Chapter 16
Cluster Analysis

Abstract Cluster analysis segments a customer database so that customers within segments are similar, and collectively different from customers in other segments. Similarity is in terms of the "clustering variables," which may be psychographics, demographics, or transaction measures such as RFM. The clusters often have rich interpretations with strong implications for which customers should be targeted with a particular offer or marketed to in a certain way. This chapter discusses the details of cluster analysis, including measures of similarity, the major cluster methods, and how to decide upon the number of clusters and their interpretation.

16.1 Introduction

Marketers have used cluster analysis to segment the market for a long time. It allows marketers to group their customers into several homogeneous clusters such that customers in the same cluster are similar in terms of their demographic and behavioral characteristics while customers across different clusters are different.

Cluster analysis is often confused with classification tasks. Both techniques segment subjects into several groups. But they are different in their goals. In classification we know the number of groups and the segment membership of each subject. The goal is to predict the segment membership of a new subject once a classification model is estimated. On the other hand, we do not have any predefined segments in clustering. Its objective is simply to group subjects into several homogeneous clusters. Using the terminology of machine learning, classification is a typical task of *directed* knowledge discovery while clustering is an example of *undirected* knowledge discovery.

In a classification task we have a dependent variable. For example, a bank wants to evaluate the credit risks of card applicants in order to decide whom to issue their credit cards. In order to develop the forecasting model, we first analyze the existing customer data for which each customer is already known to be either a defaulter or not. An appropriate model such as a discriminant

analysis or logistic regression is applied to the data, taking customer's credit status (i.e., defaulter or not) as the dependent variable and his/her demographic and behavioral characteristics as the independent variables. Upon estimating model parameters, we can classify new card applicants into either defaulters or non-defaulters by obtaining information on their demographic and behavioral characteristics.

On the other hand, there are no pre-classified groups in clustering. We do not make any distinction between dependent and independent variables. Our goal is to group subjects into an arbitrary number of homogeneous segments in terms of their characteristics or variables. Because of the subjectivity involved in determining the number of clusters and selecting the clustering algorithm with a similarity measure, clustering is often considered as an exploratory data analytic technique.

Because of its exploratory nature, there is always the question of whether the cluster analysis has produced the "correct" segmentation scheme. A practical answer is that there are multiple ways to segment the market. The question is whether a given segmentation scheme is managerially useful. Certainly the interpretability of the segmentation scheme makes it more useful. But in a database marketing context, a crucial consideration is whether a given segmentation scheme can be used for targeting. We discuss this issue in Sect. 16.3.2. It is noteworthy that most of the clustering techniques are *algorithms* that try to group customers into groups so that customers within groups are similar to each other and collectively different from clusters in other groups. There is no underlying statistical model or theory that hypothesizes a true underlying grouping that needs to be discovered. This also clouds the issue of whether the obtained solution is "correct." With most clustering methods, sources of error and variation are not formally considered. Hence, clustering results will be sensitive to outliers or noise points. The exception to this is probabilistic clustering, which is described in Sect. 16.2.3.3.

16.2 The Clustering Process

Conducting a cluster analysis consists of several steps:

- Select variables on which to cluster
- Select a similarity measure and scale the variables
- Select a clustering method
- Determine the number of clusters
- Conduct the cluster analysis, interpret the results, and apply them

There are several methods that can be used for each of these steps, especially involving similarity measures and clustering methods. Choice of these methods is often subjective. For example, an analyst who has found a certain similarity measure, scaling procedure, and clustering method to be successful

in the past will tend to use that approach again. It is often a good idea to try a few different approaches in one application, to see which yields the best results. The reader should recognize, however, that there are many methodological choices to be made with a cluster analysis, and no set of choices that all database marketers agree on as optimal. As mentioned above, often the choice of method hinges on the question – are the results useful?

16.2.1 Selecting Clustering Variables

The first question the analyst faces is what variables should form the basis for the clustering. That is, on which set of variables do we want to form homogeneous groups of customers? In typical applications, there are several variables available. These include:

- *Benefits sought:* These are measures of what benefits in a product or service customers deem to be important. They can be collected by asking customers directly in a survey (e.g., "How important is price to you?"), or via a conjoint analysis or other indirect measure of benefits sought.
- *Psychographics:* Attitudes and behaviors relevant to a particular product category. For example, if the product category is consumer electronics, customer self-report of whether they see themselves as innovators, opinion leaders, or "gadget freaks," are psychographics. Ownership and usage of various electronics products are also psychographics.
- *Demographics:* These include age, income, region, employment, etc.
- *Geo-demographics:* These include variables inferred by where the customer lives. For example, the US census makes publicly available average income, age, home ownership, etc., for fairly small geographic units such as "census blocks."
- *Behavior:* These include recency, frequency, and monetary value behaviors measured from the company's customer file. These can be measured by department (e.g., frequency of purchasing men's clothes, women's clothes, children's clothes, accessories). Behavior can also include sales channels through which the customer buys.
- *Competitive measures:* These include share of wallet, competitor preferences, etc.
- *Customer value:* These might include responsiveness to marketing, lifetime value, customer lifetime duration, customer risk, etc.

The choice of variables on which to cluster might depend on the application. For example, for a new product application, clustering on benefits sought might be most useful. For a cross-selling application, clustering on channel usage or RFM for various departments might be useful. For a customer tier program, clustering on customer value might be useful.

One strategy is to use one of the above sets of variables for the clustering variables, and then the other variables for the "discrimination variables."

These variables can aid in interpretation and in classifying new customers to clusters. The reason not to mix types of variables as clustering variables is the interpretation may become difficult, and also, different types of variables are most likely measured on different scales, which brings up a scaling problem (discussed in Sect. 16.2.2.3). In Sect. 16.3, we will show a hypothetical example to illustrate the roles of clustering and discrimination variables.

A practical question is, given we know the type of variables on which we want to cluster, how many variables should we use? This issue, the problem called "variable selection," has been well studied in classical regression. Omitting relevant variables results in biased parameter estimates while including irrelevant variables leads to overfitting the model (Greene 1997). However, the selection of clustering variables has rarely been studied. Technically speaking, one can cluster on any number of variables. However, often it is advantageous to keep the number of variables relatively small (say 5–15) to aid in interpretation.[1]

16.2.2 Similarity Measures

The goal of clustering is to group customers into several homogeneous clusters. Customers in the same cluster are supposed to be similar while subjects across different clusters are dissimilar. We need to be more precise. How do you define the similarity between customers?

Clustering begins with selecting a similarity measure and a set of variables regarding which the similarity is calculated. Care should be taken in selecting a similarity measure since different measures often lead to different clustering results. It is recommended to apply several different similarity measures and check the stability of clustering results. Unstable clustering results may imply that the clusters found do not provide a meaningful or usable segmentation scheme.

16.2.2.1 Distance-Type Similarity Measures

Broadly speaking, there are two types of similarity measures: distance type and matching type. Distance types of similarity measures are more appropriate when variables are measured on a common metric, so that the similarity between two customers can be measured as their distance in a metric space. For example, three customers are plotted in Fig. 16.1. Each customer is represented as a point in two-dimensional space. Expressed in matrix terminology,

[1] It is also recommended to study correlation structure among candidate clustering variables. Including highly correlated variables in a cluster analysis has the effect of weighting up an underlying dimension. Hence, if two variables are highly correlated, either one of them should be deleted.

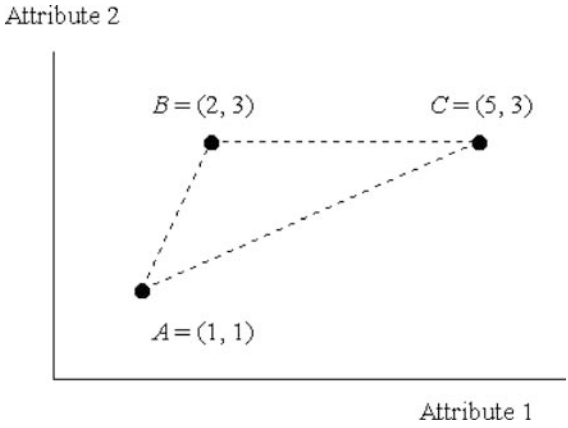

Fig. 16.1 Defining similarity*.
*Points A, B, and C represent customers, in particular their values for Attributes 1 and 2.

we have the 3×2 data matrix where there are three customers and two attributes describing each customer. The distance between Customers A and B is the shortest among all three pairwise inter-point distances. Hence, we may say that Customer A is similar to Customer B in terms of their attributes (e.g., income and age). We are implicitly using the distance between points or subjects as the similarity measure.

The most popular distance similarity measure is the Euclidean distance between points (Hair et al. 1992). The Euclidean distance between two p-dimensional subjects $\mathbf{x} = [x_1, \ldots, x_p]'$ and $\mathbf{y} = [y_1, \ldots, y_p]'$ is defined as

$$d(\mathbf{x}, \mathbf{y}) = [(x_1 - y_1)^2 + \ldots + (x_p - y_p)^2]^{1/2} = [(\mathbf{x} - \mathbf{y})'(\mathbf{x} - \mathbf{y})]^{1/2} \quad (16.1)$$

Each subject is represented in two-dimensional space ($p = 2$) in Fig. 16.1. Hence, the Euclidean distance between Customers A and B is $\sqrt{5} = [(1 - 2)^2 + (1 - 3)^2]^{1/2}$. Similarly, the distance between Customers A and C is $2\sqrt{5}$ and the distance between B and C is 3. The distance between Customers A and B is the shortest. So A and B are said to be the most similar.

The Minkowski metric generalizes the concept of the Euclidean distance. The Minkowski distance between two p-dimensional subjects is given by

$$d(\mathbf{x}, \mathbf{y}) = \left[\sum\nolimits_{i=1}^{p} |x_i - y_i|^m\right]^{1/m} \quad (16.2)$$

where it becomes the Euclidean distance when m is equal to two. On the other hand, the Minkowski metric with $m = 1$ is called the city-block distance. For example, the city-block distance between Customers A and B in Fig. 16.1 is the sum of the horizontal and the vertical distance, which in this case would be 3 ($= 1 + 2$).

16.2.2.2 Matching-Type Similarity Measures

Geometric distance is not meaningful for categorical variables. Instead, we can use the degree of matching to measure the similarity between customers when they are represented by a set of categorical characteristics. For example, two customers are treated as similar if both of them are students.

More formally, we measure the similarity between two customers as the degree of matching, specifically, the ratio of the number of matching attributes to the total number of attributes considered. For a simple example, suppose that two customers are represented by the presence (coded as 1) or absence (coded as 0) of five attributes. If Customer A's attributes can be represented as $\mathbf{x} = [0, 0, 1, 1, 1]'$ and Customer B's as $\mathbf{y} = [1, 0, 1, 1, 0]'$, then there are three matches out of five attribute comparisons and, so the similarity between Customers A and B is 0.6.

There are some variants of matching-type similarity measures such as assigning differential weighting on matching from mismatching cases. For example, when there are an unusually large number of zeros in the data, we assign a large weight on the matches of ones. Similarly, when there are a lot of ones, a large weight is assigned on the matches of zeros. More sophisticated similarity measures are discussed in Chapter 14.

One question is what to do if some of the clustering variables are metric, in which case a distance type similarity measure would be appropriate, and some are categorical, in which case a matching metric may be appropriate. There is no clear answer here. One practical solution is to code the categorical variables as 0–1 and use them together with the metric variables and a distance-type similarity measure. This will "work" in that the distance measure can be calculated, but we would be mixing variables measured in different units and this raises a scaling problem (discussed in the next section). Another possibility is to use only one type of clustering variable (e.g., benefits sought), that are all scaled the same way (in this case, with a common metric), so the problem of mixing metric and categorical variables doesn't arise.

16.2.2.3 Scaling and Weighting

Even metric variables are frequently measured in different units, and this can distort the cluster analysis results. For an example, if we multiply one variable by a large number, say 100, then the similarity measure will be dominated by the value of the variable. Hence, it is advisable to rescale the variables such that a percentage change of one variable is not more significant than the same percentage change of another variable in similarity calculation.

The scaling problem essentially comes from the different variances measured in each variable. In order to avoid the scaling problem due to different units of measurement, one approach is to make the variances of all variables to be the same. There are two remedies commonly used in

practice. The first approach is to rescale all the variables to range from zero to one. If X_i is the original variable and X_i^* is the scaled variable, $X_i^* = (X_i - X_{\min})/(X_{\max} - X_{\min})$ where $X_{\min}$ is the minimum and $X_{\max}$ is the maximum observed value for the original variable. Alternatively, we can standardize all the variables so that the rescaled variables have the common means of zero and the common variances of one. If X_i is the original variable, the rescaled variable $Z_i = (X_i - \bar{X})/\sigma_X$ where $\bar{X}$ is the mean of the original variable and σ_X is the corresponding standard deviation.[2]

While the above at first seems to solve the scaling problem, it may limit the results in important ways. For example, assume we have measured benefits sought on a 5-point scale. Further, assume that Price Importance has a mean of 3.0 and a standard deviation of 1.0. That suggests that customers vary considerably in the importance they attach to price. Now assume that Quality Importance also has a mean of 3.0 but a standard deviation of 0.5. This means that customers do not vary so much in the importance they attach to quality. If we standardize these variables, they will both have a variance of one, but we are losing important information, that in fact customers vary a lot on the importance of price, and not so much on the importance of quality. This example assumes the same units of measurement, but the same problem can arise when standardizing variables measured in different units. Let's say that we also include a measure of Service Importance based on the number of calls the customer has made to the call center. Let's say the mean of this variable is 10 with a standard deviation of 10. This means there is wide variation in this variable, and the variable is skewed to the right (since the number of calls cannot be less than zero). Standardizing this variable puts it on an equal footing with Quality Importance because both variables have the same variance. This obscures the fact that customers really all feel roughly the same about quality, but vary a lot in the importance they attach to service. This "true" variation on service importance could be an important factor in defining segments that we will miss out by equalizing the variance of all variables.

There are no easy answers to the re-scaling issue. One possibility is only to use as clustering variables that are scaled the same way, and not standardize (e.g., use measures of benefits sought measured on a 5-point scale). However, this is not always possible. For example, demographics are naturally measured on different scales (e.g., age and income). It is for this reason that cluster analysis truly is an exploratory technique. One can try various rescaling procedures, as well as distance measures, examine how it changes the nature and interpretation of the clusters, then make a decision of which clustering solution to use.

---

[2] Variables in database marketing (e.g., monetary value) are often highly skewed. And right skewed variables tend to produce many tiny clusters and a couple of big ones, which is not desirable. With skewed variables, it is recommended to take logs and then standardize the variables.

Another way to deal with the scaling issue, as well as take into account the managerial importance of different clustering variables, is to consider weighting each of the clustering variables. For example, if we believe that the income variable is much more important than the age of household head in deciding the similarities among households, it is reasonable to assign a large weight on the income variable by multiplying it by a large number (at least larger than one). However, in contrast to the objectivity of the scaling solution, finding the appropriate weights is a rather subjective task. We suggest that the weighting be considered only when you have *a priori* reasons based on previous studies or your own experience. In addition, if weighting is to be used, we suggest that a number of different weights (e.g., various multiples on the income variable) be tried and the corresponding clustering outcomes be compared in terms of interpretation and managerial relevance.

16.2.3 Clustering Methods

The goal of clustering is to group subjects into an arbitrary number of segments such that customers in the same segment are similar in their characteristics while customers across different segments are dissimilar. However, it is not a simple task to find the optimal clustering solution. For example, there exist millions of ways to cluster only 20 customers. There is one way to form one cluster with 20 customers. There are 524,287 ways to group 20 customers into two clusters! There are much more ways for three clusters, and so on.[3] A theoretical method of finding the optimal clustering solution may be to enumerate all possible clustering solutions and select the best one. However, it is practically impossible to list all possible solutions. Hence, researchers have developed heuristic methodologies that may not necessarily find the optimal solution (if indeed there is a single conclusion what could call optimal) but acceptable ones.

There are a number of algorithms available for clustering. As shown in Fig. 16.2, they are broadly classified into two groups: hierarchical and nonhierarchical clustering technique.

Hierarchical clustering develops a tree-like structure either by serially merging clusters (agglomerative method) or by successively dividing clusters (the divisive method). Given n customers, agglomerative hierarchical starts with the n cluster solution, where each customer is his or her own cluster. It then produces an $(n-1)$ cluster solution by combining the two most similar customers, an $(n-2)$ cluster solution, and so on. This merging is continued until all customers are classified into a single cluster. Divisive hierarchical clustering proceeds in the opposite direction. It starts with

[3] The number of ways of sorting n subjects into k nonempty groups can be computed by a Stirling number of the second kind that is given by $(1/k!)\sum_{x=0}^{k}(-1)^{k-x}{}_kC_x x^n$.

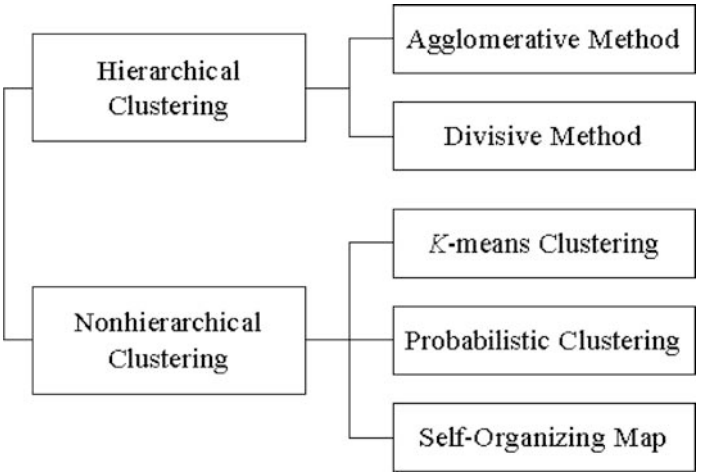

Fig. 16.2 Types of clustering methods.

the one-cluster solution consisting of all n customers, and then divides n customers into two clusters such that customers in one cluster are dissimilar to the customers in the other cluster. This division is continued until each customer becomes one cluster. We will cover agglomerative clustering in more detail in the next subsection. However, divisive method is not covered here since the algorithm is very similar to the decision trees described in Chapter 17.

More recently developed, nonhierarchical clustering techniques find the cluster solution given the number of clusters pre-specified by the user. Even though there are several methods that can be classified as nonhierarchical clustering, we will cover K-means clustering, probabilistic clustering and self-organizing map. K-means clustering starts with randomly assigning n customers into K clusters and successively improves the partitions by changing the cluster membership of each customer. Probabilistic clustering can be considered as a probabilistic version of K-means technique that overcomes its several limitations, at the cost of making various assumptions. Finally, self-organizing maps (SOM) is a special type of neural network model that can be useful in detecting clusters. It has several features that are similar to a typical neural network model and some other features that are similar to K-means clustering.

16.2.3.1 Agglomerative Clustering

Marketers have used agglomerative clustering algorithm for a long time in segmenting their customers. We describe the common algorithmic structure of agglomerative clustering even though there are a number of variants such

Table 16.1 Example of the agglomerative clustering algorithm

(a) 5-cluster solution

	1	2	3	4	5
1	0				
2	2	0			
3	5	6	0		
4	6	7	3	0	
5	10	11	8	7	0

(b) 4-cluster solution

	(12)	3	4	5
(12)	0			
3	5	0		
4	6	3	0	
5	10	8	7	0

(c) 3-cluster solution

	(12)	(34)	4
(12)	0		
(34)	5	0	
4	10	7	0

(d) 2-cluster solution

	(1234)	5
(1234)	0	
5	7	0

as linkage methods, variance methods and so on (Johnson and Wichern 1982). Agglomerative clustering starts with n clusters given n customers, that is, it considers each customer its own cluster, and then successively merges customers in terms of their similarities until all subjects are classified into a single cluster. The algorithm can best be understood by an example.

Suppose we have five customers and calculate pairwise similarities using one of the similarity definitions discussed in the previous section. The resulting similarity matrix is given in Table 16.1(**a**). Only the lower triangular of the matrix is shown because of its symmetric property (the similarity of Customer A to Customer B is the same thing as the similarity of Customer B to A). Table 16.1(**a**) shows that Customers 1 and 2 are the most similar (or the nearest) among all pairs. Hence, we merge these two customers into a cluster named (12). Now we have four clusters that consist of the cluster (12), the cluster (3), the cluster (4), and the cluster (5).

We update the similarity or distance matrix since a new cluster, cluster (12), has been created. The distances among the cluster (3), the cluster (4) and the cluster (5) stay the same. However, the distances between the new cluster (12) and the rest of the clusters should be recalculated. There are three different ways to define the distance between clusters: a simple linkage, the complete linkage and the average linkage. The simple linkage defines the distance as the shortest one among cluster members. For example, the cluster (12) has two customers. The distance between Customer 1 and the cluster (3) is 5 and the distance between Customer 2 and the cluster (3) is 6. Hence, the simple linkage distance between the cluster (12) and the cluster (3) is 5. The distances between the cluster (12) and the rest of the clusters can be similarly calculated. Table 16.1(**b**) shows the distance matrix of 4-cluster solution using a simple linkage method. On the other hand, a complete linkage selects the farthest distance. Hence, the distance between the cluster (12) and the cluster (3) is 6 by the complete linkage. Finally, an average linkage takes the average distance. If we use the average linkage, the distance between the cluster (12) and the cluster (3) is 5.5.

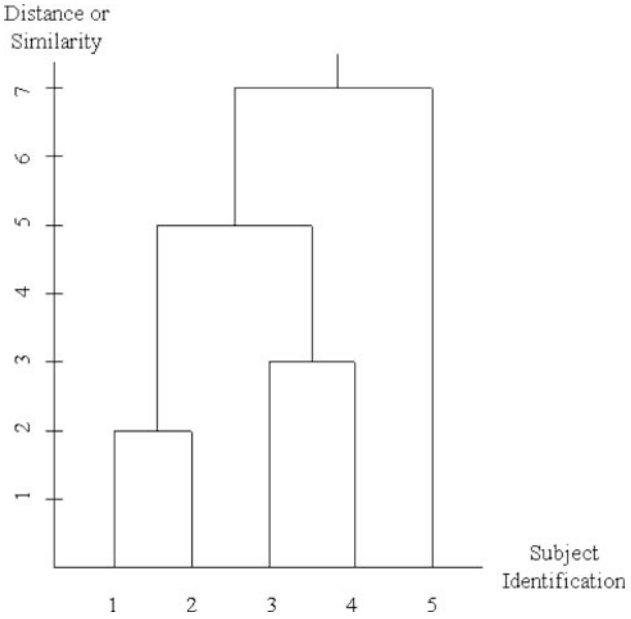

Fig. 16.3 Dendrogram for agglomerative clustering example.

From now on we limit our discussion to a simple linkage method. Table 16.1(**b**) suggests that the cluster (3) and the cluster (4) should be merged next since the distance between these two is the shortest among four clusters. Merging these two, we form the cluster (34), and now have three clusters: cluster (12), cluster (34) and cluster (5). Table 16.1(**c**) shows the updated distance matrix for the 3-cluster solution that indicates the next merging is between cluster (12) and cluster (34). The 2-cluster solution consists of cluster (1234) and cluster (5) and its updated distance matrix is shown in Table 16.1(**d**). Finally, merging these two clusters results in a single cluster including all five customers.

We often summarize the iterations from n-cluster solution to a single cluster by a tree-like diagram called dendrogram. The Greek word "dendron" means a tree. Figure 16.3 shows the dendrogram for the example with five customers. The x-axis in the dendrogram represents the identification of each customer while the y-axis shows the overall picture of cluster formation and distance information when merging. The dendrogram is often used as a tool for choosing the number of clusters relevant to the user. The distances at which clusters are combined can be calculated from the dendrogram. We choose the number of clusters at which the distance increase is suddenly large. However, as the number of customers becomes larger, the dendrogram becomes quite messy. Moreover, it is rather subjective to determine the number of clusters using the dendrogram. We discuss more formal procedures in determining the number of clusters in Sect. 16.2.4.

Agglomerative clustering is simple to apply and easy to understand. However, it has a critical limitation because of its tree-like structure. Also, once customers are incorrectly clustered in the earlier stage of the clustering process, the solution will be seriously biased, the error propagates through the rest of the tree, and re-clustering is not allowed.

16.2.3.2 *K*-Means Clustering

The K-means may be the most popular clustering method among data miners. Given the number of clusters k specified by the user, the algorithm starts with randomly choosing k points among n customers that become the initial cluster centers. Each of the remaining customers is assigned to the one of k cluster centers according to the Euclidean distance. Once all customers are grouped into k clusters, new cluster "centers" are calculated, typically as the mean values for each of the clustering variables for each cluster, and each subject is reassigned according to the distances to these new cluster centers. We stop iterations when no more reassignments occur.

We study the K-means algorithm in more detail by a simple example. Suppose we have four customers whose characteristics are represented in two attributes, say income and age. Table 16.2(**a**) shows the attribute values for each of these four customers. If we want to create two clusters (or $k = 2$), we randomly select two customers and their attribute values become the initial cluster centers. Suppose that Customers B and C be selected. Customer A is clustered into Customer C because its distance to Customer B is 10 and its distance to Customer C is $\sqrt{68} = 8.2$. Similarly, Customer D is merged into Customer B because its distance to Customer B is shorter than to Customer C. As a result, we have two clusters, the cluster (AC) and the cluster (BD), after the first iteration.

Once all subjects are assigned into two clusters, we calculate the "centers", or "centroids", of these two clusters shown in Fig. 16.2(**b**). Note that the centroid for a given cluster is simply the vector of mean characteristics across the cluster members. Now we check the possibility of reassignment by evaluating the distances between each subject and the new cluster centroids. Customer A should not be moved because his or her distance to the centroid of the cluster (AC) is $\sqrt{17} = 4.1$ while his or her distance to the centroid of the cluster (BD) is $\sqrt{122} = 11.0$. However, Customer C should be reassigned to cluster (BD) because his or her distance to the centroid of the cluster (BD) is $\sqrt{10} = 3.2$ that is shorter than the customer's distance to the centroid of the cluster (AC), $\sqrt{17} = 4.1$. Using the same method, Customers B and D are should stay in their current cluster. In result, we now have two new clusters, the cluster (A) and the cluster (BCD), after the second iteration.

Figure 16.2(**c**) shows the centroids of the new clusters. Given the new two clusters and their centroids, we again evaluate whether each customer should be moved or not. None of the customers should move. Hence, we stop the

Table 16.2 Example of the K-means algorithm

(a) Characteristics of four subjects

Customers	x_1	x_2
A	13	3
B	3	3
C	5	1
D	1	1

(b) Clusters and their centroids for the first iteration

Clusters	**Centroid** (x_1)	**Centroid** (x_2)
(AC)	9	2
(BD)	2	2

(c) Clusters and their centroids for the second iteration

Clusters	**Centroid** (x_1)	**Centroid** (x_2)
(A)	13	3
(BCD)	3	1.7

algorithm at this iteration, and conclude that our final two-cluster solution is the cluster (A) and the cluster (BCD).

The K-means method is more appropriate than agglomerative clustering for large data application because it is computationally faster. The K-means is linear in the number of observations, while agglomerative is often cubic, depending on the dissimilarity measure. Moreover, it does not require a large amount of storage space in computer memory since its algorithm does not need to save the distance matrices. However, the final solution of the K-means clustering depends on its initial condition. In our example, we randomly selected k initial points or centroids.[4] Different initial centroids might result in a different final solution. Therefore, it is recommended to generate several starting points and compare the corresponding final solutions. Finally, K-means clustering algorithm implicitly assumes spherically shaped clusters with a common error variance. Hence, it tends to produce equal-sized clusters (Everitt 1993).

16.2.3.3 Probabilistic Clustering

Both agglomerative and K-means clustering have a critical limitation in assigning subjects into clusters. The models assume that each subject belongs to only one cluster. They do not allow any statements on the strength of a subject's cluster membership. Because of their deterministic nature, these methods are sensitive to outliers and do not perform well with overlapping clusters, i.e., when a customer is partially in one cluster and partially in

[4] Alternatively, we can randomly partition all subjects into k initial clusters. However, this method is not exempted from the same problem.

another. Attracting renewed attention from researchers, probabilistic clustering overcomes this problem by incorporating uncertainty about a customer's cluster membership. Another attractive aspect of probabilistic clustering is that it assumes there is a true underlying set of clusters, and the task is to uncover them. This is conceptually more pleasing than other methods that are really just common-sense algorithms for grouping customers together but can't guarantee the solution is unique or correct in any real sense.

Probabilistic clustering has other advantages as well. It does not require the scaling of variables. For example, when working with normal distributions with unknown variances, the results will be the same irrespective of whether the observed variables are normalized or not (Magidson and Vermunt 2002). In addition, it includes a more statistically justifiable way of determining the number of clusters and testing the validity of the clustering results. Probabilistic clustering is also called soft clustering, mixture-model clustering, model-based clustering, or latent class cluster analysis.

Probabilistic clustering assumes that the data to be clustered are generated from a finite mixture of underlying probability distributions in which each component distribution represents a cluster (Fraley and Raftery 1998). Let $\mathbf{x}_i$ be the vector of characteristics for Customer i. If Customer i is a member of cluster s, its conditional probability distribution or "density" is represented by $f_s(\mathbf{x}_i|\theta_s)$ where θ_s are the parameters that describe the conditional density. Then the unconditional probability distribution or density for the Customer $i(i = 1, \ldots, n)$ can be written as

$$g(\mathbf{x}_i|\theta) = \sum_{s=1}^{S} \pi_s f_s(\mathbf{x}_i|\theta_s) \tag{16.3}$$

where S is the number of clusters and π_s is the prior probability that the Customer i belongs to cluster s. Or π_s can be considered to be the size of the cluster s. Note that $\pi_s \geq 0$ for all s and $\sum_{s=1}^{S} \pi_s = 1$.

Most of studies on the specification of the Equation 16.3 assume that $f_s(\mathbf{x}_i|\theta_s)$ is multivariate normal (Banfield and Raftery 1993; Cheeseman and Stutz 1995; Dasgupta and Raftery 1998).[5] Hence, the parameters θ_s consist of a mean vector μ_s and a covariance matrix Σ_s. The most general model requires the estimation of means, variances, and covariances for all clusters. However, as the number of characteristics and/or clusters increase, the number of parameters to be estimated increases significantly. Hence, a number of researchers have proposed simpler models by restricting the potential values for the parameters in Σ_s.

An interesting restrictive model is the "local independence" model in which all within-cluster covariances are assumed to be zero, or Σ_s is assumed to be diagonal matrix. This model is not very restrictive as it sounds. The characteristics are (locally) independent within the given cluster. The observed

[5] On the other hand, latent class models assume that $f_s(\mathbf{x}_i|\theta_s)$ is Bernoulli or class-conditional distributions (Bartholomew 1987).

characteristics can still be correlated globally. Another interesting constraint is to assume that Σ_s is the same across all clusters (Banfield and Raftery 1993; Vermunt and Migidson 2000).

The estimation of the probability clustering is typically based on the EM algorithm (Dempster et al. 1977; Tanner 1993). The EM algorithm is a general approach to maximum likelihood in the presence of incomplete data. In probability clustering, the complete data for Customer i are considered to be $y_i = (\mathbf{x}_i, \mathbf{z}_i)$. The vector $\mathbf{z}_i = (z_{i1}, \ldots, z_{iS})$ represents the missing data, where z_{is} equals 1 if Customer i belongs to cluster s and 0 otherwise. Hence, the probability density of Customer i's data, $\mathbf{x}_i$, given $\mathbf{z}_i$ becomes $\prod_{s=1}^{S} f_s(\mathbf{x}_i|\theta_s)^{z_{is}}$. Each $\mathbf{z}_i$ is assumed to be independent and identically distributed as a multinomial distribution of one draw on S categories with probabilities $\pi_1, \ldots, \pi_S$. The resulting complete-data log-likelihood is

$$\ell(\theta_s, \pi_s, z_{is}|\mathbf{x}_i) = \sum_{i=1}^{n} \sum_{s=1}^{S} z_{is}[\log \pi_s f_s(\mathbf{x}_i|\theta_s)] \tag{16.4}$$

The complete-data log-likelihood represented by Equation 16.4 is maximized using an iterative EM algorithm. It iterates between an E-step in which values of $\hat{z}_{is} = E(z_{is}|\mathbf{x}_i, \theta_1, \ldots, \theta_S)$ are computed from the data with the current parameter values, and an M-step in which the complete-data log-likelihood, with each z_{is} replaced by its current conditional expectation $\hat{z}_{is}$, is maximized with respect to the parameters. The algorithm starts with initial guesses for $\hat{z}_{is}$. The E-step and the M-step are iterated until a convergence criterion of the researcher's choice is met.

As mentioned, probability clustering has several advantages over agglomerative or K-means clustering. However, it is not widely used among database marketers because of its computational difficulties. The optimization methods for probability clustering have storage and time requirements that grow at a faster than linear rate relative to the size of the initial partition (Fraley and Raftery 1998). Hence, it is not suitable for clustering a large number of customers.

The most popular commercial software implementing the probability clustering is included in the S-PLUS package as the function *mclust*. Several researchers have also improved the function *mclust* and written codes to interface with S-PLUS (e.g., see www.stat.washington.edu/fraley/mclust/soft.shtml). Alternative software to implement probability clustering is *Latent GOLD* by Statistical Innovations. It implements the probability clustering models assuming a mixture of normals, multinomials and others. It also estimates latent-class regression models.

16.2.3.4 Self-Organizing Maps (SOM)

There is a type of neural network model called the self-organizing map (SOM) that can be employed for a clustering task. Proposed by Tuevo Kohonen in

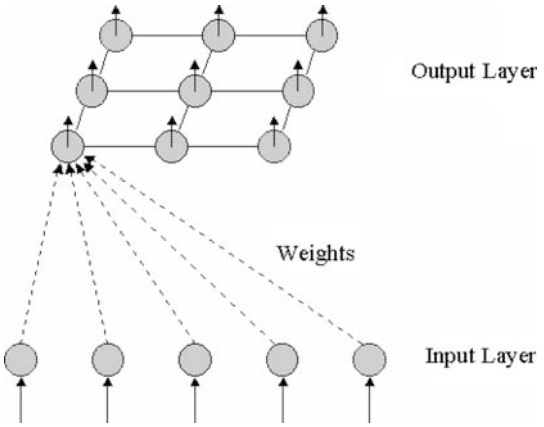

Fig. 16.4 Architecture of a self-organizing map.

1982 (Kohonen 1982), the SOM was originally used for image and sound, and recently applied to clustering people. Like other neural network models, the SOM has an input layer and an output layer (see Chapter 18). Each unit (or cluster) in the output layer is connected to units (or attributes) in the input layer and the strength of this connection is measured by a weight. However, the SOM is fundamentally different from other neural network models in that its goal is to identify no pre-specified "dependent variable" for the output layer. The SOM is looking for unknown patterns in the data. Using the terminology of machine learning, the SOM is developed for unsupervised learning. Hence, there are no training or pre-classified examples like other clustering algorithms.

Figure 16.4 shows the architecture of the SOM. Its input layer has five units, implying that each customer is represented by five attributes. Represented in a 3×3 two-dimensional grid, its output layer has nine units or clusters.[6] The output layers in the SOM are generally arranged in two-dimensional grid or lattice such as in Fig. 16.4. Each output unit is expected to become a prototype for a cluster of customers with one homogeneous class of input data vectors. The units in the output layers are not directly connected to each other. However, this grid-like structure allows output units to be related each other. Two adjacent units in the grid represent two similar clusters. If more nodes are in between two units (or clusters), they are meant to be more dissimilar.

Each unit in the output layer is connected to all units in the input layer even though we did not draw every line in Fig. 16.4. That is, the SOM can be

[6] Since the output layers in the SOM are typically arranged in 2-dimentional grid, the number of initial output units is restricted to certain numbers (e.g., 3×3 or 5×4). However, an SOM identifies fewer clusters than it has output units. Units with no hits or with very few hits are discarded. Hence, the final number of clusters chosen can be any numbers.

interpreted as a topology-preserving mapping from input space onto the two-dimensional grid of output units (Vesanto and Alhoniemi 2000). The number of data vectors available for training the SOM corresponds to the number of customers. In Fig. 16.4, each customer has been evaluated on five attributes. Each connection line has an associated weight, which is portrayed by a set of five lines, each with its own weight, connecting each output unit to each input variable.

The SOM computes the weights according to learning rules. The units in the output layer (through weight adjustments) get trained to learn about the patterns of the input data values. During the training each output unit (or cluster) competes to take "responsibility" for one particular observed input data vector. Only the winning output unit (and its neighbors) is allowed to learn by adjusting its weights for a given input data vector. Whether an output unit qualifies as the winner depends on the similarity of its current weight vector $\mathbf{w}$ and the data vector $\mathbf{x}$. More specifically, let the weight vector of output unit k be $\mathbf{w}_k = [w_{k1}, \ldots, w_{kJ}]$ where J is the number of attributes in the input layer. At each training iteration, a data vector $\mathbf{x}_i$ for customer i is randomly selected from the input data, and the similarity between the weight vector of the output unit k and the data vector is measured by their Euclidean distance.

$$\|\mathbf{x}_i - \mathbf{w}_k\| = \left[\sum\nolimits_{j=1}^{J} (x_{ij} - w_{kj})^2\right]^{1/2} \tag{16.5}$$

The distances between the data vector $\mathbf{x}_i$ and each of the units in the output layer are computed. The output unit with the smallest distance becomes the best-matching or winning unit for the given data vector $\mathbf{x}_i$.

Before the data vector is drawn, the weight vectors are initialized. At each iteration, a training data vector ($\mathbf{X}_i$) is randomly drawn from the input data, and the best-matching unit is determined by calculating the distances between the data vector and the weights for each of the units in the output layer (the $\mathbf{W}_i$'s). The weight vectors of the winning unit along with its neighboring units are updated to move closer to the data vector. As a result, each input data vector gradually belongs to one output unit as the weight updating repeats. More specifically, the weight vector of output unit i is updated according to the following rule.

$$w_{kj}(t+1) = \begin{cases} w_{kj}(t) + \lambda[x_{ij} - w_{kj}(t)] & if\ k \in N(k') \\ w_{kj}(t) & otherwise \end{cases} \tag{16.6}$$

where t is the iteration number, λ is the learning constant with $0 < \lambda < 1$, and $N(k')$ is the set of output units consisted of the winning unit k' and its neighbors (Kohonen 1994). Equation 16.6 implies that only the weights of the winning unit and its neighbors are adjusted. As a result, similar output units will be located closer to each other and similar data vectors occupy positions closer to each other than less similar ones. Even though there are many

ways to specify the updating rule, Equation 16.6 is the simplest. For more sophisticated updating rules, see Kohonen (1995) or Vesanto and Alhoniemi (2000).

16.2.4 The Number of Clusters

Determining the appropriate number of clusters is one of the most difficult problems in clustering. Sometimes managerial judgment is critical in deciding the number of clusters even though this tends to be subjective. For example, the relative sizes of the clusters should be large enough to be managerially meaningful. The clusters with a couple of subjects may be treated as outliers and ignored. As another example, marketing managers often limit the number of clusters because the implementation cost may be beyond their budgets or the fine-tuned segmentation strategy may not be feasible.

We now focus our attention on more formal ways to determine the number of clusters. The methods for determining the number of clusters depend on the clustering algorithm being used and there are several criteria available. However, Milligan and Cooper (1985) showed that the procedure by Calinski and Harabasz (1974) performed the best among 30 different criteria. Calinski and Harabasz suggested the following criterion to determine the number of clusters:

$$G(k) = (n-k)(T-W)/(k-1)W \tag{16.7}$$

where k is the number of clusters, n is the number of customers, W is the square sum of the distances of the customers to the center of its cluster, and T is the square sum of the differences of each customer to the average customer, essentially, the center of the full data. The optimal number of cluster can be determined by selecting k which returns the maximum value for $G(k)$, because in that case, W, or the distances between customers and the center of their clusters, is relatively small compared to T, the distances between customers and the center of the entire data.

For probabilistic clustering, there is a more formal way of determining the optimal number of clusters. Once we apply different numbers of clusters, we select the number of clusters that will minimize the *BIC* (Bayesian Information Criterion) proposed by Schwarz (1978). The *BIC* of the probabilistic clustering with s clusters can be written as

$$BIC_S = -2\log l_s + m_s \log n \tag{16.8}$$

where m_s is the number of estimated parameters for the model with s clusters and $\log l_s$ is the corresponding log-likelihood. As the number of clusters increase, the log-likelihood will increase. However, the penalizing term m_s

increases at the same time. We choose the model with the optimal number of clusters that will minimize the *BIC*.

16.3 Applying Cluster Analysis

16.3.1 Interpreting the Results

The results of a cluster analysis are interpreted by examining the means for each cluster of the clustering variables, and also examining the means of any other variable, i.e., "discrimination variables," not included in the clustering routine. For example, we may cluster based on benefits sought but have a host of other variables, for example demographics, that we use for interpretation.

Table 16.3 shows a hypothetical example of a cluster analysis based on a survey of 500 customers in the personal home computer market. The clustering was based on benefits sought; all variables measured on a 7-point scale. In addition, there are several demographic and psychographic variables not used in the cluster analysis, but available as discrimination variables.

Interpreting the clusters is a subjective but interesting task that often adds insight into the nature of the market. In this case, customers in Cluster 1 are very concerned with "ease of use" and "technical support," and customers in this cluster are not very likely to own a home computer. This cluster might be called "Novices." Novices would be an attractive segment because they

Table 16.3 Hypothetical cluster analysis results for home computer market

	Means on clustering variables		
	Cluster 1 ("Novices")	**Cluster 2 ("Family")**	**Cluster 3 ("Heavy Users")**
Speed	2.4	3.4	5.4
Capacity	2.7	3.3	6.1
Ease of use	5.3	5.1	2.1
Aesthetics	1.2	5.7	2.3
Reliability	4.3	3.3	5.5
Technical Support	6.6	3.3	4.0
% of sample	30%	15%	55%
	Means on discrimination variables		
Age (years)	45.4	47.3	35.1
Children present (%)	10%	48%	29%
Income (K, in $)	45.2	50.1	35.2
Use for work	20%	10.1%	45.6%
Currently own computer	22%	56.1%	75.2%
Current Mac users	10%	11%	10%

do not own a computer but fit the age and income profile of a customer who could use a computer (e.g., comparing to Cluster 2). However, the Novices might be expensive to serve because they care about ease of use and technical support, so could end up calling the company's customer service center too often.

Customers in Cluster 2 care a lot about ease of use and aesthetics, and is dominated by families with children. We might call this the "Family" segment. Probably the children present leads to the importance of ease of use, and noting the slightly older age of customers in this cluster, perhaps those children are teenagers, where aesthetics of the computer could be important. The Family segment might be attractive for a company like Apple, a company that excels in aesthetics and design.

Customers in the third cluster cares a lot about speed, capacity, and reliability, and customers in this cluster use a computer at work as well as own their own computer. This might be called the "Heavy User" segment. This segment would be attractive to a company that can excel on technical specifications such as speed, and provide capacity and reliability at low cost. In addition, the cluster analysis classifies 55% of customers in this segment, so it is the largest segment. Also, the Heavy User segment might be expected to want to trade up their computer as often as possible to the latest, fastest, home computer available.

Note that the interpretations are subjective and make use of both the clustering and discrimination variables. Note also there are clear managerial implications in that companies with particular strengths could plausibly target one of these clusters, but probably not all of them (at least with the same product).

16.3.2 Targeting the Desired Cluster

Note that the cluster analysis was based on a survey of 500 consumers. Let's say the company decided to target the Family segment. The next task – the task of database marketing – would be to figure out how to reach these customers. The measurement of benefits sought is unique to the survey. There are probably no lists available of large numbers of consumers who have answered the same benefits sought questions. However, lists are available of customers that provide measures of the discrimination variables, since these are mostly demographics and usage behaviors. For example, to target the Family segment, one would compile a list of customers that contained most if not all of the discrimination variables. One might have to purchase different lists and merge them together, or have a company such as Vente (http://lists.venteinc.com/market) compile the list. One could next proceed in two ways. One would be to select from the list customers who tend to fit the profile on the discrimination variables for the Family segment. This could

be done heuristically (e.g., select from the list households that own a home computer and have children present).

Another approach would be to estimate a predictive model based on the cluster analysis sample of 500, and apply it to the larger list for which the discrimination variables are available (see also Chapter 10). For example, one could use a logistic regression on the sample of 500 to determine whether or not the customer is in the Family segment, or use a multinomial logit to predict which of the three segments the customer is in. The dependent variable would be cluster membership. The independent variables would be the discrimination variables. Note this model would be estimated on the 500 customers because these are the customers for whom we know cluster membership. However, once we have the multinomial logit model, we can use it to "score" the entire customer list because we have the discrimination variables in that list. In this way, we could assign all 5,000,000 customers to benefits sought segments.

The above example shows how a rich set of measures obtained from a small sample can be used to identify and target customers among a larger set. The key to the success of this strategy is the existence of discrimination variables that are available for large numbers of customers. The small sample has available the clustering variables and the discrimination variables, while the compiled list has the discrimination variables. A predictive model or heuristic selection procedure allows the database marketer to infer the cluster membership of customers of the compiled list.

While this procedure is very valuable, it is possible that the clustering variables might be available for a large list and hence the above process might not be necessary. For example, consider the case of a company wanting to start a customer tier program. The company may cluster analyze its customers based on various measures of customer value (LTV, responsiveness to marketing, duration, RFM variables, etc.). It would be impractical to run the cluster analysis of all its 5,000,000 customers. So the analyst would run the cluster analysis on say 2000 customers. Then the rest of the customers could be assigned to a cluster by directly calculating its similarity to each cluster. This is obviously the most desirable situation. The small-sample – predictive model – compile list – score list approach obviously has more steps and relies on coming up with a good predictive model and being able to compile a list of many customers with data on the discrimination variables. However, many customer lists are available, and there are companies that specialize in list compilation, so especially for a customer acquisition scenario, the approach makes good sense.

The above illustrates how cluster analysis can be used to interpret segments, make a targeting decision, and then use database marketing to target potential members of the sought-after segment. The contribution of cluster analysis is extremely important to this process. It provides a rich portrait of how the market might be segmented, and often through the

discrimination variables, how they might be reached. While as we've discussed in this chapter, there are many different methods we can use to form the clusters, the real "validation" of the technique is in the managerial value of interpreting the clusters, and the targetability and ultimately the profitability of the segments realized through list creation and predictive modeling.