11장 군집분석

덕성여자대학교 정보통계학과

김 재희

11.1 서론

▶ 분류(classification)는 인간의 기본적인 개념적 활동(human conceptual activity).

▶ 과학 분야에서 보면 분류체계는 이론 발전에 필요한 개념 형성에 있어 중요한 과정이 되며 관찰이나 실험을 통해 얻은 개체들을 분류하는 것이 연구 목표가 되기도 한다.

▶ 1939년 Tryon이 ‘cluster analysis'라는 용어를 처음으로 사용하였으며 그 이후 군집화하기 위한 다양한 방법과 알고리즘(algorithm)이 개발되었다. 1963년 생물학자인 Robert Sokal과 Peter Sneath가 쓴 “Principles of Numerical Taxonomy"는 군집화 방법 개발에 중요한 자극이 되었다. Sokal과 Sneath는 생물학적 분류를 위해 생명체에 대한 유사성을 측정하여 유사성이 큰 것들은 동일한 군집을 형성하며, 군집의 패턴이 인식된 후에는 새로운 개체를 패턴 인식을 통해 분류할 수 있다고 가정하였다.

▶ 그 이후로 과학 분야에서 군집분석 응용 결과들이 많이 나왔으며 (1) 컴퓨터의 발달 (2) 과학에서 분류의 중요성 증가 등으로 인하여 군집분석에 대한 연구가 더욱 증가하게 되었다.

▶ 사회과학분야에서도 군집분석에 대한 관심은 크게 증가되고 있다. 예를 들어 인류학분야에서 데이터에 근거한 인류학적 분류, 심리학분야에서 심리시험결과에 의거한 집단 분류, 사회학에서 사회경제활동지표를 근거로 한 계급 분류 등 통계적 분류 분석에 대해 관심이 증가하고 있다.

▶ 군집분석은 시스템을 표현하는 데이터로부터 구조를 찾아내고 통계적 특성이 서로 다른 군집으로 분리할 수 있는지를 알아내는 것이라고 볼 수 있다. 따라서 구체적인 군집분석 방법에 따라 군집화 결과에 차이가 날 수 있다.

▶ 군집분석(cluster analysis)에서는 군집의 개수나 구조에 대한 가정없이 다변량 데이터로부터 거리 기준에 의해 자발적인 군집화를 유도한다. 군집분석의 첫 번째 목적은 적절한 군집으로 나누는 것이고 두 번째 목적은 각 군집의 특성, 군집간의 차이 등에 대한 탐색적 연구를 하는 것이다.

11.2 유사성 측도

▶ 군집분석에서는 각 관측벡터에 대해 어느 관측벡터와 같은 군집으로 묶일 수 있는지를 판단해야한다.

즉 관측벡터간의 유사성(similarity) 또는 근접성(proximity)을 측정해야한다.

▶ 유사성에 대한 편리한 측도로 관측벡터간의 거리(distance)를 이용할 수 있다.

거리가 가까울수록 유사성이 크고, 거리가 멀수록 비유사성(dissimilarity)이 큰 사실을 군집화 단계에서 적용할 수 있다.

▶ p-차원의 두 벡터 $\boldsymbol{x}=(x_1,...,x_p)'$, $\boldsymbol{y}=(y_1,...,y_p)'$ 간의 몇 가지 알려진 거리

(1) 유클리드 거리(Euclidean distance)

$$d(\boldsymbol{x},\boldsymbol{y})=\sqrt{(x_1-y_1)^2+...+(x_p-y_p)^2}=\sqrt{(\boldsymbol{x}-\boldsymbol{y})'(\boldsymbol{x}-\boldsymbol{y})}$$

(2) 표준화 거리 또는 통계적 거리(statistical distance)

$$d(\boldsymbol{x},\boldsymbol{y})=\sqrt{(\boldsymbol{x}-\boldsymbol{y})'\boldsymbol{D}^{-1}(\boldsymbol{x}-\boldsymbol{y})}$$

여기서 $\boldsymbol{D}=diag\{s_{11},...,s_{pp}\}$는 표본분산행렬이다.

(3) 민코우스키(Minkowski) 거리

$$d(\boldsymbol{x},\boldsymbol{y})=\left[\sum_{i=1}^{p}|x_i-y_i|^m\right]^{1/m}$$

여기서 $m>0$.

(4) 마할라노비스(Mahalanobis) 거리

$$d(\boldsymbol{x},\boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})'\boldsymbol{S}^{-1}(\boldsymbol{x}-\boldsymbol{y})}$$

여기서 $\boldsymbol{S} = \{s_{ij}\}$는 표본공분산행렬.

(5) 캔버라(Canberra) 거리

$$d(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{p} \frac{|x_i - y_i|}{(x_i + y_i)}$$

(6) 체비셰프(Chebychev) 거리

$$d(\boldsymbol{x},\boldsymbol{y}) = \max_i |x_i - y_i|$$

(7) 맨하탄(Manhattan) 거리

$$d(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{p} |x_i - y_i|$$

11.3 계층적 군집 방법

▶ 처음에 n개의 군집으로부터 시작하여 점차 군집의 개수를 줄여나가는 방법으로 계층적인 군집 방법(hierarchical clustering method)에 대해 소개한다.

▶ 군집분석에서는 관측벡터간의 거리뿐만 아니라 군집간 거리에 대한 정의가 필요하다. 본 절에서는 군집간 거리에 대한 정의에 따라 최단연결법, 최장연결법, 평균연결법, Ward 방법을 설명하고자 한다.

▶ 군집단계에 대한 그래프적 표현으로 나무구조그림(tree diagram, dendrogram)을 들 수 있으며, 유사성이 큰 가까운 이웃을 병합하는 방법으로 군집의 단계를 보여준다.

11.3.1 최단연결법

최단연결법(single linkage : nearest neighbor)에 의한 두 군집 C_1과 C_2의 거리는

$$d\{C_1, C_2\} = \min\{d(x,y) | x \in C_1,\ y \in C_2\}$$

로 두 군집간의 최단 거리를 군집간 거리로 정의한다.

최단연결법으로 군집화하는 방법은 거리행렬 $D = \{d_{ik}\}$로부터 최단 거리의 쌍 U, V를 찾아 한 군집으로 병합하는 것이다.

(UV)로 묶인 군집과 군집 W와의 거리는

$$d\{(UV)W\} = \min\{d_{UW},\ d_{VW}\}$$

로 구한다.

≪예제 11.1≫ 5개 개체의 거리행렬 D부터 최단연결법으로 군집을 만들어 보자.

$$D = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{bmatrix} 0 & & & & \\ 9 & 0 & & & \\ 3 & 7 & 0 & & \\ 6 & 5 & 9 & 0 & \\ 11 & 10 & 2 & 8 & 0 \end{bmatrix} \end{array}$$

① 개체 개수 만큼의 군집수로부터 시작한다.

가장 거리가 가까운 쌍을 찾는다.

$$\min\{d_{ik}\} = d_{53} = 2$$

이므로 5번 개체와 3번 개체를 묶어 군집(35) 로 한다.

② 군집 (35)와 나머지 개체 1, 2, 4와의 거리를 계산하다.

$$d\{(35)1\} = \min\{d_{31}, d_{51}\} = \min\{3, 11\} = 3$$
$$d\{(35)2\} = \min\{d_{32}, d_{52}\} = \min\{7, 10\} = 7$$
$$d\{(35)4\} = \min\{d_{34}, d_{54}\} = \min\{9, 8\} = 8$$

4개 군집간의 거리행렬은 다음과 같이 구해진다.

$$\begin{matrix} & (35) & 1 & 2 & 4 \end{matrix}$$
$$\boldsymbol{D}_1 = d_{ik} = \begin{bmatrix} 0 & & & \\ 3 & 0 & & \\ 7 & 9 & 0 & \\ 8 & 6 & 5 & 0 \end{bmatrix}$$

위의 거리행렬로부터 (35)와 (1)이 가장 가까우므로 같은 군집 (1(35))로 묶인다.

③ 군집(1(35))와 나머지 개체 2, 4와 의 거리를 계산하여 거리행렬을 구한다.

$$d\{(135)2\} = \min\{d_{12}, d_{(35)2}\} = \min\{9,7\} = 7$$
$$d\{(135)4\} = \min\{d_{14}, d_{(35)4}\} = \min\{6,8\} = 6 \qquad d\{(2)(4)\} = 5$$

이므로 군집간의 거리행렬을 다음과 같이 얻게 된다.

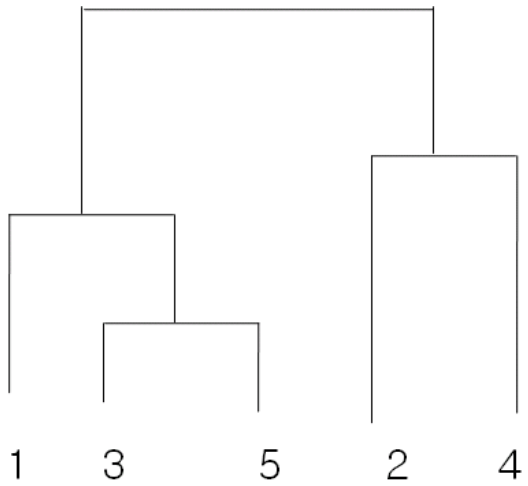
$$\begin{matrix} & 1(35) & 2 & 4 \end{matrix}$$
$$\boldsymbol{D}_2 = \mathrm{d_{ik}} = \begin{bmatrix} 0 & & \\ 7 & 0 & \\ 6 & 5 & 0 \end{bmatrix}$$

위 거리행렬로부터 개체 2와 4가 가장 가까우므로 군집(24) 로 묶을 수 있다.

④ 군집(1(35))와 군집(24)의 거리를 구해보자.

$$d\{(135)(24)\} = \min\{d_{(135)2}, d_{(135)4}\} = \min\{7,6\} = 6$$

⑤ 나무구조그림을 그려보자.



2개의 군집을 원한다면 : (135)와 (24).

3개의 군집을 원한다면 : (135), (2)와 (4).

4개의 군집을 원한다면 : (1), (35), (2)와 (4)

로 군집을 형성할 수 있다.

11.3.2 최장연결법

최장연결법(complete linkage: farthest neighbor)에서의 두 군집 C_1과 C_2의 거리는

$$d\{C_1, C_2\} = \max\{d(x,y) | x \in C_1,\ y \in C_2\}$$

로 두 군집 간의 최장 거리를 군집간 거리로 정의한다.

최장연결법으로 군집화하는 방법은 거리행렬 $D = \{d_{ik}\}$ 로부터 최단 거리의 쌍 U, V를 찾고 한 군집으로 병합한다.

(UV)로 묶인 군집과 군집 W와의 거리는

$$d\{(UV)W\} = \max\{d_{UW},\ d_{VW}\}$$

로 구한다.

≪예제 11.2≫ 5개 개체의 거리행렬 D로부터 최장연결법으로 군집을 만들어 보자.

$$D = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{bmatrix} 0 & & & & \\ 9 & 0 & & & \\ 3 & 7 & 0 & & \\ 6 & 5 & 9 & 0 & \\ 11 & 10 & 2 & 8 & 0 \end{bmatrix} \end{array}$$

① 개체 개수 만큼의 군집수로부터 시작한다.

가장 거리가 가까운 쌍을 찾는다.

$$\min\{d_{ik}\} = d_{53} = 2$$

이므로 5번 개체와 3번 개체를 묶어 군집(35)로 한다.

② 군집 (35)와 나머지 개체 1, 2, 4 와의 거리를 계산한다.

$$d\{(35)1\} = \max\{d_{31}, d_{51}\} = \max\{3, 11\} = 11$$
$$d\{(35)2\} = \max\{\mathrm{d}_{32}, \mathrm{d}_{52}\} = \max\{7, 10\} = 10$$
$$d\{(35)4\} = \max\{d_{34}, d_{54}\} = \max\{9, 8\} = 9$$
$$d_{12} = 9 \qquad d_{14} = 6 \qquad d_{24} = 5$$

4개 군집간의 거리행렬은 다음과 같이 구해진다.

$$\begin{matrix} & (35) & 1 & 2 & 4 \end{matrix}$$
$$\boldsymbol{D}_1 = \begin{bmatrix} 0 & & & \\ 11 & 0 & & \\ 10 & 9 & 0 & \\ 9 & 6 & 5 & 0 \end{bmatrix}$$

위의 거리행렬로부터 개체 2와 4가 가장 가까우므로 같은 군집(24)로 묶인다.

③ 군집(35))와 군집 (24)와 나머지 개체 1과의 거리를 계산하여 거리 행렬을 구한다.

$$d\{(35)1\} = \max\{d_{31}, d_{51}\} = \max\{3, 11\} = 11$$
$$d\{(35)(24)\} = \max\{d_{(35)2}, d_{(35)4}\} = \max\{10, 9\} = 10$$
$$d\{1(24)\} = \max\{d_{12}, d_{14}\} = \max\{9, 6\} = 9$$

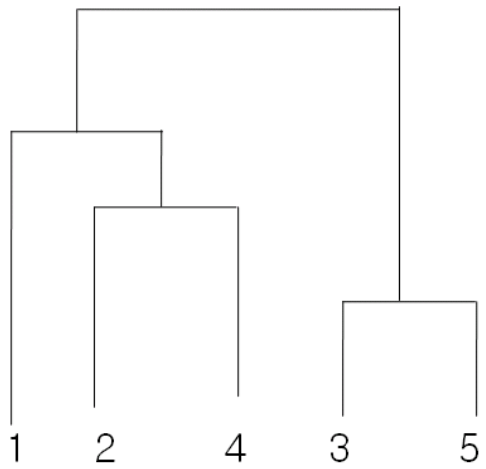
$$\begin{matrix} & (35) & 1 & (24) \end{matrix}$$
$$\boldsymbol{D}_2 = \begin{bmatrix} 0 & & \\ 11 & 0 & \\ 10 & 9 & 0 \end{bmatrix}$$

위 거리행렬로부터 개체 1과 군집(24)가 가장 가까우므로 군집(1(24))로 묶을 수 있다.

④ 군집((35))와 군집(1(24))의 거리를 구해보자.

$$d\{(35)(1(24))\} = \max\{d_{(35)1}, d_{(35)(24)}\} = \max\{11, 10\} = 11$$

⑤ 나무구조그림을 그려보자.



2개의 군집을 원한다면 : (124)와 (35).

3개의 군집을 원한다면 : (1), (24)와 (35).

4개의 군집을 원한다면 : (1), (2), (4)와 (35)으로 군집을 형성할 수 있다.

11.3.3 평균연결법

평균연결법(average linkage)에서는 두 군집 C_1과 C_2의 거리를 군집에 속한 모든 개체들 간의 거리의 평균으로 다음과 같이 정의한다:

$$d\{C_1, C_2\} = \frac{1}{n_1 n_2} \sum_i \sum_j d_{ij}$$

여기서 n_1은 C_1에 속한 개체 수이고 n_2는 C_2에 속한 개체 수이다.

≪예제 11.3≫ 5개 개체의 거리행렬 $\boldsymbol{D}$ 로부터 평균연결법으로 군집을 만들어 보자.

$$\boldsymbol{D} = \begin{matrix} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} \begin{matrix} \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{bmatrix} 0 & & & & \\ 9 & 0 & & & \\ 3 & 7 & 0 & & \\ 6 & 5 & 9 & 0 & \\ 11 & 10 & 2 & 8 & 0 \end{bmatrix} \end{matrix}$$

① 개체 개수 만큼의 군집수로부터 시작한다.

가장 거리가 가까운 쌍을 찾는다.

$$\min\{d_{ik}\} = d_{53} = 2$$

이므로 5번 개체와 3번 개체를 묶어 군집(35)로 한다.

② 군집 (35)와 나머지 개체 1, 2, 4 와의 거리를 계산한다.

$$d\{(35),1\} = \frac{1}{2 \cdot 1}(d_{31} + d_{51}) = \frac{1}{2}(3+11) = 7$$

$$d\{(35),2\} = \frac{1}{2 \cdot 1}(d_{32} + d_{52}) = \frac{1}{2}(7+10) = 8.5$$

$$d\{(35),4\} = \frac{1}{2 \cdot 1}(d_{34} + d_{54}) = \frac{1}{2}(9+8) = 8.5$$

$$d_{12} = 9$$
$$d_{14} = 6$$
$$d_{24} = 5$$

4개 군집간의 거리행렬은 다음과 같이 구해진다.

$$\boldsymbol{D}_1 = \begin{array}{c} (35) \\ 1 \\ 2 \\ 4 \end{array} \begin{bmatrix} 0 & & & \\ 7 & 0 & & \\ 8.5 & 9 & 0 & \\ 8.5 & 6 & 5 & 0 \end{bmatrix}$$

위의 거리 행렬로부터 개체 2와 4가 가장 가까우므로 같은 군집(24)로 묶인다.

③ 군집(35), 군집(24)와 나머지 개체 1과의 거리를 계산하여 거리행렬을 구한다.

$$d\{(35),(24)\} = \frac{1}{2 \cdot 2}\left(d_{32} + d_{52} + d_{34} + d_{54}\right) = \frac{1}{4}(7 + 10 + 9 + 8) = 8.5$$

$$d\{(35),1\} = 7$$

$$d\{(24),1\} = \frac{1}{2 \cdot 1}\left(d_{21} + d_{41}\right) = \frac{1}{2}(9 + 6) = 7.5$$

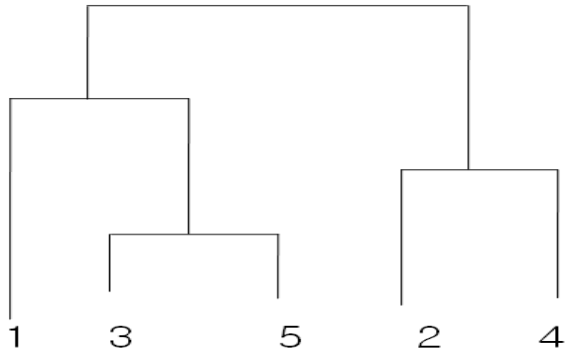
$$D_2 = \begin{matrix} (35) \\ 1 \\ (24) \end{matrix} \begin{bmatrix} 0 & & \\ 7 & 0 & \\ 8.5 & 7.5 & 0 \end{bmatrix}$$

거리행렬 D_2로부터 개체 1과 군집(35)가 가장 가까우므로 군집(1(35))로 묶을 수 있다.

④ 군집(135)와 군집(24)의 거리를 구해보자.

$$d\{(135)(24))\} = \frac{1}{3 \cdot 2}(d_{12} + d_{14} + d_{32} + d_{34} + d_{52} + d_{54})$$
$$= \frac{1}{6}(9 + 6 + 7 + 9 + 10 + 8) = 8.17$$

⑤ 나무구조그림을 그려보자.



2개의 군집 : (135)와 (24).

3개의 군집을 원한다면 : (1), (24)와 (35).

4개의 군집을 원한다면 : (1), (2), (4)와 (35).

로 군집을 형성할 수 있다.

11.3.4 Ward의 계층적 군집 방법

▶ Ward(1963)는 군집내 제곱합 증분과 군집간 제곱합을 고려한 계층적 군집 방법을 제안.군집간 정보의 손실을 최소화하도록 군집화를 하는데 여기서 군집간의 정보란 편차제곱합 ESS(error sum of squares)로 나타낸다.

▶ 군집 A와 군집 B의 군집내 거리(within-cluster distance)는 군집내 제곱합으로

$$ESS_A = \sum_{j=1}^{n_A} (\boldsymbol{X}_{Aj} - \overline{\boldsymbol{X}_{\boldsymbol{A}}})'(\boldsymbol{X}_{Aj} - \overline{\boldsymbol{X}_{\boldsymbol{A}}}) = \sum_{j=1}^{n_A}\sum_{t=1}^{p} (X_{Ajt} - \overline{X_{At}})^2$$

$$ESS_B = \sum_{j=1}^{n_B} (\boldsymbol{X}_{\boldsymbol{Bj}} - \overline{\boldsymbol{X}_{\boldsymbol{B}}})'(\boldsymbol{X}_{Bj} - \overline{\boldsymbol{X}_{\boldsymbol{B}}}) = \sum_{j=1}^{n_B}\sum_{t=1}^{p} (X_{Bjt} - \overline{X_{Bt}})^2$$

군집 A와 군집 B를 합친 경우 군집내 제곱합은

$$ESS_{AB} = \sum_{j=1}^{n_{AB}} (\boldsymbol{X}_{ABj} - \overline{\boldsymbol{X}_{\boldsymbol{AB}}})'(\boldsymbol{X}_{ABj} - \overline{\boldsymbol{X}_{\boldsymbol{AB}}}) = \sum_{j=1}^{n_{AB}}\sum_{t=1}^{p} (X_{ABjt} - \overline{X_{ABt}})^2$$

여기서 $\overline{\boldsymbol{X}_{\boldsymbol{A}}}$ 와 $\overline{\boldsymbol{X}_{\boldsymbol{B}}}$는 각 군집에서의 평균관측값벡터

$$\overline{X_{AB}} = \frac{n_A \overline{X_A} + n_B \overline{X_B}}{n_A + n_B}$$

은 합친 군집의 중심으로 군집 A와 군집 B간의 중심으로 표현된다.

군집 A와 군집 B는 군집형성으로부터 생기는 편차제곱합의 증분

$$I_{AB} = ESS_{AB} - (ESS_A + ESS_B)$$

을 최소화하도록 형성된다.

I_{AB}를 다시 정리해보면

$$\begin{aligned} I_{AB} &= n_A (\overline{X_A} - \overline{X_{AB}})' (\overline{X_A} - \overline{X_{AB}}) + n_B (\overline{X_B} - \overline{X_{AB}})' (\overline{X_B} - \overline{X_{AB}}) \\ &= \frac{n_A n_B}{n_A + n_B} (\overline{X_A} - \overline{X_B})' (\overline{X_A} - \overline{X_B}) \\ &= \frac{(\overline{X_A} - \overline{X_B})' (\overline{X_A} - \overline{X_B})}{\frac{1}{n_A} + \frac{1}{n_B}} \end{aligned}$$

▶ I_{AB}를 최소화하는 것은 군집간 거리(between cluster distance)를 최소화하는 것과 같다.• 군집 A와 군집 B가 멀리 떨어져 있을수록 병합하면서 생기는 I_{AB}가 크고 군집 A와 군집 B가 가까울수록 I_{AB}가 작게 되어 정보의 손실이 작게 된다.

•개체와 군집 중심과의 편차제곱합 ESS가 작을수록 군집내 개체가 모여있음을 알 수 있다.

▶ 각 군집의 편차제곱합의 합을 ESS로 정의하며 군집의 수가 g일 때

$$ESS = \sum_{i=1}^{g} ESS_i$$

여기서 i번째 군집의 편차제곱합 (n_i: i 번째 군집 크기)

$$ESS_i = \sum_{j=1}^{n_i} (\boldsymbol{X}_{ij} - \overline{\boldsymbol{X}}_i)'(\boldsymbol{X}_{ij} - \overline{\boldsymbol{X}}_i) = \sum_{j=1}^{n_i}\sum_{t=1}^{p} (X_{ijt} - \overline{X_{it}})^2$$

$\boldsymbol{X}_{ij}$: i번째 군집에 속한 j번째 관측벡터, $\overline{\boldsymbol{X}_i}$: i번째 군집에서의 평균관측값벡터

$\overline{X_{it}} = \frac{1}{n_i}\sum_{j=1}^{n_i} X_{ijt}$ 는 i번째 군집의 t번째 변수의 평균이다.

11.4 비계층적 군집 방법

▶ n개의 개체를 g개의 군집으로 나눌 수 있는 모든 가능한 방법을 점검하여 최적의 군집을 형성하는 것이 전략이다. 그러나 n과 g에 따라 계산량이 많아지므로 간단한 방법으로 다음의 K-평균법(K-means method)을 간략하게 소개하기로 한다.

▶ K-평균법은 전체 개체를 K 개의 군집으로 나누는 방법으로 계층적 군집 방법과는 달리 한 개체가 속해있던 군집에서 다른 군집으로 이동하는 재배치(reallocation)가 가능하다. 초기값에 의존하는 방법으로 군집의 초기치 선택이 최종 군집 선택에 영향을 미치므로 몇 가지 방법으로 초기치를 선택해보고 비교할 필요가 있다.

① 초기 씨앗(seed) 역할을 하는 개체를 g개 선택하여 군집의 초기값으로 준다. 이 때 씨앗 g개를 랜덤하게 선택할 수도 있고 서로 가장 멀리 떨어지도록 선택할 수 도 있다. 또한 이들 씨앗 간에 최소 떨어진 거리를 요구하여 이 요구에 맞게 선택할 수도 있다.

② 나머지 개체들은 각 군집의 초기값과 거리를 계산하여 가까운 초기값과 같은 군집을 형성하게 한다. 2개 이상의 개체가 모인 군집이 형성되면 씨앗은 군집 중심(cluster centroid)으로 대체된다.

③ 모든 개체가 군집으로 할당된 후 다른 군집의 중심과 거리를 계산한다. 개체가 속해 있는 군집 중심과의 거리가 다른 군집중심과의 거리 보다 더 크면 개체를 다른 군집으로 옮긴다.

④ 옮긴 후 두 군집의 중심은 다시 구하게 되며 다시 각 군집 중심과의 거리를 계산한다.

■ 이와 같은 과정을 반복하여 더 이상 개체의 군집간 이동이 없을 때 멈추고 이를 최적의 군집으로 결정한다.

■ K-평균법은 계층적 군집 방법과 같이 쓰일 수 있다. 예를 들어 계층적 군집 방법으로 적절한 군집의 개수를 정한 후 K-평균법을 사용해 개체들을 재배치 할 수도 있다.

이 씨앗들은 군집이 형성되어가면서 군집의 중심값으로 대체된다.

11.5 군집 개수의 결정

▶ 계층적 군집 방법에서는 나무구조그림에서 수평선을 이용하여 가지를 잘라내므로써 g개의 군집을 결정할 수 있다. 병합되는 과정에서 군집간의 거리 차이에 큰 변화를 보이는 경우를 고려하여 군집의 개수를 택하게 된다.

▶ Ward 방법의 경우에서는 군집개수에 대한 ESS의 증분을 고려하여 급격한 변화가 일어나는 위치에 해당하는 부분에서 군집의 개수를 결정한다.

▶ 데이터의 상황을 최적화시키는 군집의 개수는 데이터 분석 경험과 데이터 특성상의 중요성, 기준하고자 하는 통계량의 급격한 변화 등을 고려해 결정하게 되지만 최선의 선택을 위한 통계적 방법은 아직은 없다고 보아도 좋다.

▶ 참고로 Mojena(1977)가 제안한 방법: 나무구조그림에서 각 단계의 거리를 이용하여

$$\alpha_j > \overline{\alpha} + k s_\alpha, \qquad j = 1,2 \ldots, n-1$$

α_j는 $n, n-1, \ldots, 1$로 군집의 개수가 줄어들면서 생겨나는 군집간 거리이고 $\overline{\alpha}$는 이들의 평균이며 s_α는 α_j들의 표준편차이다. k는 상수값으로 2.75, 3.5, 1.25 등의 값들이 제안되었다.

11.6 군집의 타당성

▶ 군집 결과에 대한 타당성과 안정성에 대한 검정으로 교차타당성(cross-validation)을 이용한 방법을 생각해 볼 수 있다. 데이터를 A, B 두 개의 부분으로 랜덤하게 분류해 놓은 다음 각 부분에서 따로 군집분석을 한 후 합쳐서 군집분석한 결과와 비교하여 비슷하면 결과에 대한 안정성을 볼 수 있다고 판단한다.

그 외에도 다음과 같은 방법으로 타당성에 대한 점검을 할 수 있다.

① A 부분의 개체들로 g개의 군집으로 나눈다.

② B 부분의 개체들을

(a) A의 군집에, 예를 들어 군집 중심을 이용해, 배치해 본다.

(b) A 부분에 행했던 같은 방법으로 B 부분의 개체들에 대해 군집을 나누어 본다.

③ (a) 결과와 (b) 결과를 비교해본다.

11.7 변수들에 대한 군집

▶ n개의 개체들에 대한 군집 문제 외에 p개 변수들에 대한 군집을 구하고자 할 경우

▶ 변수들의 유사성, 비유사성을 나타내주는 상관계수행렬을 이용해 변수들을 군집으로 나눌 수 있다.

$1-r_{ij}^2$을 사용해 거리 측도에 대한 조건을 만족시키도록 한 다음

$\boldsymbol{A}^* = \left\{a_{ij}^*\right\}$를 거리행렬로 놓고

앞서 설명한 군집 방법들을 이용해 변수들의 군집을 찾을 수 있다.

11.8 모형기반 군집 방법

▶ Scott and Symons (1971)가 제안

Banfield and Raftery (1993), Fraley and Raftery (2002) 등이 발전시킨 방법으로 모형 기반 군집방법(model-based clustering)을 설명하고자 한다.

▶ 모집단이 G개의 군집으로 구성.

k번째 군집에 속한 $p-$차원 관측벡터 $\boldsymbol{x}$의 밀도함수는 $f_k(\boldsymbol{x},\boldsymbol{\theta})$라고 가정.

$\boldsymbol{\gamma}=(\gamma_1,\cdots,\gamma_n)'$: 여기서 $\boldsymbol{x_i}$가 k번째 군집에 속하였으면 $\gamma_i=k$이다.

▶ 가능도함수는

$$L(\boldsymbol{\theta},\boldsymbol{\gamma})=\prod_{i=1}^{n} f_{\gamma_i}(x_i;\theta_{\gamma_i})$$

가능도를 최대화하는 $\boldsymbol{\theta}=(\theta_1,\cdots,\theta_G)'$ 와 $\boldsymbol{\gamma}=(\gamma_1,\gamma_2,...,\gamma_n)'$ 를 선택하게 된다.

데이터가 속한 군집은 내재하는 확률분포로부터 형성되었다고 가정하고

혼합 모형(mixture model)

$$L_{mix}(\boldsymbol{\theta},\boldsymbol{\gamma}) = \prod_{i=1}^{n}\sum_{k=1}^{G}\tau_k f_k(\boldsymbol{x_i}|\boldsymbol{\theta_k})$$

을 고려한다. 여기서 $\boldsymbol{\theta}=(\theta_1,\cdots,\theta_G)'$는 모수벡터

τ_k는 관측벡터가 k 번째 군집에 속할 확률이며 $\tau_k \geq 0,\ \sum_{k=1}^{G}\tau_k = 1$.

▶ 관측벡터가 다변량 정규분포를 따른다고 가정하는 Gaussian 혼합모형을 고려.

$f_k(\boldsymbol{x},\boldsymbol{\theta})$: 평균벡터 $\boldsymbol{\mu_k}$와 공분산행렬 $\boldsymbol{\Sigma_k}$를 갖는 k 번째 군집의 다변량 정규밀도함수

가능도함수 :

$$L(\boldsymbol{\theta},\boldsymbol{\gamma}) = const \cdot \prod_{k=1}^{G}\prod_{i\in E_k}|\boldsymbol{\Sigma_k}|^{1/2}\exp\left\{-\frac{1}{2}(\boldsymbol{x_i}-\boldsymbol{\mu_k})'\boldsymbol{\Sigma_k}^{-1}(\boldsymbol{x_i}-\boldsymbol{\mu_k})\right\}$$

여기서 $E_k = \{i;\gamma_i = k\}$ 이다. $\boldsymbol{\mu_k}$의 최대가능도추정량은 $\overline{\boldsymbol{x_k}} = n_k^{-1}\Sigma_{i\in E_k}\boldsymbol{x_i}$.

n_k는 E_k에 속한 원소의 개수.

▶ μ_k에 대한 최대우도추정량인 $\overline{x_k}$ 로 대체하여 로그가능도함수를 나타내면

$$l(\theta,\gamma) = \text{const} - \frac{1}{2}\sum_{k=1}^{G}\left\{tr(W_k\Sigma_k^{-1}) + n_k \log|\Sigma_k|\right\}$$

여기서 const는 상수이고 W_k는 k번째 군집의 표본교차곱행렬

$$W_k = \sum_{i\in E_k}(x_i - \overline{x_k})(x_i - \overline{x_k})'$$

▶ 각 군집을 형성하는 기하학적인 특징(shape, volume, orientation)은 공분산행렬 Σ_k에 의해 결정된다. (Banfield and Raftery(1993))

고유값 분해에 의하여 공분산행렬이 대표성을 갖는 일반적인 구조 :

$$\Sigma_k = \lambda_k D_k A_k D_k'$$

여기서 D_k는 고유벡터의 직교행렬,

A_k는 각 원소가 Σ_k의 고유값을 비례적으로 취하는 대각행렬, λ_k는 스칼라.

D_k는 군집의 orientation을 결정. A_k는 군집의 shape을 결정. λ_k는 군집의 volume을 결정.

▶ 예상되는 군집 개수 G가 정해지면 가능한 군집 개수 $1 \leq k \leq G$ 에 대해 $(\tau_k, \mu_k, \Sigma_k)$ 가 EM 알고리듬에 의해 추정된다.

EM 알고리듬은 E(expectation) 단계와 M(maximization) 단계로 이루어지며,

E 단계에서는 주어진 조건하에서 관측벡터가 각 군집에 속할 확률을 구하고

M 단계에서는 주어진 상황에서 모수가 추정된다.

각 개체가 최대 확률로 해당 그룹에 할당될 때 EM 알고리듬 결과로 수렴하게 된다.

모형 선택시 BIC(Bayesian Information Criterion)를 계산하여
BIC 값이 최대가 되는 군집 개수를 최종 모형으로 선택할 수 있다.

$$BIC = 2\text{loglikelihood}(x, \theta_k^*) - (\text{no of parameters})\log n \quad .$$

이와 같은 계산과정은 R 시스템에서는 mclust(Fraley and Raftery, 1998) 패키지로 제공되고 있으며 이를 활용하여 결과를 얻을 수 있다.

▶표 11.1 Yeung et al. (2001)가 제안 다섯 개의 모형

모형	설명
equal volume spherical model	$\Sigma_k = \lambda I$ 가장 제한된 모형으로 모수의 개수가 가장 적음
unequal volume spherical model	$\Sigma_k = \lambda_k I$ volume을 결정하는 λ가 군집마다 다르므로 서로 다른 volume의 구형군집들이 형성됨
unconstraint model	$\Sigma_k = \lambda_k D_k A_k D_k'$ 가장 일반적인 모형이지만 모수가 최대개수로 추정되어야 한다는 단점이 있음
elliptical model	$\Sigma_k = \lambda D A D'$ 각 군집은 타원형이지만 모두 동일한 shape, volume, orientation을 가짐
diagonal model	$\Sigma_k = \lambda_k B_k$ 여기서 B_k는 $\lvert B_k \rvert = 1$ 을 만족하는 대각행렬 기하학적으로 한 축에 일직선으로 세워진 타원형의 군집들과 일치하게 됨

≪예제 11.4≫ 1975년 미국 대도시의 강력범죄에 관한 자료에 대해 통계적 분석을 하고자한다.

[표 11.2] 미국 주요도시 강력범죄 발생률 자료(인구 100,000명당)

번호	도시명	murder	rape
1	Atlanta	16.5	24.8
2	Boston	4.2	13.3
3	Chicago	11.6	24.7
4	Dallas	18.9	34.2
5	Denver	6.9	41.5
6	Detroit	13.0	35.7
7	Hartford	2.5	8.8
8	Honolulu	3.6	12.7
9	Houston	16.8	26.6
10	Kansas City	10.8	43.2
11	Los Angeles	9.7	51.8
12	New Orleans	10.3	39.7
13	New York	9.4	19.4
14	Portland	5.9	23.0
15	Tucson	5.1	22.9
16	Washington	12.5	27.6

[프로그램 11.1] 범죄자료에 대한 계층적 군집분석

```
crime=read.csv("C:/data/crime.csv", header=T)
crime
attach(crime)
x=crime[, 3:4]
dx=round(dist(x), digits=2)   # 표 11.2 distance
matrix
dx
D2= dist(x, method ="manhattan")
D2
```

```
##############################################
#          Hierarchical cluster ananlysis          #
##############################################
hc1=hclust(dist(x)^2,method="single" )   # 최단연결법
  plot(hc1,labels=city,hang=-1, main="dandrogram:single")
#dendrogram

hc2=hclust(dist(x)^2,method="complete" )  # 최장연결법
  plot(hc2,labels=city,hang=-1, main="complete linkage")

hc3=hclust(dist(x)^2,method="ward" )      # Ward 방법
  plot(hc3,labels=city,hang=-1, main="Ward Method")

hc4=hclust(dist(x)^2,method="average" )   # 평균연결법
  plot(hc4,labels=FALSE,hang=-1, main="Average linkage")
```

```
cl.num=2    # number of clusters
colnames(x)=c("murder","rape")

hc1.result=cutree(hc2,k=cl.num)
   plot(x, pch=hc1.result)
   text(x,labels=city, adj=0, cex=0.5, main="single")

hc2.result=cutree(hc2,k=cl.num)
   plot(x, pch=hc2.result)
   text(x,labels=city, adj=0, cex=0.5, main="complete")

hc3.result=cutree(hc3,k=cl.num)
   plot(x, pch=hc3.result)
   text(x,labels=city, adj=0, cex=0.5, main="Ward")
```

▶표 11.3 거리 행렬

(i) 유클리드 거리행렬

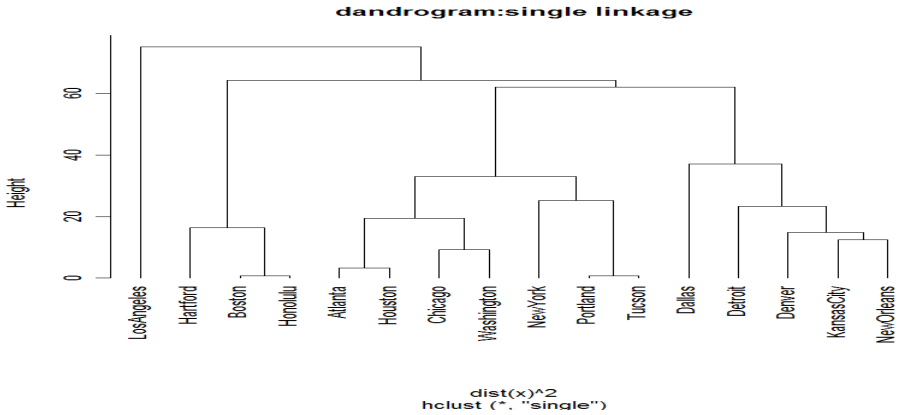
```
> dx=round(dist(x), digits=2)   # distance matrix
> dx
      1     2     3     4     5     6     7     8     9    10    11    12    13    14    15
2  16.84
3   4.90 13.59
4   9.70 25.55 11.98
5  19.26 28.33 17.45 14.05
6  11.45 24.07 11.09  6.09  8.42
7  21.26  4.81 18.32 30.23 32.99 28.88
8  17.69  0.85 14.42 26.39 28.99 24.85  4.05
9   1.82 18.32  5.54  7.88 17.89  9.86 22.83 19.17
10 19.26 30.62 18.52 12.11  4.25  7.82 35.39 31.34 17.65
11 27.84 38.89 27.17 19.86 10.67 16.43 43.60 39.57 26.18  8.67
12 16.14 27.10 15.06 10.21  3.85  4.83 31.87 27.82 14.62  3.54 12.11
13  8.92  8.02  5.74 17.59 22.24 16.69 12.65  8.86 10.32 23.84 32.40 20.32
14 10.75  9.85  5.95 17.16 18.53 14.55 14.60 10.55 11.48 20.79 29.05 17.27  5.02
15 11.56  9.64  6.74 17.84 18.69 15.04 14.34 10.31 12.27 21.09 29.26 17.59  5.54  0.81
16  4.88 16.53  3.04  9.19 14.99  8.12 21.29 17.36  4.41 15.69 24.36 12.30  8.77  8.04  8.77
```

(ii) 맨하탄 거리행렬

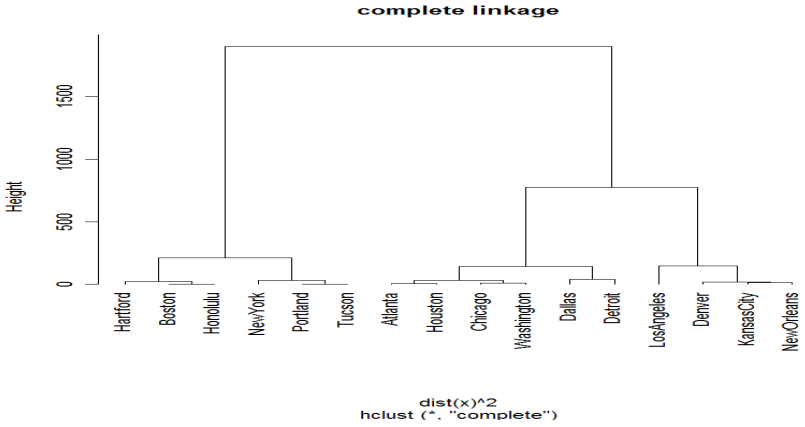
```
> D2 <- dist(x, method ="manhattan")
> D2
      1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
2  23.8
3   5.0 18.8
4  11.8 35.6 16.8
5  26.3 30.9 21.5 19.3
6  14.4 31.2 12.4  7.4 11.9
7  30.0  6.2 25.0 41.8 37.1 37.4
8  25.0  1.2 20.0 36.8 32.1 32.4  5.0
9   2.1 25.9  7.1  9.7 24.8 12.9 32.1 27.1
10 24.1 36.5 19.3 17.1  5.6  9.7 42.7 37.7 22.6
11 33.8 44.0 29.0 26.8 13.1 19.4 50.2 45.2 32.3  9.7
12 21.1 32.5 16.3 14.1  5.2  6.7 38.7 33.7 19.6  4.0 12.7
13 12.5 11.3  7.5 24.3 24.6 19.9 17.5 12.5 14.6 25.2 32.7 21.2
14 12.4 11.4  7.4 24.2 19.5 19.8 17.6 12.6 14.5 25.1 32.6 21.1  7.1
15 13.3 10.5  8.3 25.1 20.4 20.7 16.7 11.7 15.4 26.0 33.5 22.0  7.8  0.9
16  6.8 22.6  3.8 13.0 19.5  8.6 28.8 23.8  5.3 17.3 27.0 14.3 11.3 11.2 12.1
```

[결과 11.1] 계층적 군집분석 결과

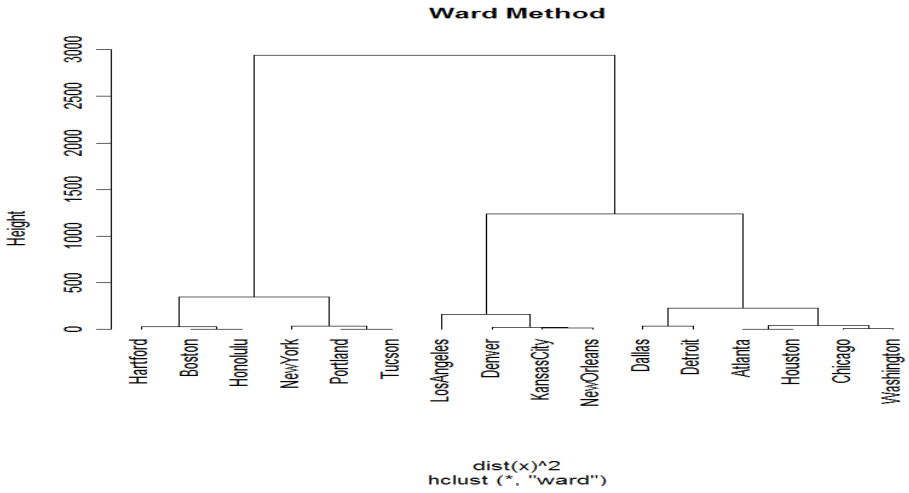
(1) 최단연결법을 이용한 경우 덴드로그램

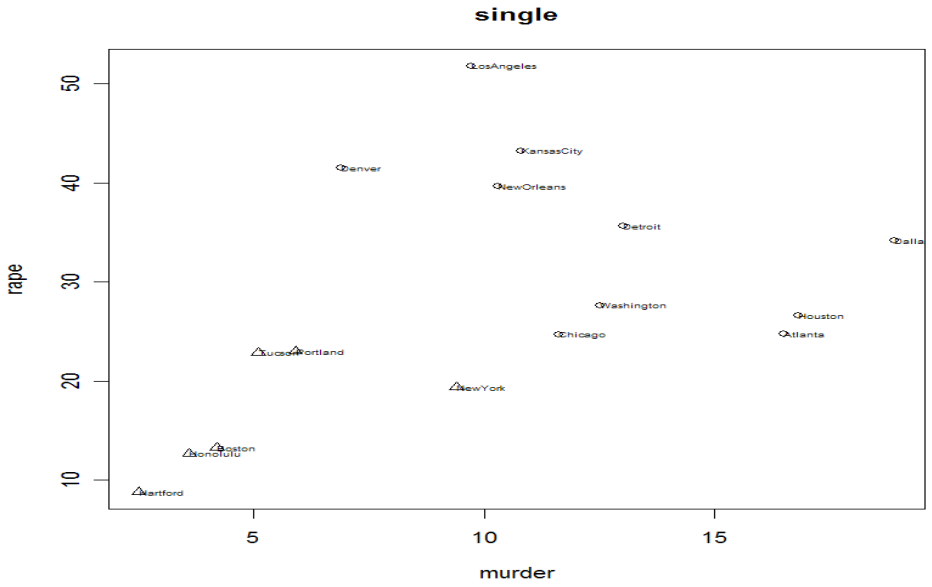


(2) 최장연결법을 이용한 경우 덴드로그램

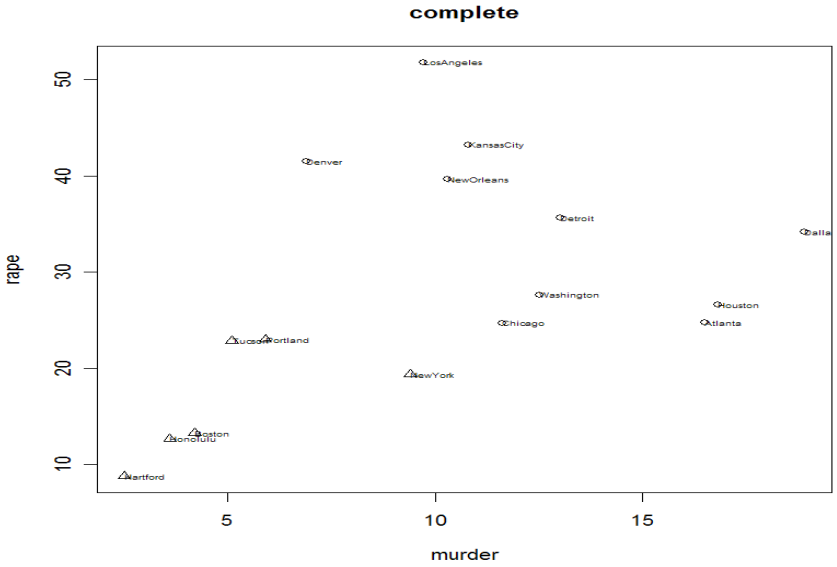


(3) Ward 방법을 이용한 경우 덴드로그램

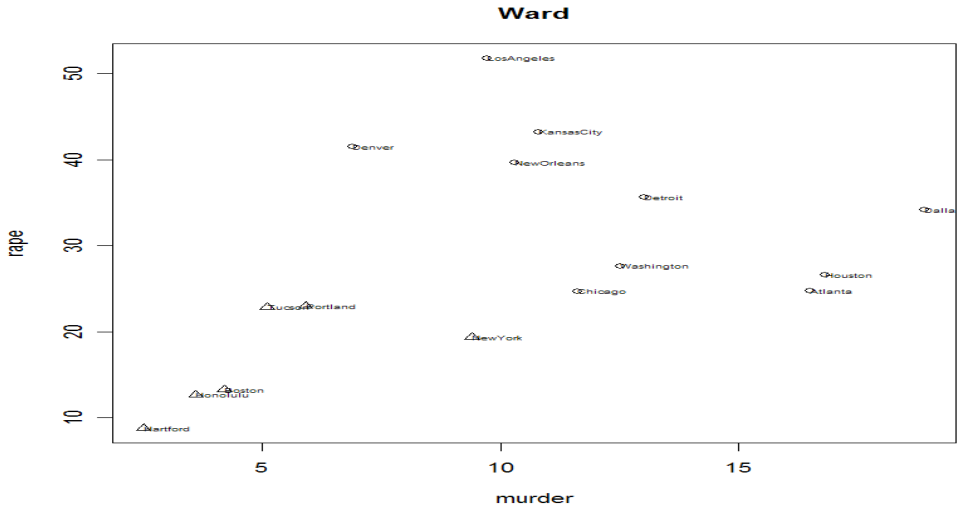




[그림 11.1] 최단연결법 이용 결과 군집



[그림 11.2] 최장연결법 이용 결과 군집



[그림 11.3] Ward 방법 이용 결과 군집

[프로그램 11.3] 범죄자료에 대한 K-means 군집분석

```
#####################################################
#          K-means  clustering                      #
#####################################################
crime_k=kmeans(x,centers=3)   # 3개 군집
attributes(crime_k)
crime_k$cluster

### grouping ###
clus=cbind(city,x,crime_k$cluster)
clus1=clus[(clus[,4]==1),]
clus1
clus2=clus[(clus[,4]==2),]
clus2
clus3=clus[(clus[,4]==3),]
clus3
kc=table(crime_k$cluster)   ## number of each cluster
 kc
plot(x, pch=crime_k$cluster,col=crime_k$cluster, main="K-means
clustering")
 text(x,labels=city, adj=0, cex=0.5)
ccent(x,crime_k$cluster)    # clusterwise info
```

[결과 11.3] K-means 군집분석 수행결과

```
> clus=cbind(city,x,crime_k$cluster)
> clus1=clus[(clus[,4]==1),]
> clus1
          city murder rape crime_k$cluster
4       Dallas   18.9 34.2               1
5       Denver    6.9 41.5               1
6      Detroit   13.0 35.7               1
10 KansasCity   10.8 43.2               1
11 LosAngeles    9.7 51.8               1
12 NewOrleans   10.3 39.7               1
```

```
> clus2=clus[(clus[,4]==2),]
> clus2
          city murder rape crime_k$cluster
1      Atlanta   16.5 24.8               2
3      Chicago   11.6 24.7               2
9      Houston   16.8 26.6               2
13     NewYork    9.4 19.4               2
14    Portland    5.9 23.0               2
15      Tucson    5.1 22.9               2
16 Washington   12.5 27.6               2
```

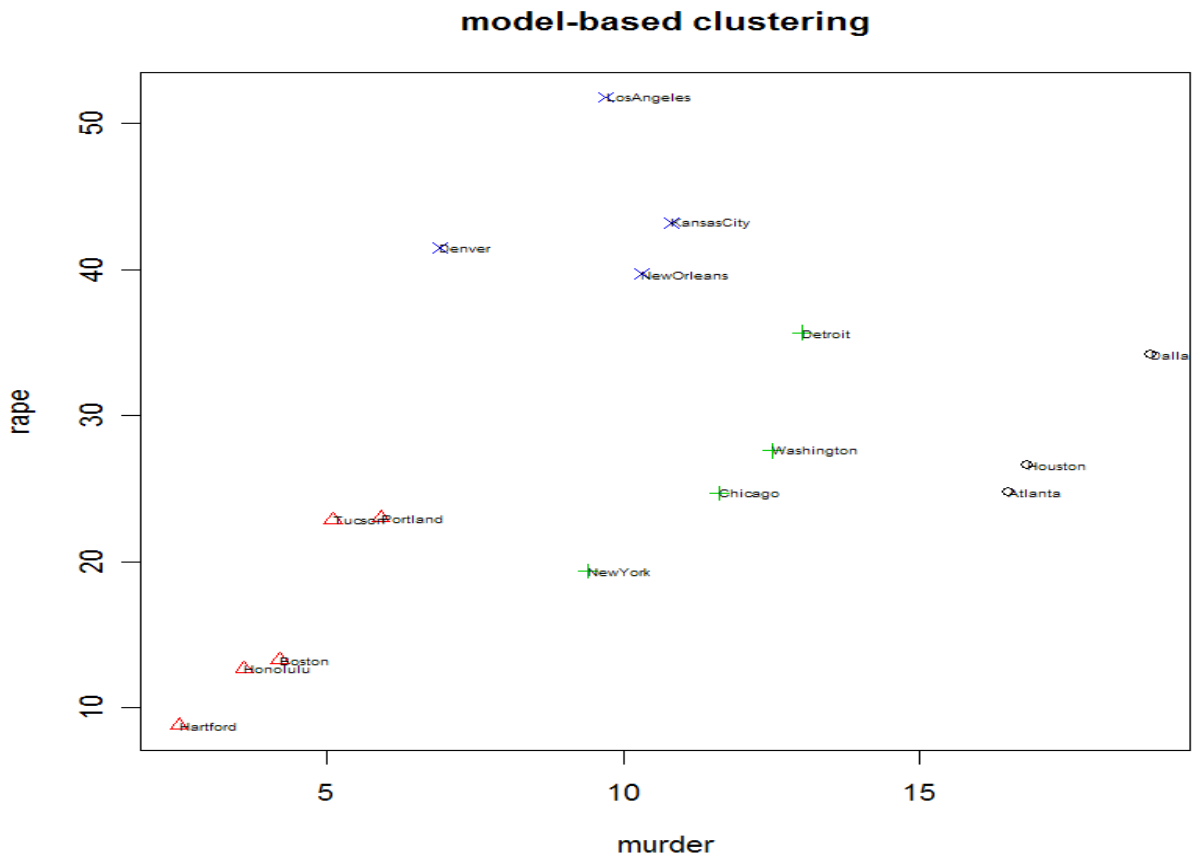
```
> clus3=clus[(clus[,4]==3),]
> clus3
      city murder rape crime_k$cluster
2   Boston    4.2 13.3               3
7 Hartford    2.5  8.8               3
8 Honolulu    3.6 12.7               3
> table(crime_k$cluster)    # number of each cluster
1 2 3
6 6 4
>plot(x,pch=crime_k$cluster,col=crime_k$cluster,main="K-means clustering")
>  text(x,labels=city, adj=0, cex=0.5)
> ccent(x,crime_k$cluster)    # clusterwise info
              1        2         3
murder 11.60000 11.11429  3.433333
rape   41.01667 24.14286 11.600000
```

[프로그램 11.4] 범죄자료에 대한 모형기반 군집분석

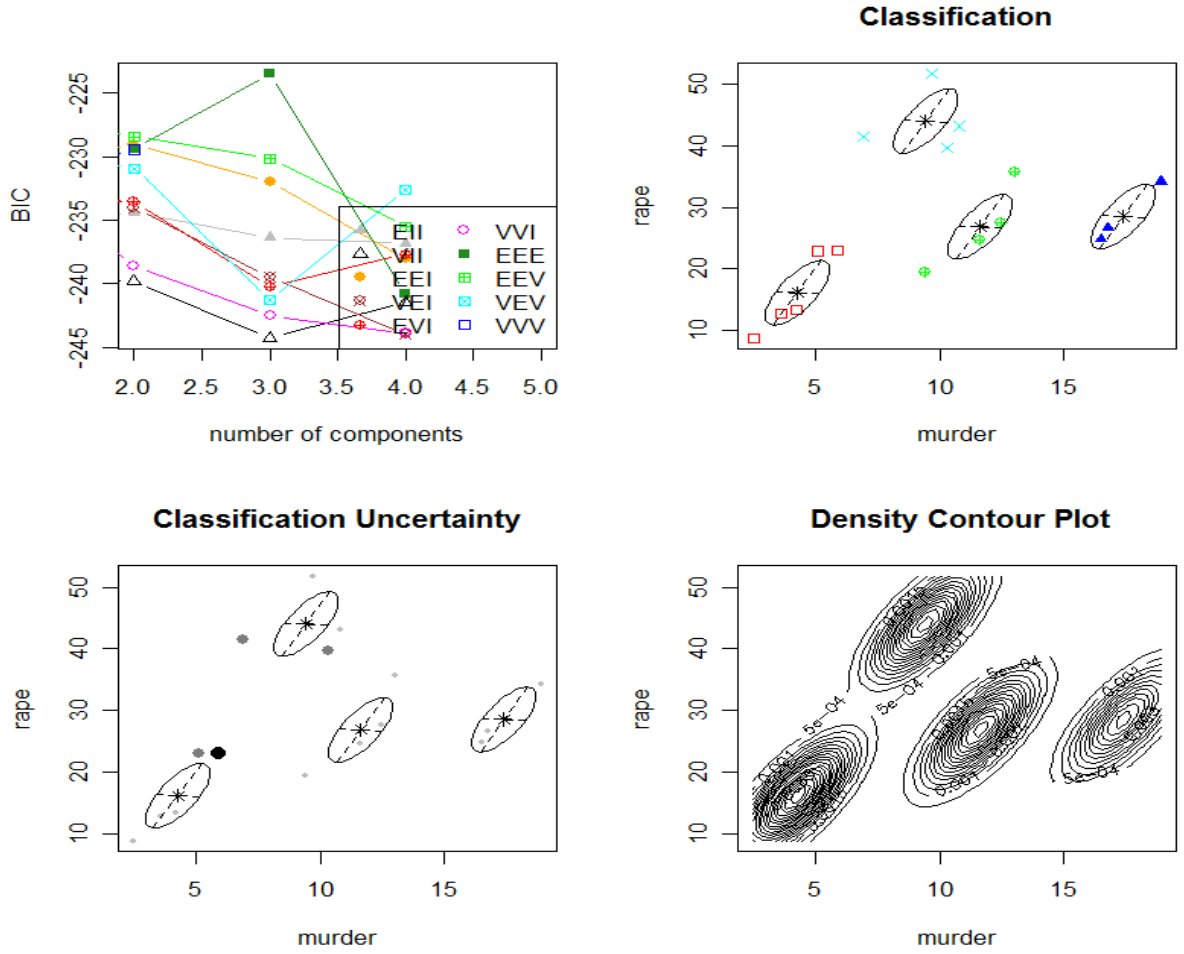
```
#####################################################
#          model-based  clustering                  #
#####################################################
library(mclust)
crime_mc=Mclust(x, 2:5)  # 군집개수 2~5개 사이 적절한 군집 결정
crime_mc
attributes(crime_mc)
crime_mc$classification  # 군집번호

mc=table(crime_mc$classification)   # number of each cluster
mc
plot(x, pch=crime_mc$classification,col=crime_mc$classification,
main="model-based clustering")
 text(x,labels=city, adj=0, cex=0.5)     # 그림 11.4

par(mfrow=c(2,2))
plot(crime_mc, data=crime[, 3:4])        # 그림 11.5
```

[결과 11.4] 모형기반 군집분석 수행결과

```
> crime_mc = Mclust(x, 2:5)
> crime_mc
 best model: EEE with 4 components
> crime_mc$classification
 [1] 1 2 3 1 4 3 2 2 1 4 4 4 3 2 2 3
> ccent(x, crime_mc$classification)
                1     2      3      4
murder 17.40000  4.26 11.625  9.425
rape   28.53333 16.14 26.850 44.050
> mc=table(crime_mc$classification)   # number of each cluster
> mc
1 2 3 4
3 5 4 4
```



[그림 11.4] 모형기반 방법 이용 결과 군집



[그림 11.5] crime 데이터의 Mclust 적용후 군집 관련 그림

왼쪽 위: 공분산행렬 형태에 따른 BIC, 오른쪽 위: 군집 형성 그림

왼쪽 아래:공분산행렬에 따른 uncertainty, 오른쪽 아래:확률밀도함수 등고선