Cluster analysis (Chapter 14)

In cluster analysis, we determine clusters from multivariate data. There are a number of questions of interest:

1. How many distinct clusters are there?
2. What is an optimal clustering approach? How do we define whether one point is more similar to one cluster or another?
3. What are the boundaries of the clusters? To which clusters do individual points belong?
4. Which variables are most related to each other? (i.e., cluster variables instead of observations)

Cluster analysis

In general, clustering can be done for multivariate data. Often, we have some measure of similarity (or dissimilarity) between points, and we cluster points that are more similar to each other (or least dissimilar).

Instead of using high-dimensional data for clustering, you could also use the first two principal components, and cluster points in the bivariate scatterplot.

Cluster analysis

For a cluster analysis, there is a data matrix

$$\mathbf{Y} = \begin{pmatrix} \mathbf{y}_1' \\ \mathbf{y}_2' \\ \vdots \\ \mathbf{y}_n' \end{pmatrix} = (\mathbf{y}_{(1)}, \ldots, \mathbf{y}_{(p)})$$

where $\mathbf{y}_{(j)}$ is the column corresponding to the jth variable. We can either cluster the rows (observation vectos) or columns (variables). Usually, we'll be interested in clustering the rows.

Cluster analysis

A standard approach is to make a matrix of the pairwise distances or dissimilarities between each pair of points. For n observations, this matrix is $n \times n$. Euclidean distance can be used, and is

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})'(\mathbf{x} - \mathbf{y})} = \sqrt{\sum_{k=1}^{p} (x_j - y_j)^2}$$

if you don't standardize. To adjust for correlations among the variables, you could use a standardized distance

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

Recall that these are Mahalonobis distances. Other measures of distance are also possible, particularly for discrete data. In some cases, a function $d(\cdot, \cdot)$ might be chosen that doesn't satisfy the properties of a distance function (for example, if it is a squared distance). In this case $d(\cdot, \cdot)$ is called a dissimilarity measure.

Cluster analysis

Another choice of distances is the Minkowski distance

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{j=1}^{p} |x_j - y_j|^r \right)^{1/r}$$

which is equivalent to the Euclidian distance for $r = 2$. If data consists of integers, $p = 2$ and $r = 1$, then this is the city block distance. I.e., if you have a rectangular grid of streets, and you can't walk diagonally, then this measures the number of blocks you need to get from point (x_1, x_2) to (y_1, y_2).

Other distances for discrete data are often used as well.

Cluster analysis

The distance matrix can be denoted $\mathbf{D} = (d_{ij})$ where $d_{ij} = d(\mathbf{y}_i, \mathbf{y}_j)$. For example, for the points

$$(x, y) = (2, 5), (4, 2), (7, 9)$$

there are $n = 3$ observations and $p = 2$, and we have (using Euclidean distance)

$$d((x_1, y_1), (x_2, y_2)) = \sqrt{(2-4)^2 + (5-2)^2} = \sqrt{4+9} = \sqrt{13} \approx 3.6$$
$$d((x_1, y_1), (x_3, y_3)) = \sqrt{(2-7)^2 + (5-9)^2} = \sqrt{25+16} = \sqrt{41} \approx 6.4$$
$$d((x_2, y_2), (x_3, y_3)) = \sqrt{(4-7)^2 + (2-9)^2} = \sqrt{9+49} = \sqrt{58} \approx 7.6$$

Thus, using Euclidean distance

$$\mathbf{D} \approx \begin{pmatrix} 0 & 3.6 & 6.4 \\ 3.6 & 0 & 7.6 \\ 6.4 & 7.6 & 0 \end{pmatrix}$$

However, if we use the city block distance, then we get

$$\begin{aligned} d((x_1, y_1), (x_2, y_2)) &= |2 - 4| + |5 - 2| = 5 \\ d((x_1, y_1), (x_3, y_3) &= |2 - 7| + |5 - 9| = 9 \\ d((x_2, y_2), (x_3, y_3) &= |4 - 7| + |2 - 9| = 10 \end{aligned}$$

$$\mathbf{D} \approx \begin{pmatrix} 0 & 5 & 9 \\ 5 & 0 & 9 \\ 9 & 10 & 0 \end{pmatrix}$$

In this case, the ordinal relationships of the magnitudes are the same (the closest and farthest pairs of points are the same for both distances), but there is no guarantee that this will always be the case.

Cluster analysis

Another thing that can change a distance matrix, including which points are the closest, is the scaling of the variables. For example, if we multiply one of the variables (say the x variable) by 100 (measuring in centimeters instead of meters), then the points are

$$(200, 5), (400, 2), (700, 9)$$

and the distances are

$$d((x_1, y_1), (x_2, y_2)) = \sqrt{(200 - 400)^2 + (5 - 2)^2} = \sqrt{200^2 + 9} = 200.0225$$

$$d((x_1, y_1), (x_3, y_3)) = \sqrt{(200 - 700)^2 + (5 - 9)^2} = \sqrt{500^2 + 16} = 500.018$$

$$d((x_2, y_2), (x_3, y_3)) = \sqrt{(400 - 700)^2 + (2 - 9)^2} = \sqrt{300^2 + 49} = 300.0817$$

Here the second variable makes a nearly negligible contribution, and the relative distances have changed, so that on the original scale, the third observation was closer to the second than to the first observation, and on the new scale, the third observation is closer to the first observation. This means that clustering algorithms will be sensitive to the scale of measurement, such as Celsius versus Farenheit, meters versus centimeters versus kilometers, etc.

Cluster analysis

The example suggests that scaling might be appropriate for variables measured on very different scales. However, scaling can also reduce the separation between clusters. What scientists usually like to see is well separated clusters, particularly if the clusters are later to be used for classification. (More on classificaiton later....)

Cluster analysis: hierarchical clustering

The idea with **agglomerative** hierarchical clustering is to start with each observation in its own singleton cluster. At each step, two clusters are merged to form a larger cluster. At the first iteration, both clusters that are merged are singleton sets (clusters with only one element), but at subsequent steps, the two clusters merged can each have any number of elements (observations).

Alternatively, **divisive** hierarchical clustering treats all elements as belonging to one big cluster, and the cluster is divided (partitioned) into two subsets. At the next step, one of the two subsets is then further divided. The procedure is continued until each cluster is a singleton set.

Cluster analysis: hierarchical clustering

The aggomerative and divisive hierarchical clustering approaches are examples of **greedy algorithms** in that they do the optimal thing at each step (i.e., something that is locally optimal), but that this doesn't guarantee producing a globally optimal solution. An alternative might be to consider all possible sets of $g \geq 1$ clusters, for which there are

$$N(n, g) = \frac{1}{g!} \sum_{k=1}^{g} \binom{g}{k} (-1)^{g-k} k^n$$

which is approximatley $g^n/g!$ for large n. The number of ways of clustering is then

$$\sum_{g=1}^{n} N(n, g)$$

For $n = 25$, the book gives a value of $\geq 10^{19}$ for this number. So it is not feasible (and never will be, no matter fast computers get) to evaluate all possible clusterings and pick the best one.

Cluster analysis

One approach for clustering is called **single linkage** or **nearest neighbor** clustering. Even if the distance between two points is Euclidean, it is not clear what the distance should be between a point a set of points, or between two sets of points. For single linkage clustering, we use an agglomerative approach, merging the two clusters that have the smallest distance, where the distance between two sets of observations, A and B is

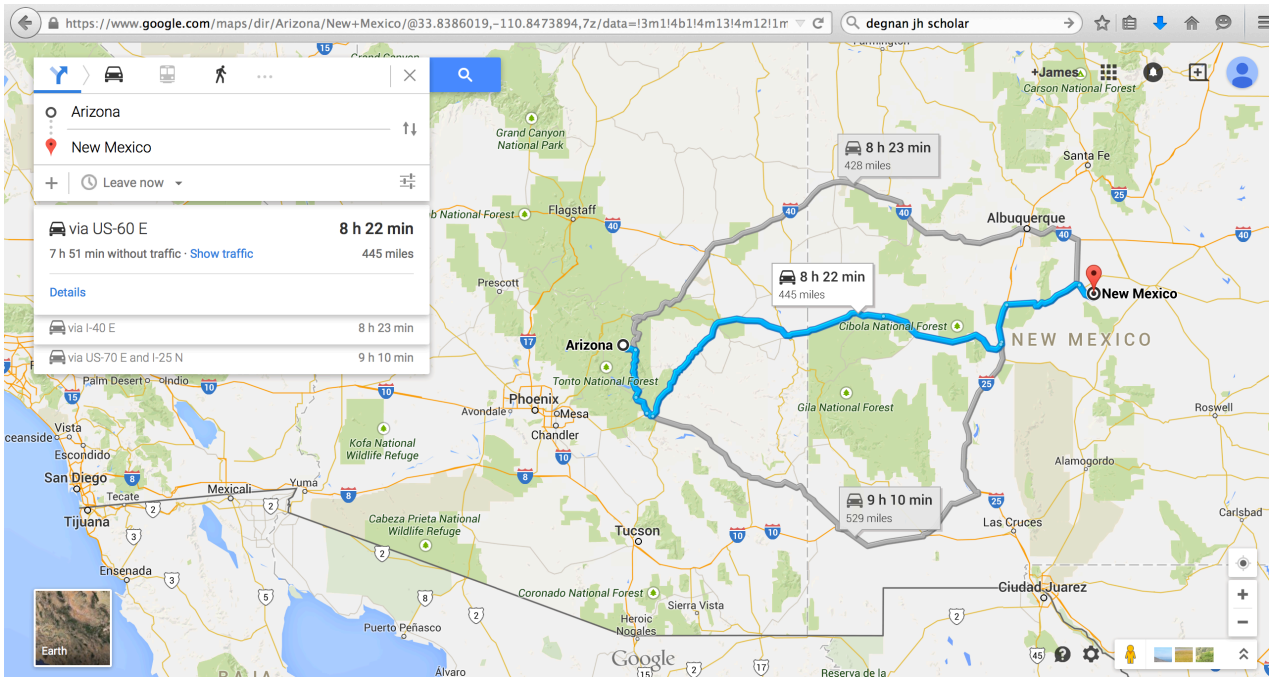
$$d(A, B) = \min\{\mathbf{y}_i, \mathbf{y}_j\}, \text{ for } \mathbf{y}_i \in A \text{ and } \mathbf{y}_j \in B$$

Cluster analysis: single linkage

As an analogy for the method, think about the distance between two geographical regions. What is the distance between say, New Mexico and California? One approach is to take the center of mass of New Mexico and the center of mass of California, and measure the distance. Another approach is to see how far it is from the western edge of NM to a southeastern part of CA. The single linkage approach is taking the latter approach, looking at the minimum distance from any location in NM to any location in CA. Similarly, if you wanted to know the distance from the US to the Europe, you might think of NY to Paris rather than say, St. Louis to Vienna, or San Diego to Warsaw.

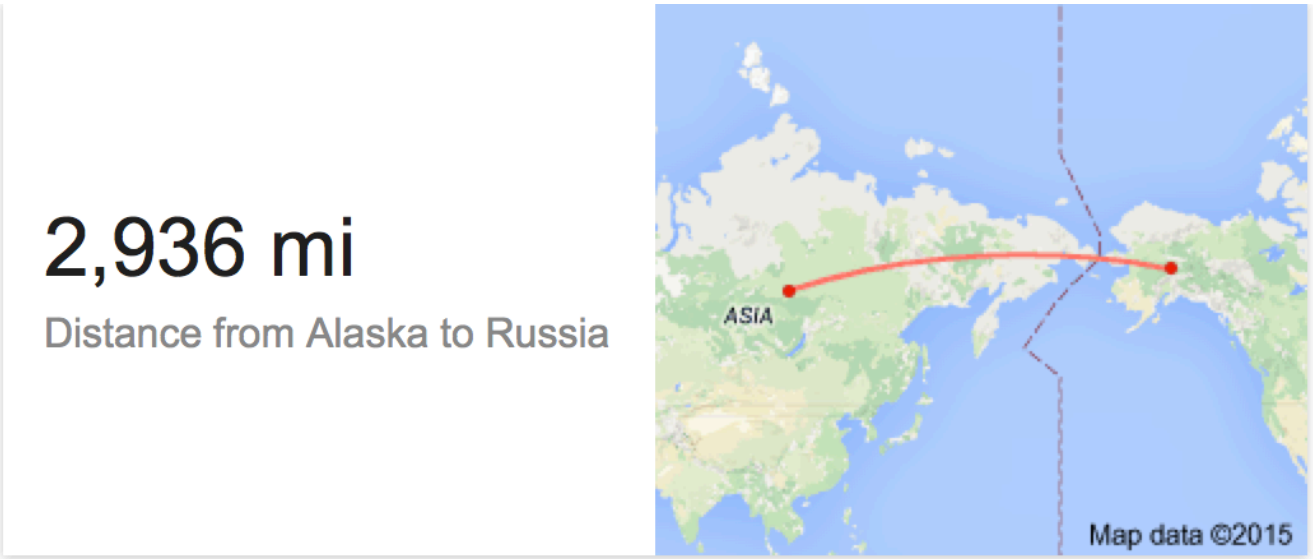
Cluster analysis

The distance from NM to AZ?



Cluster analysis

The distance from Alaska to Russia?



According to Wikipedia, "Big Diomede (Russia) and Little Diomede (USA) are only 3.8 km (2.4 mi) apart."

Cluster analysis: example with crime data

Table 14.1. City Crime Rates per 100,000 Population

City	Murder	Rape	Robbery	Assault	Burglary	Larceny	Auto Theft
Atlanta	16.5	24.8	106	147	1112	905	494
Boston	4.2	13.3	122	90	982	669	954
Chicago	11.6	24.7	340	242	808	609	645
Dallas	18.1	34.2	184	293	1668	901	602
Denver	6.9	41.5	173	191	1534	1368	780
Detroit	13.0	35.7	477	220	1566	1183	788
Hartford	2.5	8.8	68	103	1017	724	468
Honolulu	3.6	12.7	42	28	1457	1102	637
Houston	16.8	26.6	289	186	1509	787	697
Kansas City	10.8	43.2	255	226	1494	955	765
Los Angeles	9.7	51.8	286	355	1902	1386	862
New Orleans	10.3	39.7	266	283	1056	1036	776
New York	9.4	19.4	522	267	1674	1392	848
Portland	5.0	23.0	157	144	1530	1281	488
Tucson	5.1	22.9	85	148	1206	756	483
Washington	12.5	27.6	524	217	1496	1003	793

Cluster analysis: example with crime data

We'll consider an example of cluster analysis with crime data. Here there are seven categories of crime and 16 US cities. The data is a bit old, from the 1970s, when crime was quite a bit higher. Making a distance matrix results in a 16×16 matrix. To make things easier to do by hand, consider a subset of the first 6 cities. Note that we now have $n = 6$ observations and $p = 7$ variables. Having $n < p$ is not a problem for cluster analysis.

City	Distance between Cities					
Atlanta	0	536.6	516.4	590.2	693.6	716.2
Boston	536.6	0	447.4	833.1	915.0	881.1
Chicago	516.4	447.4	0	924.0	1073.4	971.5
Dallas	590.2	833.1	924.0	0	527.7	464.5
Denver	693.6	915.0	1073.4	527.7	0	358.7
Detroit	716.2	881.1	971.5	464.5	358.7	0

Cluster analysis: example with crime data

As an example of computing the distance matrix, the squared distance between Detroit and Chicago (which are geographically fairly close) is

$$\begin{aligned} d^2(\text{Detroit,Chicago}) = &(13-11.6)^2 + (35.7-24.7)^2 + (477-340)^2 \\ &+ (220-242)^2 + (1566-808)^2 + (1183-609)^2 \\ &+ (788-645)^2 = 971.5271^2 \end{aligned}$$

So the distance is approximately 971.5

Cluster analysis: example with crime data

The first step in the clustering is to pick the two cities with the smallest cities and merge them into a set. The smallest distance is between Denver and Detroit, and is 358.7. We then merge them into a cluster $C_1 = \{(\text{Denver,Detroit})\}$. This leads to a new distance matrix

Atlanta	0	536.6	516.4	590.2	693.6
Boston	536.6	0	447.4	833.1	881.1
Chicago	516.4	447.4	0	924.0	971.5
Dallas	590.2	833.1	924.0	0	464.5
C_1	693.6	881.1	971.5	464.5	0

Cluster analysis: example with crime data

The new distance matrix is 5×5, and the rows and columns for Denver and Detroit have been replaced with a single row and column for cluster C_1. Distances between singleton cities remain the same, but making the new matrix requires computing the new distances, $d((\text{Atlanta}, C_1))$, $d((\text{Boston}, C_1))$, etc. The distance from Atalanta to C_1 is the minimum of the distances from Atlanta to Denver and Atlanta to Detroit, which is the minimum of 693.6 (distance to Denver) and 716.2 (distance to Detroit), so we use 693.6 as the distance between Atlanta and C_1.

The next smallest distance is between Boston and Chicago, so we create a new cluster, $C_2 = \{(\text{Boston,Chicago})\}$.

Cluster analysis: example with crime data

The updated matrix is now 4×4. The distance between C_1 and C_2 is the minimum between all pairs of cities with one in C_1 and one in C_2. You can either compute this from scratch, or, using the the previous matrix, think of the distance between C_1 and C_2 as the minimum of $d(C_1, \text{Boston})$ and $d(C_1, \text{Chicago})$. This latter recursive approach is more efficient for large matrices.

Atlanta	0	516.4	590.2	693.6
C_2	516.4	0	833.1	881.1
Dallas	590.2	833.1	0	464.5
C_1	693.6	881.1	464.5	0

Cluster analysis: example with crime data

Atlanta	0	516.4	590.2
C_2	516.4	0	833.1
C_3	590.2	833.1	0

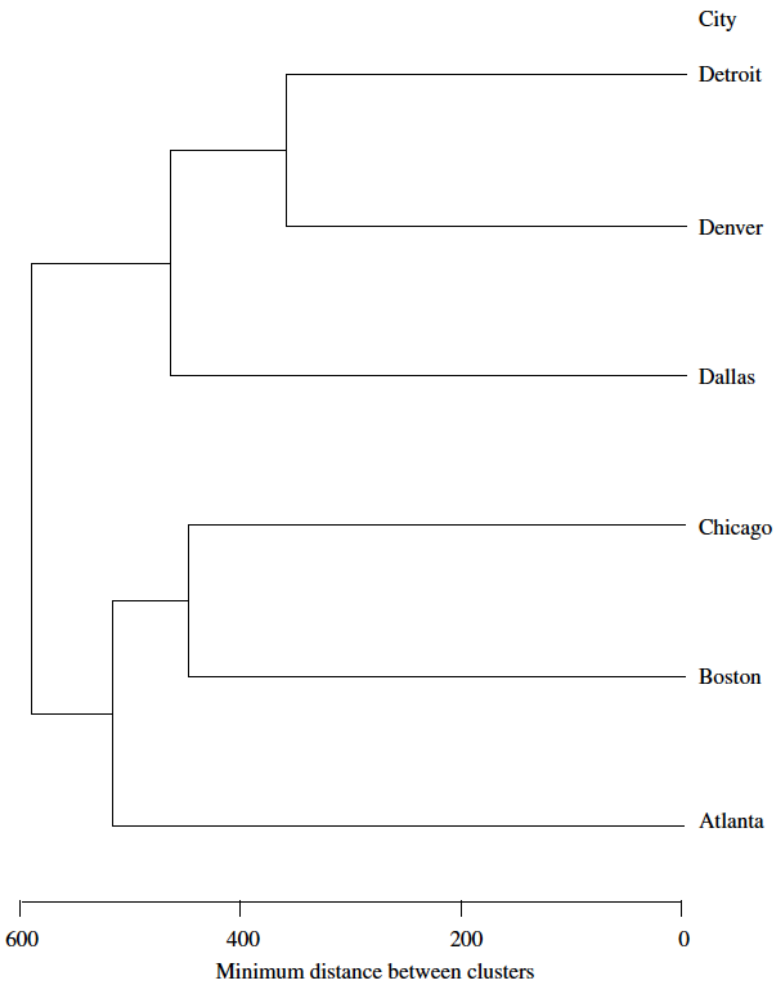
The smallest distance is 516.4, which defines $C_4 = \{\text{Atlanta}, C_2\}$. The distance matrix for C_3 and C_4 is

C_3	0	590.2
C_4	590.2	0

Cluster analysis: example with crime data

At the last step, once you have two clusters, they are joined without any decision having to be made, but it is still useful to compute the resulting distance as 590.2 rather than 833.1 so that we can draw a diagram (called a **dendrogram**) to show the sequence of clusters.

Cluster analysis: example with crime data

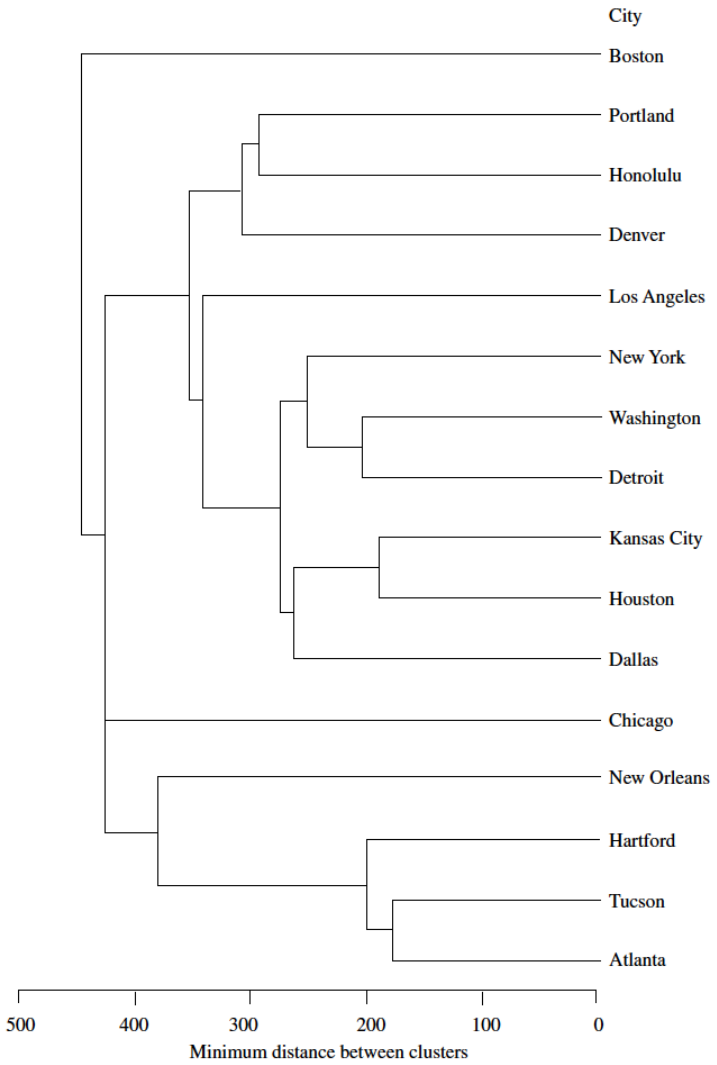


Cluster analysis: example with crime data

The diagram helps visualize which cities have similar patterns of crime. The patterns might suggest hypotheses. For example, in the diagram, the top half of the cities are west of the bottom half of the cities, so you might ask if there is geographical correlation in crime patterns?

Of course, this pattern might not hold looking at all the data from the 16 cities.

Cluster analysis: example with crime data



Cluster analysis: complete linkage and average linkage

With **complete linkage**, the distance between two clusters is the maximum distance between all pairs with one from each cluster. This is sort of like a worst-case scenario distance. (i.e., if one person is in AZ and one in NZ, the distance is treated as the farthest apart that they might be).

With **average linkage**, the distance between two clusters is the average distance between all pairs with one from each cluster.
For the crime data, the subset of six cities results in the same clustering pattern for all three types of linkage. Note that the first cluster is necessarily the same for all three methods regardless of the data. However the dendrogram differs between single linkage versus complete or average linkage. Complete linkage and average linkage lead to the same dendrogram pattern (but different times).

Cluster analysis: example with crime data

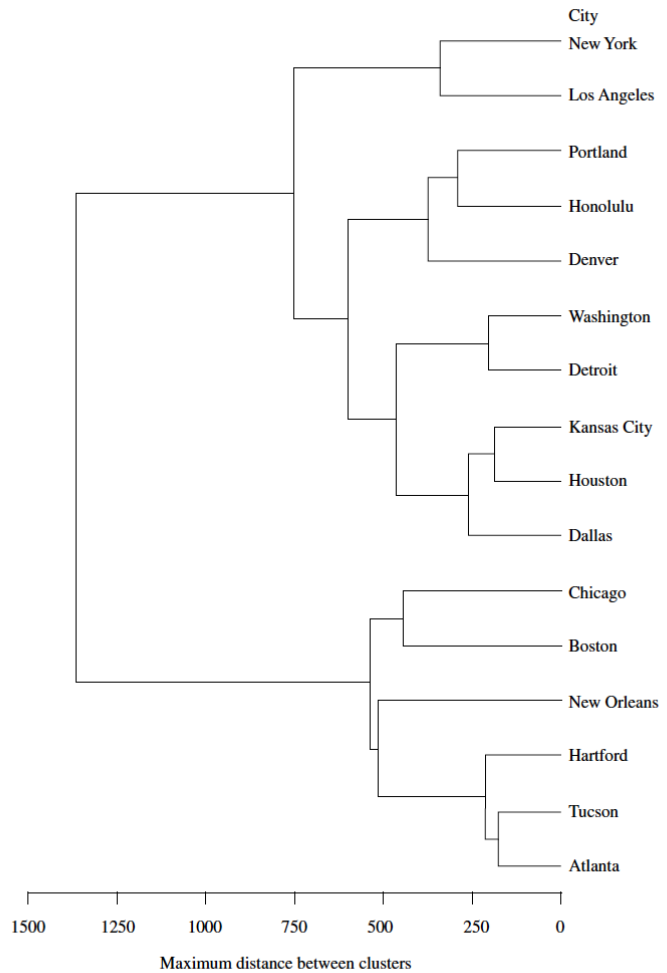


Figure 14.5. Dendrogram for complete linkage of the complete city crime data of Table 14.1 [see Example 14.3.3(b)].

Cluster analysis: example with crime data

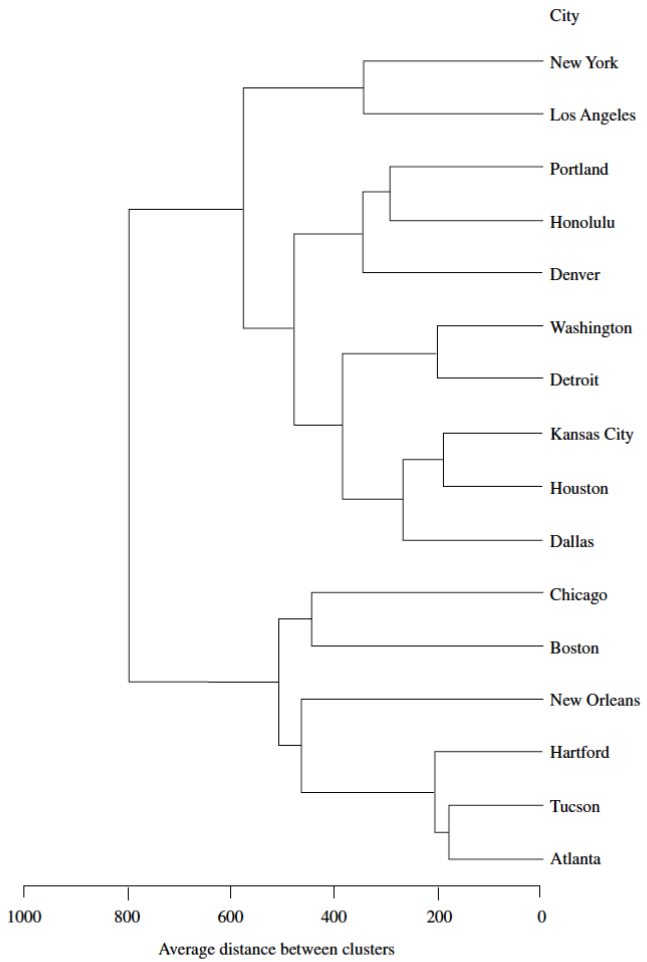


Figure 14.6. Dendrogram for average linkage clustering of the data in Table 14.1 (see Example 14.3.4).

Cluster analysis: centroid approach

When using centroids, the distance between clusters is the distance between mean vectors

$$D(A, B) = d(\bar{\mathbf{y}}_A, \bar{\mathbf{y}}_B)$$

where

$$\bar{\mathbf{y}}_A = \frac{1}{n_A} \sum_{i \in A} \mathbf{y}_i$$

When two clusters are joined, the new centroid is

$$\bar{\mathbf{y}}_{AB} = \frac{1}{n_A + n_B} \sum_{i \in A \cup B} \mathbf{y}_i = \frac{n_A \bar{\mathbf{y}}_A + n_B \bar{\mathbf{y}}_B}{n_A + n_B}$$

Cluster analysis: median approach

The median approach weights different clusters differently so that each cluster gets an equal weight instead of clusters with more elements getting more weight. For this approach, the distance between two clusters is

$$D(A, B) = \frac{1}{2}\overline{\mathbf{y}}_A + \frac{1}{2}\overline{\mathbf{y}}_B$$

Cluster analysis: example with crime data

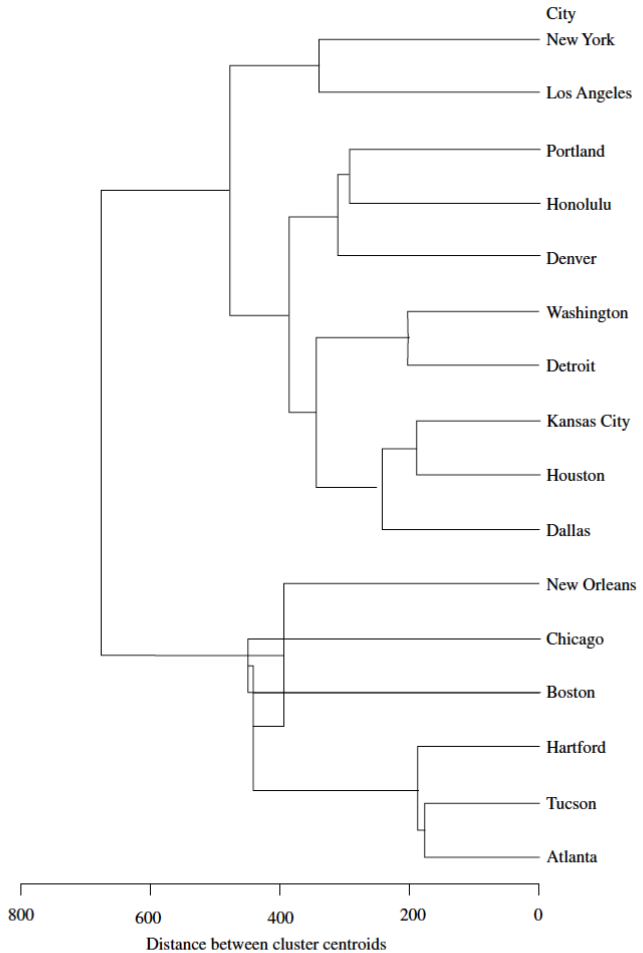


Figure 14.7. Dendrogram for the centroid clustering of the complete city crime data in Table 14.1 (see Example 14.3.5).

Cluster analysis

A variation on the centroid method is Ward's method which computes the sums of squared distances within each cluster, SSE_A and SSE_B as

$$SSE_A = \sum_{i \in A} (\mathbf{y}_i - \bar{\mathbf{y}}_A)'(\mathbf{y}_i - \bar{\mathbf{y}}_A)$$

$$SSE_B = \sum_{i \in B} (\mathbf{y}_i - \bar{\mathbf{y}}_A)'(\mathbf{y}_i - \bar{\mathbf{y}}_A)$$

and the between sum of squares as

$$SSE_{AB} == \sum_{i \in A \cup B} (\mathbf{y}_i - \bar{\mathbf{y}}_{AB})'(\mathbf{y}_i - \bar{\mathbf{y}}_A)$$

Two clusters are joined if they minimize $I_{AB} = SSE_{AB} - (SSE_A + SSE_B)$. That is, over all possible clusters, A, B at a given step, merge the two clusters that minimize I_{AB}. The value of I_{AB} when A and B are both singltons is $\frac{1}{2}d^2(\mathbf{y}_i, \mathbf{y}_j)$, so essentially a squared distance.

This is an ANOVA-inspired method and results in being more likely to result in smaller clusters being agglomerated than the centroid method. For this data, Ward's method results in 6 two-city cluster, whereas the centroid method results in 5 two-city clusters. Different methods might have different properties in terms of the sizes of clusters they produce.

Cluster analysis

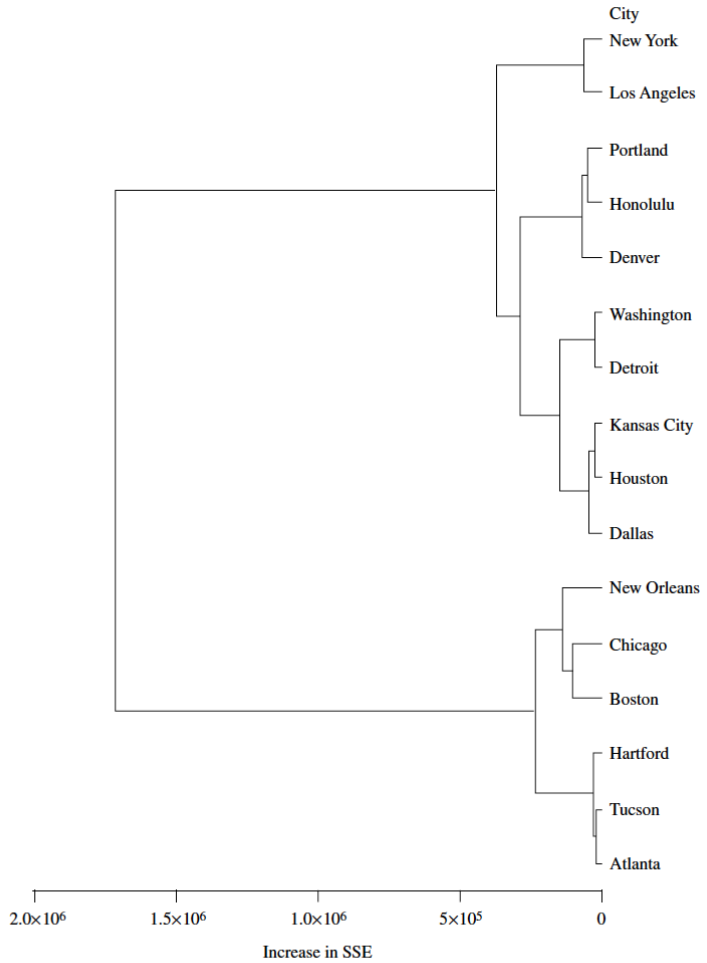


Figure 14.9. Dendrogram for Ward's method applied to the complete city crime data in Table 14.1 (see Example 14.3.7).

To unify all of these methods, the **flexible beta method** gives a generalization for which the previous methods are special cases. Let the distance from a recently formed cluster AB to another cluster C be

$$D(C, AB) = \alpha_A D(A, C) + \alpha_B D(B, C) + \beta D(A, B) + \gamma |D(A, C) - D(B, C)|$$

where $\alpha_A + \alpha_B + \beta = 1$. If $\gamma = 0$ and $\alpha_A = \beta_B$, then the constraint that $\alpha_A + \alpha_B + \beta = 1$ means that different choices of β determine the clustering, which is where the name comes from. The following parameter choices lead to the different clustering methods:

Cluster analysis: example with crime data

Table 14.2. Parameter Values for (14.20)

Cluster Method	α_A	α_B	β	γ
Single linkage	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$
Complete linkage	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
Average linkage	$\frac{n_A}{n_A+n_B}$	$\frac{n_B}{n_A+n_B}$	0	0
Centroid	$\frac{n_A}{n_A+n_B}$	$\frac{n_B}{n_A+n_B}$	$\frac{-n_An_B}{(n_A+n_B)^2}$	0
Median	$\frac{1}{2}$	$\frac{1}{2}$	$-\frac{1}{4}$	0
Ward's method	$\frac{n_A+n_C}{n_A+n_B+n_C}$	$\frac{n_B+n_C}{n_A+n_B+n_C}$	$\frac{-n_C}{n_A+n_B+n_C}$	0
Flexible beta	$(1-\beta)/2$	$(1-\beta)/2$	$\beta(<1)$	0

Cluster analysis

Crossovers occurred in some of the plots. This occurs when the distances between later mergers are smaller than distances at earlier mergers. Clustering methods for which this cannot occur are called **monotonic** (that is distances are non-decreasing).

Single linkage and complete linkage are monotonic, and the flexible beta family of methods are monotonic if $\alpha_A + \alpha_B + \beta \geq 1$

Cluster analysis

Clustering methods can be **space conserving**, **space contracting**, or **space dilating**. Space contracting means that larger clusters tend to be formed, so that singletons are more likely to cluster with non-singleton clusters. This is also called chaining, and means that very spread out observations can lead to one large cluster. Space dilating means that singletons tend to cluster with other singletons rather than with non-singleton clusters. These properties affect how balanced or unbalanced trees are likely to be. Space conserving methods are neither space-contracting nor space-dilating.

Single linkage clustering is space contracting whereas complete linkage is space dilating. Flexible beta is is space contracting for $\beta > 0$, space dilating for $\beta < 0$,and space-conserving for $\beta = 0$.

To be space-conserving, if clusters satisfy

$$D(A, B) < D(A, C) < D(B, C)$$

(think of points spread out on a line so that A is between B and C but closer to B than C), then

$$D(A, C) < D(AB, C) < D(B, C)$$

Single linkage violates the first inequality because $D(AB, C) = min\{D(A, C), D(B, C)\} = D(A, C)$. And complete linkage violates the second inequality because $D(AB, C) = \max\{D(A, C), D(B, C)\} = D(B, C)$.

Cluster analysis

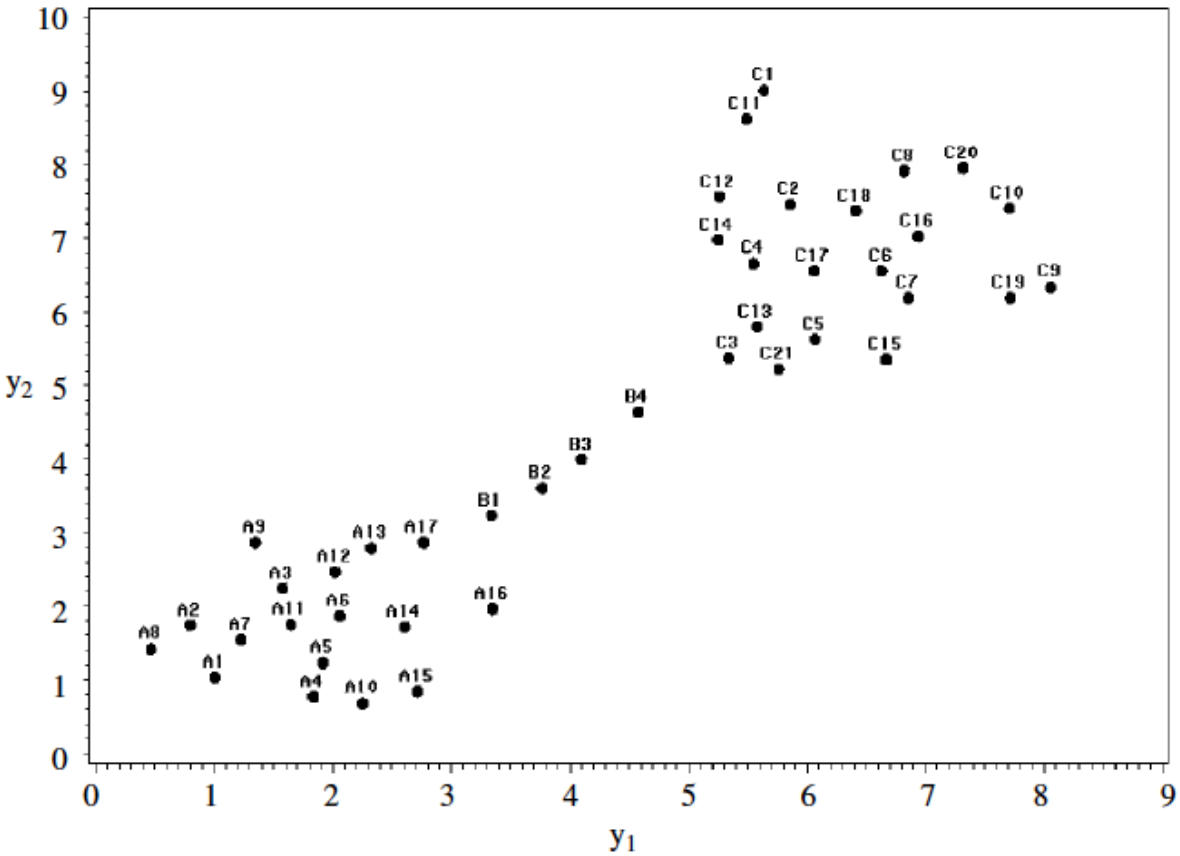


Figure 14.12. Two distinct clusters with intervening individuals.

Cluster analysis

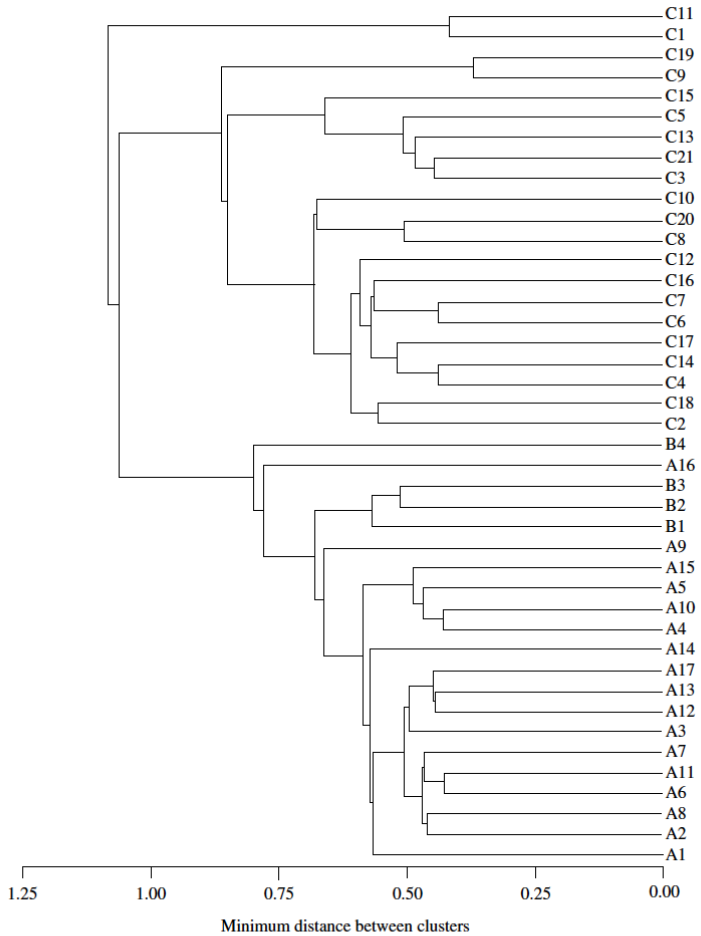


Figure 14.13. Single linkage clustering of the data in Figure 14.12.

Cluster analysis

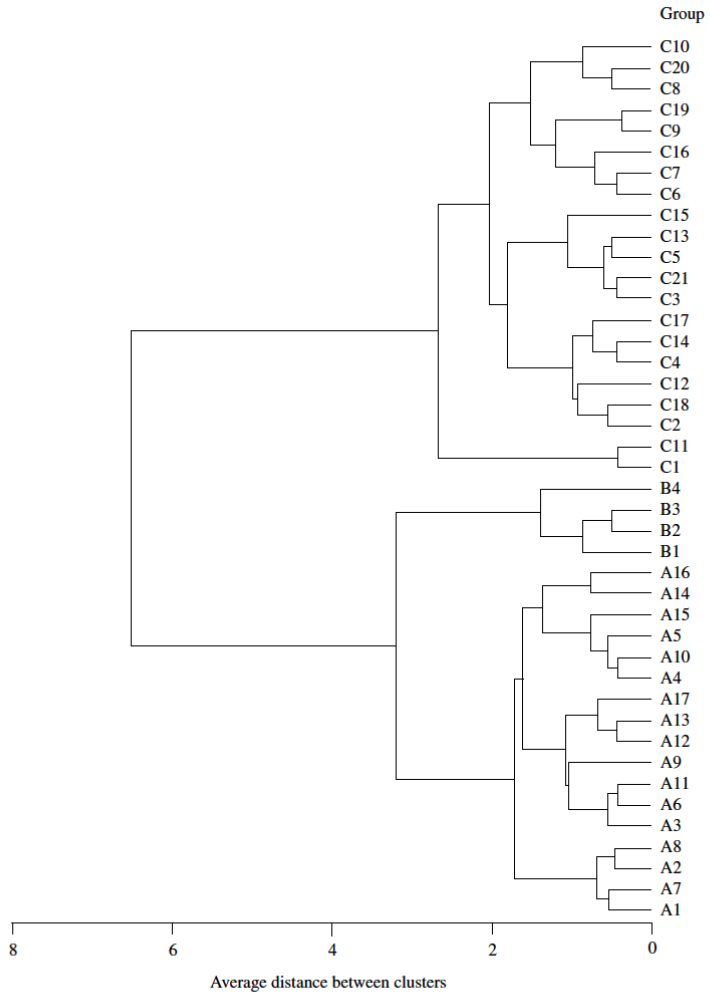


Figure 14.14. Average linkage clustering of the data in Figure 14.12.

Cluster analysis

The average linkage approach results in more two-observation clusters (14 versus 12), and results in the B group all clustering together, whereas for single linkage, $B4$ is outside $\{A1, \ldots, A17, B1, B2, B3\}$.

Cluster analysis

Effect of variation. For the average linkage approach, the distance between two clusters increases if the variation in one of the clusters increases, even if the centroid remains the same. Furthermore, distance based on single linkage can decrease while the distance based on average linkage can increase.

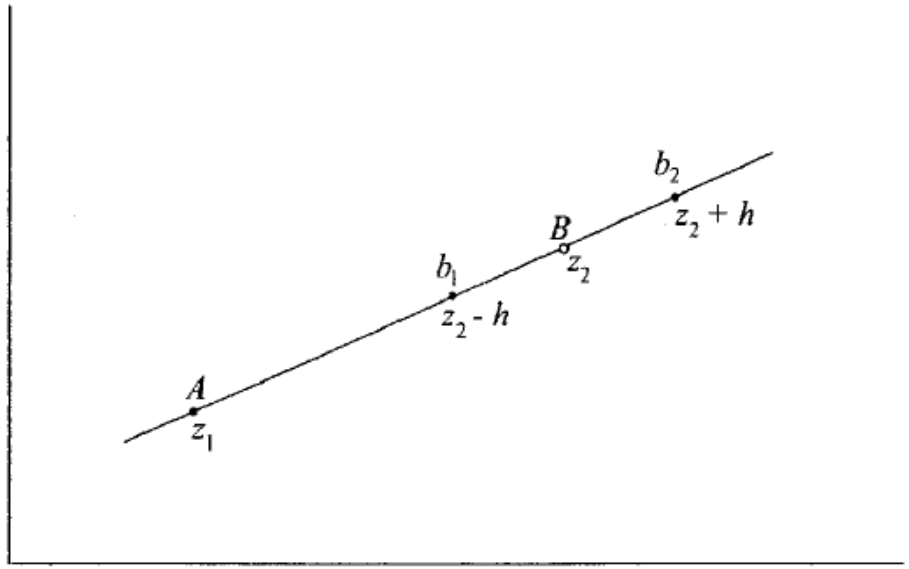


Figure 14.15. Clusters in a single dimension.

Effect of variation.
Suppose A has a single point at (0,0) and B has two points at (4,0) and (6,0). Then the average squared distance is

$$[(4-0)^2 + (6-0)^2]/2 = 52/2 = 26$$

whereas the average squared distance if B has two points at (5,0) is $(5^2 + 5^2)/2 = 25 < 26$. The actual distance are then $\sqrt{25} < \sqrt{26}$. If instead B has points (3,0) and (7,0), then the average squared distance is $(3^2 + 7^2)/2 = 58/2 = 29$.

Cluster analysis

For a divisive technique where you cluster on one qunatitative variable, you can consider all partitions of n observations into n_1 and n_2 observations for groups 1 and 2, with the only constraint being that $n_1 + n_2 = n$ with $n_i \geq 1$. Assuming that group 1 is at least as large as group 2, there are $\lfloor n/2 \rfloor$ choices for the group sizes. For each group size, there are $\binom{n}{n_1}$ of picking which elements belong to group 1 (and therefore also to group 2). For each such choice, you can find the the groups that minimize

$$SSB = n_1(\overline{y}_1 - \overline{y})^2 + n_2(\overline{y}_2 - \overline{y})$$

Each subcluster can then be divided again repeatedly until only singleton clusters remain.

Cluster analysis

For binary variables, you could instead cluster on one binary variable at a time. This is quite simple as it doesn't require computing a sum of squares. This also corresponds how you might think of animal taxonomy: Animals are either cold-blooded or warm-blooded. If warm blooded, they either lay eggs or don't. If they lay eggs, then they are monotremes (platypus, echidna). If they don't lay eggs, then they either have pouches or don't (marsupials versus placental mammals). And so forth. This type of classification is appealing in its simplicity, but the order of binary variables can be somewhat arbitrary.

Cluster analysis

There are also non-hierarchical methods of clustering, including partitioning by k-means clustering, using mixtures of distributions, and density estimation.

For partitioning, initial clusters are formed, and in the process, items can be reallocated to different clusters, whereas in hierarchical clustering, once an element is in a cluster, it is fixed there. First select g elements (observations) to be used as seeds. This can be done in several ways

1. pick g items at random
2. pick the first g items in the data set
3. find g items that are furthest apart
4. partition the space into a grid and pick g items from different section of the grid that are roughly equally far apart
5. pick items in a grid of points and create artificial observations that are equally spaced.

Cluster analysis

For all of these methods, you might want to constrain the choices so that the seeds are sufficiently far apart. For example, if choosing point randomly, then if the second choice is too close to the first seed, then pick a different random second seed.

For these methods, the number g must be given in advance (the book uses g rather than k), and sometimes a cluster analysis is run several times with different choices for g. An alternative method is to specify a minimum distance between points. Then pick the first item in the data set (you could shuffle the rows to randomize this choice). Then pick the next observation that is more than the minimum distance from the first. Then pick the next observation that is more than the minimum distance from the first two, etc. Then the number of seeds will emerge and be a function of the minimum distance chosen. In this case, you could re-run the analysis with different minimum distances which result in different values for g.

Cluster analysis

Once the seeds are chosen, each point in the data set is assigned to the closest seed. That is for each point that isn't a seed, a distance is chosen (usually Euclidean) and the distance between each non-seed and the seed is computed. Then each non-seed is assigned to the seed with the smallest distance.

Once the clusters are chosen, the centroids are computed, and distances between each point and the centroids of the g clusters are computed. If a point is closer to a different centroid than its current centroid, then it is reallocated to a different cluster. This results in a new set of clusters, for which new centroids can be computed, and the process can be reiterated. The reallocation process should eventually converge so that points stop being reallocated.

Cluster analysis

You could also combine the k-means approach with hierarchical clustering as a way of finding good initial seeds. If you run the hierarchical clustering first, then choose some point at which it has g clusters (it initially has n clusters, then one cluster at the end of the process, so at some point it will have g clusters). You could then compute the centroids of these clusters and start the reallocation process. This could potentially improve the clustering that was done by the hierarchical method.

An issue with k-means clustering is that it is sensitive to the initial choice of seeds. Consequently, it is reasonable to try different starting seeds to see if you get similar results. If not, then you should be less confident in the resulting clusters. If the clusters are robust to the choice of starting seeds, this suggests more structure and natural clustering in the data.

Cluster analysis

Clustering is often combined with other techniques such as principal components to get an idea of how many clusters there might be. This is illustrated with an example looking at sources of protein in European countries.

Cluster analysis

Table 14.7. Protein Data

Country	Red Meat	White Meat	Eggs	Milk	Fish	Cereals	Starchy Foods	Nuts	Fruits/Veg.
Albania	10.1	1.4	.5	8.9	.2	42.3	.6	5.5	1.7
Austria	8.9	14.0	4.3	19.9	2.1	28.0	3.6	1.3	4.3
Belgium	13.5	9.3	4.1	17.5	4.5	26.6	5.7	2.1	4.0
Bulgaria	7.8	6.0	1.6	8.3	1.2	56.7	1.1	3.7	4.2
Czech.	9.7	11.4	2.8	12.5	2.0	34.3	5.0	1.1	4.0
Denmark	10.6	10.8	3.7	25.0	9.9	21.9	4.8	.7	2.4
E. Germany	8.4	11.6	3.7	11.1	5.4	24.6	6.5	.8	3.6
Finland	9.5	4.9	2.7	33.7	5.8	26.3	5.1	1.0	1.4
France	18.0	9.9	3.3	19.5	5.7	28.1	4.8	2.4	6.5
Greece	10.2	3.0	2.8	17.6	5.9	41.7	2.2	7.8	6.5
Hungary	5.3	12.4	2.9	9.7	.3	40.1	4.0	5.4	4.2
Ireland	13.9	10.0	4.7	25.8	2.2	24.0	6.2	1.6	2.9
Italy	9.0	5.1	2.9	13.7	3.4	36.8	2.1	4.3	6.7
Netherlands	9.5	13.6	3.6	23.4	2.5	22.4	4.2	1.8	3.7
Norway	9.4	4.7	2.7	23.3	9.7	23.0	4.6	1.6	2.7
Poland	6.9	10.2	2.7	19.3	3.0	36.1	5.9	2.0	6.6
Portugal	6.2	3.7	1.1	4.9	14.2	27.0	5.9	4.7	7.9
Romania	6.2	6.3	1.5	11.1	1.0	49.6	3.1	5.3	2.8
Spain	7.1	3.4	3.1	8.6	7.0	29.2	5.7	5.9	7.2
Sweden	9.9	7.8	3.5	24.7	7.5	19.5	3.7	1.4	2.0
Switzerland	13.1	10.1	3.1	23.8	2.3	25.6	2.8	2.4	4.9
UK	17.4	5.7	4.7	20.6	4.3	24.3	4.7	3.4	3.3
USSR	9.3	4.6	2.1	16.6	3.0	43.6	6.4	3.4	2.9
W. Germany	11.4	12.5	4.1	18.8	3.4	18.6	5.2	1.5	3.8
Yugosloslavia	4.4	5.0	1.2	9.5	.6	55.9	3.0	5.7	3.2

Cluster analysis: how many clusters?

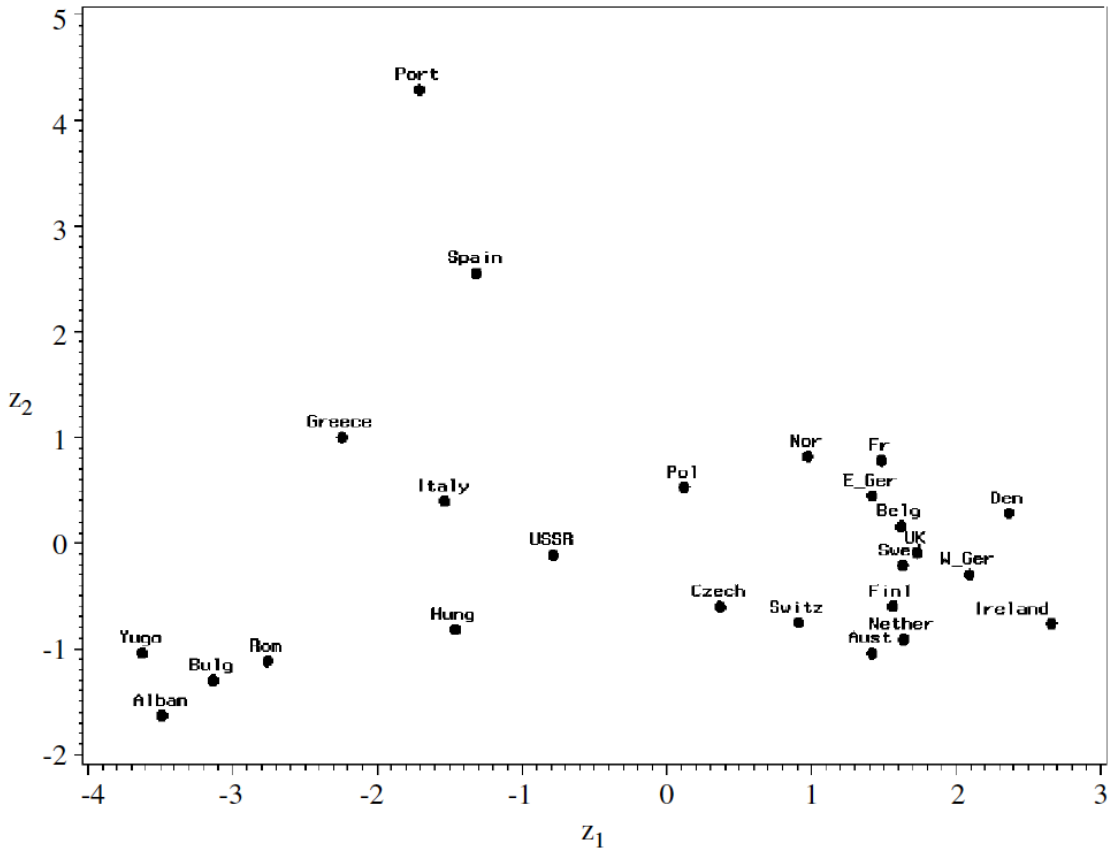


Figure 14.16. First two principal components z_1 and z_2 for the protein data in Table 14.7.

Cluster analysis: how many clusters?

The book suggests that the PCA indicates at least 5 clusters. This isn't obvious to me, it seems like it could be 3–5 to me. But we can use $g = 5$ for the number of clusters. You can reanalyze (in homework) with different numbers of clusters. The book considers four methods of picking starting seeds:

1. Select at random g observations that are at least a distance r apart.
2. Select the first g observations that are at least a distance r apart.
3. Select the g observations that are mutually farthest apart.
4. Use the g centroids from the g-cluster solution from the average linkage (hierarchical) clustering method.

Cluster analysis: k means example

Table 14.8. k-Means Cluster Solution for Seeds Chosen at Random

Country	Cluster	Distance from Centroid	Country	Cluster	Distance from Centroid
Portugal	1	1.466	Sweden	4	1.594
Spain	1	1.466	E. Germany	4	1.966
Netherlands	2	1.123	Norway	4	2.031
Austria	2	1.217	France	4	2.621
Czech.	2	1.385	Romania	5	1.066
Switzerland	2	1.657	Yugoslavia	5	1.701
Poland	2	1.914	Bulgaria	5	1.741
Ireland	3	1.334	Italy	5	2.092
UK	3	1.821	Hungary	5	2.443
Finland	3	2.261	USSR	5	2.613
Belgium	4	1.201	Albania	5	2.725
W. Germany	4	1.405	Greece	5	2.741

Cluster analysis: k means example

Table 14.9. k-Means Cluster Solution Using the First Five Observations as Seeds

Country	Cluster	Distance from Centroid	Country	Cluster	Distance from Centroid
Albania	1	.000	Romania	4	1.415
Netherlands	2	.648	Bulgaria	4	1.587
Austria	2	1.000	Yugoslavia	4	1.784
W. Germany	2	1.087	Italy	4	1.898
Switzerland	2	1.489	Greece	4	2.450
Belgium	3	1.368	Poland	5	1.709
Sweden	3	1.462	Czech.	5	1.956
Denmark	3	1.666	USSR	5	2.218
Ireland	3	1.832	E. Germany	5	2.285
Norway	3	1.927	Spain	5	2.344
UK	3	2.076	Hungary	5	2.558
Finland	3	2.341	Portugal	5	3.859
France	3	2.629			

Cluster analysis: k means example

Table 14.10. k-Means Cluster Solution Using as Seeds the Five Observations Furthest Apart

Country	Cluster	Distance from Centroid	Country	Cluster	Distance from Centroid
Romania	1	.601	France	2	2.358
Yugoslavia	1	1.159	Poland	2	2.405
Bulgaria	1	1.435	UK	2	2.537
Albania	1	2.421	Greece	3	1.075
Hungary	1	2.540	Italy	3	1.075
Belgium	2	.956	Portugal	4	1.466
W. Germany	2	1.012	Spain	4	1.466
Netherlands	2	1.416	Norway	5	1.054
Austria	2	1.663	Sweden	5	1.191
Czech.	2	1.706	Finland	5	1.545
Switzerland	2	1.713	Denmark	5	1.708
Ireland	2	1.839	USSR	5	2.780
E. Germany	2	2.042			

Cluster analysis: k means example

Table 14.11. k-Means Cluster Solution Using Seeds from Average Linkage

Country	Cluster	Distance from Centroid	Country	Cluster	Distance from Centroid
Romania	1	.970	Norway	2	2.287
Yugoslavia	1	1.182	UK	2	2.354
Bulgaria	1	1.339	France	2	2.600
Albania	1	1.970	Finland	2	2.683
Belgium	2	1.152	Greece	3	1.075
W. Germany	2	1.245	Italy	3	1.075
Netherlands	2	1.547	Portugal	4	1.466
Sweden	2	1.604	Spain	4	1.466
Ireland	2	1.744	Czech.	5	1.337
Denmark	2	1.766	Poland	5	1.579
Switzerland	2	1.831	USSR	5	1.964
Austria	2	2.037	Hungary	5	2.023
E. Germany	2	2.251			

Cluster analysis: k means example

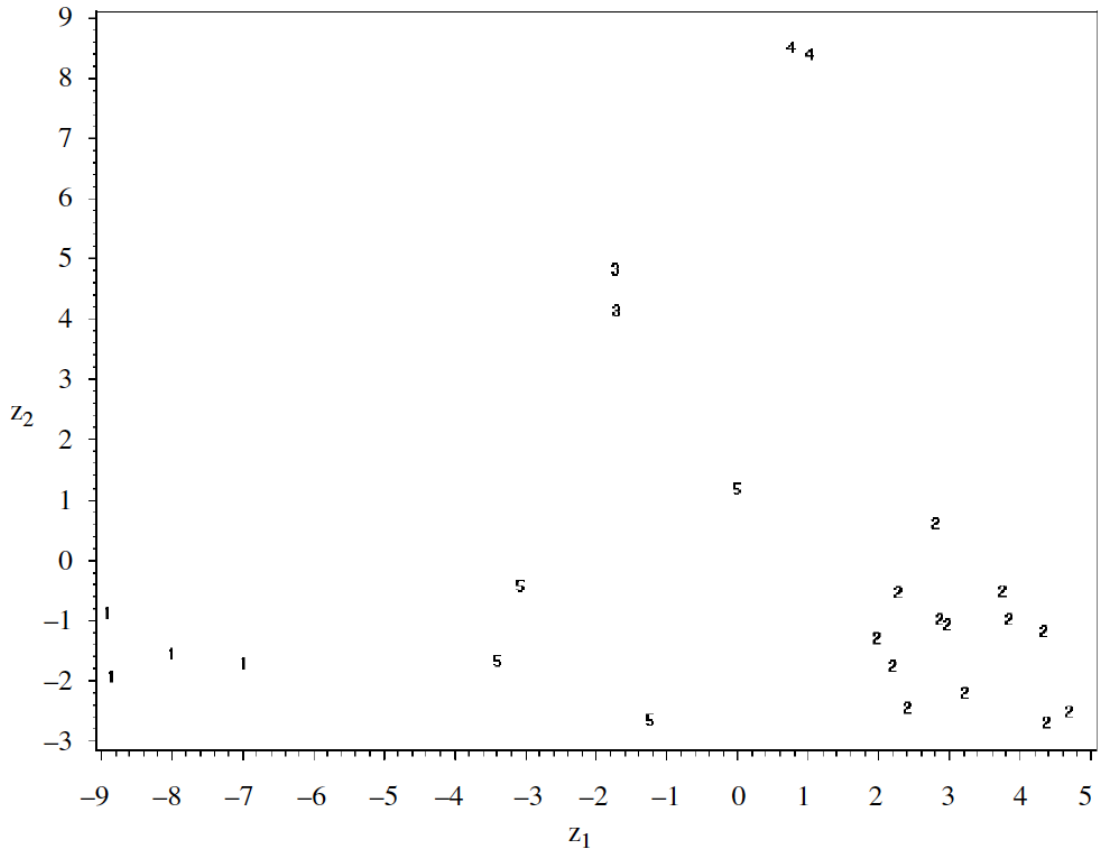


Figure 14.20. First two discriminant functions z_1 and z_2 for the clusters in Table 14.11.

Cluster analysis based on MANOVA

A different approach is motivated by MANOVA but isn't used as often. The idea is that once you have clusters assigned, you have multivariate data from g groups. Then you could think of doing MANOVA to see if the groups are different. Part of MANOVA is the generation of the $\mathbf{E}$ and $\mathbf{H}$ matrices which are the within and between cluster sums of squares. So we could pick clusters (once the number g has been fixed) to minimize some function of these matrices. Possible criteria are

1. minimize tr$\mathbf{E}$
2. minimize $|\mathbf{E}|$
3. maximimize tr$(\mathbf{E}^{-1}\mathbf{H})$

Cluster analysis in R

k-measn clustering can be done with the `kmeans()` function in R. For the European protein data, use

```
> x <- read.table("protein.txt",header=T)
> x2 <- x[,2:10] # x2 has numeric data only, not country names
> cluster <- kmeans(x2,centers=5)
```

The `centers` argument can either be the number of clusters (here $g = 5$) or a set of seed vectors, which you would have to compute by hand if you want some other method than randomly chosen observations. If the number of clusters is given, then the starting seeds are randomly chosen.

Cluster analysis in R

To create a new data frame of countries with their cluster assignment, you can do this

```
> cluster2 <- data.frame(x$country,cluster$cluster)
> cluster2
      x.country cluster.cluster  # weird variable names
1       Albania                4
2       Austria                5
3       Belgium                5
> colnames(cluster2) <- c("country","cluster")
> cluster2
       country cluster
1       Albania       4
2       Austria       5
3       Belgium       5
```

Cluster analysis in R

To sort the countries by the cluster number

```
> cluster2[order(cluster2$cluster),]
          country cluster
6         Denmark       1
8         Finland       1
15         Norway       1
20         Sweden       1
4        Bulgaria       2
18        Romania       2
25 Yugosloslavia       2
7        EGermany       3
17       Portugal       3
19          Spain       3
1         Albania       4
5          Czech.       4
10         Greece       4
11        Hungary       4
13          Italy       4
```

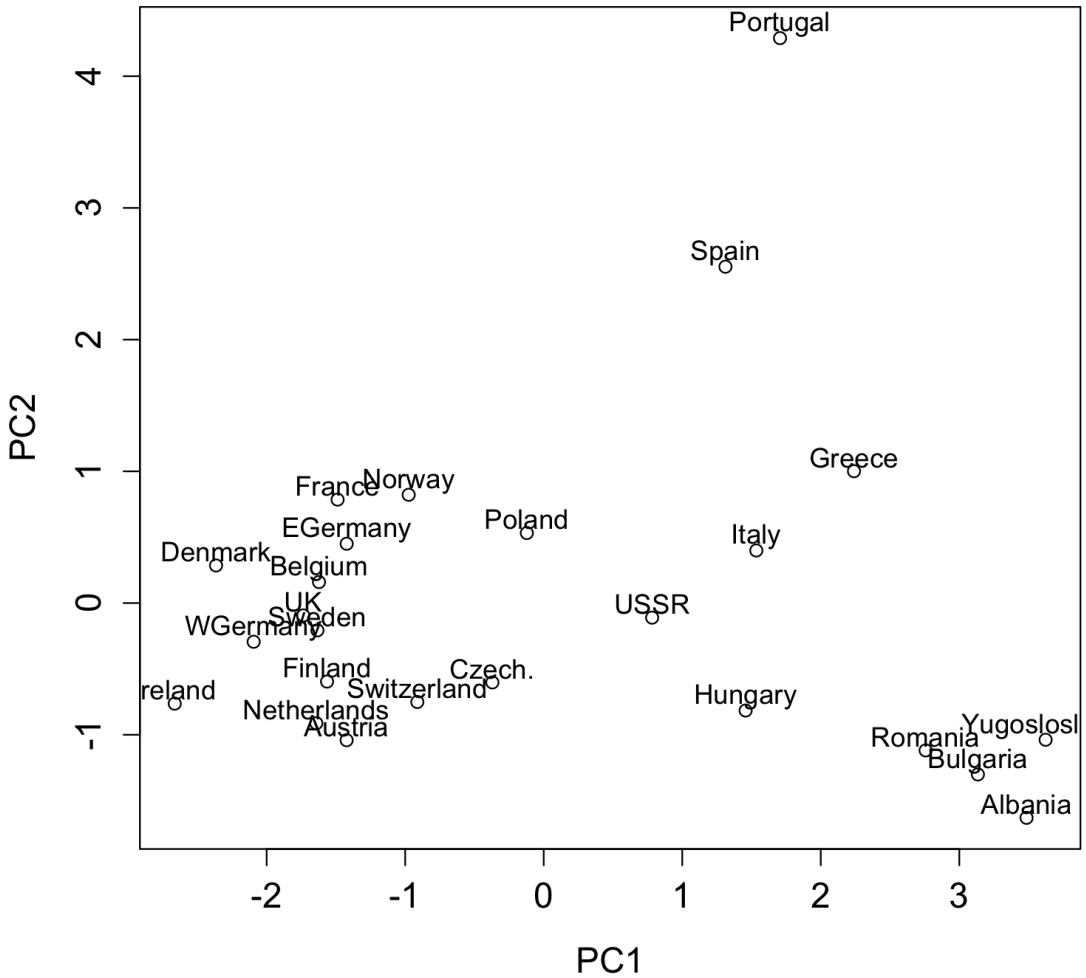
Plotting

To plot in R, we might first try plotting the principal compoents with the names of the countries.

```
> b <- prcomp(scale(x[,2:10]))
> plot(b$x[,1],b$x[,2],xlab="PC1",ylab="PC2",cex.lab=1.3,cex.a
> text(b$x[,1], b$x[,2]+.1, labels=x$country,cex=1)
```

Here I added 0.1 to the y-coordinate of the country name to avoid having the label right on top of the point, which makes the label and the point hard to read. Another approach is to just plot the label, and use `type=''n''` in the plot statement so that you initially generate an empty plot.

Cluster analysis



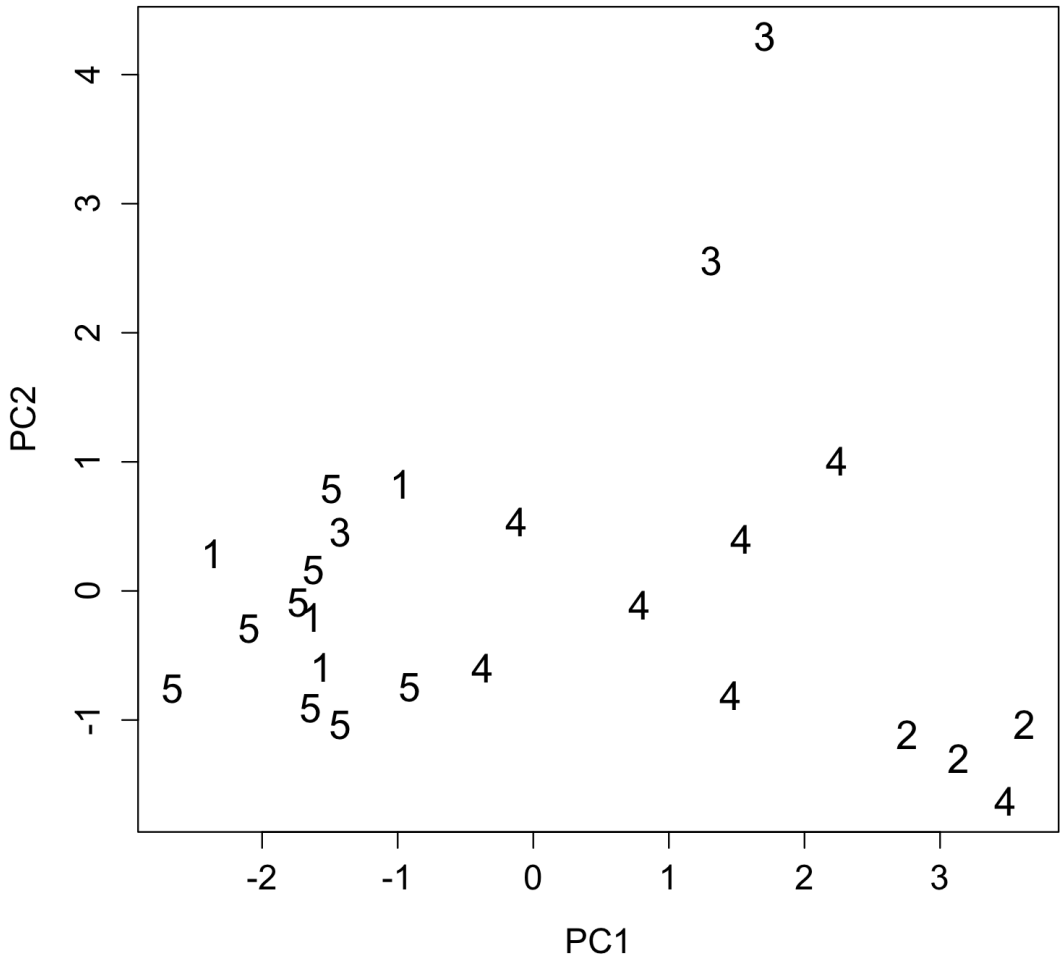
Plotting

To plot just the cluster number type

```
plot(b$x[,1],b$x[,2],xlab="PC1",ylab="PC2",cex.lab=1.3,cex.axi
pch=paste(cluster2$cluster))
```

Here the `pch` option gives the plotting symbol. If you use `pch=15` you get a square, for example. Instead of a plotting symbol code, you can put customized strings, which is what I did here. To convert the numeric cluster numbers to a string, I used `paste()` which is a string function that can sort of copy and paste strings together as objects.

Cluster analysis:



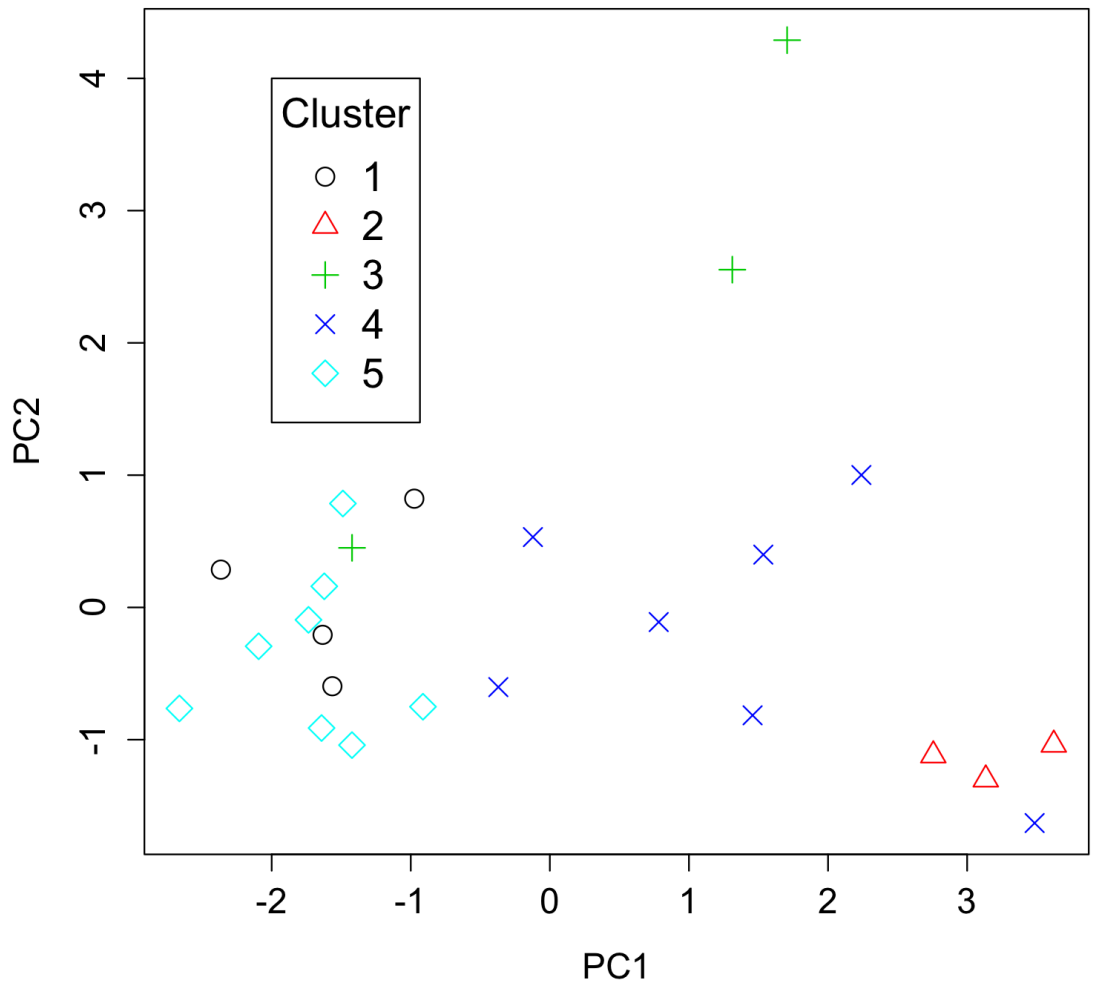
Cluster analysis

Another possibility...

```
> plot(b$x[,1],b$x[,2],xlab="PC1",ylab="PC2",cex.lab=1.3,
cex.axis=1.3,pch=cluster2$cluster,cex=1.5,
col=cluster2$cluster)
> legend(-2,4,legend=1:5,col=1:5,pch=1:5,cex=1.5,
title="Cluster")
```

Instead of picking default values, you can customize the color choice plot character choices as vectors such as `col=c(‘‘red","blue",‘‘pink",...)` With geographic data, you can get kind of intricate....

Cluster analysis:



Cluster analysis

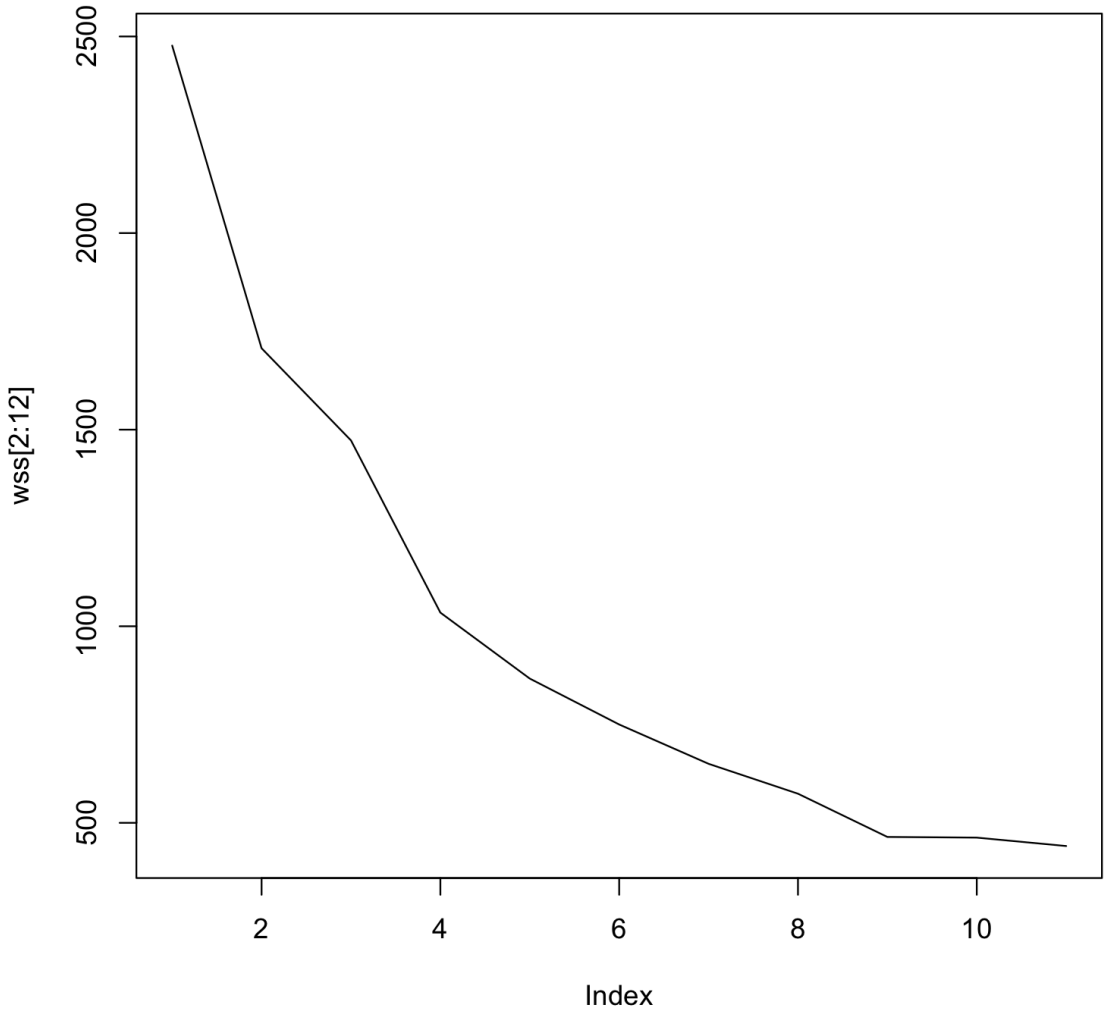
Another approach for determining the number of clusters is to perform the cluster analysis using different numbers of clusters and plot the within groups sums of squares against the number of clusters. If the number of clusters is too low, then sums of squared distances (to the centroid) will be high for some clusters. If the number of clusters is very high, then these sums of squares will be close to zero. A scree plot can be used, and the bend in the plot will suggest where there is little improvement in adding more clusters.

Cluster analysis

To illustrate this approach in R,

```
wss <- 1:12
for (i in 2:12) wss[i] <- sum(kmeans(x2,
   centers=i)$withinss)
plot(wss)
```

Cluster analysis:



Cluster analysis

There is a slight bend at 4 clusters and another at 9. There isn't an obvious elbow in this graph, though, so it isn't obvious how to decide how many clusters should be used.

Text data as an example

Suppose you have an unstructured text file, such as a plain text (or html code) for one of Shakespeare's plays. We want to turn this into data, specifically word frequencies.

Obviously, the data isn't arranged into nice rectangular arrays of columns with equal numbers of rows. Someting we can do is read in the data line by line. For html data, we also want to strip away the html tags.

Text data as an example

Here, I found a play of Shakespeare's from http://shakespeare.mit.edu.

I downloaded the webpage as html for the play *All's Well That Ends Well*, one of his comedies.
You can view the play scene by scene or as an entire play in one webpage, which is what I did, then downloaded the webpage (save as....), which gave me the html code. It looks like this:

Web page view

All's Well

Shakespeare homepage | A

ACT I

SCENE I. Rousillon. The COUNT's palace.

Enter BERTRAM, the COUNTESS of Rousillon, HELENA, and LAFEU, all in black

COUNTESS

In delivering my son from me, I bury a second husband.

BERTRAM

And I in going, madam, weep o'er my father's death
anew: but I must attend his majesty's command, to
whom I am now in ward, evermore in subjection.

LAFEU

You shall find of the king a husband, madam; you,
sir, a father: he that so generally is at all times
good must of necessity hold his virtue to you; whose

```
<A NAME=speech2><b>BERTRAM</b></a>
<blockquote>
<A NAME=1.1.2>And I in going, madam, weep o’er my father’s death</
<A NAME=1.1.3>anew: but I must attend his majesty’s command, to</A
<A NAME=1.1.4>whom I am now in ward, evermore in subjection.</A><b
</blockquote>

<A NAME=speech3><b>LAFEU</b></a>
<blockquote>
<A NAME=1.1.5>You shall find of the king a husband, madam; you,</A
<A NAME=1.1.6>sir, a father: he that so generally is at all times
<A NAME=1.1.7>good must of necessity hold his virtue to you; whose
<A NAME=1.1.8>worthiness would stir it up where it wanted rather</A
<A NAME=1.1.9>than lack it where there is such abundance.</A><br>
</blockquote>
```

We want to read in the html file but get rid of the html code. The html code is in angled brackets, so basically we want to get rid of the angled brackets and anything inside the angled brackets. Stuff that you want tends to be not within the brackets.

First, we can read in the html code line by line. This creates a data set where there is only one column, and each column is a wide string of text. This can be accomplished using `readLines()`. First

```
> x <- readLines("shake1.html")
> head(x)
[1] "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 Transitional//E
[2] " \"http://www.w3.org/TR/REC-html40/loose.dtd\">"
[3] " <html>"
[4] " <head>"
[5] " <title>All's Well That Ends Well: Entire Play"
[6] " </title>"
```

To remove html code, we'll use the following function which I found online at stackoverlow.com. Basically it removes characters that match the pattern of having balanced open and closed angle brackets with anything in between, and replaces it with nothing.

```
head(y)
[1] "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 Transitional//E
[2] " \"http://www.w3.org/TR/REC-html40/loose.dtd\">"
[3] " "
[4] " "
[5] " All's Well That Ends Well: Entire Play"
[6] " "
```

```
> source("shake.r")
> words1
z
     NA     the       i     and      to     you      of
   5726    1458    1384    1240    1028     964     918
     my    that      in      it      is     not     his
    756     652     600     560     554     500     466
   your    lord      me     for    have      be     but
    436     414     402     400     392     386     370
    her parolles    this    with    will      so bertram
    352     348     346     326     314     308     258
 helena    king    what   lafeu   shall   first      do
    246     244     238     232     230     228     208
     if     our     all     was countess    thou      by
    206     200     194     192     190     190     188
    are    good     she   which      we    well     thy
    186     180     178     174     172     172     168
   know    thee      am    from    more  second      at
    166     156     152     142     140     136     134
```