Chapter 6 Continued: Partitioning Methods

- *Partitioning methods* fix the number of clusters k and seek the best possible partition for that k.
- The goal is to choose the partition which gives the optimal value for some *clustering criterion*, or objective function.
- In reality, we cannot search all possible partitions to try to optimize the clustering criterion, but the algorithms are designed to search intelligently among the partitions.
- For a fixed k, partitioning methods are able to investigate far more possible partitions than a hierarchical method is.
- In practice, it is recommended to run a partitioning method for several choices of k and examine the resulting clusterings.

K-means Clustering

- The goal of K-means, the most well-known partitioning method, is to find the partition of n objects into k clusters that minimizes a *within-cluster sum of squares* criterion.
- In the traditional K-means approach, “closeness” to the cluster centers is defined in terms of squared Euclidean distance, defined by:

$$d_E^2(\mathbf{x}, \bar{\boldsymbol{x}}_{\boldsymbol{c}}) = (\mathbf{x} - \bar{\boldsymbol{x}}_{\boldsymbol{c}})'(\mathbf{x} - \bar{\boldsymbol{x}}_{\boldsymbol{c}}) = \sum_m (x_{im} - \bar{x}_{cm})^2,$$

  where $\mathbf{x} = (x_1, \ldots, x_q)'$ is any particular observation and $\bar{\boldsymbol{x}}_{\boldsymbol{c}}$ is the centroid (multivariate mean vector) for, say, cluster c.

K-means Clustering (Continued)

- The goal is to minimize the sum (over all objects within all clusters) of these squared Euclidean distances:
$$WSS = \sum_{c=1}^{k} \sum_{i \in c} d_E^2(\mathbf{x}_i, \bar{\boldsymbol{x}}_{\boldsymbol{c}})$$

- In practice, K-means will not generally achieve the global minimum of this criterion over the whole space of partitions.

- In fact, only under certain conditions will it achieve the local minimum (Selim and Ismail, 1984).

The K-means Algorithm

- The K-means algorithm (MacQueen, 1967) begins by randomly allocating the n objects into k clusters (or randomly specifying k centroids).
- One at a time, the algorithm moves each object to the cluster whose centroid is closest to it, using the measure of closeness $d_E^2(\mathbf{x}, \bar{\boldsymbol{x}}_{\boldsymbol{c}})$.
- When an object is moved, the centroids are immediately recalculated for the cluster gaining the object and the cluster losing it.
- The method repeatedly cycles though the objects until no reassignments of objects take place.
- The final clustering result will somewhat depend on the initial configuration of the objects.
- In practice, it is good to rerun the algorithm a few times (with different starting points) to make sure the result is stable.
- The `R` function `kmeans` performs K-means clustering.

Ward's Method

- The method of Ward (1963) is a hybrid of hierarchical clustering and K-means.
- It begins with n clusters and joins clusters together, one step at a time.
- At each step, the method searches over all possible ways to join a pair of clusters so that the K-means criterion WSS is minimized for that step.
- It begins with each object as its own cluster (so that $WSS = 0$) and concludes with all objects in one cluster.
- The `R` function `hclust` performs Ward's method if the option `method = 'ward'` is specified.

K-medoids Clustering

- The K-medoids algorithm (Kaufman and Rousseeuw, 1987) is a robust alternative K-means.
- It attempts to minimize the criterion

$$Crit_{Med} = \sum_{c=1}^{k} \sum_{i \in c} d(\mathbf{x}_i, \mathbf{m}_c)$$

  where $\mathbf{m}_c$ is a *medoid*, or “most representative object,” for cluster c.
- The algorithm begins (in the “build step”) by selecting k such representative objects.
- It proceeds by assigning each object to the cluster with the closest medoid.
- Then (in the “swap step”), if swapping any non-medoid object with a medoid results in a decrease in the criterion $Crit_{Med}$, the swap is made.
- The algorithm stops when no swap can decrease $Crit_{Med}$.

K-medoids Clustering (Continued)

- Like K-means, the K-medoids algorithm does not globally minimize its criterion in general.
- The `R` function `pam` in the `cluster` package performs K-medoids clustering.
- An advantage of K-medoids is that (unlike `kmeans`) the function can accept a dissimilarity matrix, as well as a raw data matrix.
- This is because the criterion to be minimized is a direct sum of pairwise dissimilarities between objects.
- The `pam` function also produces tools called the *silhouette plot* and *average silhouette width* to guide the choice of k (see examples).

Specialized Partitioning Methods

- The K-medoids algorithm is computationally infeasible for very large n ($n > 5000$ or so).
- The R function `clara` (**C**lustering **Lar**ge **A**pplications) is designed as a large-sample version of `pam`.
- With `clara`, the medoids are calculated using randomly selected subsets of the data.
- The build-step and swap-step are carried out on the subsets rather than the entire data set.
- *Fuzzy Cluster Analysis* (implemented by `fanny` in R) assumes each object can have *partial* membership in several clusters.
- Rather than assigning each object to only one cluster, it assigns a “membership coefficient” for each cluster to an object that reflects the “degree of membership” of the object to that cluster.

Objective Methods to Determine the Number of Clusters k

- At some point we need to choose a single value of k to get a clustering solution.
- A variety of criteria have been proposed to pick the best value of k.
- The *average silhouette width* is based on the difference between the average dissimilarity of objects to other objects in their own cluster and the average dissimilarity of objects to the objects in a "neighbor cluster."
- The larger the average silhouette width, the better the clustering of the objects.
- We could calculate the average silhouette width for clusterings based on several values of k and choose the k with the largest average silhouette width.
- The *silhouette* function in the *cluster* package of R gives the average silhouette width for any clustering result and distance matrix.

Other Methods to Determine the Number of Clusters

- Another criterion for choosing k is the *Dunn index*, which is implemented with the `dunn` function in the `clValid` package.

- Especially with K-means clustering, a common way to choose k is to plot the within-cluster sum-of-squares WSS for the K-means partitions for a variety of choices of k.

- As k increases, the corresponding WSS will decrease, and at some point will level off.

- The “best” choice of k usually occurs near the “elbow” in this plot.

Model-based Clustering

- Neither hierarchical nor partitioning methods assume a specific statistical model for the data.
- They are strictly exploratory tools, and no formal inference about a wider population is possible.
- *Model-based clustering* assumes that the population generating the data consists of k subpopulations, which correspond to the k clusters we seek.
- Therefore, the distribution for the data is assumed to be composed of k densities.
- This idea was originally proposed by Scott and Symons (1971) but fully developed in recent years by Banfield and Raftery (1993) and Fraley and Raftery (2002).

Clustering Model Setup

- Let $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_n]'$ be a vector of cluster labels, such that $\gamma_i = j$ if observation $\mathbf{x}_i$ is from the j-th subpopulation.

- Suppose the subpopulation densities are denoted by $f_j(\mathbf{x}; \boldsymbol{\theta_j})$, where $\boldsymbol{\theta_j}$ contains the set of unknown parameters for the j-th density.

- Then the likelihood, given the observed data, is:

$$L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k, \boldsymbol{\gamma} | \mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{i=1}^{n} f_{\gamma_i}(\mathbf{x}_i; \boldsymbol{\theta}_{\gamma_i}).$$

- Fitting the model amounts to choosing $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k, \boldsymbol{\gamma}$ to maximize this likelihood.

- The estimated $\boldsymbol{\gamma}$ is the “clustering vector” that defines which cluster each object is assigned to.

The Multivariate Normality Assumption

- We may assume that each subpopulation ($j = 1, \ldots, k$) follows a multivariate normal density having mean vectors $\boldsymbol{\mu}_j$ and covariance matrices $\boldsymbol{\Sigma}_j$, for $j = 1, \ldots, k$, as its parameters.
- Then the likelihood becomes

$$L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k, \boldsymbol{\gamma}) \propto \prod_{j=1}^{k} \prod_{i \in f_j} |\boldsymbol{\Sigma}_j|^{1/2} \exp\Big[-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_j)' \boldsymbol{\Sigma}_j^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j)\Big].$$

- The MLE of $\boldsymbol{\mu}_j$ is $\bar{\boldsymbol{x}}_j$, the sample mean vector for the observations in subpopulation j.

The Multivariate Normality Assumption (continued)

- Replacing $\boldsymbol{\mu}_j$ with $\bar{\boldsymbol{x}}_j$, the log-likelihood function is a constant plus

$$-\frac{1}{2}\sum_{j=1}^{k} trace(\mathbf{W}_j\mathbf{\Sigma}_j^{-1} + n\ln|\mathbf{\Sigma}_j|),$$

  where $\mathbf{W}_j$ is a matrix containing the sums of squares and cross products of variables for observations in subpopulation j.

- We can assume a certain structure for the covariance matrices $\mathbf{\Sigma}_j$ ($j = 1, \ldots, k$) and then determine computationally the value of $\boldsymbol{\gamma}$ that maximizes this (log) likelihood.

Possible Covariance Structures

- We could consider a few possible covariance structures.
- A simple (maybe unrealistic!) assumption is that each subpopulation has the same covariance structure *and* that all the $\mathbf{\Sigma}_j = \sigma^2\mathbf{I}$.
- In this case, $\boldsymbol{\gamma}$ is chosen so that the total within-group sum-of-squares $trace(\sum_{j=1}^{k} \mathbf{W}_j)$ is minimized.
- This tends to produce clusters that are spherical and roughly of equal size.
- A slightly more complicated assumption is that each subpopulation has the same covariance structure, i.e., $\mathbf{\Sigma}_j = \mathbf{\Sigma}$ for all $j = 1, \ldots, k$.
- This tends to produce clusters that are elliptical with roughly the same directional slope.

Other Covariance Structures

- An extremely unrestrictive assumption is that each subpopulation may have a completely different covariance structure, $\Sigma_j, j = 1, \ldots, k$.
- This may produce clusters that are different in size, shape, and orientation.
- We might consider assumptions that are less restrictive than the equal-covariances assumption yet more parsimonious than the unstructured-covariances assumption.
- The covariance structure we assume leads to a clustering solution in which the sizes, shapes, and orientations of the clusters might be the same or different.
- In practice, the `R` function `Mclust` in the `mclust` package considers many such models, letting the covariance assumptions *and* the number of clusters k vary.
- Usually the *Bayesian information Criterion* (BIC) is used to choose the best of all these competing models and thus determine the model-based clustering result.

Clustering Binary Data

- When the q variables measured on each observation are *binary* (e.g., representing the presence or absence of some characteristic), the objects may still be clustered based on a distance measure.

- Suppose, for each individual ($i = 1, \ldots, n$), we let the binary variable X_{ij} (for $j = 1, \ldots, q$) take the value 0 or 1.

- Then two individuals have a “match” on a binary variable if both individuals have the same value for that variable (either both 0 or both 1).

- Otherwise, the two individuals are said to have a “mismatch” on the binary variable.

- Calculating squared Euclidean distances $\sum_{j=1}^{q}(X_{ij} - X_{i'j})^2$ between each pair of rows of this sort of data matrix of 0’s and 1’s amounts to counting the total number of mismatches for each pair of objects.

- Once we calculate the distances, we can input them into a standard clustering algorithm like K-medoids or a hierarchical method.

Meaning of Matches and Mismatches for Binary Data

- Using squared Euclidean distance essentially treats 0-0 matches and 1-1 matches as equally important. Is this appropriate?

- It depends on the situation: If the binary variable is measuring a very rare (or very common) characteristic, then a 1-1 match may be more meaningful than a 0-0 match (or vice versa).

- If $X_i = 1$ if an individual is a strict vegan and 0 otherwise, then a 1-1 match might indicate two similar individuals, but a 0-0 match would be less informative.

- If $X_i = 1$ if an individual knows how to read and 0 otherwise, then a 0-0 match might indicate two similar individuals, but a 1-1 match would be less informative.

Other Measures of Distance for Binary Data

- Define a 2×2 table counting the matches (a = total 0-0 matches, d = total 1-1 matches) and mismatches (b = total 0-1 mismatches, c = total 1-0 mismatches) for a pair of objects:

	$\mathbf{Y}_{i'}$		
$\mathbf{Y}_i$	0	1	Totals
0	a	b	$a+b$
1	c	d	$c+d$
Totals	$a+c$	$b+d$	$q = a+b+c+d$

- Defining the distance between the two objects to be $\frac{b+c}{q}$ gives equal weights to 0-0 matches and 1-1 matches.

Other Measures of Distance for Binary Data (Continued)

- Defining the distance between the two objects to be $\frac{b+c}{b+c+d}$ ignores 0-0 matches, treating them as irrelevant (vegan example?).
- Defining the distance between the two objects to be $\frac{b+c}{a+b+c}$ ignores 1-1 matches, treating them as irrelevant (reading example?).
- Several other distances measures based on a, b, c, d are possible (see Johnson and Wichern, p. 674).

Gower Dissimilarities for Clustering Mixed Data

- Sometimes we have data that are **mixed data** having different variable types.
- For example, perhaps some of the variables are numerical, others are binary or nominal, and maybe still others are ordinal (categorical with *ordered* categories).
- Gower (1971) developed a dissimilarity measure for mixed data that combine contributions to the dissimilarity from each variable.
- For any pair of individuals, we have the following rules for calculating the Gower dissimilarity between those two individuals:

Calculation of Gower Dissimilarities

- For a nominal or binary variable, the contribution is 1 if the two individuals do not have matching categories on that variable and 0 if the individuals match on that variable.

- For a numerical variable, the contribution is the absolute difference in the variable's values for the two observations, divided by the total range ($max - min$) for that variable in the data set.

- For an ordinal variable, the categories are numerically labeled $1, 2, \ldots$ and then the contribution is calculated the same way as for numerical variables.

- The overall Gower dissimilarity is the mean (possibly weighted, if desired) of the contributions of each of the variables.

Clustering Mixed Data

- The Gower dissimilarities can be calculated using the `daisy` function in R.
- The nominal variables should be saved as `factor` columns and the ordinal variables should be saved as `ordered` columns in R.
- Once we calculate the distances, we can input them into a standard clustering algorithm like K-medoids or a hierarchical method.
- This method is implemented in R on the heart disease data set.