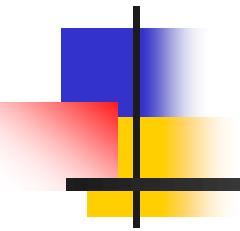


군집분석(Cluster Analysis)

개념, 알고리즘 및 응용

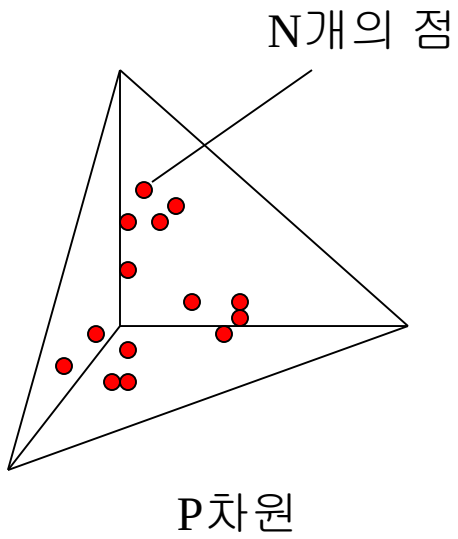
군집분석(Cluster Analysis) 개요

- 다변량 자료를 각 특성의 유사성에 따라 여러 그룹(군집 또는 집락)으로 나누는 통계적 기법중의 하나

P개

ID	성별	나이	직업	학력	본인소득	가입일	연횟수	연간금액	결제
1	1	21	7	2		9908	1	10000	1
3	2	25	2	4	3	9902	2	124000	1
4	2	25	2	4	3	9902	2	124000	1
5	1	30	2	4	7	9705	2	157000	1
6	2	47	4	1	5	9904	1	10000	1
8	1	24	7	2	2	9804	1	10000	1
14	2	22	2	4		9908	2	98000	1
15	2	22	2	4		9908	2	98000	1
16	1	19	7	2		9908	1	34500	1
17	1	37	5	4	5	9710	2	57000	1
18	1	37	5	4	5	9710	2	57000	1
19	1	39	2	6	2	9812	2	40500	1
20	1	39	2	6	2	9812	2	40500	1

N 개



- 군집의 개수, 내용, 구조가 파악되지 않은 상태에서 특성을 파악하며, 군집들 간의 관계를 분석 (탐색적 분석)
- 고객의 세분화 또는 군집 별로 추가적인 분석을 수행하기 위해 활용

유사성 측정

Clustering에서의 거리 계산

유사성

- 군집으로 묶기 위해서는 개체간에 유사한 특성을 가지고 있어야 함
- 이 유사한 특성의 정도를 나타내는 척도로 개체간의 거리를 사용하고, 거리가 상대적으로 가까운 개체들을 동일 군집으로 묶음
- 거리(distance)의 조건

$$d_{ij} \geq 0, d_{ii} = 0, i, j = 1,2,...,n$$

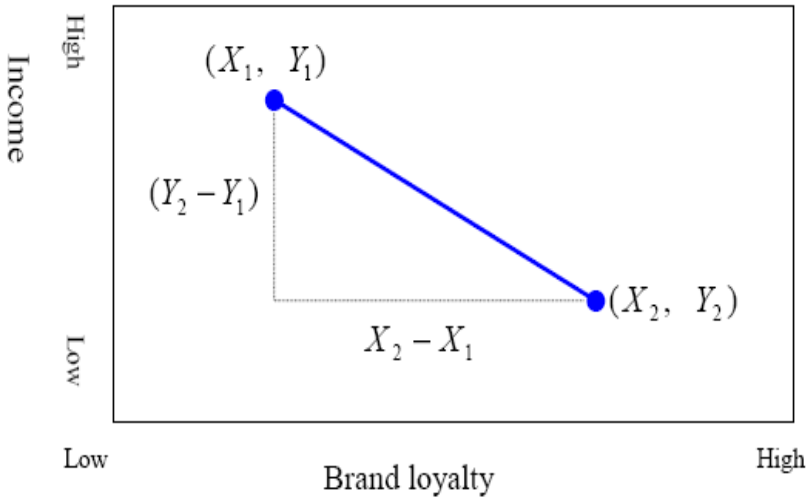
$$d_{ij} = d_{ji}, i, j = 1,2,...,n$$

$$d_{ij} + d_{jk} \geq d_{ik}, i, j, k = 1,2,...,n$$

- 개체간의 거리는 행렬을 이용하여 계산 유클리드안 거리/유클리드안 제곱 거리/시티-블록, 맨하탄 거리/코사인 거리/체비세프 거리/민코우스키 거리 등이 있음

e.g.

유클리드 거리



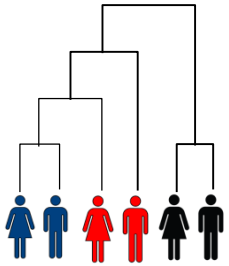
$$\text{Distance} = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$$

군집분석의 종류

군집분석

- **계층적 군집분석**: 단일, 완전, 평균, 중심결합기준, Ward 방식
- **비계층적 군집분석**: K-means, 코호넨 SOM

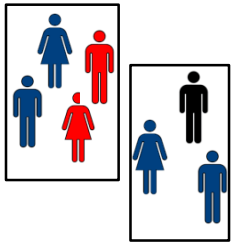
개별대상 간의 거리에 의하여 가장 가까이 있는 대상들로부터 시작하여 결합해 감으로써 나무모양의 계층구조를 형성해가는 방법



•장점: 군집이 형성되는 과정을 정확하게 파악할 수 있어 군집의 수 도출이 용이

•단점: 자료의 크기가 크면 분석하기 어려움.

군집의 수를 정한 상태에서 설정된 군집의 중심에 가장 가까운 개체를 하나씩 포함해 가는 방식으로 군집을 형성하는 방법

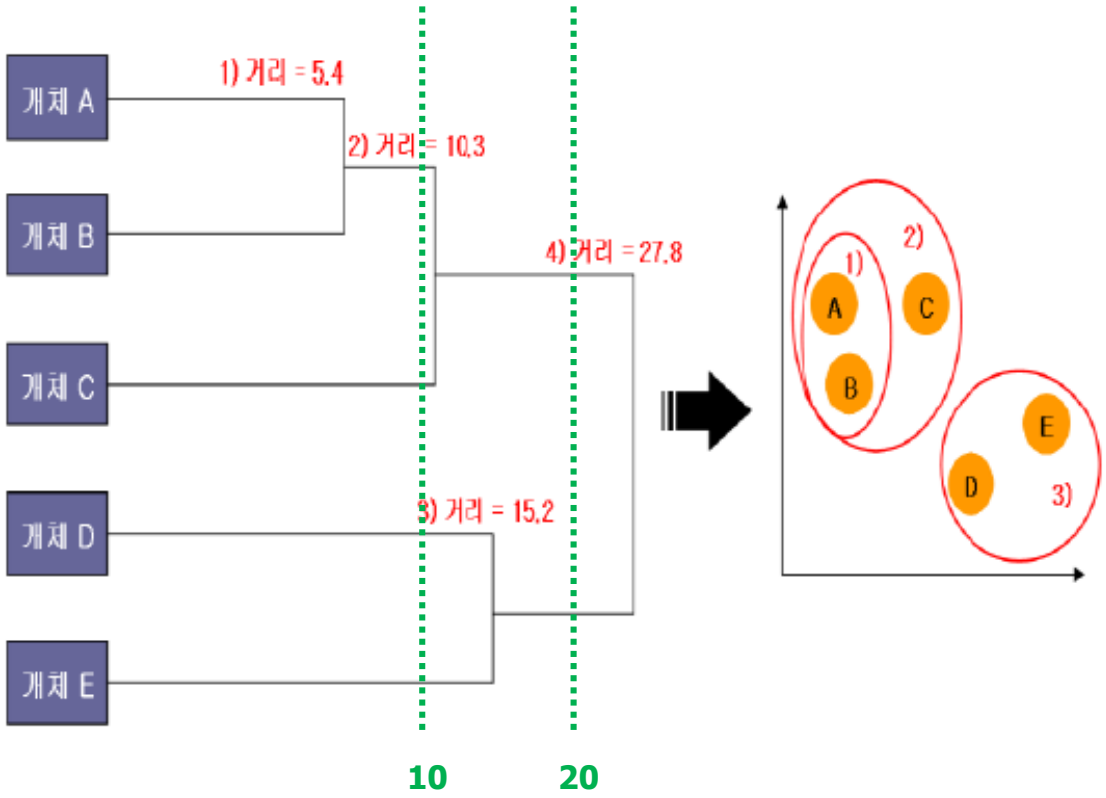


• 장점: 많은 자료를 빠르고 쉽게 분류

• 단점: 군집의 수를 미리 정해 주어야 하고, 군집을 형성하기 위한 초기값에 따라 군집결과가 달라짐.

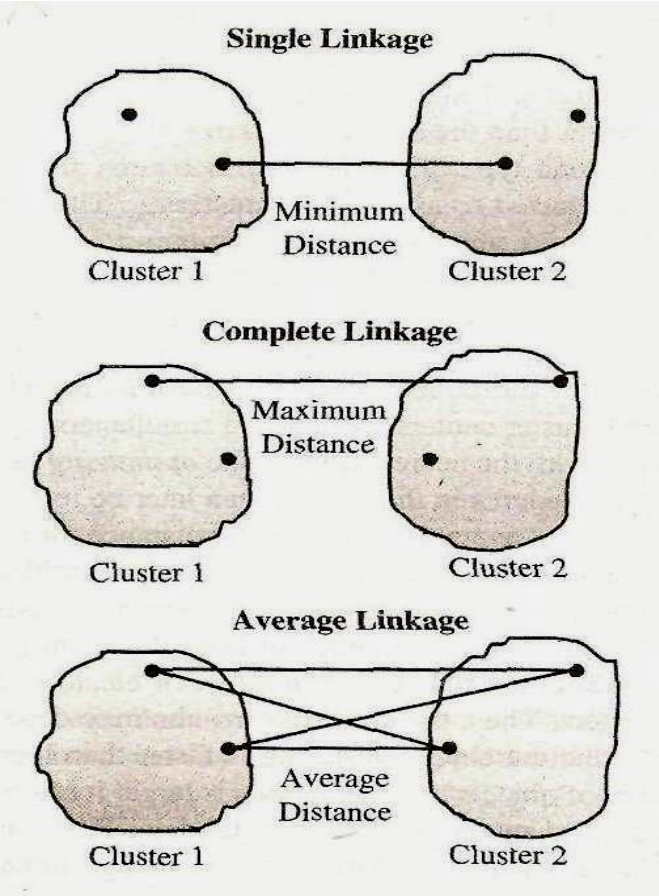
계층적 군집분석 (Hierarchical Clustering)

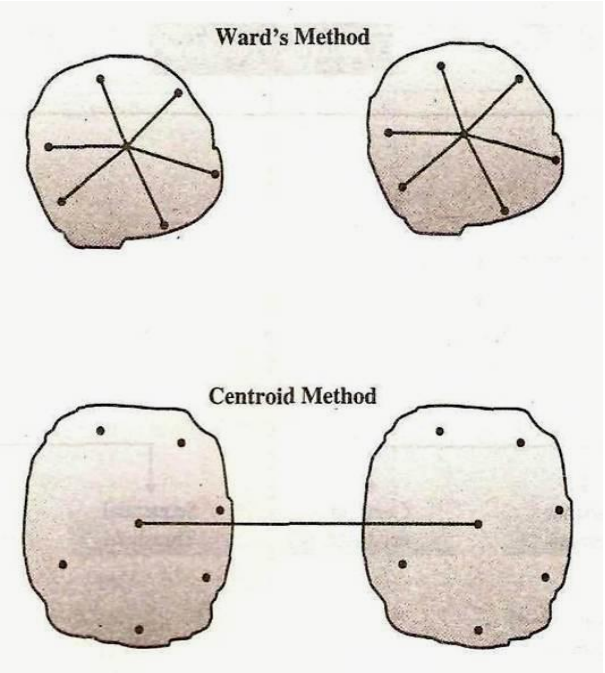
- 정확히 하나의 레코드로 구성된 군집들로 시작
- 최종적으로 모든 레코드들로 구성된 하나의 군집만이 남을 때까지, 가장 가까운 두 군집들을 점진적으로 병합해 나감



계층적 군집분석

- 병합방법(Linkage Method)





[Source: www.slideshare.net/neerajkaushik/cluster-analysis]

활용 예시: Retail Market

- 고객들을 군집화하여 VIP 고객군과 일반 고객군 간에 어떤 차이점이 있는지 파악
- 일반 고객들 중에서 VIP 고객 그룹에 더 가까운 고객을 대상으로 교차판매 마케팅 전략을 전개할 수 있음
- 전체 매장 또는 가맹점 가운데 유사한 성향을 보이는 매장끼리 군집화하여 차별화된 관리 가능

군집분석의 활용사례 (1/5)

산업(대)	산업(중)	주제(상황)	변수	결과
유통	백화점	• 쇼핑 성향을 통해 고객들을 군집화하고자 함.	• [7점 척도] 쇼핑은 흥미 있음, 쇼핑은 당신의 소득에 악영향을 끼침, 쇼핑을 하면서 외식을 즐김, 쇼핑 시 최고 제품을 구입하기 위한 노력, 쇼핑에 관심이 없음.	3개의 군집으로 도출됨. 군집1. 쇼핑의 흥미, 쇼핑을 하면서 외식을 즐김의 평균이 높음. 쇼핑에 관심이 없음은 평균이 낮음. → 쇼핑 애호가 군 군집2. 쇼핑의 흥미, 쇼핑을 하면서 외식을 즐김의 평균이 낮음. 쇼핑에 관심이 없음의 평균이 높음. → 냉담한 소비자 군 군집3. 쇼핑은 가계에 악영향, 쇼핑 시 최고의 상품을 구입하기 위한 노력의 평균이 높음. → 경제적인 소비자 군
서비스	호텔	• 호텔 종사원의 특성을 분류하고자 함.	• 사회적 책임활동, 조직몰입, 근속연수, 연령, 학력, 성별, 결혼여부	2개의 군집으로 도출됨. 군집1. 조직몰입, 근속연수, 연령의 평균이 높음. 군집2. 조직몰입은 높으나 근속연수, 연령의 평균이 낮음.

군집분석의 활용사례 (2/5)

산업(대)	산업(중)	주제(상황)	변수	결과
유통	백화점	• 고객 등급화	• 나이, 성별, 주소, 주거형태, 집 평수, 백화점 첫 이용 날짜, 구매일자, 항목, 구매액수, 결제수단, 첫 구매 시기	1. A백화점의 고객은 4개의 등급으로 분류됨. 2. 기존고객들의 등급 별 특성을 도출 후 신규고객을 유입하기 위한 방안을 모색함.
서비스	골프장	• 만족 유형을 이용한 집단 분류	• [골프 연습장의 만족척도] 시설, 요금, 대인서비스의 요인을 요인점수로 환산	5개의 군집으로 도출됨. 1. 시설 만족군 2. 전반적 만족군 3. 비용 만족군 4. 대인서비스 만족군 5. 전반적 불만족군
	커피 전문점	• 이용실태를 분석하여 집단 분류	• 테이크아웃 전문점 이용횟수, 이용 목적, 구입한 음식의 용도, 1회 평균 지출액	2개의 군집 중 군집1은 메뉴의 다양성, 매장 기기 및 기물의 청결성 속성에 높은 중요도를 나타냄. -> 다양한 메뉴 개발, 청결 서비스 전략을 수행해야 함.

군집분석의 활용사례 (3/5)

산업(대)	산업(중)	주제(상황)	변수	결과
공공	군수산업	• 여군의 새로운 군복 치수 결정	• 가슴, 목, 어깨둘레, 소매 바깥솔기, 목에서 엉덩이까지의 길이 등	20가지의 형태로 구성된 의복치수 군집이 도출됨
제조	목욕세제	• 마케팅전략을 도출하기 위해 고객을 분류하고자 함.	• 묶음판매 프로모션 구매 여부, 구매한 브랜드 수, 연속해서 구매한 브랜드 수, 구매거래 수, 인구통계량 정보	4개의 군집으로 도출됨 군집1. 구매가 가장 많이 일어나는 집단이긴 하나 구매하는 브랜드 수나 연속해서 구매하는 브랜드 건수가 많음. → 브랜드에 대한 충성도가 높은 집단으로 보긴 어려움. 군집2. 프로모션 반응율 가장 낮음. 군집3. 구매거래의 수에 비해 브랜드의 수가 적고 연속해서 구매한 브랜드의 건수가 많이 나타남. 군집4. 35세 이상의 여성이 주를 이룸.

군집분석의 활용사례 (4/5)

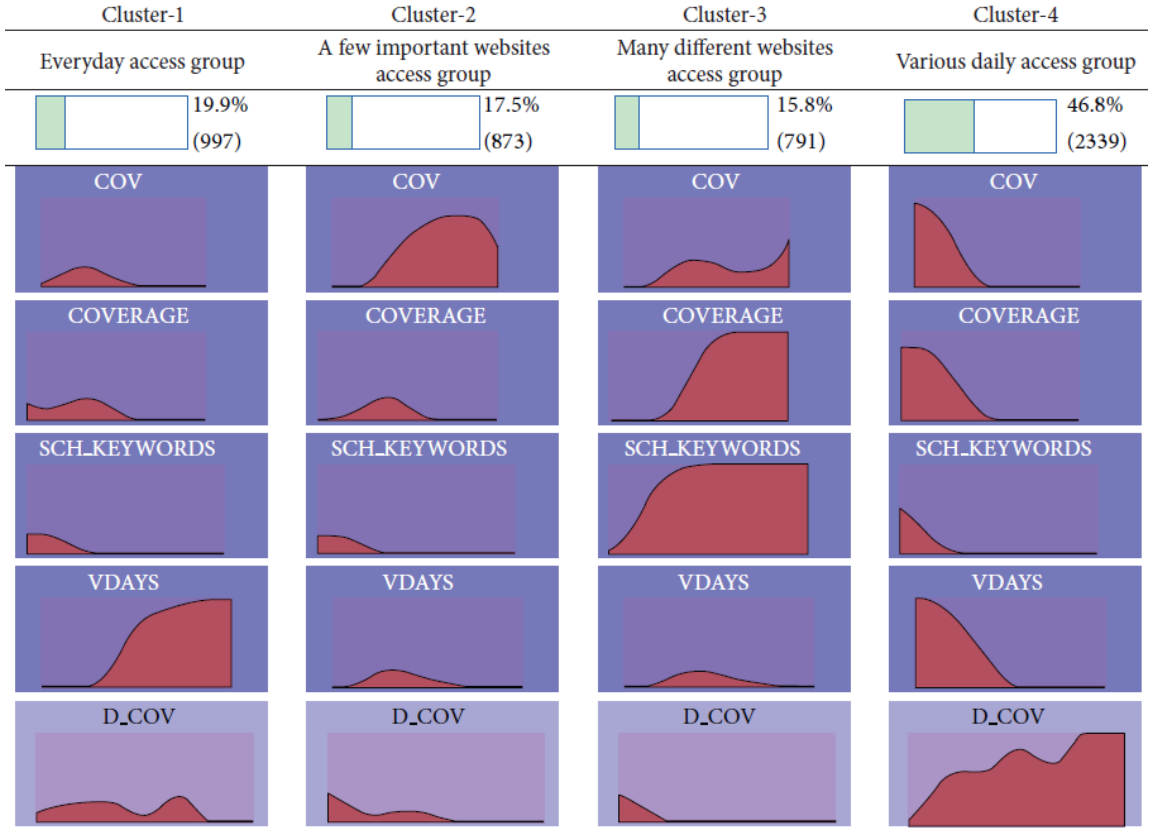
산업(대)	산업(중)	주제(상황)	변수	결과
서비스	온라인 게임	• 고객의 행위 특성을 분류	• 총 사용 시간, 총 사용 횟수, 에러 횟수, 한달 평균 사용시간	그리드 5X7의 군집이 가장 적합함. 각 군집의 특성별로 행위 유도 방안을 제시함. (접속횟수의 증가, 평균 사용시간의 증가, 총 사용시간의 증가, 접속횟수 및 총 사용시간의 증가, 사용 오류의 감소)
금융	은행	• 주택담보대출을 받은 고객의 특성을 파악하기 위해 분류	• 집의 감정가, 사용 가능한 신용 잔액, 주어진 대출금액, 연령, 결혼 상태, 자녀의 수, 가구 소득, 입출금 시스템, 신용카드 시스템	주택담보대출을 받은 고객의 군집엔 [대학에 진학하는 자녀들을 둔 고객층]에 많은 고객이 포함, 그들은 개인 계좌 뿐 아니라 기업 계좌도 보유하고 있음. → 자녀들이 대학을 가서 집을 떠나면 대출금을 이용하여 새로운 사업을 시작할 기회를 엿봄. → 이를 통해 은행은 집의 빈자리를 이용한 사업을 꾸려나갈 부모들을 목표로 하는 새로운 마케팅 프로그램을 생성함.

군집분석의 활용사례 (5/5)

산업(대)	산업(중)	주제(상황)	변수	결과
금융	은행	• 인터넷 뱅킹 고객의 특성요인을 통해 고객을 군집화하고자 함.	• 유용성, 사용편의성, 신뢰성, 위험성, 실제사용	4개의 군집으로 도출됨. 군집1. 편의성 요인을 높일 수 있는 마케팅 전략이 필요함. 군집2. 타 군집군에 비해 인터넷 뱅킹 이용률이 매우 저조함 → 충성도를 높이는 전략이 필요함. 군집3. 자사의 인터넷 뱅킹 이용이 안전하다는 인식을 지속적으로 알려 주는 마케팅 전략이 필요함. 군집4. 핵심 우량 고객군 → 차별화된 고객관리가 필요함. (ex. 은행 수수료 감면 or 다양한 개인별 금융정보 서비스를 실시할 수 있음)
	증권	• 균형 포트폴리오를 구성하기 위해 투자대상 기업을 분류	• 수익(일별, 주별 또는 월별), 가격변동률, 베타, 자본총액 등	도출된 서로 다른 군집으로부터 주식을 선택하여 위험을 분산

군집분석의 평가 (1/2)

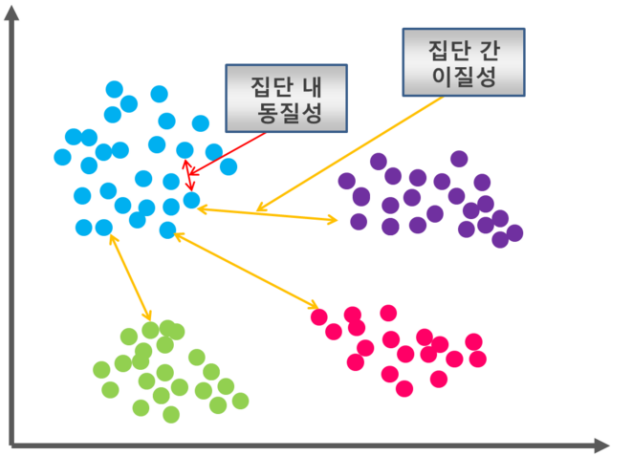
- 군집의 설명 가능성
  - 군집분석에 사용된 각 변수들에 대해 각 군집의 요약통계량(평균, 최소, 최대 등)을 구한다.
  - 해석에 기초해서 각 군집에 어울리는 이름 또는 라벨을 할당한다.



군집분석의 평가 (2/2)

- 군집의 안정성
  - 입력 값이 약간 달라지더라도 군집이 유의하게 변하지 않아야 좋은 결과이다.
  - 데이터를 2개로 분할한 후 한 쪽 데이터에 근거하여 형성된 군집이 나머지 데이터에 얼마나 동일하게 잘 적용되는지(일치하는지) 살펴본다.
  - [관련 문헌] 허명회 & 이용구, “클레멘타인을 활용한 K-평균 군집화 결과의 재현성 평가”, SPSS White Paper, 2003.

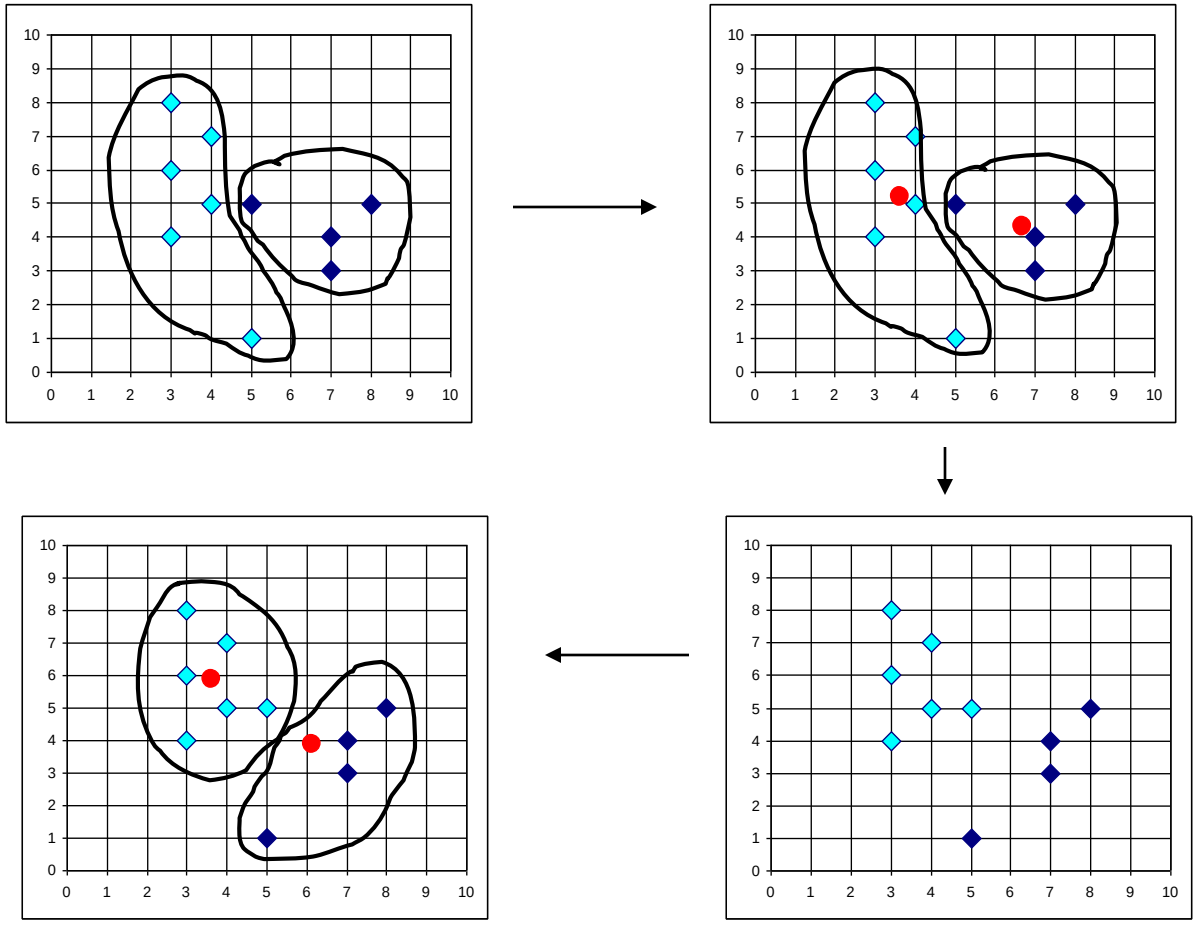
- 군집의 분리
  - 군집의 분리가 타당한지를 살펴보기 위해 군집 내 분산(cluster cohesion)에 대한 군집 간 분산(cluster separation)의 비를 검토한다.
  - 좋은 결과는 각 군집 내 분산은 최소로, 군집 간 분산은 최대로 되는 것이다.
  - 대표적인 평가지표
    - Silhouette measure
    - Akaike information criterion (AIC)
    - Bayesian information criterion (BIC)
    - Deviance information criterion (DIC)



K-평균 군집분석(K-Means algorithm)

- 단계 0 : 사전적으로 군집의 수 K를 지정한다.
- 단계 1 : 각 군집에 1개의 군집 중심을 임의로 정한다.
  (보통 서로 상당히 떨어진 개체를 선택함)
- 단계 2 : 모든 개체를 각각 가장 가까운 군집 중심에
  배속시킨다.
- 단계 3 : 각 군집의 중심을 산출한다.
- 단계 4 : 단계 2와 단계3을 변화가 거의 없을 때까지
  (보통 10회 이하) 반복한다.

K-평균 군집분석 과정



K-평균 군집분석 예제

▶1단계 최초 군집화

개체	변수1	변수1
1	1,0	1,0
2	1,5	2,0
3	3,0	4,0
4	5,0	7,0
5	3,5	5,0
6	4,5	5,0
7	3,5	4,5
8	6,0	2,0
9	5,0	3,0
10	6,5	3,0
11	7,0	3,5

최초 임의의 중심점 생성

(1-1.5)^2 + (1-2)^2
=1.25

개체	군집1의 중심(개체1)과의 거리 (√)	군집2의 중심(개체4)과의 거리 (√)	군집3의 중심(개체8)과의 거리 (√)
2	1,25	37,25	20,25
3	13	13	13
5	22,25	6,25	15,25
6	28,25	4,25	11,25
7	18,25	9,25	12,5
9	28,25	6,25	3,25
10	34,25	18,25	1,25
11	42,25	16,25	3,25

최초 임의의 중심점과 개체간의 거리계산

군집	개체	변수1	변수2	평균 (변수1, 변수2)	
1	1	1,0	1,0	1,84	2,34
	2	1,5	2,0		
	3	3,0	4,0		
2	4	5,0	7,0	4,125	5,375
	5	3,5	5,0		
	6	4,5	5,0		
	7	3,5	4,5		
3	8	6	2	6,125	2,88
	9	5	4,5		
	10	6,5	3		
	11	7	3,5		

최초 발생한 그룹들의 평균을 생성

▶2단계 반복

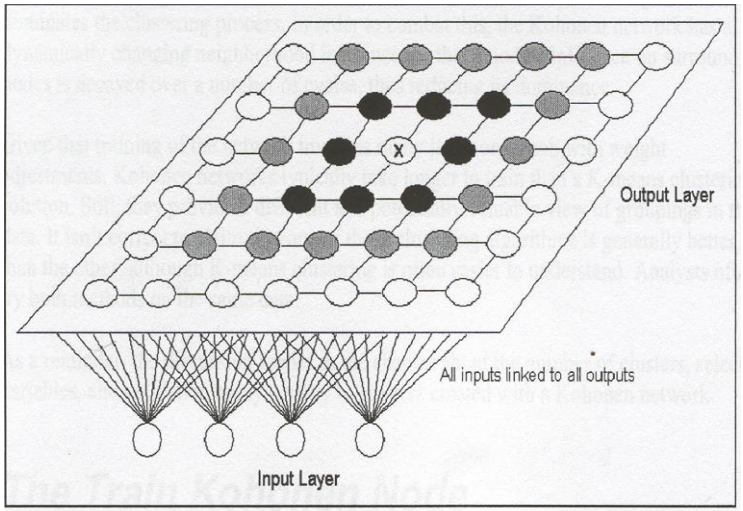
군집중심이 **(1,1) →(1.84, 2.34)**로 이동하였고,
이 중심을 기준으로 다시 군집간 개체 거리 계산을 하여, 변화가 없거나, 특정한 반복 수만큼 수행하고 멈추며, 멈추는 그 시점의 평균값이 곧 군집 중심이 되며, 그 중심으로부터의 거리가 최종 유사성 척도가 됨.

Kohonen SOM 군집분석 (1/7)

SOM (Self-Organizing Map)은 자기조직화 지도라는 것으로 관측개체들을 스스로 조직화하여 지도의 형태로 뿌려주는 신경망 기법이다. SOM (자기조직화 지도)개념도는 2개의 층으로 이루어져 있으며, 첫 번째 층이 입력 층(input layer), 두 번째 층은 출력 층(output layer)으로 이루어진 2차원 격자(grid)로 되어 있다.

Kohonen SOM Clustering

Kohonen SOM의 개념도

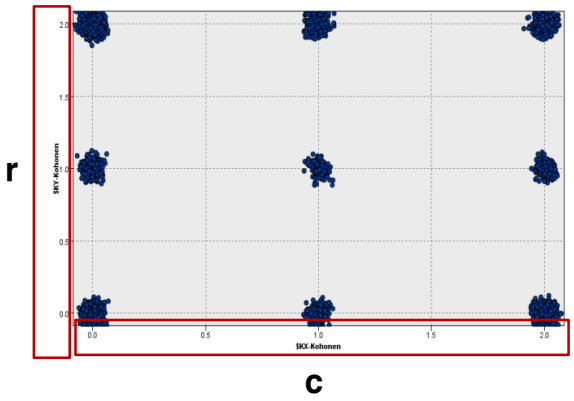


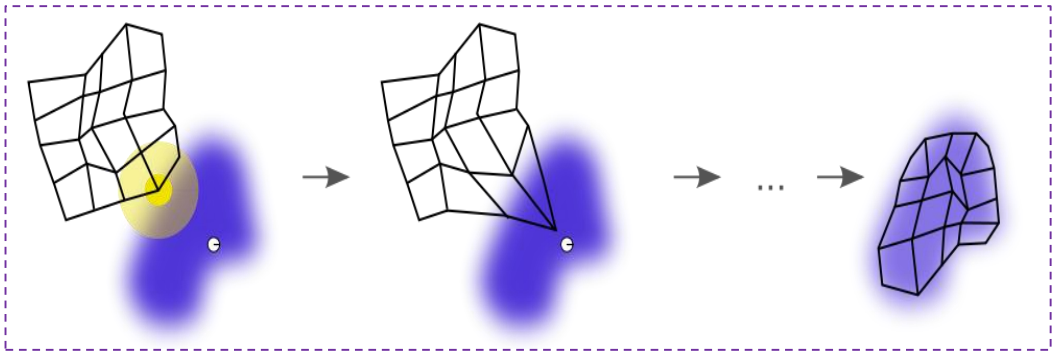
Kohonen SOM의 사전작업

- SOM에 사용할 변수를 미리 지정해야 한다.
- 사용 변수들은 주 알고리즘의 적용에 앞서 표준화 되어야 한다.
- 2차원 SOM의 경우 그리드의 크기를 미리 결정한다.

Kohonen SOM 군집분석 (2/7)

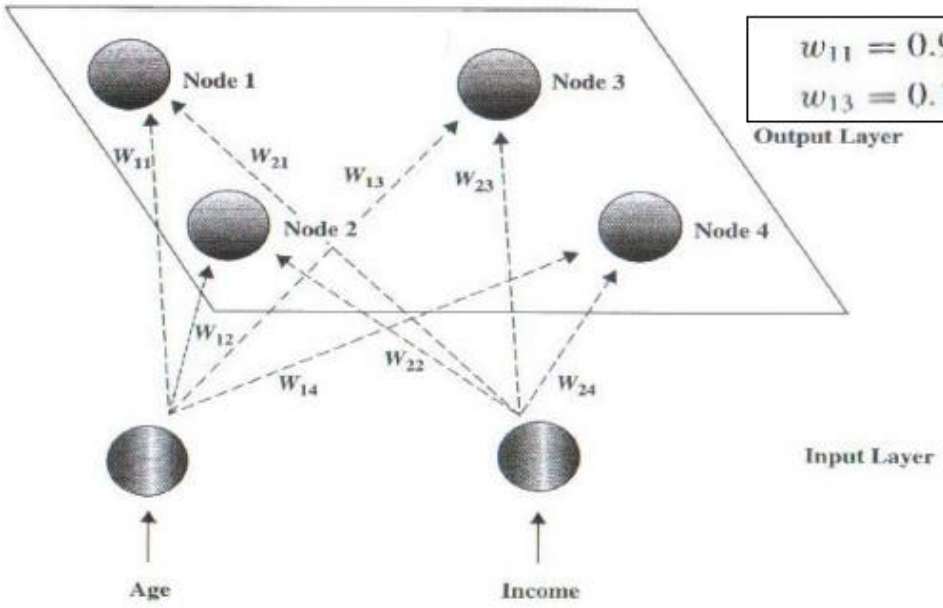
Kohonen Method	
Step 1	2차원 그리드 상에 출력 노드(r × c)들을 깔아 놓는다. [initialization]디폴트10 × 7
Step 2	입력벡터(입력노드)를 표준화(Standardization) 한다. (범위: 0-1)
Step 3	각 레코드는 가장 유사한 출력노드, 중량 벡터(weight vector)를 찾아 간다. : winning node (or winner)
Step 4	각 레코드는 그리드 상의 winner와 그 이웃의 노드들을 동일한 방향으로 학습(업데이트)시킨다.
Step 5	이웃의 범위와 업데이트의 정도를 점차 줄임으로써 수렴 해를 얻는다.





Kohonen SOM 군집분석 (3/7)

- 예제)
  - 고객의 나이와 수입으로 군집화
  - 출력 층: 2×2 격자



초기 가중치 값

$w_{11} = 0.9$	$w_{21} = 0.8$	$w_{12} = 0.9$	$w_{22} = 0.2$
$w_{13} = 0.1$	$w_{23} = 0.8$	$w_{14} = 0.1$	$w_{24} = 0.2$

(Source: Larose, D.T., Discovering Knowledge in Data: An Introduction to Data Mining, 2004.)

Kohonen SOM 군집분석 (4/7)

- 1단계: 경쟁

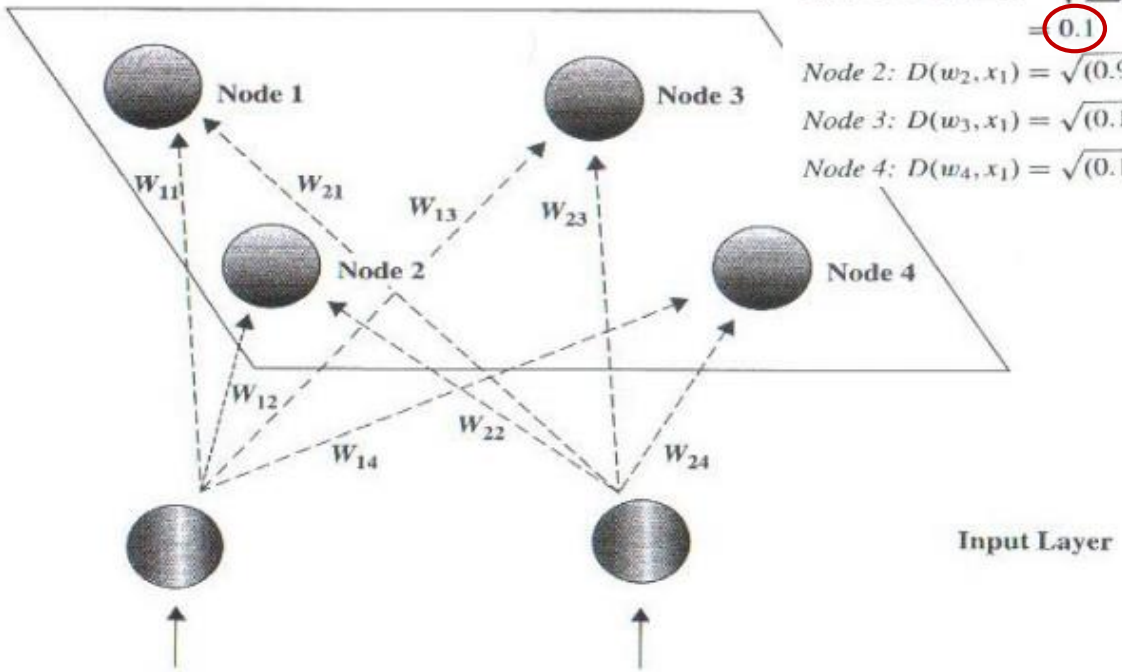
$$w_{11} = 0.9 \quad w_{21} = 0.8 \quad w_{12} = 0.9 \quad w_{22} = 0.2$$
$$w_{13} = 0.1 \quad w_{23} = 0.8 \quad w_{14} = 0.1 \quad w_{24} = 0.2$$

$$\text{Node 1: } D(w_1, x_1) = \sqrt{\sum_i (w_{i1} - x_{1i})^2} = \sqrt{(0.9 - 0.8)^2 + (0.8 - 0.8} = 0.1$$

$$\text{Node 2: } D(w_2, x_1) = \sqrt{(0.9 - 0.8)^2 + (0.2 - 0.8)^2} = 0.61$$

$$\text{Node 3: } D(w_3, x_1) = \sqrt{(0.1 - 0.8)^2 + (0.8 - 0.8)^2} = 0.70$$

$$\text{Node 4: } D(w_4, x_1) = \sqrt{(0.1 - 0.8)^2 + (0.2 - 0.8)^2} = 0.92$$



1	$x_{11} = 0.8$	$x_{12} = 0.8$	Older person with high income
2	$x_{21} = 0.8$	$x_{22} = 0.1$	Older person with low income
3	$x_{31} = 0.2$	$x_{32} = 0.9$	Younger person with high income
4	$x_{41} = 0.1$	$x_{42} = 0.1$	Younger person with low income

(Source: Larose, D.T., Discovering Knowledge in Data: An Introduction to Data Mining, 2004.)

Kohonen SOM 군집분석 (5/7)

- 2단계: 조정

$$w_{ij,new} = w_{ij,current} + \alpha(x_{ni} - w_{ij,current})$$

학습률 α=0.5라고 가정하면

$$\text{For } age: \quad w_{11,new} = w_{11,current} + 0.5(x_{11} - w_{11,current})$$
$$= 0.9 + 0.5(0.8 - 0.9) = 0.85$$
$$\text{For } income: \quad w_{21,new} = w_{21,current} + 0.5(x_{12} - w_{21,current})$$
$$= 0.8 + 0.5(0.8 - 0.8) = 0.8$$

(Source: Larose, D.T., Discovering Knowledge in Data: An Introduction to Data Mining, 2004.)

Kohonen SOM 군집분석 (6/7)

- SOM은 어떻게 패턴을 인식하나?
  - 승리한 단위에 대한 가중치뿐만 아니라 승리한 단위 바로 근처(neighborhood)에 있는 단위들의 가중치 역시 입력에 대한 응답을 강화하도록 조정
    - 근접도(Neighborliness) 매개변수
      - 근접한 단위들의 범위와 조정의 정도 조절
    - 서로 비슷한 군집들이 더 가까이 뭉치도록 해줌
    - 단위들의 집단이 하나의 군집이 될 수 있도록 해줌

(Source: 경영데이터마이닝 by K-MOOC)

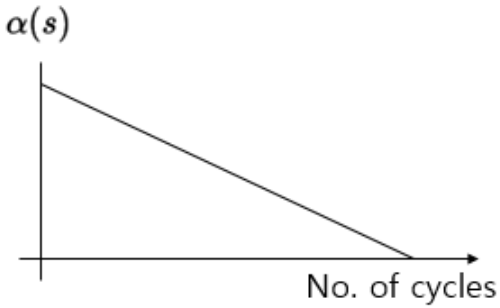
Kohonen SOM 군집분석 (7/7)

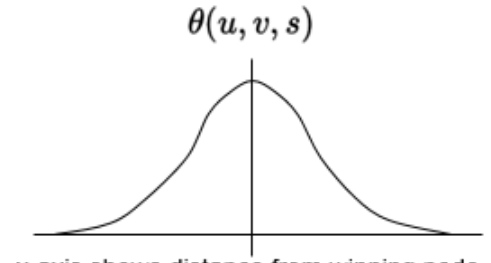
Formal Algorithm

1. Randomize the map's nodes' weight vectors
2. Traverse each input vector in the input data set
   1. Traverse each node in the map
      1. Use the Euclidean distance formula to find the similarity between the input vector and the map's node's weight vector
      2. Track the node that produces the smallest distance (this node is the best matching unit, BMU)
   2. Update the nodes in the neighborhood of the BMU (including the BMU itself) by pulling them closer to the input vector

$$W_v(s+1) = W_v(s) + \theta(u,v,s) \cdot \alpha(s) \cdot (D(t) - W_v(s))$$

3. Increase s and repeat from step 2 while s < λ

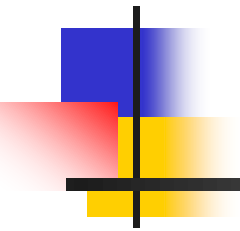




–x-axis shows distance from winning node

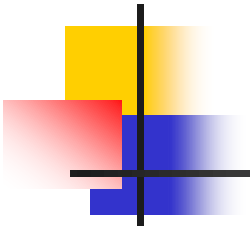
–y-axis shows 'degree of neighbourhood' (max. 1)

For more detailed SOM examples,
refer to https://genome.tugraz.at/MedicalInformatics2/SOM.pdf



군집분석(Clustering Analysis)

R을 활용한 군집분석 실습



K-means Clustering with R

- No package installation
- Related functions
  - kmeans()
  - plot()
  - ggplot(), gplot() – ggplot2 package
- Other R functions for K-means clustering
  - kcca{flexclust}
  - cclust{flexclust}
  - cclust{cclust}
  - Kmeans{amap}

K-means Clustering with R

```
install.packages("ggplot2"); library(ggplot2)
cdata <- read.delim("Cluster.txt", stringsAsFactors=FALSE)

# 군집수를 4로 하는 k-means clustering
set.seed(1)
km <- kmeans(subset(cdata, select=-c(ID)), centers=4)
str(km)
```

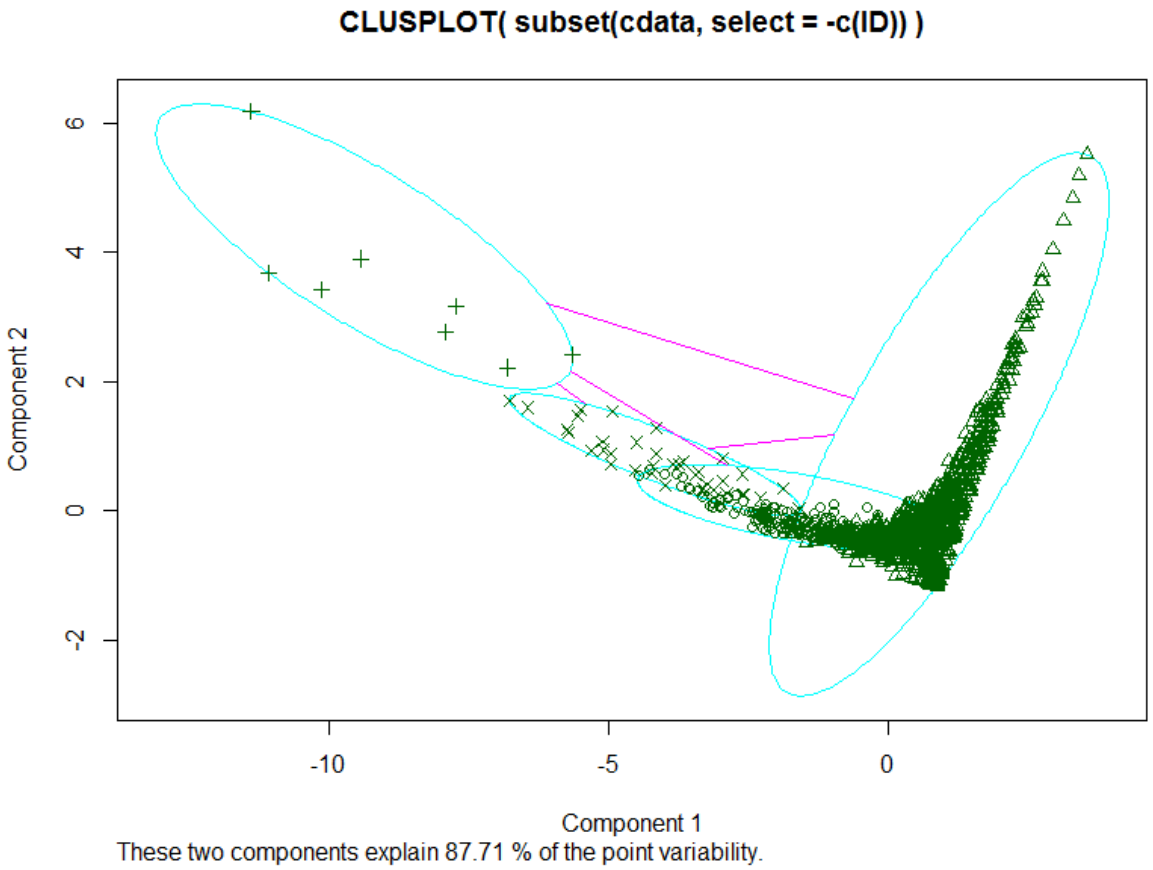
```
List of 9
 $ cluster     : Named int [1:1000] 2 2 2 2 2 2 1 2 2 2 ...
  ..- attr(*, "names")= chr [1:1000] "1" "2" "3" "4" ...
 $ centers     : num [1:4, 1:4] 1.28e+06 2.55e+05 1.13e+07 3.42e+06 3.26e+01 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:4] "1" "2" "3" "4"
  .. ..$ : chr [1:4] "MONEY" "VISIT" "CROSS" "API"
 $ totss       : num 1.65e+15
 $ withinss    : num [1:4] 2.80e+13 2.77e+13 1.42e+14 3.39e+13
 $ tot.withinss: num 2.32e+14
 $ betweenss   : num 1.42e+15
 $ size        : int [1:4] 181 770 8 41
 $ iter        : int 3
 $ ifault      : int 0
 - attr(*, "class")= chr "kmeans"
```

```
km
```

```
K-means clustering with 4 clusters of sizes 181, 770, 8, 41

Cluster means:
       MONEY      VISIT     CROSS       API
1  1284818.8  32.607735 10.966851  7.232044
2   255051.4   8.124675  3.654545 34.877922
3 11323243.8 101.125000 20.750000  0.750000
4  3421840.0  62.146341 15.390244  3.024390
```

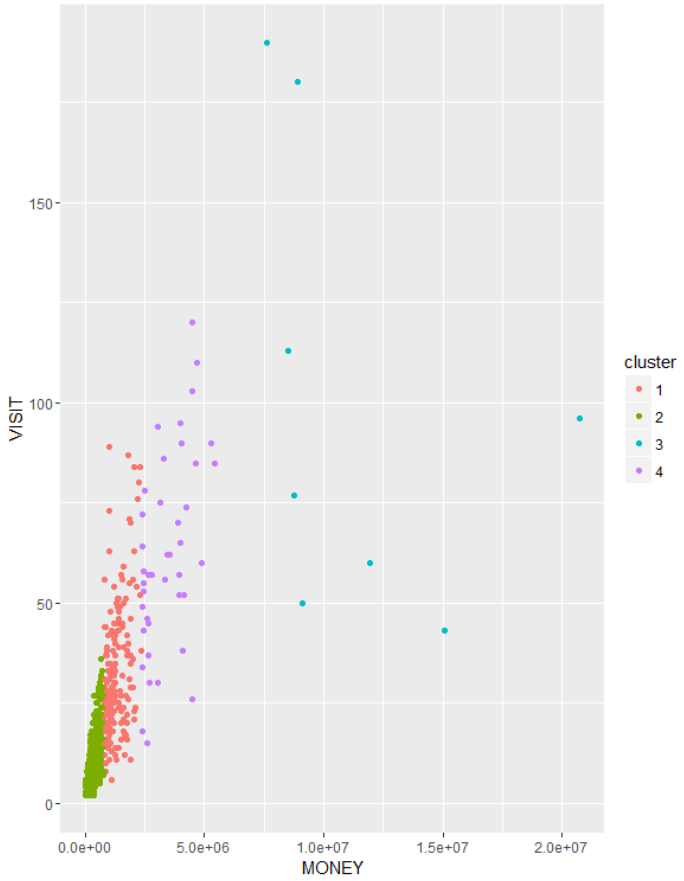
K-means Clustering with R

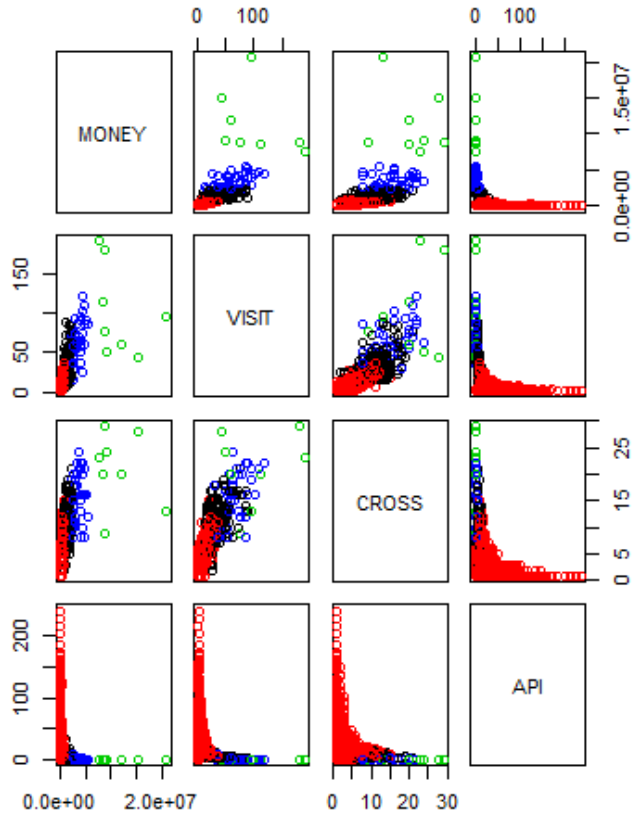
- `# 군집의 반경과 관계를 2차원으로 도식`
- `install.packages("cluster"); library("cluster")`
- `clusplot(subset(cdata, select=-c(ID)), km$cluster)`



K-means Clustering with R

```
# 군집의 분포를 도식
cdata$cluster <- as.factor(km$cluster)
qplot(MONEY, VISIT, colour=cluster, data=cdata)                    # left
plot(subset(cdata, select=-c(ID,cluster)), col=km$cluster)         # right
```

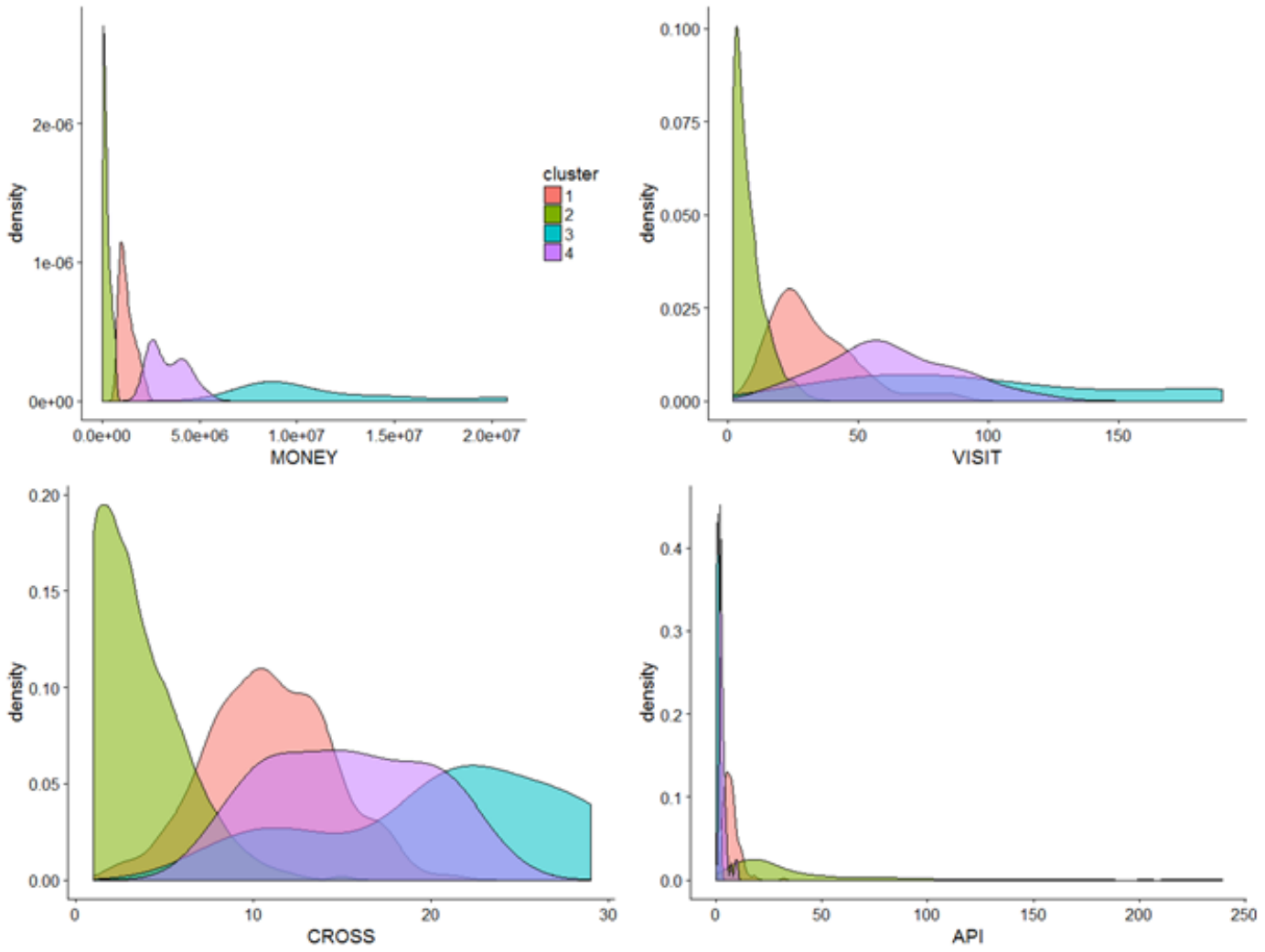




Cluster comparison & interpretation

```
# 특정 군집화변수에 대한 군집별 밀도를 도식: 방법1

install.packages("gridExtra"); library(gridExtra)
install.packages("scales"); library(scales)

p1 <- qplot(MONEY, fill=cluster, alpha=.5, data=cdata, geom="density") +
scale_alpha(guide="none")

p2 <- qplot(VISIT, fill=cluster, alpha=.5, data=cdata, geom="density") +
theme(legend.position="none")

p3 <- qplot(CROSS, fill=cluster, alpha=.5, data=cdata, geom="density") +
theme(legend.position="none")

p4 <- qplot(API, fill=cluster, alpha=.5, data=cdata, geom="density") +
theme(legend.position="none")

grid.arrange(p1, p2, p3, p4, ncol=2, nrow=2)
```

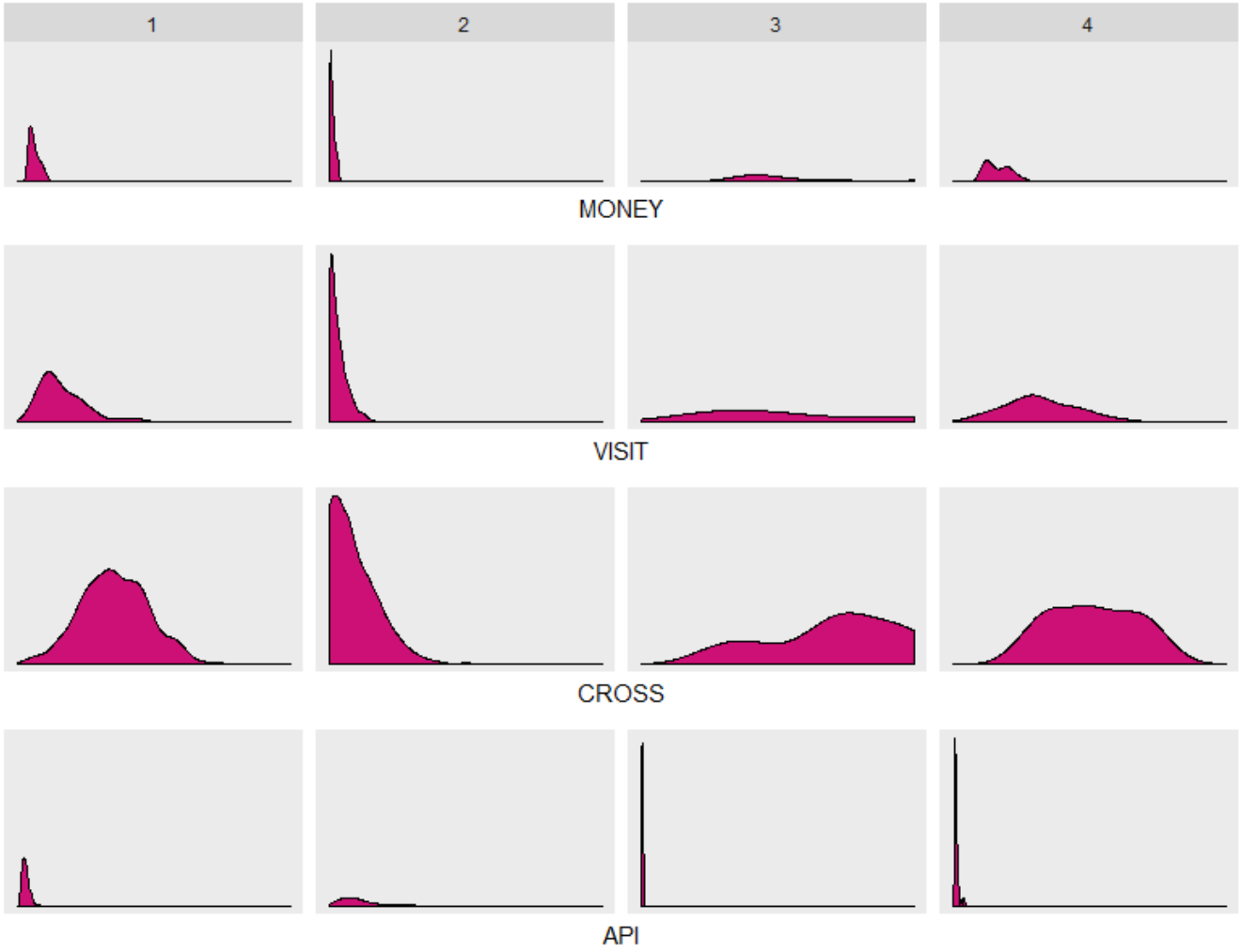
Cluster comparison & interpretation



Cluster comparison & interpretation

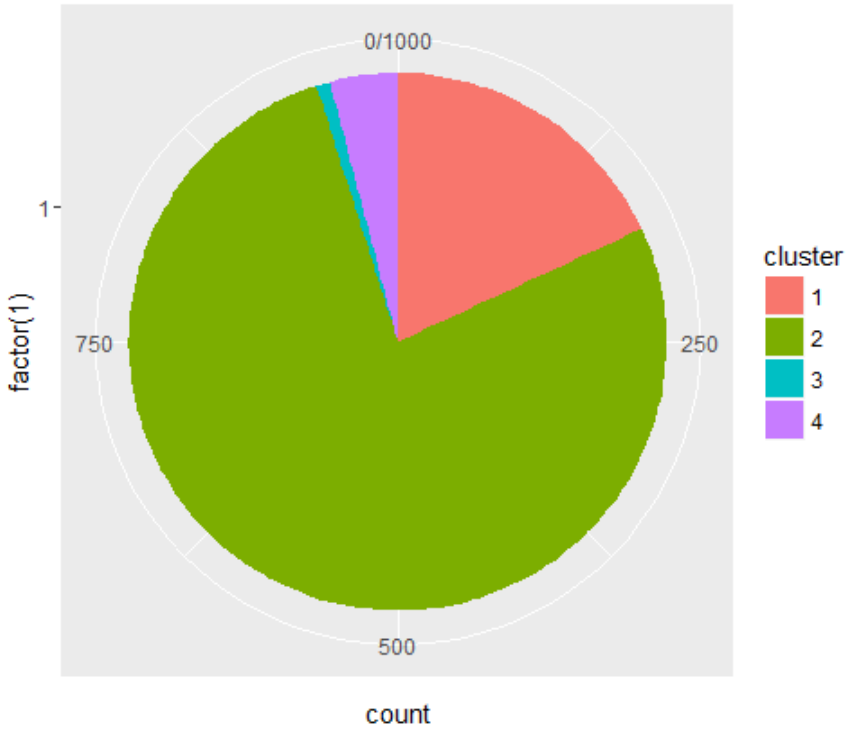
```
# 군집별로 각 군집화변수의 밀도를 도식: 방법2

p1 <- ggplot(cdata, aes(MONEY)) + geom_density(fill='deeppink3', adjust=1)
+ facet_grid(. ~ cluster) + scale_x_continuous(breaks=NULL) +
scale_y_continuous("", breaks=NULL)

p2 <- ggplot(cdata, aes(VISIT)) + geom_density(fill='deeppink3', adjust=1)
+ facet_grid(. ~ cluster) + scale_x_continuous(breaks=NULL) +
scale_y_continuous("", breaks=NULL) + theme(strip.text.x=element_blank())

p3 <- ggplot(cdata, aes(CROSS)) + geom_density(fill='deeppink3', adjust=1)
+ facet_grid(. ~ cluster) + scale_x_continuous(breaks=NULL) +
scale_y_continuous("", breaks=NULL) + theme(strip.text.x=element_blank())

p4 <- ggplot(cdata, aes(API)) + geom_density(fill='deeppink3', adjust=1) +
facet_grid(. ~ cluster) + scale_x_continuous(breaks=NULL) +
scale_y_continuous("", breaks=NULL) + theme(strip.text.x=element_blank())

grid.arrange(p1, p2, p3, p4, ncol=1, nrow=4)
```

Cluster comparison & interpretation



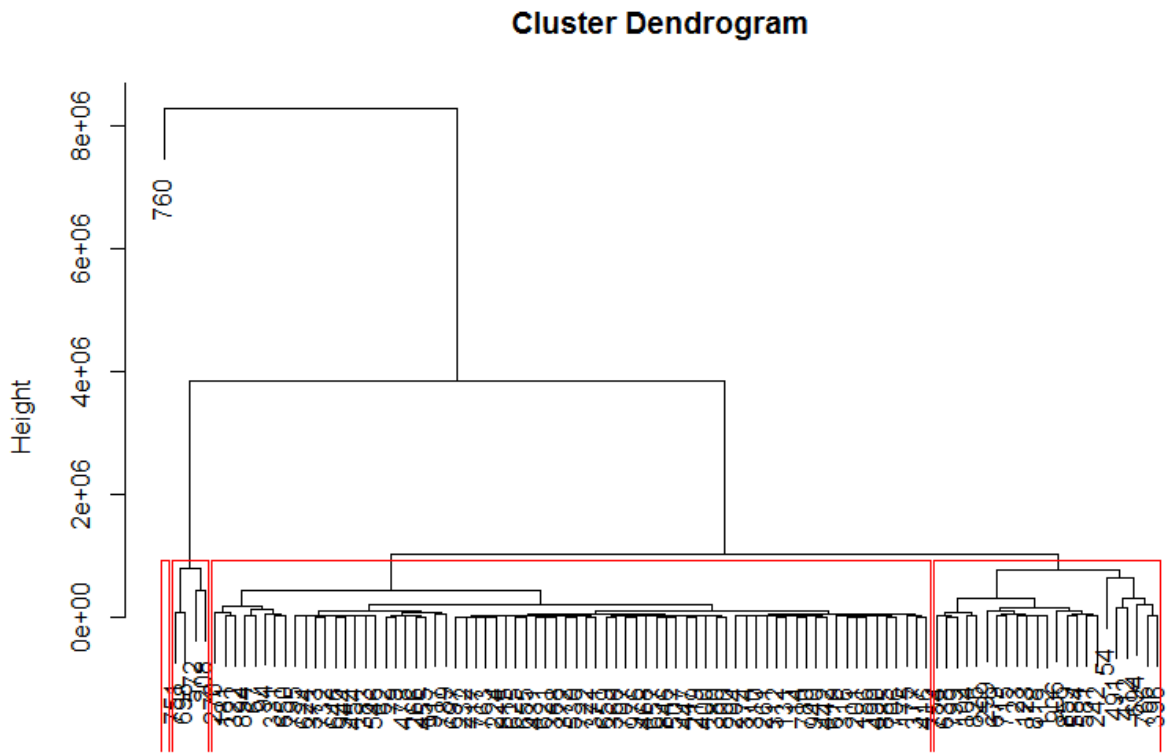
Cluster comparison & interpretation

```
# 군집의 크기를 도식
x <- ggplot(cdata, aes(x=factor(1), fill=cluster))
x + geom_bar(width=1) + coord_polar(theta="y")
```



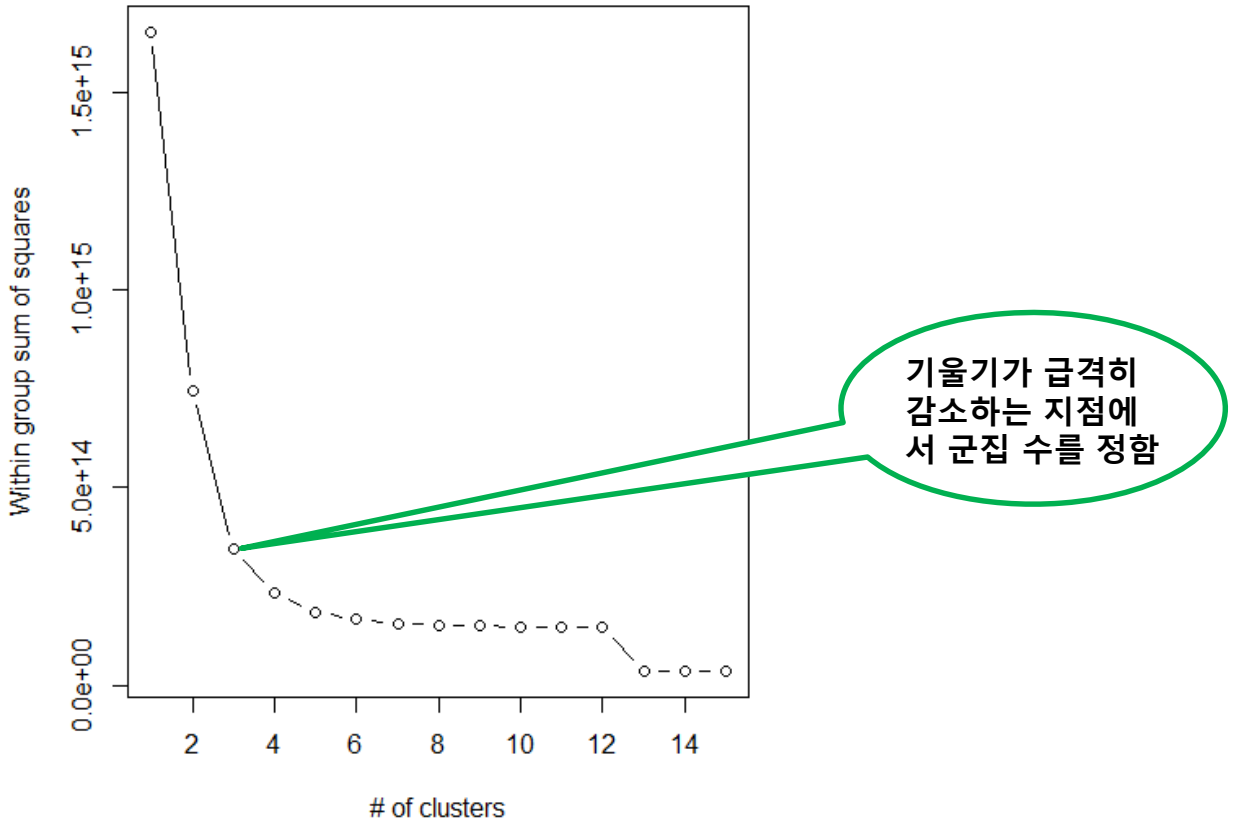
Determining the optimal number of clusters

```
# 최적의 군집 수 찾기: 방법1
set.seed(1)
sd <- cdata[sample(1:nrow(cdata),100),-1]
d <- dist(sd, method="euclidean")
fit <- hclust(d, method="ave")
plot(fit)
rect.hclust(fit, k=4, border = "red")
```



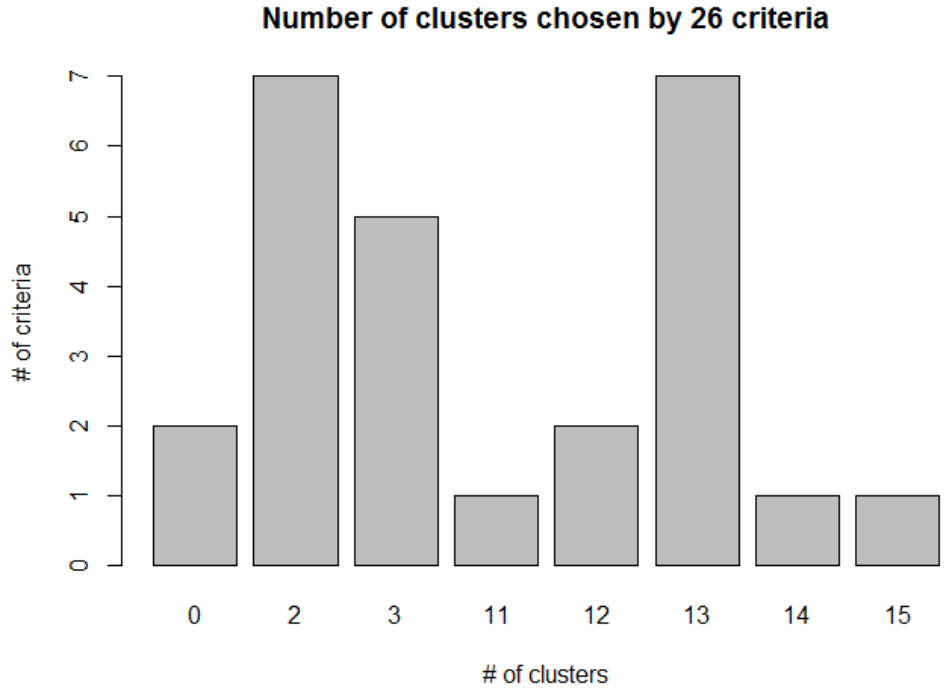
Determining the optimal number of clusters

```
# 최적의 군집 수 찾기: 방법2
wss <- 0
for(i in 1:15) wss[i] <- kmeans(cdata, centers=i)$tot.withinss
plot(1:15, wss, type="b", xlab="# of clusters", ylab="Within group
sum of squares")
```



Determining the optimal number of clusters

```
# 최적의 군집 수 찾기: 방법3
install.packages("NbClust"); library("NbClust")
nc = NbClust(subset(cdata, select=-c(ID,cluster)), min.nc=2, max.nc=15, method="kmeans")
barplot(table(nc$Best.nc[1,]), xlab="# of clusters", ylab="# of criteria", main="Number of clusters chosen by 26 criteria")
```




```
* According to the majority rule, the best number of clusters is 2
```

SOM Clustering with R

- Using "kohonen" or "som" packages
- Related functions
  - scale() – kohonen package
  - normalize() – som package
  - som(data, xdim, ydim) – som package
  - som(data, grid=somgrid(xdim, ydim, "rectangular“)) – kohonen package
  - plot()

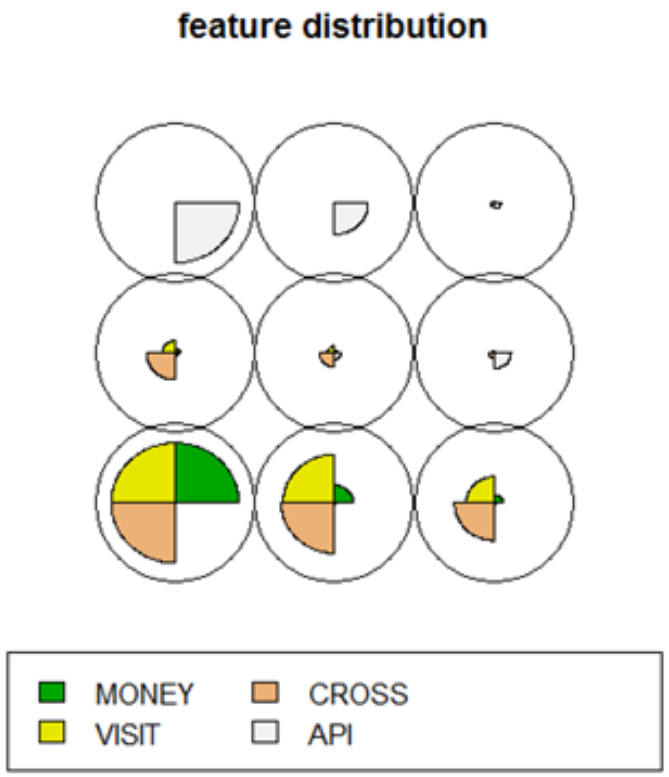
SOM Clustering with R

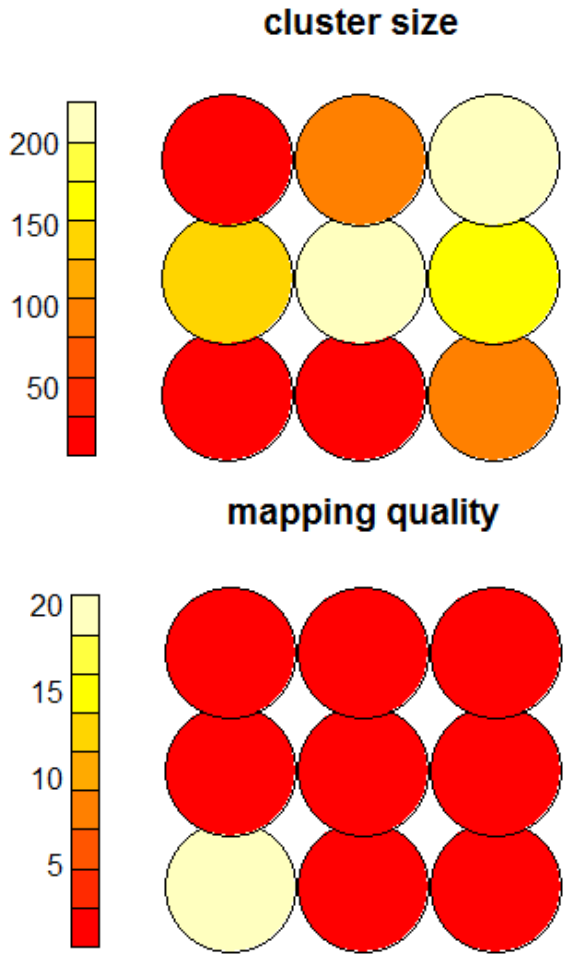
```
install.packages("kohonen"); library(kohonen)
cdata <- read.delim("Cluster.txt", stringsAsFactors=FALSE)

# 데이터 정규화
cdata.n <- scale(subset(cdata, select=-c(ID)))

# 그리드를  3 x 3으로 하는 SOM clustering
set.seed(1)
sm <- som(data = cdata.n, grid = somgrid(3, 3, "rectangular"))
str(sm)
```

```
List of 10
 $ data        : num [1:1000, 1:4] -0.227 -0.463 -0.15 -0.466 -0.414 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "MONEY" "VISIT" "CROSS" "API"
  ..- attr(*, "scaled:center")= Named num [1:4] 659823.1 15.5 5.6 28.3
  .. ..- attr(*, "names")= chr [1:4] "MONEY" "VISIT" "CROSS" "API"
  ..- attr(*, "scaled:scale")= Named num [1:4] 1.29e+06 1.92e+01 4.67 3.22e+01
  .. ..- attr(*, "names")= chr [1:4] "MONEY" "VISIT" "CROSS" "API"
 $ grid        :List of 5
  ..$ pts   : int [1:9, 1:2] 1 2 3 1 2 3 1 2 3 1 ...
  .. ..- attr(*, "dimnames")=List of 2
  .. .. ..$ : NULL
  .. .. ..$ : chr [1:2] "x" "y"
  ..$ xdim  : num 3
  ..$ ydim  : num 3
  ..$ topo  : chr "rectangular"
  ..$ n.hood: chr "square"
  ..- attr(*, "class")= chr "somgrid"
 $ codes       : num [1:9, 1:4] 8.546 2.183 0.775 0.269 -0.111 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "MONEY" "VISIT" "CROSS" "API"
 $ changes     : num [1:100, 1] 0.0202 0.021 0.0212 0.0224 0.0221 ...
 $ alpha       : num [1:2] 0.05 0.01
 $ radius      : num [1:2] 2 -2
 $ toroidal    : logi FALSE
 $ unit.classif: int [1:1000] 9 8 5 8 6 5 4 5 9 4 ...
 $ distances   : num [1:1000] 0.309 0.752 0.273 0.146 0.055 ...
 $ method      : chr "som"
```

SOM Clustering with R

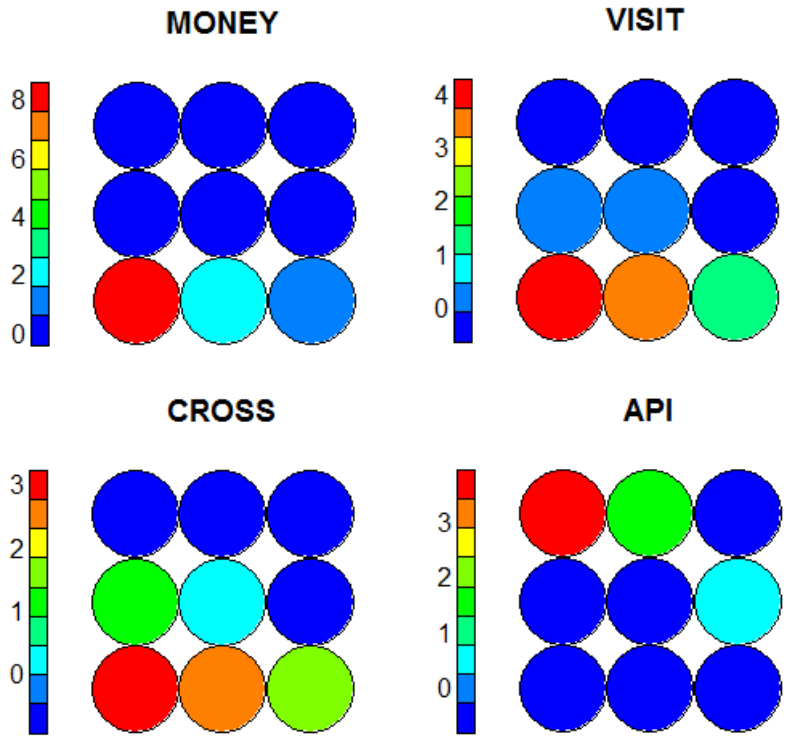
```
plot(sm, main = "feature distribution")
plot(sm, type="counts", main = "cluster size")
plot(sm, type="quality", main = "mapping quality")
```





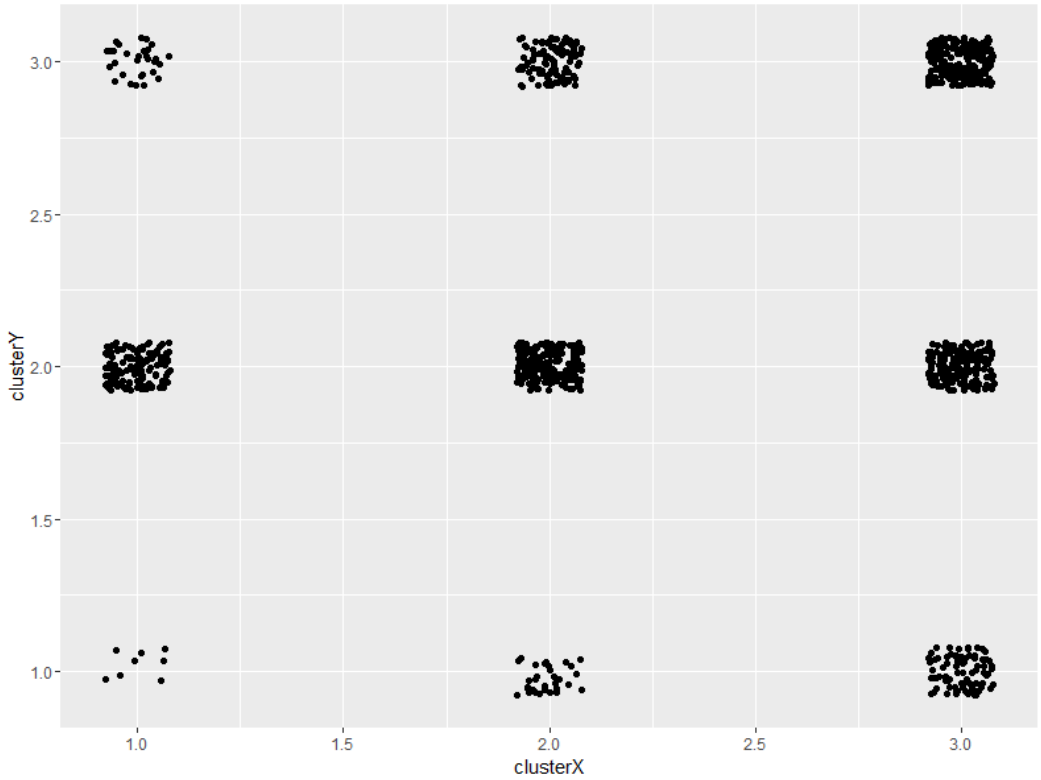
SOM Clustering with R

```
coolBlueHotRed <- function(n, alpha = 1) {
   rainbow(n, end=4/6, alpha=alpha)[n:1]
}
for (i in 1:ncol(sm$data))
   plot(sm, type="property", property=sm$codes[,i],
        main=dimnames(sm$data)[[2]][i], palette.name=coolBlueHotRed)
```



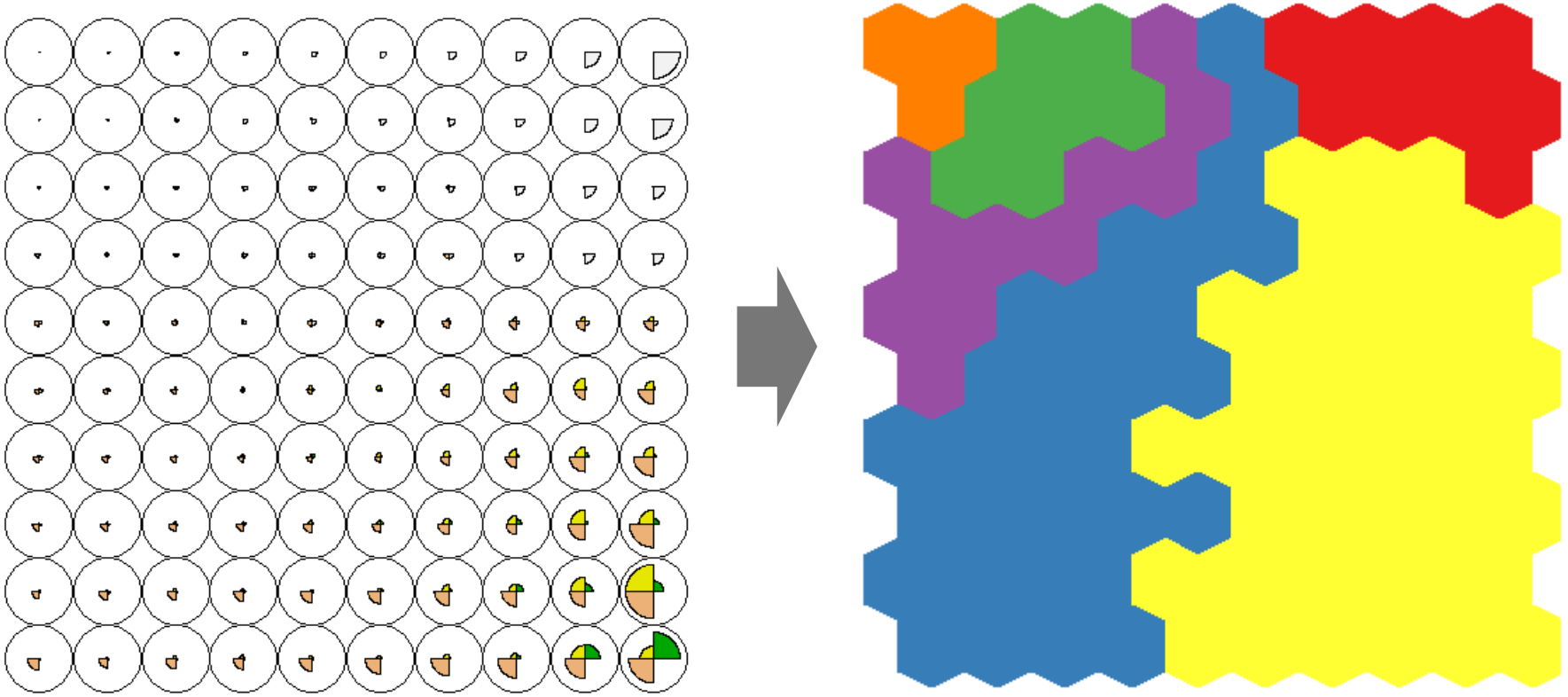
SOM Clustering with R

```
# ggplot2 패키지를 이용하여 SPSS Modeler와 유사한 Grid 도식
cdata$clusterX <- sm$grid$pts[sm$unit.classif,"x"]
cdata$clusterY <- sm$grid$pts[sm$unit.classif,"y"]
p <- ggplot(cdata, aes(clusterX, clusterY))
p + geom_jitter(position = position_jitter(width=.2, height=.2))
```



Reducing SOM complexity through cluster analysis

K-Means(k=6)을 사용하여 SOM neuron(10x10)들 간의 유사성을 도식: visSOM.R 참조



SOM Application Examples

- Example Using Ta-Feng Grocery Shopping Data
- Example Using Irish Census Data
- Source:

  http://www.slideshare.net/shanelynn/2014-0117-dublin-r-selforganising-maps-for-customer-segmentation-shane-lynn