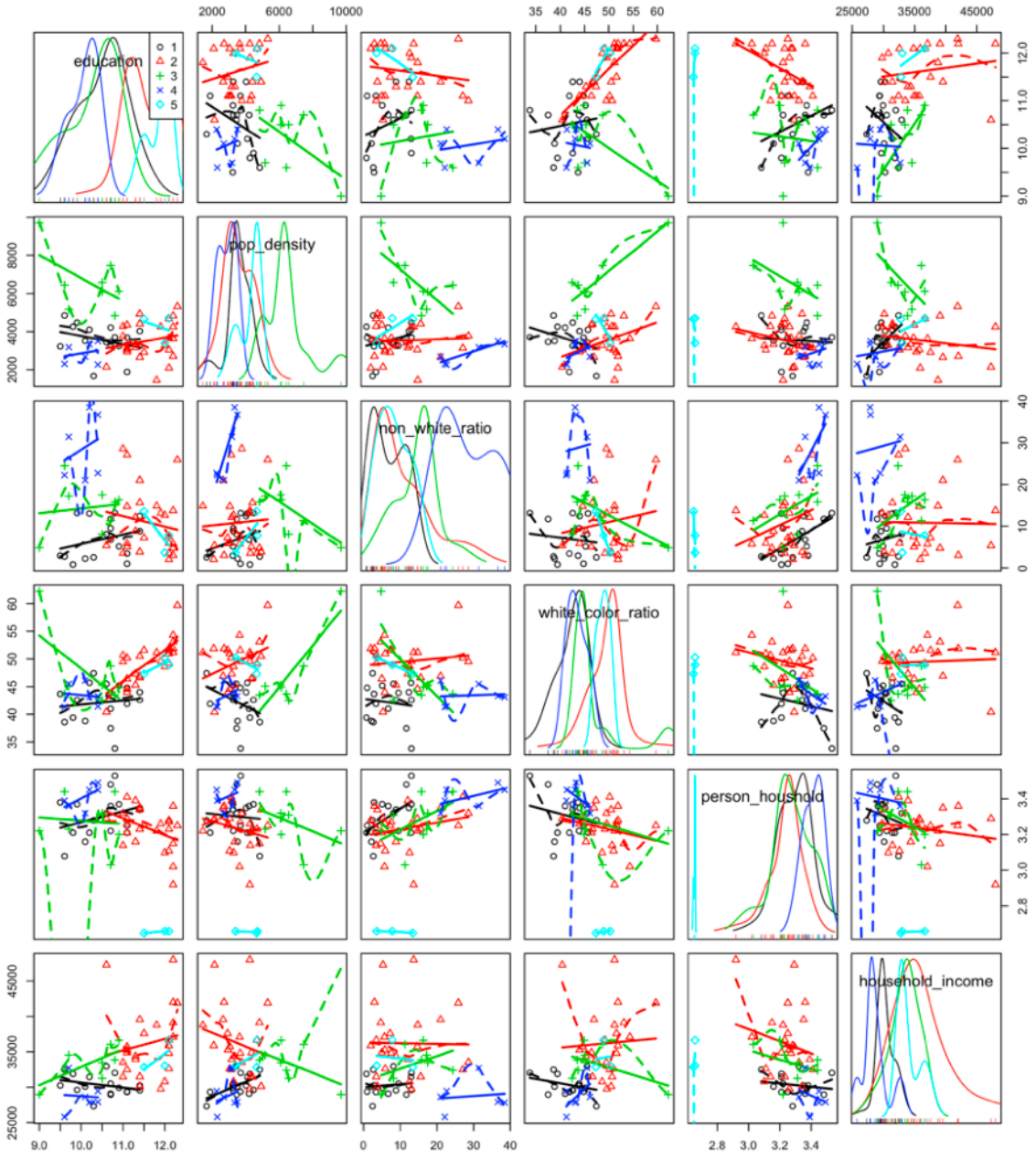
덴드로그램 : 연결법 Ward 방법, 개체 유사성은 유클리디언 표준화 거리 사용



군집화 결과 : 군집간 군집변수 평균 표준편차 계산, 군집변수 산점도

	cluster	education.mean	pop_density.mean	non_white_ratio.mean
1	1	10.51765	3539.647	6.882353
2	2	11.62000	3526.480	10.836000
3	3	10.21250	6549.000	14.425000
4	4	10.06667	2874.500	28.750000
5	5	11.86667	4248.000	8.300000

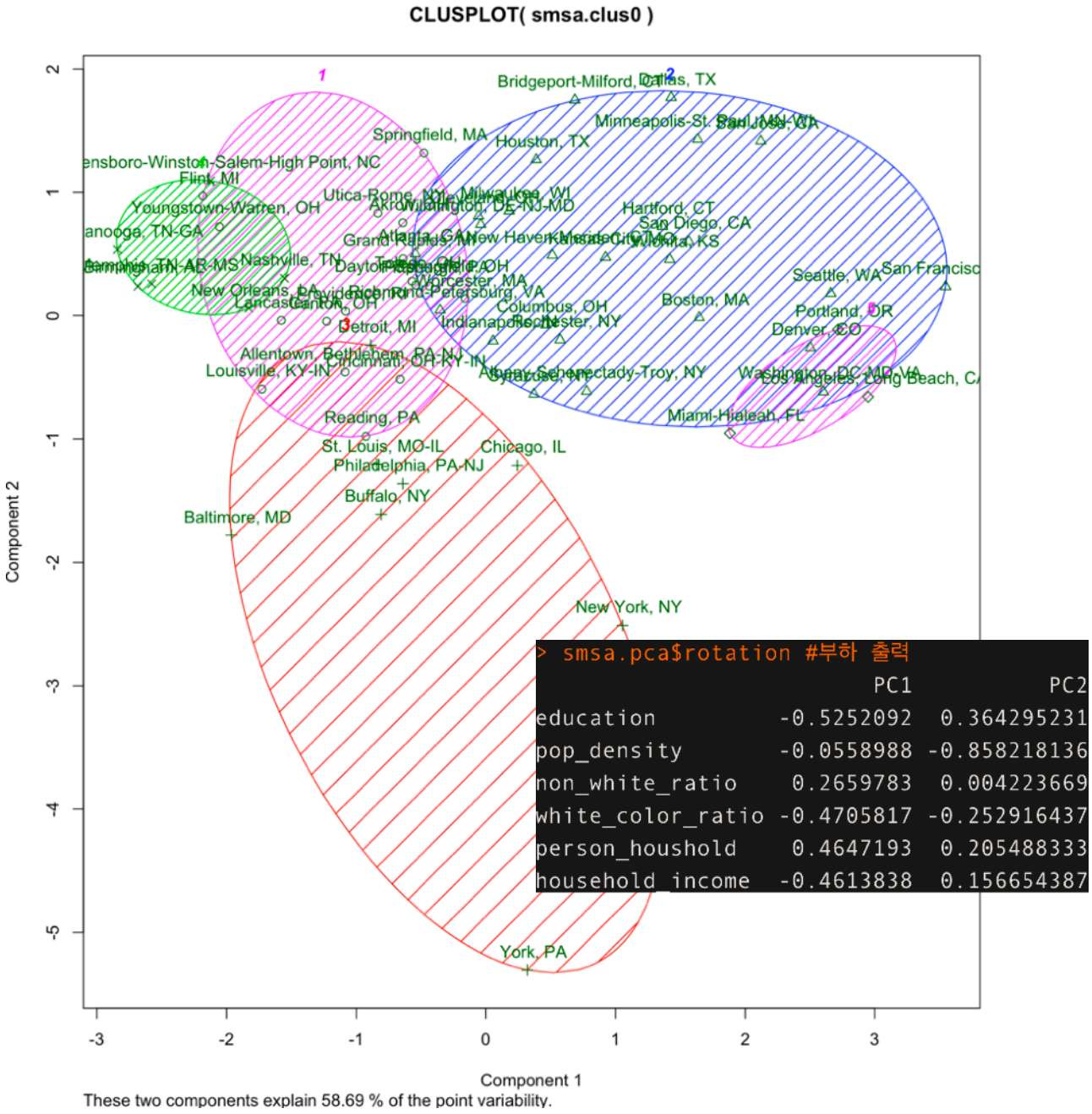
	white_color_ratio.mean	person_houshold.mean	household_income.mean
1	42.14118	3.301176	30287.53
2	49.49200	3.232400	36191.88
3	46.98750	3.273750	33386.75
4	43.45000	3.410000	28720.33
5	48.83333	2.656667	34150.67



주성분 이름 부여 후 군집 이름 부여

제1 주성분 : 교육기간, 백인비율, 가구소득 높고 가구원 수(반대부호) 낮으므로 고소득 성분 : 음의 부호이므로 낮을수록 좋은 도시 -> 살기좋은 도시(-) : 그럼에도 불구하고 주성분 부하계수 부호와 반대로 그렸음 -> 그러므로 + 값일수록 살기 좋은 도시임, 하여 다음 fviz_cluster()함수로 그린 주성분 산점도가 부하계수와 일치된 해석 가능

제2 주성분 : 인구밀도 성분 -> 인구밀집 지역(-)




```
> smsa.pca$rotation #부하 출력
                       PC1         PC2
education        -0.5252092  0.364295231
pop_density      -0.0558988 -0.858218136
non_white_ratio   0.2659783  0.004223669
white_color_ratio -0.4705817 -0.252916437
person_houshold   0.4647193  0.205488333
household_income  -0.4613838  0.156654387
```

군집 3 : 인구밀집 도시, Baltomore는 NY보다 살기 좋은 도시

군집 2, 5 : 살기 좋은 도시, 군집 4: 매우 나쁜 도시, 군집 1은 살기 보통 수준 도시

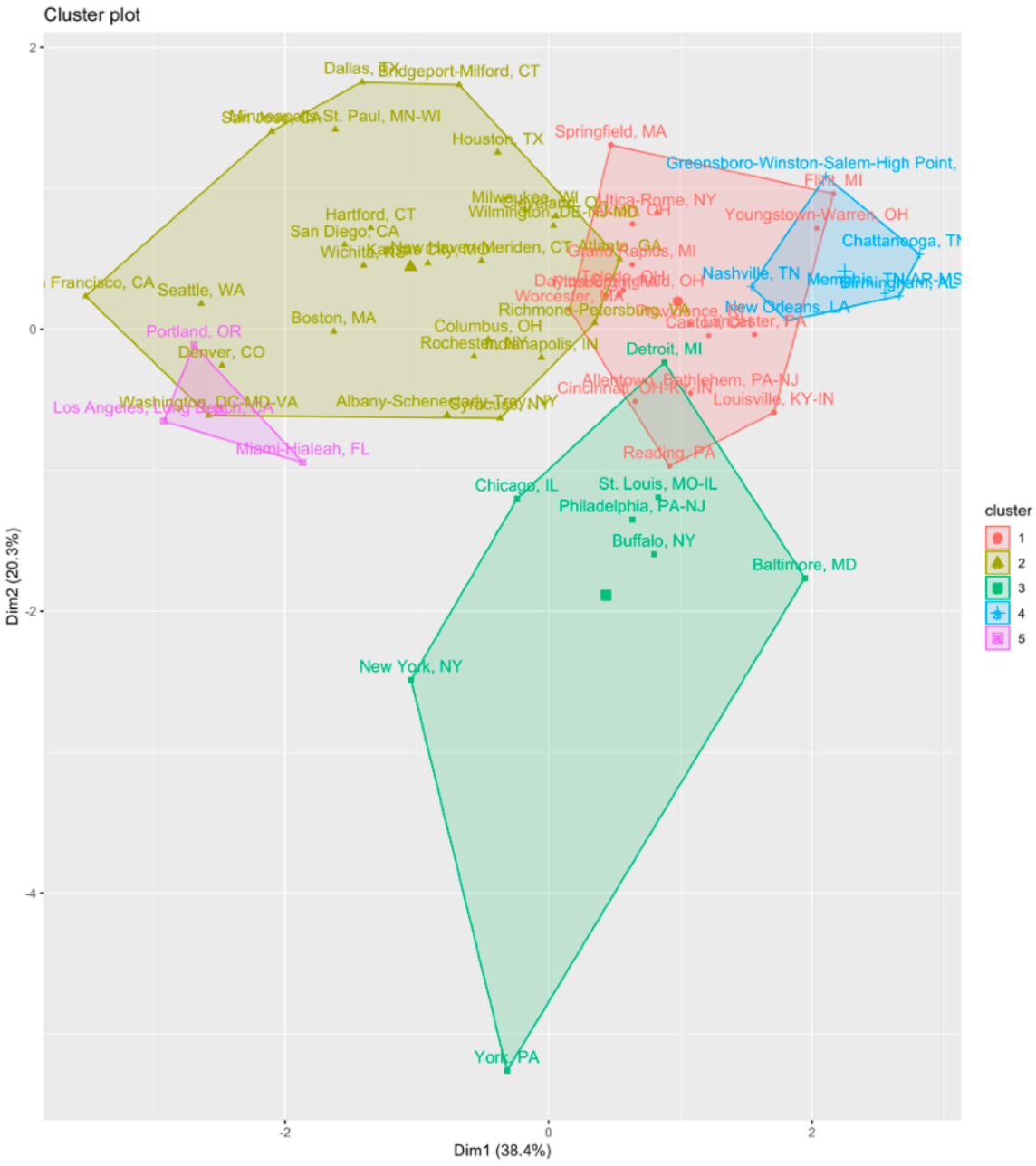
실제 군집2과 군집5는 묶는 것이 적절해 보여 4개 군집이 적절해 보인다.

Better PCA cluster plot (반드시 이 함수 이용) - 주성분 계수와 일치

```
library(factoextra)
fviz_cluster(list(data=smsa.clus0,cluster,cluster=cluster))
```

훨씬 팬시하다. 제1주성분의 부호만 달라졌을 뿐임, Dim1=-주성분1 (이 그래프가 주성분 부하계수 부호와 일치된 내용을 담고 있음), Dim2=주성분2 동일함

주성분1의 부하계수 부호를 보면 - 값이 클수록 살기 좋은 도시이므로 clusplot() 주성분 산점도와 Dim1의 부호가 반대로 되어 주성분 부하 계수 부호와 일치 -> 군집 이름 부여는 동일



Model Based to select the optimal cluster size

mclust 패키지 이용

Model based approaches assume a variety of data models and apply maximum likelihood estimation and Bayes criteria to identify the most likely model and number of clusters.

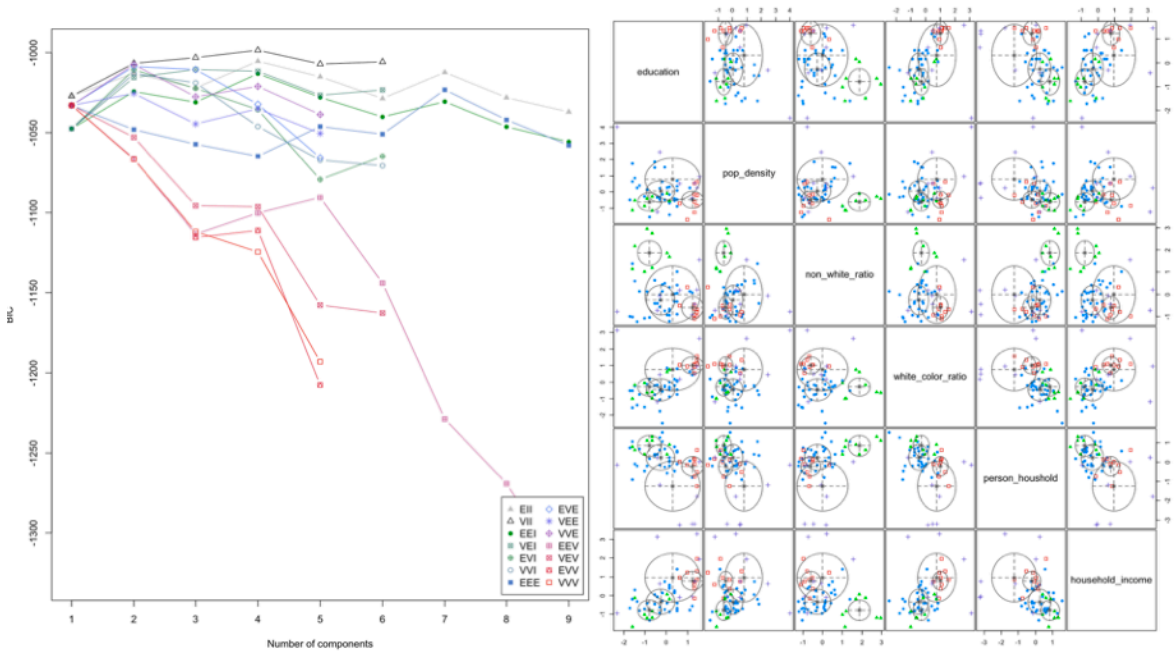
```
library(mclust)
fit <- Mclust(scale(smsa[c(7:10,12,13)]))
summary(fit) # display the best model
plot(fit) # plot results
```

모델기반 최적 군집개수는 4이다. <- 앞에서 최종 결정한(처음에는 5개로 하였지만) 군집의 개수와 일치

```
> summary(fit) # display the best model
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VII (spherical, varying volume) model with 4 components:

 log-likelihood  n df       BIC       ICL
      -436.0288 59 31 -998.4612 -1005.445

Clustering table:
 1  2  3  4
34  9  8  8
```

plot() 함수는 4개 그래프가 그려지는데 1은 최적 군집 개수 결정 그림이고 2는 군집변수 산점도로 앞에서 그린 그림과 동일하다.

```
> plot(fit) # plot results
Model-based clustering plots:

1: BIC
2: classification
3: uncertainty
4: density
```

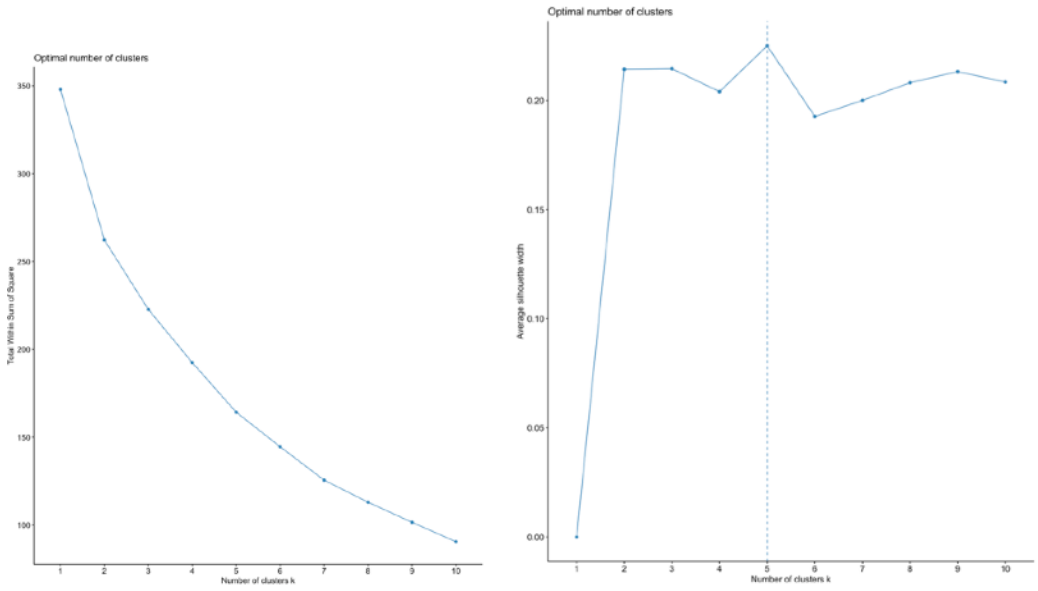


factoextra 패키지, Elbow 방법, 평균 Silhouette 방법

```
library(factoextra)
fviz_nbclust(scale(smsa[c(7:10,12,13)]), FUN = hcut, method = "wss") #Elbow Method
fviz_nbclust(scale(smsa[c(7:10,12,13)]), FUN = hcut, method = "silhouette") #Average Silhouette Method
```

elbow 방법(왼쪽 그래프)은 CCC 와 동일하게 값이 하락 속도가 느려지는 곳 -> 찾기 어려움,

Silhouette 방법 : 최적 군집 개수를 설정해 줌 - 5개임



군집 분석 - 비계층적 방법

군집의 중심이 되는 seed 점들 집합을 선택하여 그 seed 점과 유사성이 높은(거리가 가까운) 개체들을 묶는(그룹화) 방법이다. 이 방법은 다음 3가지 문제점을 갖고 있다. 1)사전에 군집(그룹) 수에 대한 예상이 필요하다. 2)개체 분류는 처음 선정한 seed 점들에 의해 영향을 많이 받고 분석자 마다 분류가 다를 가능성이 있다. 3)군집의 수와 seed 값의 위치의 결합 조건이 너무 많아 계산 소요 시간이 길다.

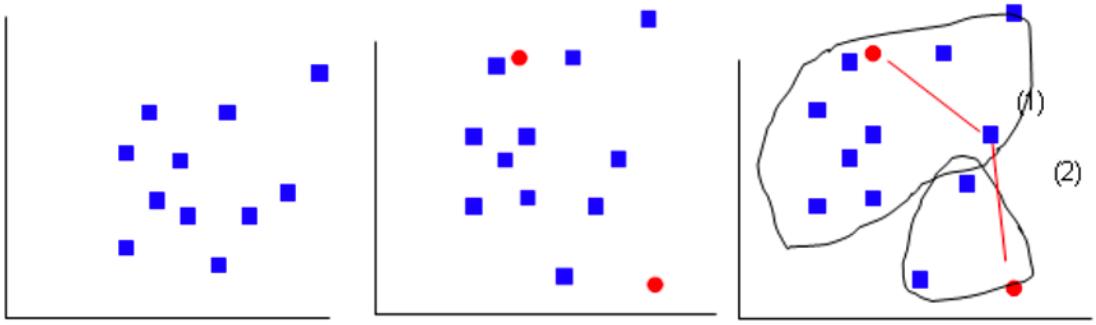
군집의 중심 결정 : 집단 내 개체 평균: **K-means** 비계층적 방법

군집의 크기 결정 (fast cluster) : 반지 름(radius) 길이

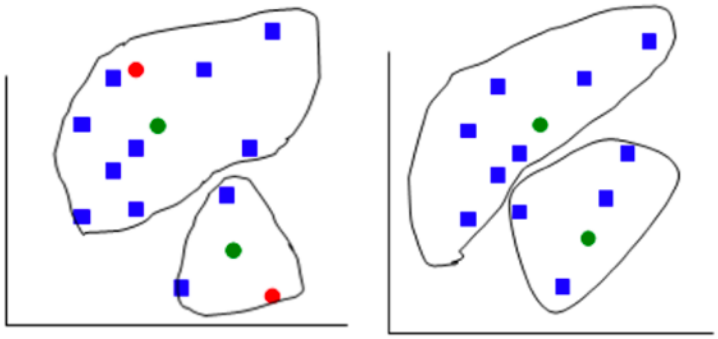
Faster Cluster 군집 분석

Faster clustering 방법은 앞 절에서 살펴보았던 계층적 군집(hierarchical clustering) 방법과는 [유사성(거리)이 가까운 개체들을 차례로 군집으로 묶어가는 방법] 달리 비계층적 군집(non-hierarchical clustering) 방법이다. 우선 seed를 정하고 이 seed에 가까운 개체들을 군집으로 묶는다. 그러므로 군집의 개수를 분석 전에 정해야 하며(number of clusters) 군집의 크기(size: 이는 radius로 설정)를 정해 주어야 한다. 비계층적 군집 방법의 순서는 다음과 같다.

(데이터) 다음의 데이터가



(데이터) (Step 1) (Step 2)



(Step 3) (Step 4)

(STEP1) 군집 seeds가 선택된다. 초기 seed의 개수는 분석자가 정해 주는 개수대로 된다. 초기 SEED 개수는 2개이다. seed 위치 ●이다.

(STEP2)개체들을 가장 가까운 군집 seed ●에 묶는다.

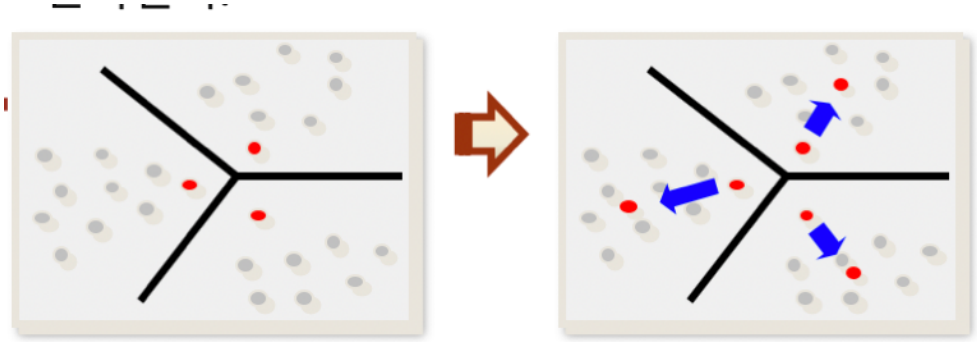
(STEP3)일단 개체 군집이 끝나면 군집 개체의 평균을 SEED ● 로 하여 개체를 다시 분류한다

(STEP4)군집이 끝나면 STEP 2)-STEP3)을 반복한다. 다음 STEP에서는 SEED가 빨강에서 초록색으로 옮겨간다. 그리고 위의 STEP을 반복한다.

Non-hierarchical clustering 방법의 또 하나의 결점은 자료 입력 순서에 따라 군집이 달라지므로 RANDOM이라는 옵션을 사용해 자료 입력 순서를 바꾸어 가며 군집하게 한다. 이처럼 FASTCLUS 방법도 하나의 방법이 있는 것은 아니다. 군집 분류 결과의 산점도(변수가 3개 이상인 경우는 주성분 분석을 이용하여 PRIN1, PRIN2를 이용한다.) 살펴보고 군집 내 개체간 거리 분산이 작은 것들을 택하면 된다.

K-평균 군집 방법

사전에 결정된 군집 수 K에 기초하여 각 관측 값을 군집의 중심들 중에서 가장 가까운 군집에 할당하는 방법이다.



단계 1 : 군집의 수 K를 결정한다.

단계 2 : 초기 K개 군집의 중심을 랜덤하게 선택한다.

단계 3 : 각 관측 값들을 가장 가까운 중심의 군집에 할당한다.

단계 4 : 새로운 군집에 할당된 관측 값들로 새로운 중심을 계산한 후 개체 군집 변동이 없을 때까지 단계 3, 4를 반복한다.

군집 수 K 값 결정

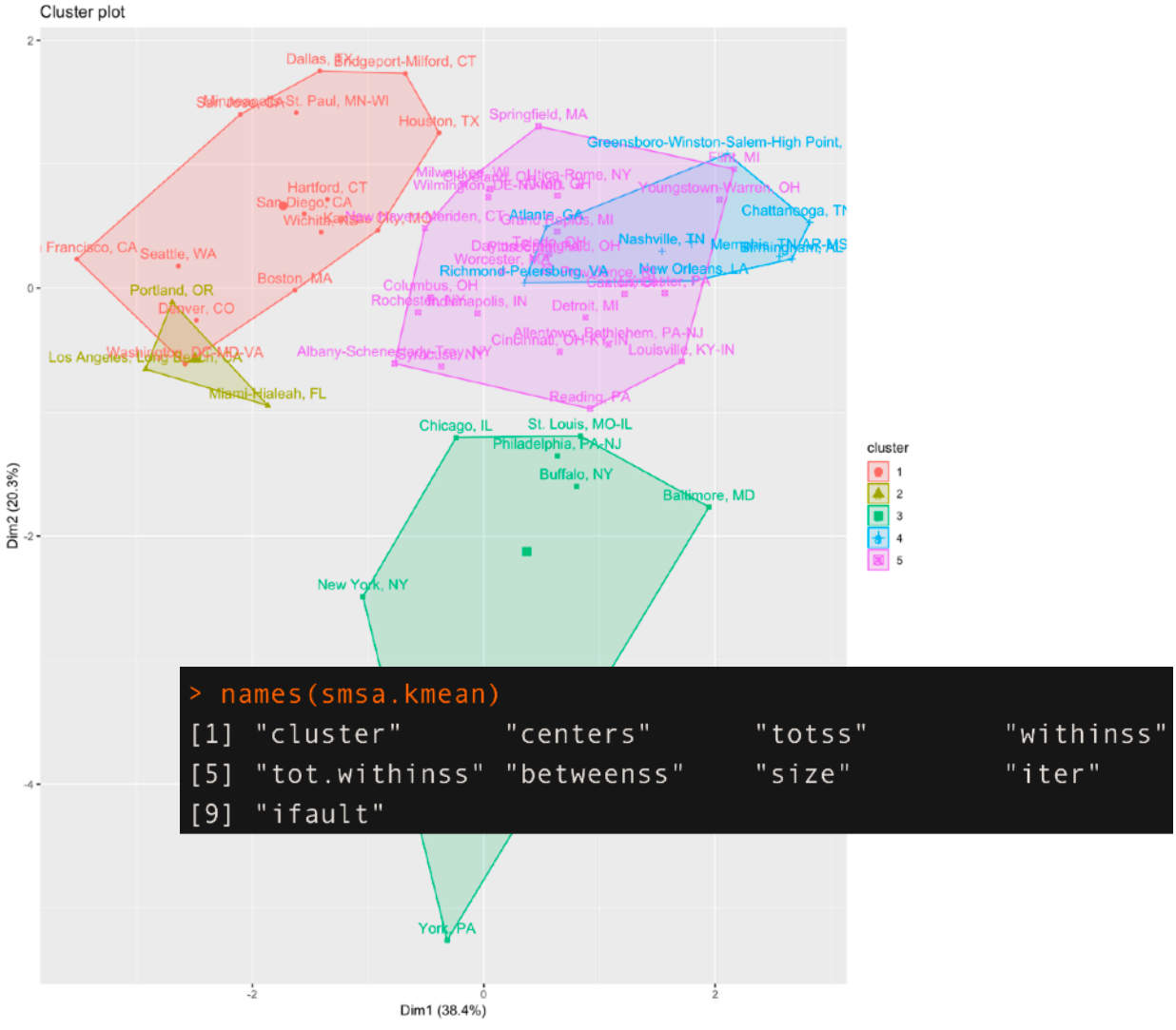
- 계층적 분석 방법에 의해 k 결정
- 여러 k 사용하여 군집간 평균 거리나 군집 내 개체 평균 거리를 활용하여 최적 k 결정

비계층적 군집분석 K-means 방법 사례 : SMSA.csv 데이터 이용

```
smsa<-read.csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/
SMSA_USA.csv')
names(smsa)
smsa.sub0<-data.frame(smsa$city_name,scale(smsa[c(7:10,12,13)]))
rownames(smsa.sub0)=smsa$city_name
smsa.kmean<-kmeans(smsa.sub0[-1],5, nstart = 25)
fviz_cluster(smsa.kmean, data=smsa.sub0[-1])
names(smsa.kmean)
```

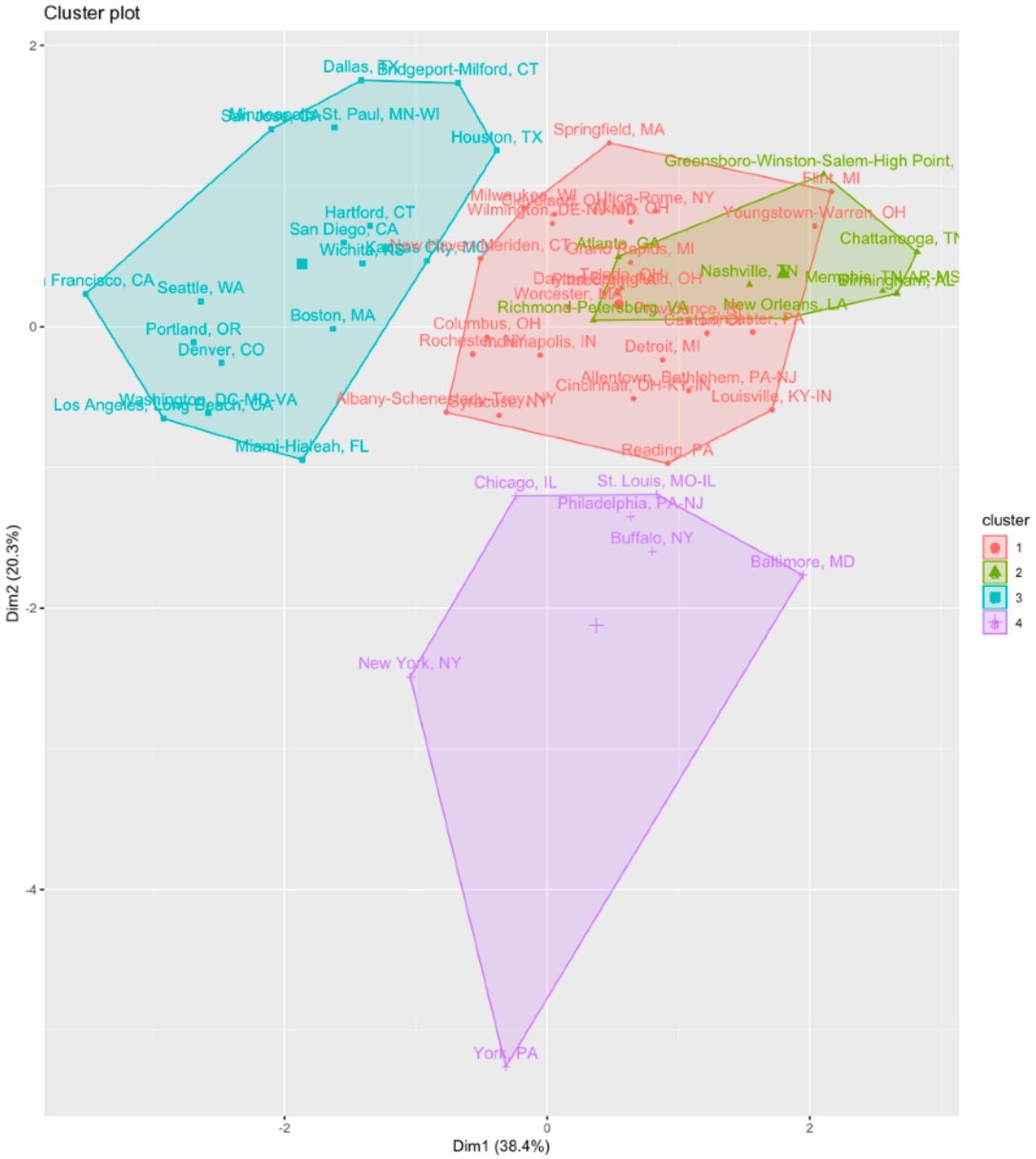
군집 번호 변수

```
> head(smsa.kmean$cluster,3)
                  Akron, OH Albany-Schenectady-Troy, NY
                          1                           1
Allentown, Bethlehem, PA-NJ
                          1
```




```
> names(smsa.kmean)
[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
```

군집의 개수 4-means 군집



군집의 중앙 위치 군집변수 값 -> 3군집 교육기간이 가장 높음

```
> smsa.kmean$centers
   education pop_density non_white_ratio white_color_ratio person_houshold
1 -0.2126725  -0.1513212      -0.4341051        -0.4933732      0.23922779
2 -0.7683273  -0.6194062       1.8503538        -0.2636132      0.83171638
3  1.1047388  -0.2917474      -0.2889440         0.8183623     -0.77897658
4 -0.9845406   2.0000897       0.2614366         0.2168320      0.01853149
  household_income
1      -0.43006801
2      -0.77377319
3       1.04047924
4       0.01626783
```

군집 이름 부여

계층적 Dim1, Dim2의 주성분과 동일함. 페이지 14의 결과와 동일

```
smsa.pca<-prcomp(smsa[c(7:10,12,13)],scale=T)
smsa.pca$sdev^2 #고유값 출력
smsa.pca$sdev^2/sum(smsa.pca$sdev^2) #고유값 비율
smsa.pca$rotation #부하 출력
```

```
> smsa.pca$rotation #부하 출력
                            PC1           PC2
education           -0.5252092  0.364295231
pop_density         -0.0558988 -0.858218136
non_white_ratio      0.2659783  0.004223669
white_color_ratio   -0.4705817 -0.252916437
person_houshold      0.4647193  0.205488333
household_income    -0.4613838  0.156654387
```

주성분 1의 상대적 계수 크기 우월 변수 : 교육기간(-), 비백인 비율(-), 가구원수(+), 가구소득(-) - 살기 좋은 도시의 지표이며 -가 큰 값일수록 살기 좋은 도시

주성분 2의 상대적 계수 크기 우월 변수 : 인구밀도(-)가 dominant, 인구밀집 도시, -값이 클수록

군집4 : 인구밀집 도시, 살기 좋은 도시 지표 중간

군집 3 : 가장 살기 좋은 도시, 인구밀집 정도 낮음

군집 1 : 살기 좋은 지표 중간 수준 도시, 인구밀집 정도 낮음

군집 2 : 살기 좋은 지표 낮은 수준 도시, 인구밀집 정도 낮음

다차원 척도법

개념

다차원 척도법(MDS: Multi Dimensional Scaling)이란 n개의 개체를 저 차원 가시적 공간(일반적으로 2차원)에 나타낼 수 있도록 하는 방법이다. 각 개체간 유사성(similarity) 혹은 거리는 저 차원으로 옮겨지더라도 원래 유사성 크기를 갖고 있어야 한다. 예를 들어 보자. Under-arm Deodorant생산 회사에서 판매 전략을 세우기 위하여 각 제품들이 서로 얼마나 유사한지(가까운지) 알아보려고 한다. 이를 위하여 소비자를 임의로 선택하여 제품의 각 분야에 대한 평가를 하게 하였다. 즉 향기, 냄새 제거 정도, 사용 편리 정도, 옷에 묻어나는 정도 등을 10점 만점으로 평가하였다. 이 점수들을 이용하여 제품의 유사성 정도를 2차원 공간에 표현하는 방법이다.

개체간 유사성을 측정하는 방법은 metric 방법과 non-metric 방법이 있다.

(1)Euclidean distance ▶ 측정형 변수 거리 (Metric 방법)

(2)각 개체의 유사성(거리)을 사람들이 평가하도록 한다. (Metric/non-Metric 방법)

(3)평가자 들이 개체를 마음대로 분류하게 하고 빈도로부터 유사성을 측정한다. (non-Metric 방법)

응용 범위를 살펴보면 다음과 같다.

(1)회사들의 이미지 측정을 통한 고객 분류

(2)소비자들이 인지하고 있는 유사한 상품 속성이나 상품 분류에 사용

(3)인구 학적 특성, 경제적 특성을 기초하여 도시간 동질성 파악

개체간의 거리 측정

다차원 척도법이란 개의 개체를 저 차원 가시적 공간(일반적으로 2차원)에 나타낼 수 있도록 하는 방법이므로 각 개체간 거리(유사성)를 측정해야 한다. 군집 분석과 유사해 보이지만 다차원 척도법은 개체의 유사성을 이차원에 표시하는 것이고 군집 분석은 개체간의 거리(유사성)가 가까운 것끼리 묶어 가는 방법이다.

측정변수 / 개체	X_1	X_2	⋯	X_p
1	x_{11}	x_{12}	⋯	x_{1p}
2	x_{21}	x_{22}	⋯	x_{2p}
⋮	⋮	⋮	⋯	⋮
n	x_{n1}	x_{n2}	⋯	x_{np}

metric 방법

metric 방법은 두 개체간의 거리(유사성)를 Euclidean distance로 나타낸다. 개체 (1, 2)의 유사성(거리)는 다음과 같이 정의된다. $D_{ij} = \sqrt{\sum_{i}^{p} (x_{1i} - x_{2i})^2}$

측정이 불가능한 경우는 각 개체간의 유사성(거리)을 리커드 척도(10점, 100점)를 이용하여 사람들이 평가하도록 하는 방법이 있다. 즉 각 , , 가 사용자들의 주관적인 평가 점수(리커드 척도)가 된다. Deodorant 제품을 분류할 경우 향기, 냄새 제거 정도, 사용 편리 정도, 옷에 묻어나는 정도 등을 10점 만점으로 평가하였다면 이 점수가 개체들간의 거리를 측정하는데 사용된다.

Non-metric 방법

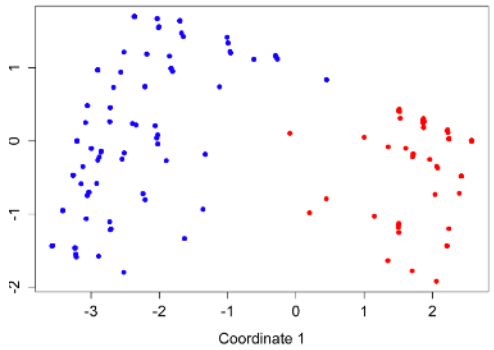
개체간의 거리를 사람들이 임의로 분류한 결과로부터 만들어진 빈도로 측정하는 방법이다. 이 방법을 이해하려면 예를 들어보는 것이 더 편리할 것이다. 50명이 20개의 개체를 임의로 분류하여 (1, 2)를 하나의 군집으로 분류한 사람이 30, (1, 20)을 하나의 군집으로 분류한 사람이 25명, (2, 20)을 하나의 군집으로 분류한 사람이 45명이라면

측정변수 / 개체	1	2	⋯	20
1	0			
2	30	0		
⋮	⋮	⋮	⋯	
20	25	45	⋯	0

이 경우 숫자가 클수록 거리는 가깝다. 즉 개체간 유사성이 높다. MDS 분석을 위하여 자료를 입력할 때는 (1-빈도/평가자수) 이것이 유사성이 된다.

기본 알고리즘

각 개체간 유사성을 측정한다. 개체의 개수가 n개인 경우 k=n(n-1)/n개 유사성 그룹이 존재한다. 다차원 공간의 개체를 2차원 공간에 표현하는 것이다.



MDS의 좌표 행렬 X는 $B = X'X$의 고유 분해 (eigen-value decomposition)에 의해 구해진다.

1) 두 개체 $X_i = (x_i, y_i), X_j = (x_j, y_j)$의 유사성을 $S_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$으로 정의하고 이를 이용하여 유사성 행렬을 정의한다. $S^2 = S_{ij}^2$

2) Double Centering 적용 : $B = -\frac{1}{2} JS^2 J, J = I - \frac{1}{n} 1'1$, n은 개체의 개수

3) B의 고유값$(\lambda_1, \lambda_2, \ldots, \lambda_n)$, 그에 대응하는 고유벡터$(e_1, e_2, \ldots, e_n)$를 구하고 이를 이용하여 고유분해를 통하여 MDS의 좌표 행렬 $X = E_k \Lambda_k^{\frac{1}{2}}$ 를 얻는다. Λ는 고유값인 대각인 대각행렬, E는 고유벡터 행렬이다. k는 표현 원하는 차원이다.

4) 개체를 m(일반적으로 2)차원으로 공간으로 줄일 경우 개체간의 거리를 구한다. 이는 물론 측정 변수 전체를 가지고 유사성을 측정한 2)와 다를 것이다. 2차원 공간으로 줄일 수 있는지를 알아 보는 것이 STRESS 값이다. 2차원 공간으로 줄이는 것이 목적이므로 Stress 검정은 하지 않는다.

$$Strain_D(x_1, x_2, \ldots, x_N) = \left(\frac{\sum_{i,j} \left(b_{ij} - \langle x_i, x_j \rangle \right)^2}{\sum_{i,j} b_{ij}^2} \right)^{1/2}$$

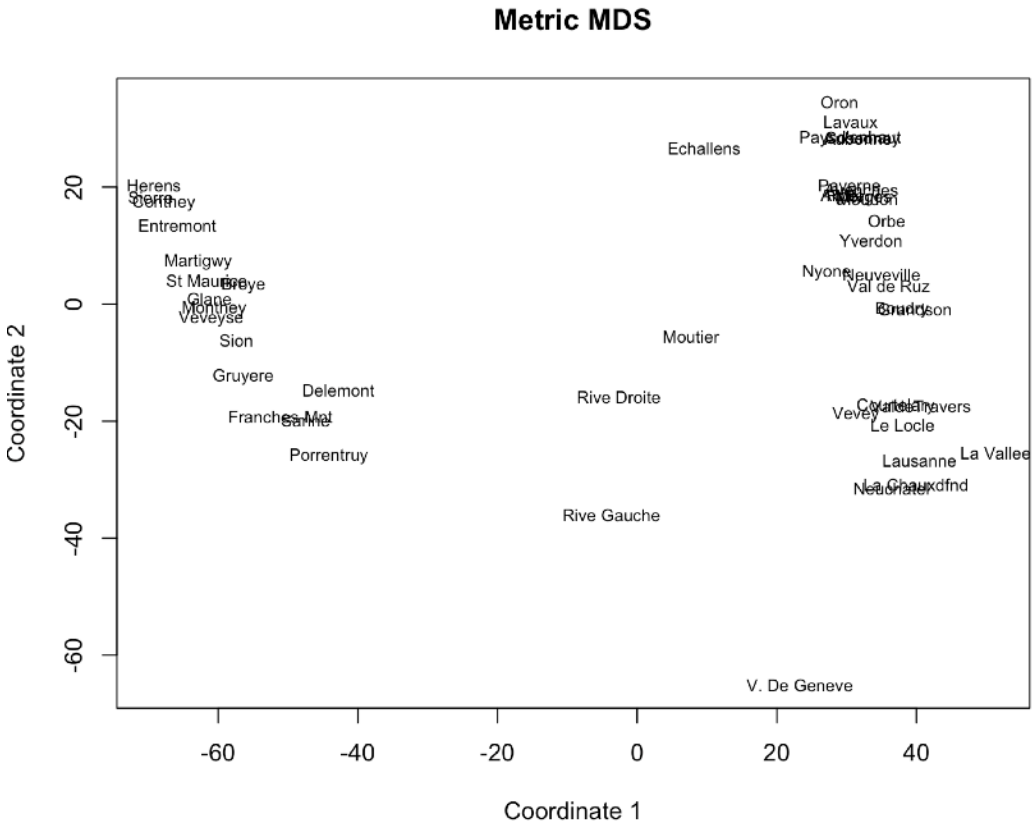
Stress	Goodness of fits
20%	Poor
10%	Fair
5%	Good
2.5%	Excellent

사례분석

swiss data that contains fertility and socio-economic (출생, 사회지표) data on 47 French speaking provinces in Switzerland.

```
##              Fertility Agriculture Examination Education Catholic
## Courtelary        80.2        17.0          15        12     9.96
## Delemont          83.1        45.1           6         9    84.84
## Franches-Mnt      92.5        39.7           5         5    93.40
## Moutier           85.8        36.5          12         7    33.77
## Neuveville        76.9        43.5          17        15     5.16
## Porrentruy        76.1        35.3           9         7    90.57
##              Infant.Mortality
## Courtelary               22.2
## Delemont                 22.2
## Franches-Mnt             20.2
## Moutier                  20.3
## Neuveville               20.6
## Porrentruy               26.6
```

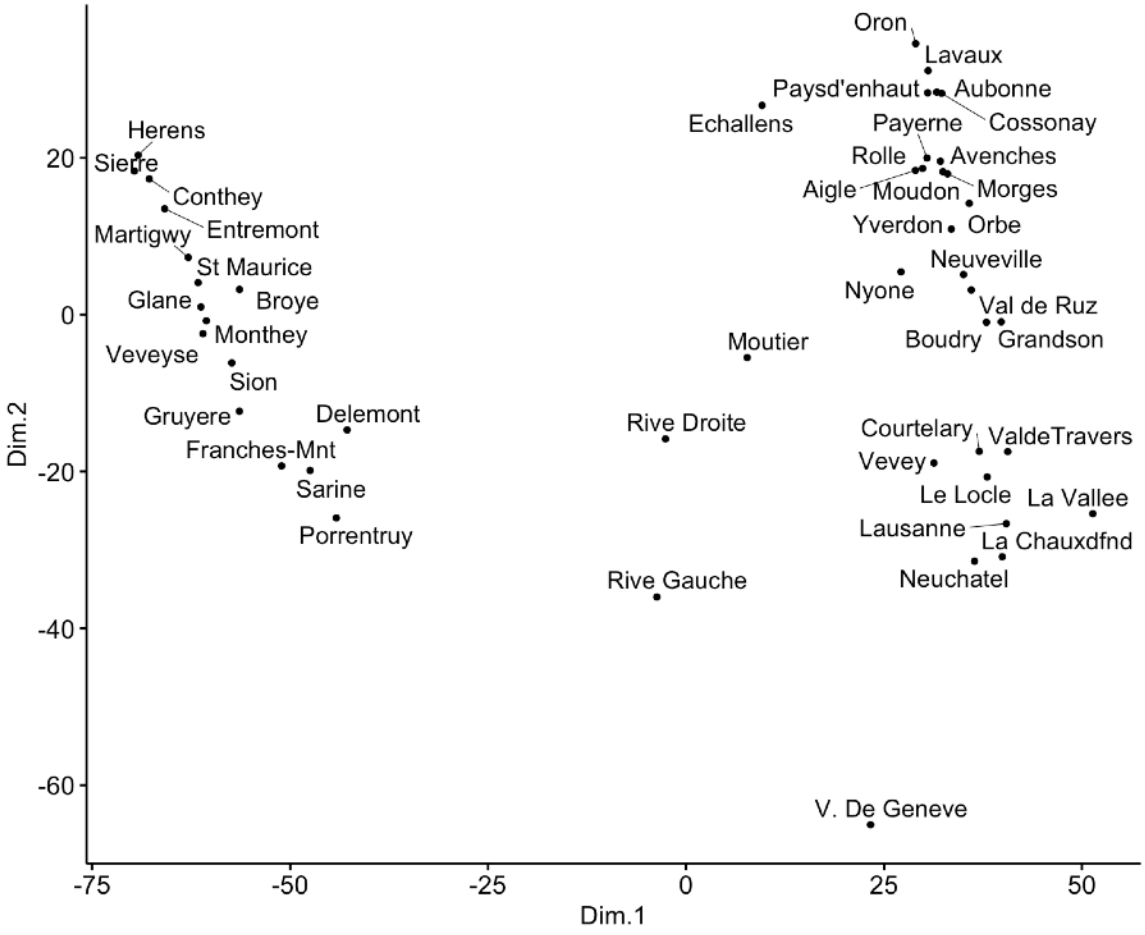
```
data(swiss)
swiss.dist<-dist(swiss) # euclidean distances between the rows
fit<-cmdscale(swiss.dist,eig=TRUE,k=2) # RESULTS OF mds
# plot solution
x <- fit$points[,1] ; y <- fit$points[,2]
plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2",main="Metric MDS", type="n")
text(x, y, labels = row.names(swiss), cex=.7)
```



```
library(magrittr); library(dplyr); library(ggpubr)
mds<-as_tibble(cmdscale(dist(swiss)))
colnames(mds) <- c("Dim.1", "Dim.2")
# Plot MDS
ggscatter(mds, x = "Dim.1", y = "Dim.2", label = rownames(swiss),size = 1, repel
= TRUE)
```

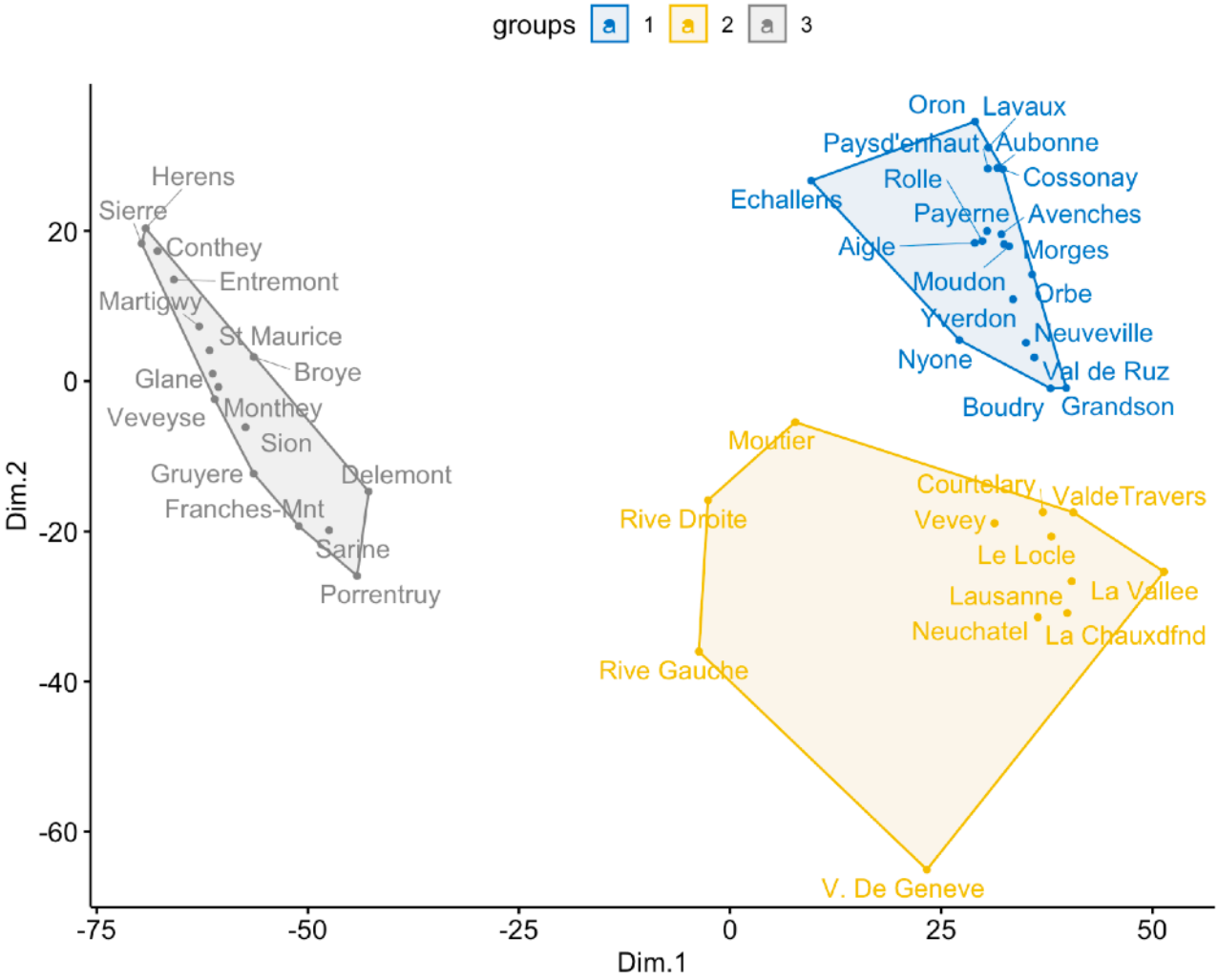
출생, 사회지표 6개 척도에 의해 유사한 지역은 Oron - Lavaux, Aubonne

Herons-Sierre, Conthey 이렇게 산점도에서 가까이 있는 도시들이 유사한 지역으로 분류하면 된다.



MDS 결과를 k-means 비계층적군집 방법 적용

```
# K-means clustering
clust<-as.factor(kmeans(mds,3)$cluster)
mds<-mutate(mds,groups = clust)
# Plot and color by groups
ggscatter(mds, x = "Dim.1", y = "Dim.2",
          label = rownames(swiss),color='groups',
          palette='jco', size = 1,
          ellipse = TRUE,ellipse.type = "convex", repel = TRUE)
```

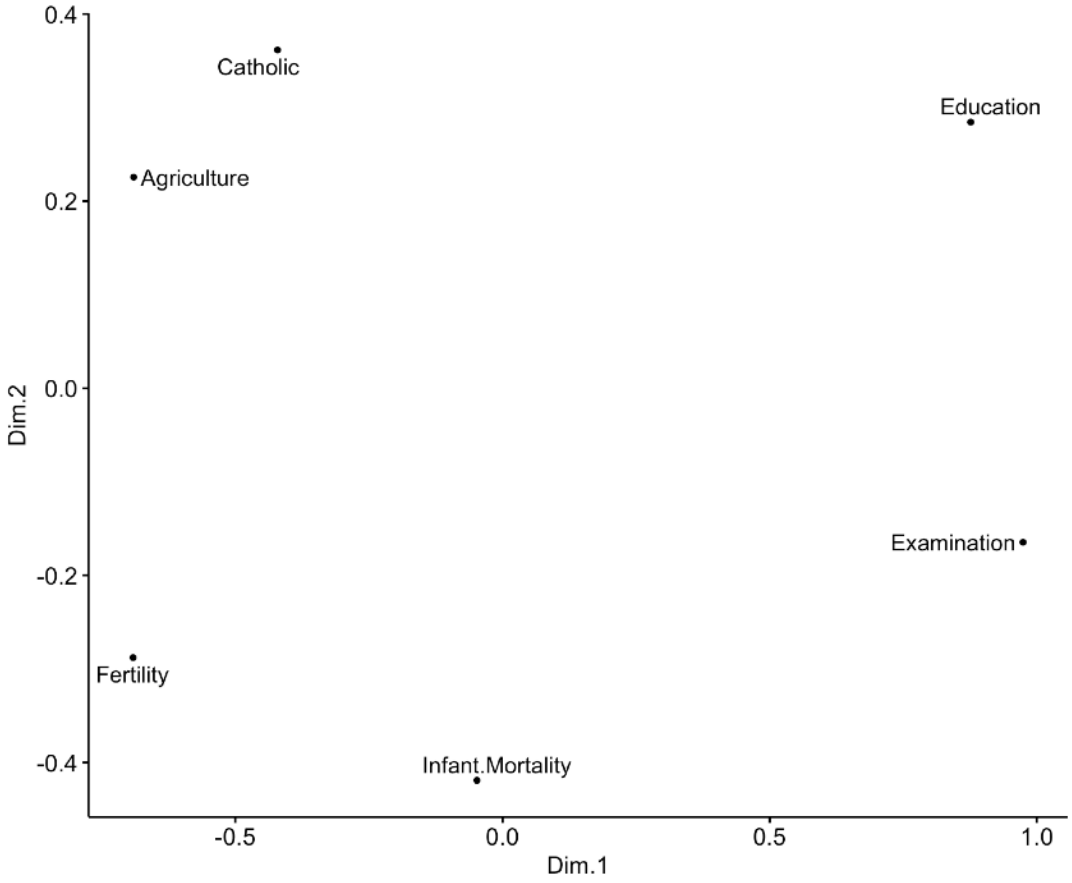


다차원척도법 활용 상관계수 그래프 표현

유사성 행렬 계산 시 (1-상관계수) 척도를 사용한 이유는 상관계수의 값이 크다는 것은 유사성이 높다는 것인데 거리 개념으로 보면 유사성이 낮기 때문에 역으로 하여 유사성 행렬을 계산해야 한다.

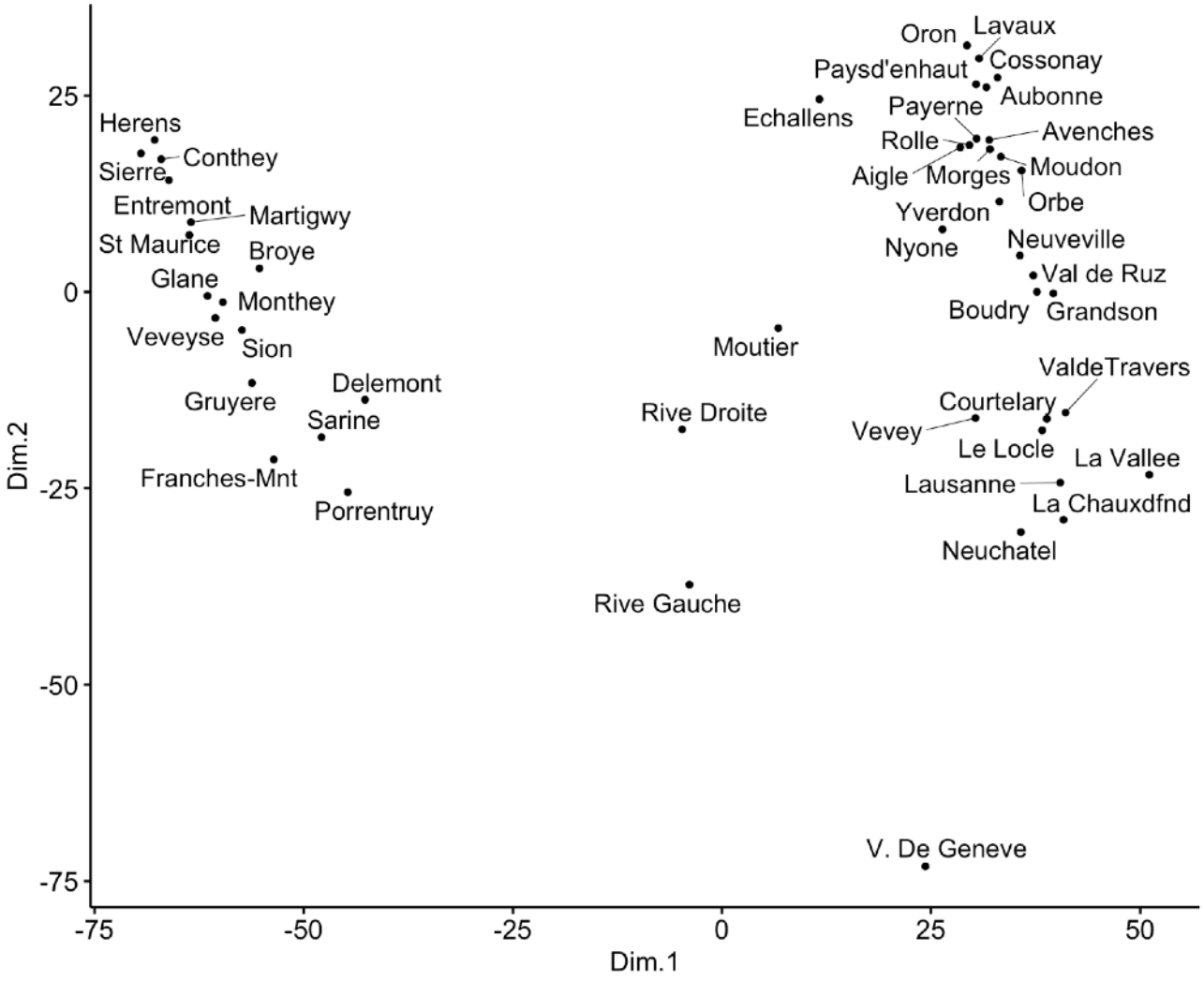
```
res.cor<-cor(swiss, method='pearson')
mds.cor<-as_tibble(cmdscale((1 - res.cor)))
colnames(mds.cor) <- c("Dim.1", "Dim.2")
ggscatter(mds.cor, x = "Dim.1", y = "Dim.2",
          size = 1,label = colnames(res.cor),repel = TRUE)
```

변수의 유사성이 표현됨 : (출생률, 유아 사망율) 유사 변수, (천주교 비율, 농업비율) 유사 변수



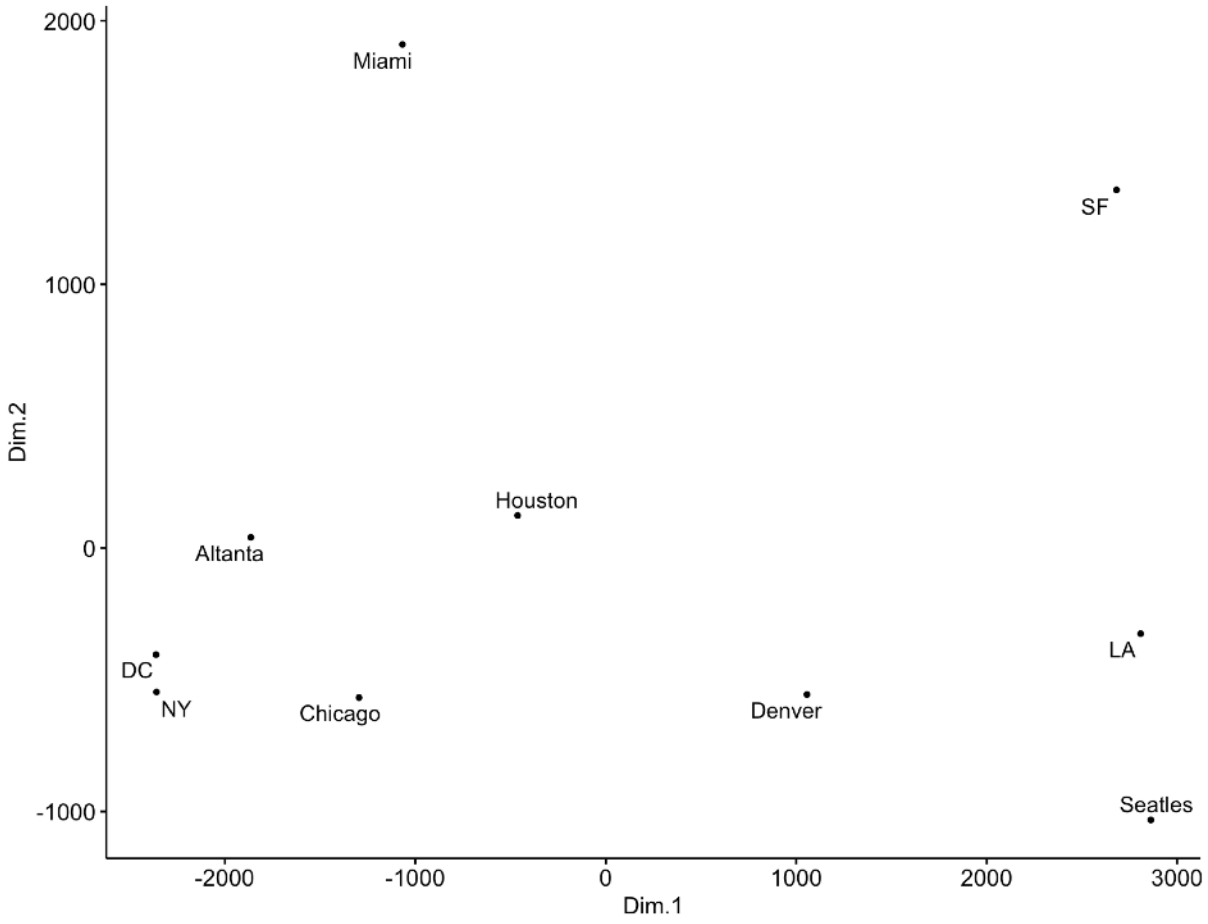
Kruskal's non-metric multidimensional scaling

```
library(MASS)
mds.non<-as_tibble(isoMDS(dist(swiss))$points)
colnames(mds.non) <- c("Dim.1", "Dim.2")
# Plot MDS
ggscatter(mds.non, x = "Dim.1", y = "Dim.2",
          label = rownames(swiss),size = 1,repel = TRUE)
```



도시 거리 행렬 다차원 척도법 표현

```
city<-read.csv('http://203.247.53.31/Stat_Notes/example_data/도시거리.csv')
rownames(city)<-city[,1]
city.df<-city[,-1]
library(magrittr);library(dplyr); library(ggpubr)
mds.city<-as_tibble(cmdscale(dist(city.df)))
colnames(mds.city) <- c("Dim.1", "Dim.2")
# Plot MDS
ggscatter(mds.city, x = "Dim.1", y = "Dim.2",
          label = rownames(city.df), size = 1, repel = TRUE)
```



본 예제는 거리 행렬만 구하면 non-metric 정성적 분류에 적용 가능하다. 한국 지붕 문양 30개의 유사성으로 군집화 하는 경우, 유사성을 측정형 척도로 측정하는 것은 가능하지 않음. 하여, 전문가 50명을 선발하여 임의로 문양 30개를 군집화 하게 한다. 문양 A,B를 동일한 그룹으로 묶은 전문가가 5명이라면 두 문양의 유사성은 5/50이다. 이렇게 하여 유사성(거리) 행렬을 구한 후 위의 방법을 적용하면 다차원 적용 그래프를 얻는다.

대응분석

개념

- 범주형 변수 범주의 유사성 표현
- 교차표(분할표)로 나타내어지는 자료의 행과 열 범주를 저차원 공간상(2차원)의 좌표로 표현하여 관계를 탐구하려는 탐색적 자료 분석 기법

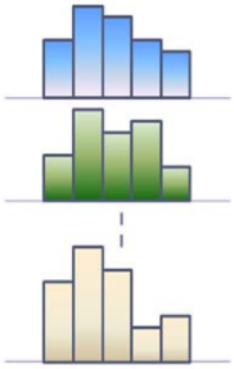
기원

- 대응분석의 수리적인 기원은 1930년대 Hirshfeld의 논문『상관관계와 분할표의 연관성』
- 대응분석의 기하적인 면은 1960년대 프랑스에서 Jean-Paul Benzecri에 의해서 발전되었다.
- 일본: 1950년대 Chikio Hayashi에 의해서 수량화 제3방법으로 개발되어 발전
- 프랑스: 1960년대 Jean-Paul Benzecri가 이끄는 자료분석 모임이 다양한 분야로부터 수집된 자료를 분석하는데 대응분석 기법을 응용하고 발전

RXC 분할표 검정

두 범주형 변수 X(R개 범주, 원인), Y(C개 범주, 결과변인)의 교차표는 다음과 같다.

X \ Y	1	2	…	C	Total
1	π11	π12	…	π1c	π1+
2	π21	π22	…	π2c	π2+
…	…	…	…	…	…
R	πr1	πr2	…	πrc	πr+
Total	π+1	π+2	…	π+c	π++



π_{ij} : (X, Y) 결합밀도함수, π_{i+}: X 주변밀도함수, π_{+j} : Y 주변밀도함수

Homogeneity (동질성)

각 행에 대해 열의 분포가 동일한가?

귀무가설 : $H_0 : \pi_{ij} = \pi_{kj} \ for \ all \ j = 1,2,\ldots,C$

Independence (독립성)

(X, Y)는 서로 독립인가? 두 변수가 독립이라면 ($P(X = x, Y = y) = P(X = x)P(Y = y)$)이므로 귀무가설은 다음과 같다.

귀무가설 : $H_0 : \pi_{ij} = \pi_{i+}\pi_{+j}$

검정통계량

동질성, 독립성 검정 모두 검정통계량은 동일하다.

(i, j) 셀의 관측빈도를 $O_{ij} = n_{ij}$ 라 하자. i-행의 행 관측빈도 합을 n_{i+}, j-열의 열 관측빈도 합을 n_{+j}, 총 관측빈도를 n_{++}라 정의하자.

만약 귀무가설이 맞다면 (i, j) 셀의 기대빈도는 $E_{ij} = \frac{n_{i+}n_{+j}}{n_{++}}$이다.

검정통계량 : $TS = \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \sim \chi^2(df = (R-1)(C-1))$

*) Cochran Theorem : 셀의 기대빈도가 5이하인 셀이 전체 셀 중 20% 미만이면 교차검정통계량은 χ^2-분포를 따른다. 만약 이를 위반하면 Fisher Exact 검정 방법을 적용한다.

[위키피디아 예제] 성별(남녀) 통계학 공부 여부 차이가 있는지 검정

	Men	Women	*Row total*
Studying	1	9	*10*
Not-studying	11	3	*14*
Column total	*12*	*12*	*24*

	Men	Women	*Row Total*
Studying	*a*	*b*	*a + b*
Non-studying	*c*	*d*	*c + d*
Column Total	*a + c*	*b + d*	*a + b + c + d (=n)*

$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{n}{a+c}} = \frac{\binom{a+b}{b}\binom{c+d}{d}}{\binom{n}{b+d}} = \frac{(a+b)!\,(c+d)!\,(a+c)!\,(b+d)!}{a!\;b!\;c!\;d!\;n!}$$

검정통계량 :

$$p = \binom{10}{1}\binom{14}{11}/\binom{24}{12} = \frac{10!\;14!\;12!\;12!}{1!\;9!\;11!\;3!\;24!} \approx 0.001346076$$ +

$$p = \binom{10}{0}\binom{14}{12}/\binom{24}{12} \approx 0.000033652$$

	Men	Women	*Row Total*
Studying	0	10	*10*
Non-studying	12	2	*14*
Column Total	*12*	*12*	*24*

= 0.00138

fisher.test(rbind(c(1,9),c(11,3)), alternative="less")

```
> fisher.test(rbind(c(1,9),c(11,3)), alternative="less")

        Fisher's Exact Test for Count Data

data:  rbind(c(1, 9), c(11, 3))
p-value = 0.00138
```

유의확률이 0.0014로 매우 유의한 차이가 있음. 아래 열 퍼센트 결과 남자의 공부 비율은 8.3%, 여자의 공부비율은 75%로 여자가 남자에 비해 공부하는 비율이 유의적으로 매우 높다.

```
> prop.table(rbind(c(1,9),c(11,3)), 2)# column
           [,1] [,2]
[1,] 0.08333333 0.75
[2,] 0.91666667 0.25
```

결과 해석

검정 결과 p-value(유의확률) 0.05보다 적으면 1)두 변수 상관관계 존재한다. 2)동일분포가 파괴된다.

관계 해석: 행 페센트 혹은 열 퍼센트에 의한 차이 해석한다.

(단점)

- R×C 셀이 많아지면 퍼센트에 의한 해석이 복잡해지고 신뢰도가 떨어짐
- 행 범주, 열 범주의 유사성 정도를 표현하지 못함

교차표 검정 예제

http://203.247.53.31/Stat_Notes/elem_stat/EDA/sports.csv

1열 : 1=초반리드 게임 패, 2=초반리드 게임 승

2열 : 게임종류 1=농구(1쿼터종료), 2=야구(3이닝종료), 3=풋볼(1쿼터종료), 4=아이스하키(1피리어드종료)

3열 : 1=후반리드 게임 패, 2=후반리드 게임 승

4열 : 게임종류 1=농구(3쿼터종료), 2=야구(7이닝종료), 3=풋볼(3쿼터종료), 4=아이스하키(2피리어드종료)

```
sport<-read.csv('http://203.247.53.31/Stat_Notes/elem_stat/EDA/sports.csv')
names(sport)
table.sport<-table(sport$Game..Early.,sport$Early)
chisq.test(table.sport)
#prop.table(table.sport) # cell percentages
prop.table(table.sport, 1) # row percentages
#prop.table(table.sport, 2) # column percentage
```

4개 변수 초반리드 승리여부, 경기종류, 후반리드 승리여부, 경기종류

```
> names(sport)
[1] "Early"        "Game..Early." "Late"         "Game..Late."
```

경기종류와 초반리드 승리여부의 차이는 있는가? 즉, 초반에 리드한 팀의 승리 가능성이 높으면 보는 사람들의 흥미는 반감될 것이다.

귀무가설 : 경기종류와 초반리드 승리여부는 차이가 없다. 경기종류와 초반리드 팀 승리여부는 독립이다.

대립가설 : 경기종류와 초반리드 승리여부는 차이가 있다.

검정 통계량 : 12.6 유의확률 0.006 매우 유의 => 귀무가설 기각하여 경기 종류에 따른 초반리드 승리여부의 차이가 있다.

```
> chisq.test(table.sport)

        Pearson's Chi-squared test

data:  table.sport
X-squared = 12.616, df = 3, p-value = 0.005544
```

야구 초반 리드 팀이 경기에 승리(=2)할 비율은 81.3% 가장 높고 풋볼이 가장 낮은 54.8%이다. 풋볼이 초반 리드가 승리로 이어질 가능성이 가장 낮아 긴장감 높은 종목이다.

```
> prop.table(table.sport, 1) # row percentages

           1         2
  1 0.3015873 0.6984127
  2 0.1875000 0.8125000
  3 0.4520548 0.5479452
  4 0.3066667 0.6933333
```

귀무가설 : 경기종류와 후반리드 승리여부는 차이가 없다. 경기종류와 후반 리드 팀 승리여부는 독립이다.

대립가설 : 경기종류와 후반리드 승리여부는 차이가 있다.

```
table.sport2<-table(sport$Game..Late.,sport$Late)
chisq.test(table.sport2)
prop.table(table.sport2,1)
```

유의확률 0.015%이므로 귀무가설 기각, 경기에 따른 후반 리드 팀 승리 비율은 차이가 있다.

```
> chisq.test(table.sport2)

        Pearson's Chi-squared test

data:  table.sport2
X-squared = 10.518, df = 3, p-value = 0.01464
```

야구 경기가 7이닝 종료 시 리드한 팀이 승리할 확률 93.4%로 가장 높고, 농구는 3쿼터 종료 승리 팀이 게임 이길 확률이 79.4%로 가장 낮다. 야구는 7이닝 이흐 역전 가능성은 6.6%로 매우 낮으므로 7이닝만 보고 가도 된다. 차가 막힐 것을 대비하여

```
> prop.table(table.sport2,1) # column percentages

             1          2
  1 0.20634921 0.79365079
  2 0.06521739 0.93478261
  3 0.22580645 0.77419355
  4 0.18750000 0.81250000
```

대응분석 방법

(i,j)셀의 빈도 $n_{ij} \geq 0$의 i번째 행의 각 열빈도 $(n_{i1}, n_{i2}, \ldots, n_{iC})$은 총빈도가 n_{i+}이고 C개 범주를 갖는 다항(Multinomial) 분포이다.

(1) 다항분포의 (i,j) 확률은 상대빈도 $f_{ij} = \frac{n_{ij}}{n_{i+}}$: 이것을 행 프로파일 (row profile)이라 정의한다.

(2) 각 행의 상대빈도 $f_{i1}, f_{i2}, \ldots, f_{iC}$를 선형계수로 (주성분 분석과 유사) 하여 좌표 계산한다.

$R_i = (\frac{f_{i1}}{f_{i+}}, \frac{f_{i2}}{f_{i+}}, \ldots, \frac{f_{iC}}{f_{i+}})$가 C 차원 가중 Euclid 공간의 좌표이다. 가중(weighted) Euclid 공간이란 에서는 두 개의 좌표 (R_i, R_j) 사이의 거리를 다음과 같이 정의한다.

$$D(R_i, R_j) = \sqrt{\frac{\sum_{c=1}^{C} (\frac{f_{ic}}{f_{i+}} - \frac{f_{jc}}{f_{j+}})^2}{f_{+c}}}$$

(3) 같은 방식으로 열 프로파일의 좌표 및 개체 거리 계산하고 행, 열 프로파일을 각각 2차원 공간에 표현한다.

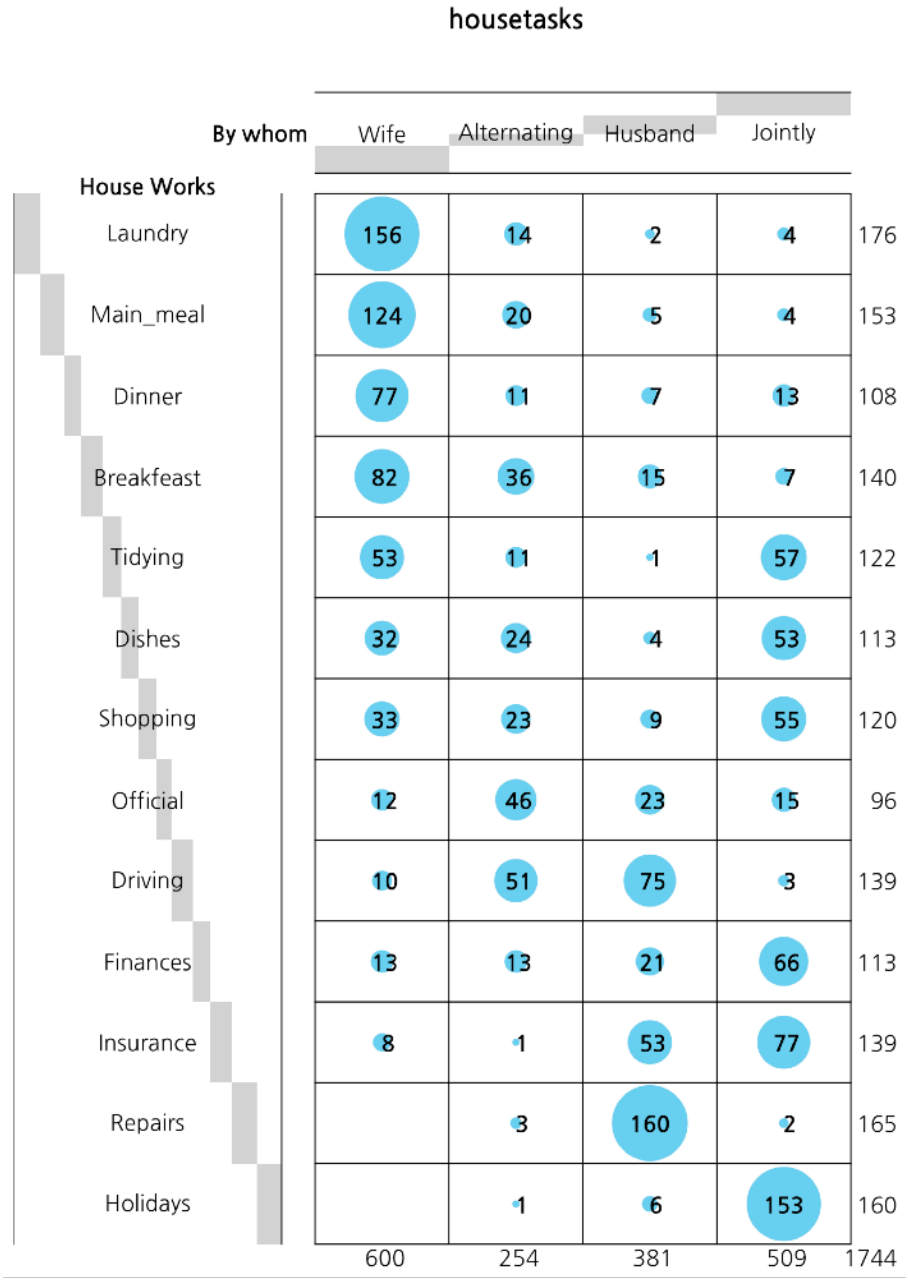
사례 : 이원 대응분석 (교차표)

내부 데이터 : housetasks

행 범주 : 집안 일, 열 범주 : 아내, 남편, 함께 jointly, 교대 alternating

```
#install.packages(c("FactoMineR", "factoextra"))
library("FactoMineR"); library("factoextra")
data(housetasks)
```

```
library("gplots")
dt <- as.table(as.matrix(housetasks)) #matrix
balloonplot(t(dt), main ="housetasks", xlab ="By whom", ylab="House Works")
```



빈도표에 대한 그래프 표현, 행과 열의 주변 marginal 확률분포(빈도)와 함께 출력하였음. 회색 면적은 빈도의 상대적 크기임.

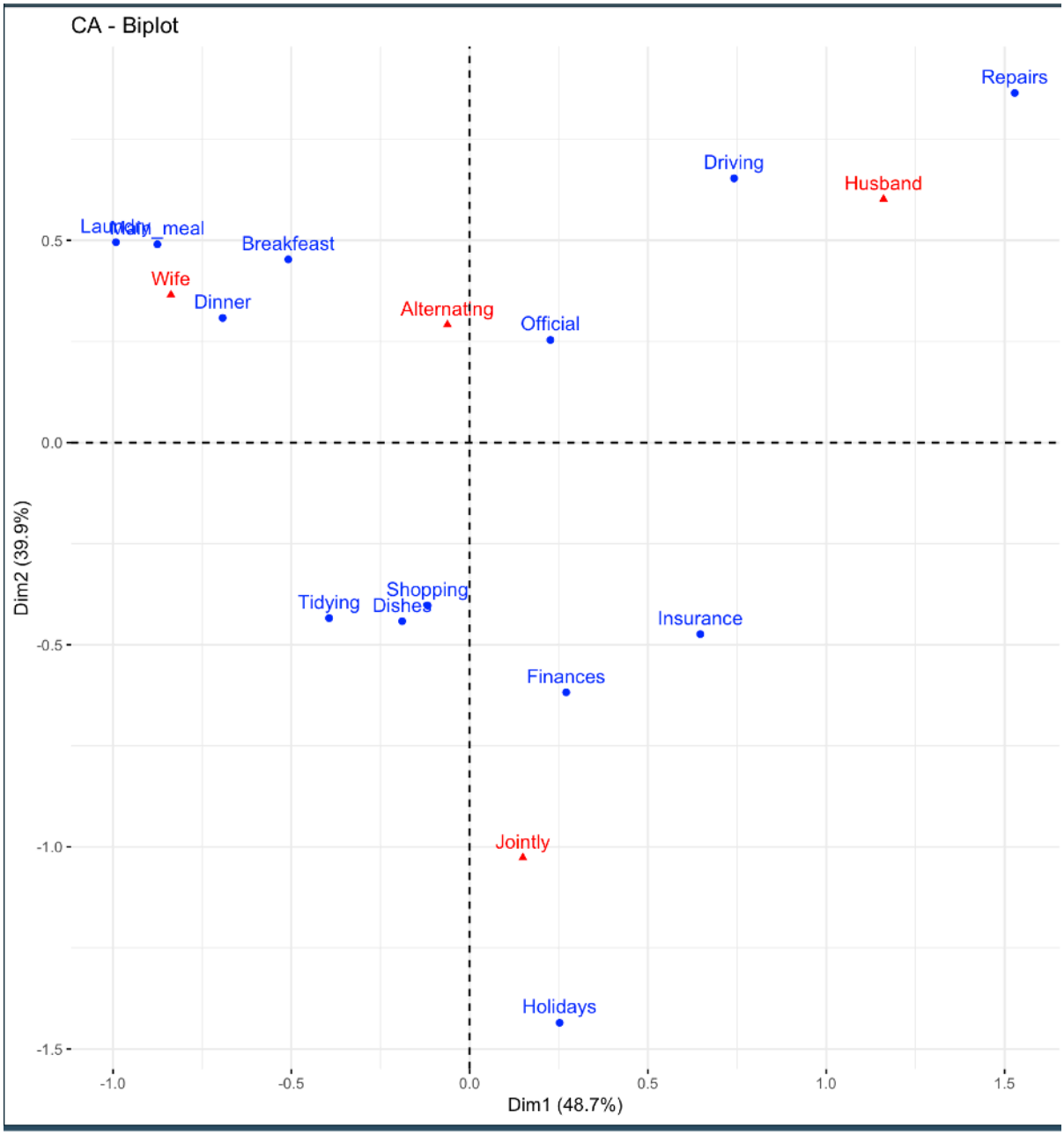
```
chisq.test(housetasks) #chi-square Independence Test
```

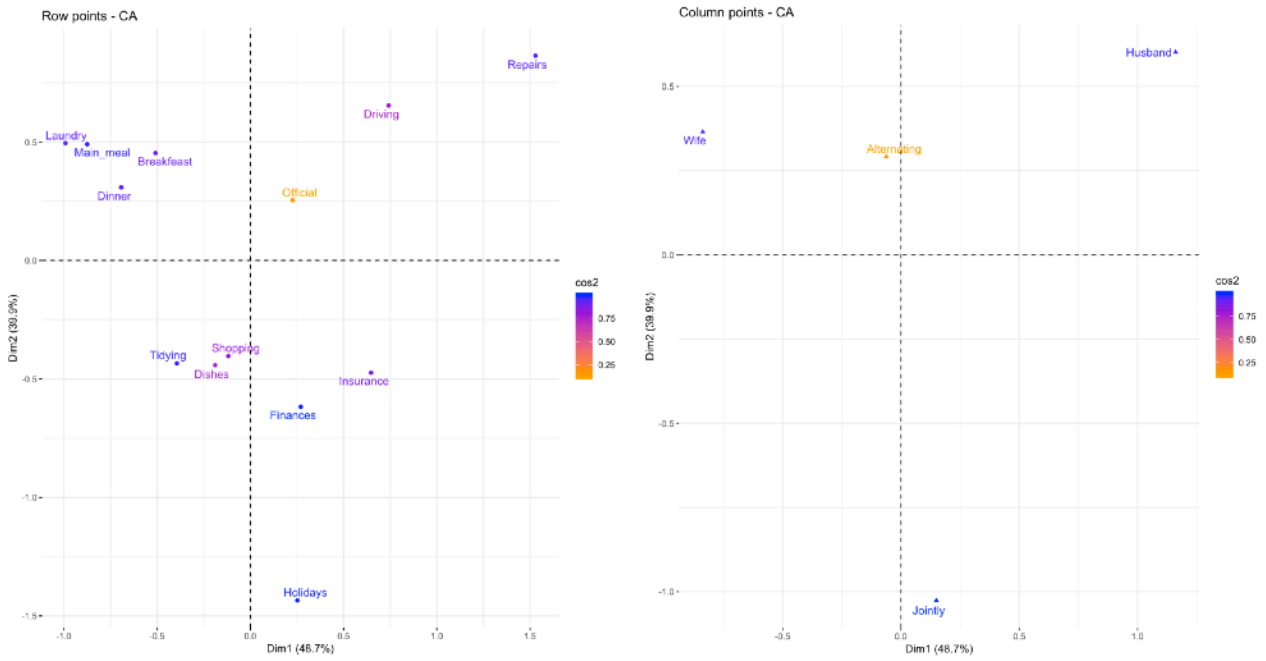
유의확률 <0.001 집안일과 역할 분담자의 관계는 매유 유의함. 아내는 세탁, 음식준비, 남편은 수리, 운전, 함께하는 일은 휴일 즐기기, 보험, 가정재무 등이다.

```
        Pearson's Chi-squared test

data:  housetasks
X-squared = 1944.5, df = 36, p-value < 2.2e-16
```

```
library("FactoMineR")
res.ca <- CA(housetasks,graph=FALSE) #results of Correspondence
fviz_ca_biplot(res.ca) #bi-plot plot

fviz_ca_row(res.ca, col.row = "cos2", gradient.cols = c("orange","blue"),repel =
TRUE) #row plot
fviz_ca_col(res.ca, col.col = "cos2", gradient.cols = c("orange","blue"),repel =
TRUE) #column plot
```

빈도표 해석 결과를 2차원 그래프에 표현함.





행과 열 주변 marginal 빈도 크기를 표현함. 열 범주 - 가사 역할 빈도 높은 주부(600), 남편(509) 블루 색으로, 행 범주 - 집안 일 빈도 높은 Laudry(176), Repairs(165), Holidays(160)가 블루, 낮은 빈도는 오렌지 색으로 표현하였음 (사실 크게 필요 없음)

이차원 축소 시 교차표의 정보가 얼마나 표현되는지 고유치에 의해 판단할 수 있음 (주성분분석과 동일함) - 1차원은 48.7%, 2차원은 39.9% 합은 88.6%로 이차원 공간에 교차표 정보가 충분히 표현된다.

```
library("factoextra")
get_eigenvalue(res.ca) #eigen values-dim explain
```

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	0.5428893	48.69222	48.69222
Dim.2	0.4450028	39.91269	88.60491
Dim.3	0.1270484	11.39509	100.00000

예제 2 : 15대 대선 후보자 지역 특표수 교차표

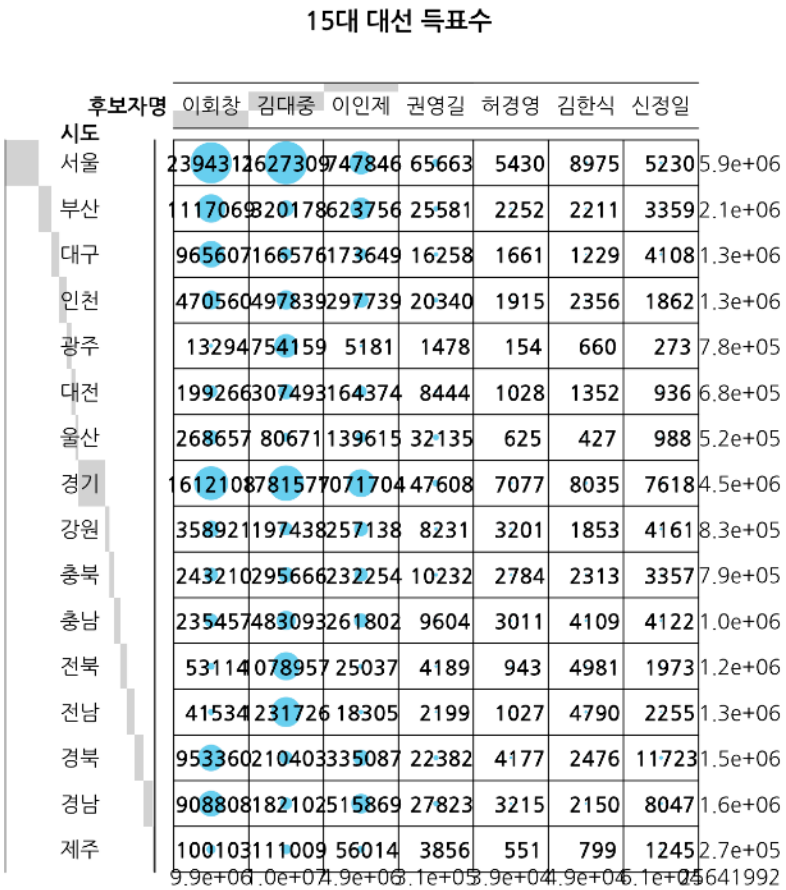
```
vote<-read.csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/vote15.csv')
rownames(vote)<-vote[,1] #시도 이름으로 행 이름 바꾸기
vote.df<-vote[-1] #시도 변수 제외
chisq.test(vote.df)
```

```
        Pearson's Chi-squared test

data:  vote.df
X-squared = 7123971, df = 90, p-value < 2.2e-16
```

=> 매우 유의하므로 지역별 후보자 득표율의 차이는 있음

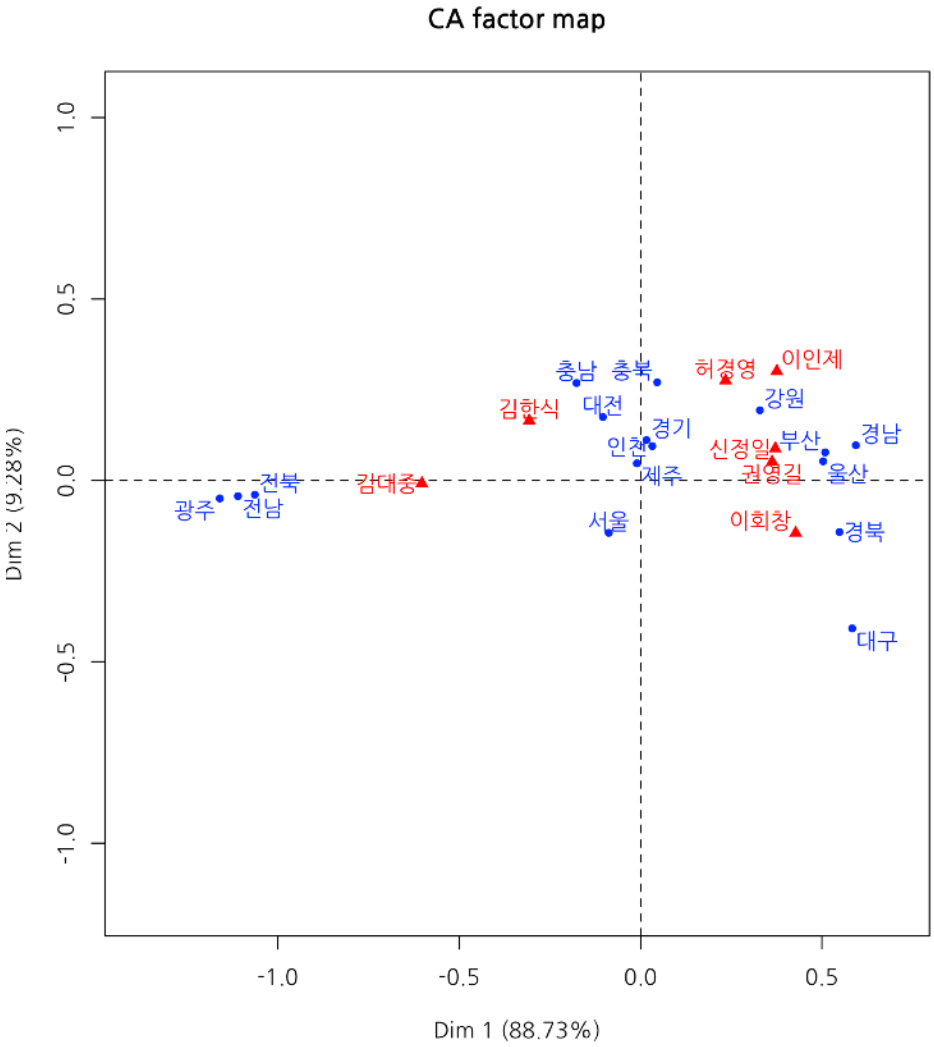
```
par(family="NanumGothic") #for MAC users
library("gplots")
balloonplot(t(vote.table), main ="15대 대선 득표수", ylab ="시도", xlab="후보자명")
```

서울 지역 김대중, 이회창 이인제, 김대중 후보는 전남, 전북ㅇ서 이회창은 경북, 경남에서 우의를 보임.



```
library("FactoMineR")
vote.ca<-CA(vote.df,graph=FALSE) #results of Correspondence
plot(vote.ca) # fviz_ca_biplot 함수는 한글 출력되지 않음 on Mac
fviz_ca_biplot(vote.ca) #bi-plot plot
```

김대중 후보는 전남, 전북, 광주 상대적 우세, 이회창은 경북, 대구, 울산, 경남, 권영길은 부산, 울산, 경남 상대적 우세를 보임. 이인제는 강원이었음.




```
library("factoextra")
get_eigenvalue(vote.ca) #eigen values-dim explain
```

```
> get_eigenvalue(vote.ca) #eigen values-dim explain
       eigenvalue variance.percent cumulative.variance.percent
Dim.1 2.465222e-01      88.73310991                    88.73311
Dim.2 2.577412e-02       9.27712510                    98.01024
```

2차원 공간에 교차표 정보가 충분히(98%) 표현됨

사례 : 다차원 대응분석 (삼차원 이상) : 패키지 예제 데이터

Tea : 티 종류 (black, green, flavored)

How : 함께 마시는 것 (alone, w/milk, w/lemon, other)

sugar : 설탕 첨가 여부 (yes, no)

Where : 티 구입 장소 (supermarket, shops, both)

```
library("FactoMineR"); library("factoextra")
data(tea)
newtea<-tea[, c("Tea", "How", "sugar", "where")]
cats<-apply(newtea, 2, function(x) nlevels(as.factor(x))); cats
```

```
> cats
  Tea   How sugar where
    3     4     2     3
```

관심 범주 변수의 수준을 출력하였음

```
mca1 = MCA(newtea, graph = FALSE)
mca1$eig
```

다중 교차표가 2차원 공간에 표현하는 경우 33%만 표현됨

```
> mca1$eig
       eigenvalue percentage of variance cumulative percentage of variance
dim 1  0.3344834              16.724168                           16.72417
dim 2  0.3256352              16.281760                           33.00593
dim 3  0.2989486              14.947429                           47.95336
dim 4  0.2523412              12.617061                           60.57042
```

```
# data frame with variable coordinates
mca1_vars_df = data.frame(mca1$var$coord, Variable = rep(names(cats), cats))
# data frame with observation coordinates
mca1_obs_df = data.frame(mca1$ind$coord)
# plot of variable categories
ggplot(data=mca1_vars_df,
       aes(x = Dim.1, y = Dim.2, label = rownames(mca1_vars_df))) +
  geom_hline(yintercept = 0, colour = "gray70") +
  geom_vline(xintercept = 0, colour = "gray70") +
  geom_text(aes(colour=Variable)) +
  ggtitle("MCA plot of variables using R package FactoMineR")
```

그린티는 티숍에서 블랙으로 슈가 없이 마신다.

얼그레이는 chain_티숍에서 밀크와 레몬, 슈가를 넣어 마신다.