

Lecture 3 Clustering

Section 3.1 Background

Section 3.2 Distance Measures

Section 3.3 Clustering Methods

3.3.1 Hierarchical Clustering

3.3.2 Partition Based Clustering

Section 3.4 Variable Clustering

Section 3.5 Clustering with SAS Enterprise Miner

Section 3.6 Case Study 1: How to Use Cluster Node

Section 3.7 Case Study 2: Variable Clustering for Logistic Regression

Appendix 3.1 Data Used in Lecture 3 Section 3.6

Appendix 3.2 Data Used in Lecture 3 Section 3.7

Appendix 3.3 Cubic Cluster Criterion

Appendix 3.4 Reference for Lecture 3

Section 3.1 Background

Suppose the data set is huge, it is very likely that the data are heterogeneous. This means that the data might fall in several distinct groups, with members within each subgroup being similar to each other but different from members of other groups. Since it is possible that there are different models or patterns in each group, it is very difficult to spot any single pattern or model to represent the whole data set. Creating clusters of similar cases reduces the model complexity within clusters that allows data mining techniques more likely to success in each cluster. Even if the data do not have natural groups, partition data into homogeneous groups can be very useful. For example, it is well known that the customer preference for their products depends on geographic and demographic factors. Thus, we can use geographic and demographic factors to group customers into several segments and to develop marketing strategy for each segment. Although customers do not form these marketing segments naturally, it is much easier to develop efficient marketing strategy for each segment separately than to one single marketing strategy to target all customers.

Clustering is one important unsupervised data-mining tool. Unlike supervised data mining tools, cluster analysis has no assumptions that are made concerning the number of clusters or cluster structure. The basic objective in clustering is to discover natural groupings of the cases or variables based on some similarity or distance (dissimilarity) measures. Although there is no target variable to be predicted, clustering technique can be used in many ways. First, it can be used in missing value imputation. For example, one can use cluster mean to impute missing value for a numerical variable instead of using overall mean because overall mean does not consider the between-variable relationship. Secondly, one can use clustering to detect outliers because outliers typically belong to clusters with only one case. Thirdly, one can use clustering to discover the characteristics of clusters if he suspects that there are meaningful groupings that may represent groups of cases. After finding these meaningful groups, one can then develop different ways to deal with each group such as target marketing that will be discussed later. Fourthly, we can use cluster analysis to partition a complex data structure into several subsets in order to give supervised data-mining techniques such as decision tree or neural network a better chance of finding a good predictive model.

Since the benefits of cluster analysis are easy to see, there are huge amount of research effort in the past several decades on cluster analysis and there are many “automatic clustering” techniques available. However, different clustering techniques lead to different types of clusters and it is very difficult to tell whether a cluster analysis exercise has been successful because cluster analysis is an unsupervised data mining exercise. To use cluster techniques effectively, we need to at least understand several important aspects of choosing clustering techniques.

Select Similarity Measure:

In order to decide whether a set of cases can be split into subgroups with members are similar to other members within the same group than members from other groups, we need to define what we mean “similar”. Central to all of the goals of cluster analysis is

the notion of the degree of similarity (or dissimilarity) between the individual objects being clustered. This can only come from *subject matter* considerations. Suppose we want to group a deck of ordinary playing cards into clusters. There are many ways to do so such as

- Two clusters: One cluster has all the face card and another cluster has all other cards.
- Four clusters: Each suit of thirteen cards forms one cluster.
- Two clusters: Red suits are in one cluster and black suits are in another cluster.
- Thirteen clusters: Each cluster has all cards that have the same face value.

Obviously, the clusters obtained are very different with different “similarity measure”. Thus, we need to know how to choose a similarity measure before selecting clustering technique.

Select the Right Number of Clusters

One important question to ask in cluster analysis is “what is the right number of clusters?”. In the well know k-mean cluster algorithm, the original choice number of k determines the number of clusters that will be found. If this number does not match to the natural structure of the data, the technique will not obtain good results. Unless there is good prior knowledge on how many clusters exist in the data, it is very difficult to choice the number of k before applying k-mean cluster technique. Typically, the miner needs to try several different k 's before finding the “right” number of k . In general, the best set of clusters is the one that does best job of keeping the distance between members in the same cluster small and the distance between members of adjacent clusters large. However, if the purpose of clustering is to detect unexpected pattern, the right number of clusters might the one that can find unexpected pattern from the data.

Cluster Interpretation

Clustering is a powerful unsupervised knowledge discovery technique, however, it has weakness and limitations. For example, if one does not know what he is looking for, one may not recognize it when one finds it. Although the clustering technique can help to find clusters, it is up to the user to interpret them. The following approaches can help the user to understand clusters:

- Use graphical tools or summary statistics to exam the within cluster distribution for each variable (the INSIGHT node)
 - Use graph tools such as box plot to study the within cluster distribution for each continuous variable
 - Use graph tools such as Mosaic plot to study the within cluster distribution for each categorical variable
 - Study the within cluster statistics for each variable
- Use other visualization tools to see how the clusters are affected by changes for each variable with normalization mean plot (Clustering node)
- Building a decision tree with the cluster label as the target variable and using it to derive rules explaining how to assign new records to the correct cluster

Section 3.2 Distance Measures (Similarity Measures)

Similar to many other data mining techniques, cluster analysis is based on distance measure between cases. Most commonly used distance measure is Minkowski metric.

Suppose $x(i) = [x_1(i), x_2(i), \dots, x_m(i)]$ and $x(j) = [x_1(j), x_2(j), \dots, x_m(j)]$ be any two cases with m variables, the Minkowski metric (L_p norm) is defined as

$$d(i, j) = \left[\sum_{k=1}^m |x_k(i) - x_k(j)|^p \right]^{1/p}.$$

For $p=2$, $d(i,j)$ becomes to the Euclidean distance. For $p=1$, $d(i,j)$ becomes to the mean absolute deviation between the two cases. For $p=\infty$, $d(i,j)$ becomes to the maximum absolute deviation between the two cases.

Minkowski metric satisfies the following properties

- $d(i,j) = d(j,i)$
- $d(i,j) > 0$ if $i \neq j$
- $d(i,j) = 0$ if $i = j$
- $d(i,j) < d(i,k) + d(j,k)$.

Since this measure of distance assume some degree of *commensurability* between the different variables. Thus, it would be effective if each variable measured using the same units and each variable is equally important. But, it is very unlikely that all variables in a data mining exercise were measured with the same unit. One way to deal with this incommensurability is to standardize the data by divided each of the variables by its sample standard deviation. In addition, if we have the idea of the relative importance of these variables, then we can weight them (after standardization) to yield the weighted standardize distance measure.

Another commonly used distance measure is the statistical distance that takes the variance-covariance structure into consideration. However, without prior knowledge of the distinct groups and the number of groups, the variance-covariance structure can not be computed. For this reason, Minkowski distance (or Euclidean distance) is often preferred for clustering purpose.

Data Layout

- **The raw data – data matrix**

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}$$

- **Proximity (Dissimilarity) Matrix**

$$\begin{bmatrix} 0 & d(1,2) & \dots & d(1,n) \\ d(2,1) & 0 & \dots & d(2,n) \\ \dots & \dots & \dots & \dots \\ d(n,1) & d(n,2) & \dots & d(n,n) \end{bmatrix}$$

Section 3.3 Clustering Methods

The best way to find clusters is to examine all possible clusters. However, it takes too long to examine all clusters even with the fastest and largest computer. Because of this problem, a wide variety of clustering algorithms have emerged that find “reasonable” clusters without having to look at all possible clusters.

3.3.1 Hierarchical Clustering

Hierarchical clustering techniques proceed by either a series of successive merges (*bottom-up*) or a series of successive divisions (*top-down*). *Agglomerative* hierarchical clustering methods start with the individual cases. Thus, there are as many clusters as cases. Most “similar” cases are first merged to form a reduced number of clusters. This is repeated until just one cluster with all cases.

Let $D = \{x(1), x(2), \dots, x(n)\}$ be n cases and $D(C_i, C_j)$ be the distance measure between any two cluster C_i and C_j . Then, an agglomerate algorithm for clustering can be described as follows:

```

for  $i = 1$  to  $n$  let  $C_i = \{x(i)\}$ 
while there is more than one cluster left do
    let  $C_i$  and  $C_j$  be the two clusters minimizing
    the distance between any two clusters;
     $C_i = C_i \cup C_j$ 
    remove  $C_j$ ;
end;
  
```

In “Single Linkage” method, the distance between two clusters is defined as

$$D_{SL}(C_i, C_j) = \min\{d(x, y) \mid x \in C_i \text{ and } y \in C_j\}.$$

The clusters formed by single linkage method will not be affected by the distance measures used if these distance measures have the same relative ordering. It also has the property that if two pairs of clusters are equidistance it does not matter which one is merged first. The overall result will be the same. Single linkage method is the only clustering method that can find *nonellipsoidal* clusters. The tendency of single linkage method to pick up long string like cluster is known as chaining. This tendency and sensitivity to outlier cases and perturbation of the data combined make it less useful in customer segmentation application.

In “Complete Linkage” method, the distance between two clusters is defined as

$$D_{CL}(C_i, C_j) = \max\{d(x, y) \mid x \in C_i \text{ and } y \in C_j\}.$$

The clusters formed by complete linkage method will not be affected by the distance measures used if these distance measures have the same relative ordering. Complete linkage method tends to find clusters to be equal size in terms of the volume of space occupied, making it particularly suitable in customer segmentation application.

In “Average Linkage” method, the distance between two clusters is defined as

$$D_{AL}(C_i, C_j) = \frac{\sum_{k=1}^{C_i} \sum_{h=1}^{C_j} d(x, y)}{n_{C_i} + n_{C_j}}.$$

The clusters formed by average linkage method will be affected by the distance measures used even if these distance measures have the same relative ordering. This makes average linkage less attractive in data mining application.

Agglomerative clustering only needs to have a distance matrix to start the clustering procedure. This means that it does not need to store all variable values for each case. Suppose we can compute the “distance” between variables, these methods can be applied in *variable clustering* as well. We will address this issue in Section 3.4. These methods mentioned here have several drawbacks.

- ❖ First, a case will stay in the same cluster once it is assigned to this cluster. This means that reallocation is not allowed in the clustering process even if one case has wrongly assigned to a cluster.
- ❖ Secondly, these methods are sensitive to outliers and “noise”. Thus, we need to try several different cluster methods and, within each method, to try several distance measures. If the outcomes from all methods are consistent with one another, perhaps a set of good clusters has been found. Also, we can add small errors to each case before applying clustering method to see how stable the clusters are.

Other than linkage methods, there are *centroid* method (that defines the distance between two cluster as the distance between their centroids) and *Ward* statistics. These will be discussed in Section 3.3.2.

Example 3.1 Single Linkage, Complete Linkage, and Average Linkage

Sometimes the data is represented directly in terms of the proximity (alikehood or affinity) between pairs of objects. These can be either similarities or dissimilarities (difference or lack of affinity). For example, in social science experiments, participants are asked to judge by how much certain objects differ from one another. Dissimilarities can then be computed by averaging over the collection of such judgments. Nevertheless, subjectively judged dissimilarities are seldom distances in the strict sense, since the triangle inequality $d(i,j) < d(i,k) + d(j,k)$ does not hold.

The distances between pairs of five cases are given below.

$$\begin{array}{c}
 1 \quad 2 \quad 3 \quad 4 \quad 5 \\
 \left[\begin{array}{ccccc}
 0 & & & & \\
 9 & 0 & & & \\
 3 & 7 & 0 & & \\
 6 & 5 & 9 & 0 & \\
 11 & 10 & 2 & 8 & 0
 \end{array} \right]
 \end{array}$$

Cluster the five cases using each procedure and draw the dendrograms and compare the results.

- (a) Single linkage hierarchical procedure.
- (b) Complete linkage hierarchical procedure.
- (c) Average linkage hierarchical procedure.

<Solutions>:

(a) (Single Linkage)

Step 1: Merge case 3 and case 5 since $\min(d_{ij}) = d_{35} = 2$

Step 2: $d_{(3,5),1} = \min(d_{31}, d_{51}) = 3$

$d_{(3,5),2} = \min(d_{32}, d_{52}) = 7$

$d_{(3,5),4} = \min(d_{34}, d_{54}) = 8$

Thus, the new distance matrix is

$$\begin{array}{c}
 (35) \quad 1 \quad 2 \quad 4 \\
 \left[\begin{array}{cccc}
 0 & & & \\
 3 & 0 & & \\
 7 & 9 & 0 & \\
 8 & 6 & 5 & 0
 \end{array} \right]
 \end{array}$$

Step 3: Merge case (3,5) and case 1 since minimum distance is 3.

Step 4: The new distance matrix is

$$\begin{array}{c}
 (135) \quad 2 \quad 4 \\
 \left[\begin{array}{ccc}
 0 & & \\
 7 & 0 & \\
 6 & 5 & 0
 \end{array} \right]
 \end{array}$$

Step 5: Merge case 2 and case 4 since the minimum distance is 5.

Step 6: Merge all cases together and the minimum distance is 6.

(b) (Complete Linkage)

Step 1: Merge case 3 and case 5 since $\min(d_{ij}) = d_{35} = 2$

Step 2: $d_{(3,5),1} = \max(d_{31}, d_{51}) = 11$

$$d_{(3,5),2} = \max(d_{32}, d_{52}) = 10$$

$$d_{(3,5),4} = \max(d_{34}, d_{54}) = 9$$

Thus, the new distance matrix is

$$\begin{array}{c} (35) \quad 1 \quad 2 \quad 4 \\ (35) \left[\begin{array}{cccc} 0 & & & \\ 1 & 11 & 0 & \\ 2 & 10 & 9 & 0 \\ 4 & 9 & 6 & 5 & 0 \end{array} \right] \end{array}$$

Step 3: Merge case 2 and case 4 since minimum distance is 5.

Step 4: The new distance matrix is

$$\begin{array}{c} (35) \quad (24) \quad 1 \\ (35) \left[\begin{array}{ccc} 0 & & \\ (24) & 10 & 0 \\ 1 & 11 & 9 & 0 \end{array} \right] \end{array}$$

Step 5: Merge case (24) and case 1 since the minimum distance is 9.

Step 6: Merge all cases together and the minimum distance is 11.

(c) (Average Linkage)

Step 1: Merge case 3 and case 5 since $\min(d_{ij}) = d_{35} = 2$

Step 2: $d_{(3,5),1} = \text{avg}(d_{31}, d_{51}) = 7$

$$d_{(3,5),2} = \text{avg}(d_{32}, d_{52}) = 8.5$$

$$d_{(3,5),4} = \text{avg}(d_{34}, d_{54}) = 8.5$$

Thus, the new distance matrix is

$$\begin{array}{c} (35) \quad 1 \quad 2 \quad 4 \\ (35) \left[\begin{array}{cccc} 0 & & & \\ 1 & 7 & 0 & \\ 2 & 8.5 & 9 & 0 \\ 4 & 8.5 & 6 & 5 & 0 \end{array} \right] \end{array}$$

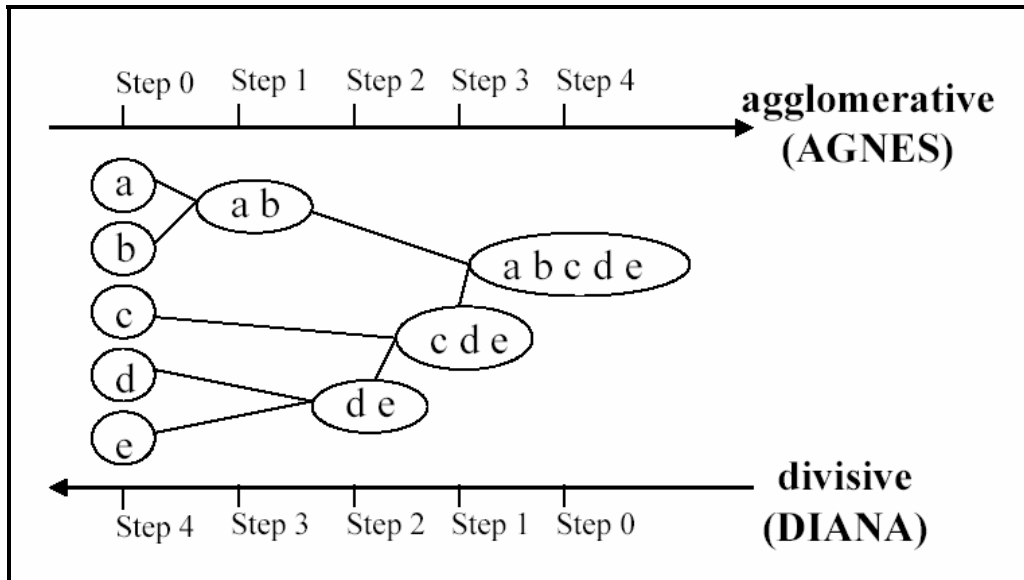
Step 3: Merge case 2 and case 4 since minimum distance is 5.

Step 4: The new distance matrix is

$$\begin{array}{c} (35) \quad (24) \quad 1 \\ (35) \left[\begin{array}{ccc} 0 & & \\ (24) & 8.5 & 0 \\ 1 & 7 & 7.5 & 0 \end{array} \right] \end{array}$$

Step 5: Merge case (35) and case 1 since the minimum distance is 7.
 Step 6: Merge all cases together and the minimum distance is 8.33.

Dendrogram diagram will be drawn in class.



3.3.2 Partition Based Clustering (Centroid Approaches)

In partition based clustering, the task is to partition data set into k disjoint clusters of cases such that the cases within each cluster are as homogeneous as possible, that is, given a set of n cases $D = \{x(1), x(2), \dots, x(n)\}$, our task is to find k clusters $C = \{C_1, C_2, \dots, C_k\}$ such that each case $x(i)$ is assigned to a unique cluster C_j .

There are many *score functions* can be used to measure the quality of clustering. Centroid method uses the distance of two cluster centroids to measure the distance between two clusters. Average method use the average distance between all pairs of points (one point from each cluster) to measure the distance between two clusters. Wald statistics use the between clusters sum of squares to measure the distance between two clusters. Once the score function is selected, it is an optimization process to find clusters. Many optimization procedures can be applied. Here, we only introduce the popular K-means clustering method.

Let $D = \{x(1), x(2), \dots, x(n)\}$ be n cases and our task is to find K clusters $C = \{C_1, C_2, \dots, C_K\}$:

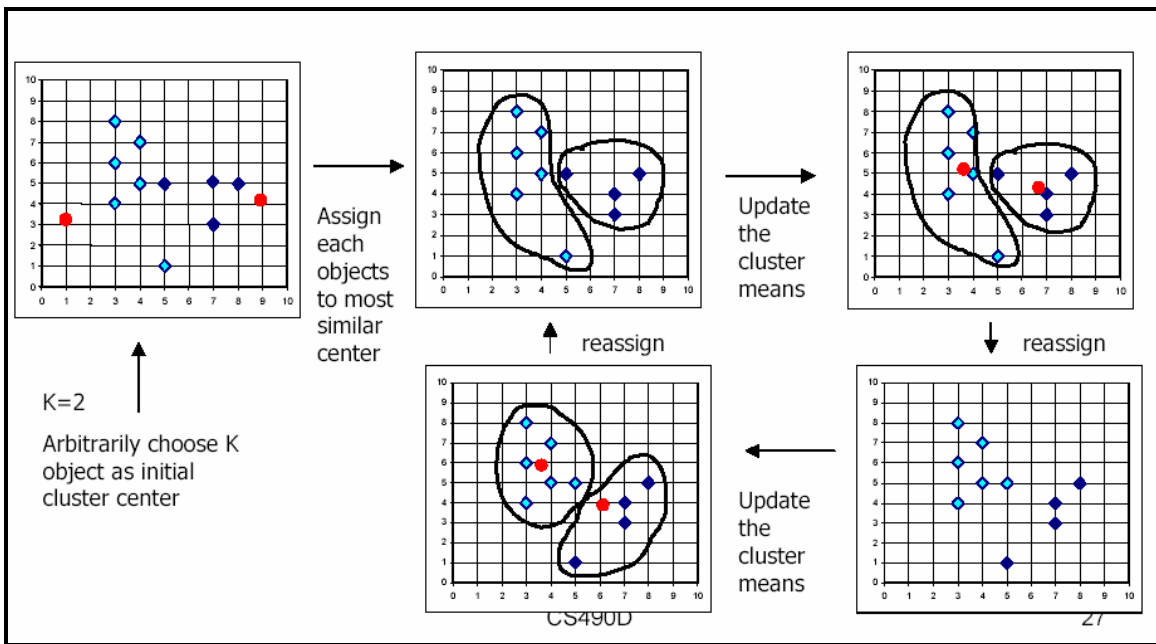
Let $\{r(k): k = 1, 2, \dots, K\}$ be K randomly selected points in D .
 for $i = 1, 2, \dots, I$;
 form clusters:
 for $k = 1, 2, \dots, K$ do

```

Ck={x∈D|d(r(k),x) ≤d(r(j),x) for all j = 1, 2, ..., K, j≠k}
end;
compute the new cluster centers;
for k = 1, 2, ..., K do
    r (k)= the vector mean of the cases in Ck
end;
end;
end;

```

The time complexity of K-means algorithm is $O(KIn)$, where I is the number of iterations. Since K , the number of clusters, is fixed in partition based clustering methods, the selection of K is very important. If the number of natural clusters is different from K , the results supplied by the partition based clustering algorithm will not be satisfactory. One way to avoid this problem is to try several different numbers of K and run the algorithm several times. Then use either R^2 or *cubic clustering criterion* (in Appendix 3.3) to select the “right” number of clusters.



Section 3.4 Variable Clustering

All methods discussed in section 3.3.1 can be used in variable clustering except the average linkage method if the distance between variables can be computed. Most popular distance measure for numerical variables is Pearson correlation coefficient. For categorical variables, we typical use the association to measure the distance between them. Actually, we can show that the correlation and association are the same for two binary variables (see Problem 1 in Assignment #3).

Example 3.2

Suppose the correlation matrix is

$$\begin{pmatrix} X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 \\ 1 & & & & & & & \\ .643 & 1 & & & & & & \\ -.103 & -.348 & 1 & & & & & \\ -0.82 & -.086 & .100 & 1 & & & & \\ -.259 & -.260 & .435 & .034 & 1 & & & \\ -.152 & -.010 & .028 & -.288 & .176 & 1 & & \\ .045 & .211 & .115 & -.164 & -.019 & -.374 & 1 & \\ -.013 & -.328 & .005 & .486 & -.007 & -.561 & -.185 & 1 \end{pmatrix}$$

Use Single linkage and complete linkage to find the clusters.

<Solutions>:

Problem 2 in Assignment #3.

Section 3.5 Clustering with SAS Enterprise Miner

The **Clustering** node in SAS Enterprise Miner uses two SAS procedures, FASTCLUS and CLUSTER. The FASTCLUS procedure is designed to handle a much larger data set than PROC CLUSTER can. The **FASTCLUS** procedure performs nonhierarchical cluster analysis. That means the clusters obtained do not have the tree structure as they do in hierarchical cluster analysis algorithm such as the CLUSTER procedure. In order to obtain hierarchy clusters for a very large data set, one can use PROC FASTCLUS to find initial clusters and then uses those initial clusters as input to PROC CLUSTER to find clusters with the tree structure.

By default, FASTCLUS procedure uses K-means method that is composed of the following four steps.

1. Partition the cases into K initial clusters
2. Proceed through the list of cases and assign each case to the cluster whose centroid is nearest.
3. Recalculate the centroid for any cluster either receiving new cases or losing cases.
4. Repeat Step 2 and Step 3 until no more reassignments take place,

K is the number of clusters that can be determined either in advance by user or as part of the clustering procedure. By default the clustering node uses **CLUSTER** procedure with

Cubic Clustering Criterion (CCC) based on a sample of 2000 observations to estimate the appropriate number of clusters (between 2 and 40). If you do not want to use the default setting to choose the number of clusters, you can change these setting in **Clusters** tab. For example, we can use the following steps to change the “cluster distance” measure to *average linkage* and keep all other options.

1. Open **Clustering** node
2. Select **Clusters** tab
3. Select **Selection Criterion** tab
4. Change the **Clustering Method** to “Average”

The choice of K is very important in K-Mean clustering algorithm. The algorithm is very likely to provide bad results if the number K does not match the natural structure of the data. Unless there is prior knowledge about the optimal value of K, the miners probably need to try several values of K before deciding the optimal value for K. In general, the best set of clusters is the one that does the best job of keeping the distance between members of the same cluster small and the distance between members of adjacent clusters large.

Other options can be changed in **Clusters** tab includes:

- Variable name for clusters (the default is `_SEGMENT_`)
- Label for Cluster Variable (the default is Cluster ID)
- Model role for Cluster Variable (the default is Group)
- The number of clusters (Check **User Specify** in Clusters Tab)
- Method used in computing the cluster distance (Average, Centroid, and Wald)
- The cutoff number for Cubic Clustering Criterion (the default is 3)
- Minimum number of clusters (the default is 2)
- Maximum number of clusters (the default is 40)
- Minimum number of cluster size (the default is the cluster size in training sample)

After the number of clusters has been decided either by user specified number or by CLUSTER procedure with a sample, the **Clustering** node to find the initial clusters with **FASTCLUS** procedure.

By default, the FASTCLUS procedure uses Euclidean distances, so the cluster centers are based on least-squares estimation. This kind of clustering method is often called a *k*-

means model, since the cluster centers are the means of the observations assigned to each cluster. However, any L_p norm can be used as the distance measure. For example, you can change the **Clustering criterion** to “Midrange” in **Seeds** tab if you want to use the L_∞ norm and you can change it to “Median” if you want to use L_1 norm. It worth knowing that **Clustering** node can not automatically compute the optimal number of clusters for you if the distance measure is other than L_2 . Other options can be changed in **Seeds** tab include

- Maximum number of iteration:

PROC FASTCLUS is designed to find good clusters (but not necessarily the best possible clusters) with only two or three passes over the data set. The initialization method of PROC FASTCLUS guarantees that, if there exist clusters such that all distances between observations in the same cluster are less than all distances between observations in different clusters, and if you tell PROC FASTCLUS the correct number of clusters to find, it can always find such a clustering without iterating. Even with clusters that are not as well separated, PROC FASTCLUS usually finds initial seeds that are sufficiently good so that only a few iterations are required. Hence, by default, PROC FASTCLUS performs only one-iteration.

- Convergent Criterion:

Iterations stop when the maximum relative change in the cluster seeds is less than or equal to the convergence criterion. The relative change in a cluster seed is the distance between the old seed and the new seed divided by a scaling factor. If the distance measure is L_2 norm, the scaling factor is the minimum distance between the initial seeds. Otherwise, the scaling factor is an L_1 scale estimate and is recomputed on each iteration. Specify the “**Convergent Criterion**” only if the maximum number of iteration is greater than 1.

- The minimum distance between seeds (radius):

No observation is considered as a new seed unless its minimum distance to previous seeds exceeds the minimum distance between seeds. The default value is 0. If you use L norm, the cluster seed is the mean value of each variable.

Section 3.6 Case Study 1: How to Use Cluster Node

This case study shows you how to use **Clustering** node to partition the cases into several clusters and how to use **Clustering Result Browser** to detect interesting patterns.

Input Data Source Node:

- Select CLUSEXAM from STA5703 library

- Change model role of LOC to “rejected” based on domain expert’s opinion
- Since there are only about 2-3% of missing values for five variables, we do not need to use **Replacement** node to impute these missing values.

Clustering Node – Variable Tab:

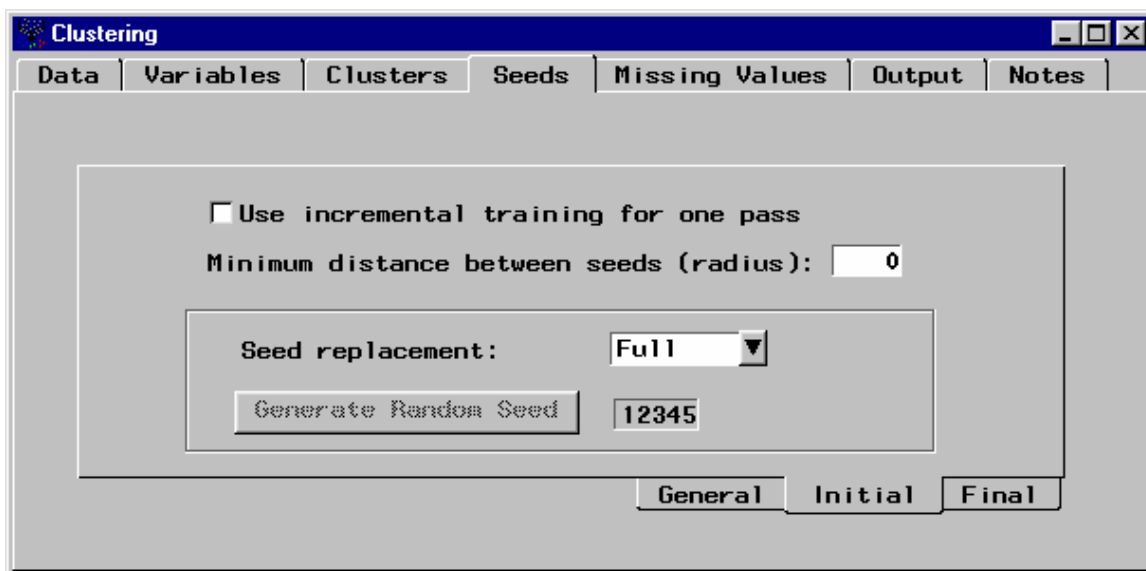
- Select “**Std Dev**” radio button because K-means clustering is very sensitive to the scale of measurement of different input variables.
- Also, clustering algorithm works well on numerical variables, i.e., we had better not to use categorical variables.
- We will use all variables in this case study because there are only a few numerical variables in the data.

Clustering Node – Clusters Tab:

- We keep default here

Clustering Node – Seeds Tab:

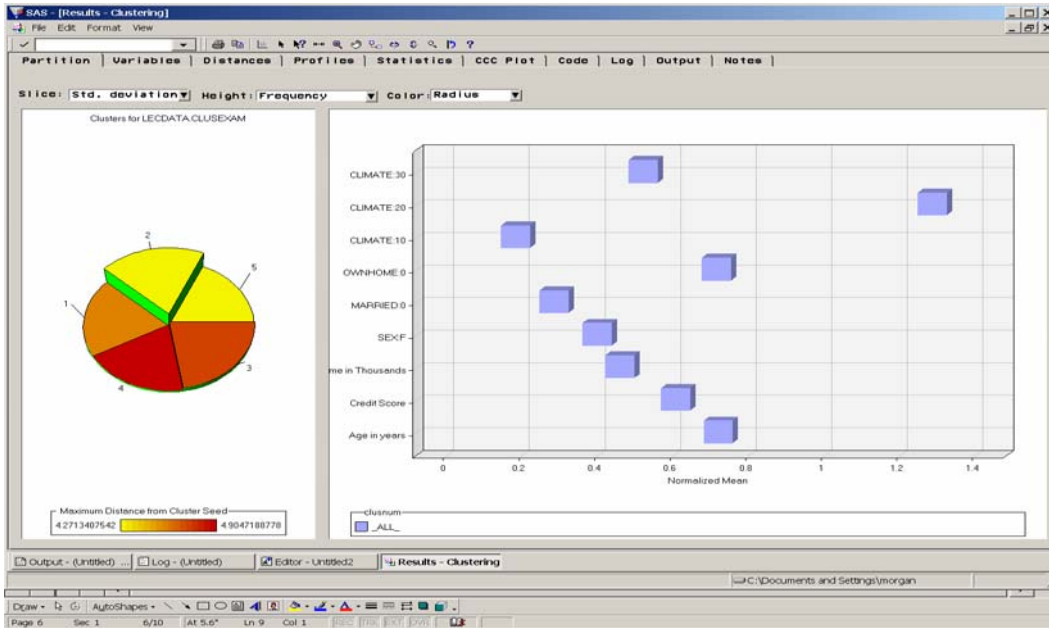
- We keep default here too



Clustering Node – Missing Values Tab:

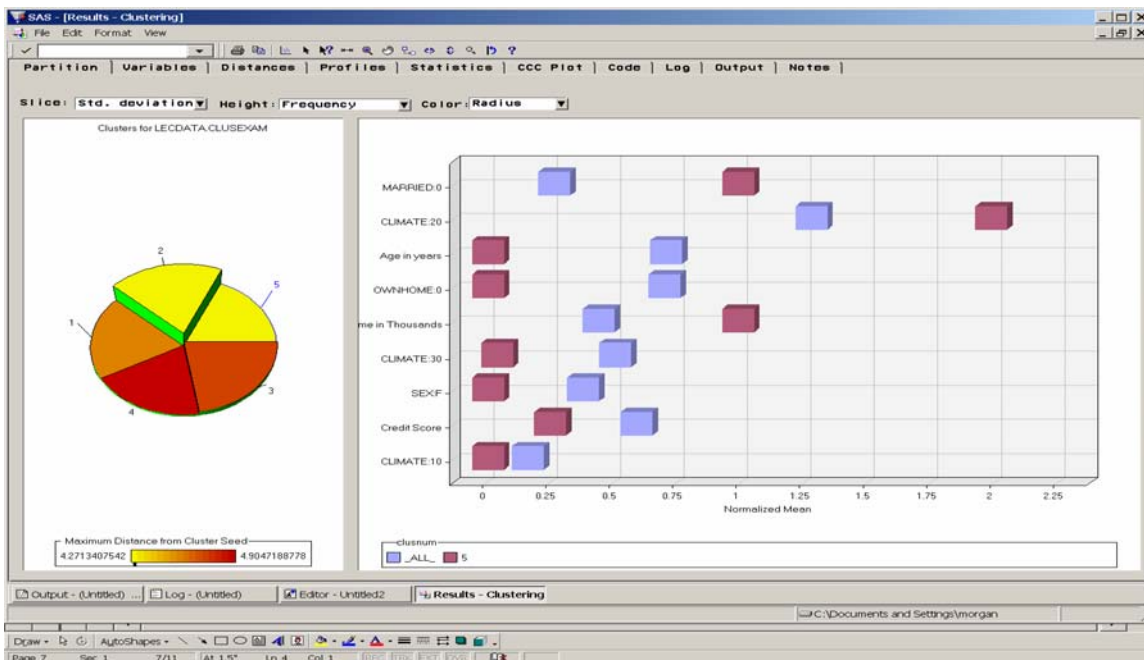
- We use cluster mean imputation to impute missing values by checking “**Imputation**” and selecting “**Mean of nearest cluster**” method (Note: “Mean of nearest cluster” and “Centroid of nearest cluster” are same if the distance measure is L_2 norm.)

Results:



There are five clusters. Cluster 2 has the most observations (high of the slice indicates the cluster size). Cluster 4 has the largest radius. Note that there are three variables associated with *CLIMATE* because *Clustering* node constructs a dummy variable for each level of a categorical variable except binary variable.

To compare the overall normalized mean for each variable with a given cluster. You can click the slide of the given cluster and refresh the graph.



We can see that the normalized means for most dimensions in cluster 5 is much lower than the overall normalized means. Some characteristics for cluster 5 are as follows:

- (1) Customers in cluster 5 have lower marriage rate (“marriage = 0” means not married)
- (2) They have higher income than average
- (3) Most of them live in climate zone 20 and very few of them are from climate zone 10 or 30.
- (4) They have a higher rate of home ownership (“ownhome=0” mean renter)
- (5) They are much younger than the average
- (6) Most of them are male

Observing the **Variables** tab, we can see the most important variable in clustering is a nominal scale variable climate zone. The next two important variables are credit score and age. And the least important variables are married status and home ownership indicator. K-Means clustering method typically put higher weight on non-numerical variables because non-numerical variables need to be transformed and rescaled before the least square distance can be calculated. To avoid the problem that the non-numerical value variables to dominate the clustering, miners should not use categorical variables to cluster cases.

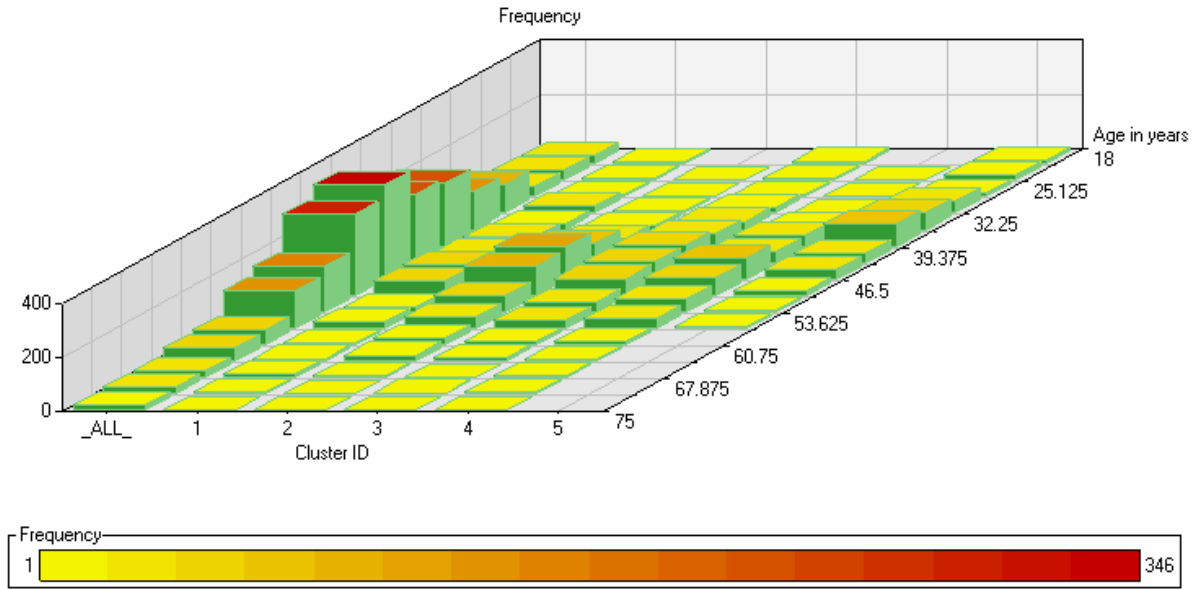
Other available utilities in Clustering Browser are

- Cluster Distance Plot and Table:

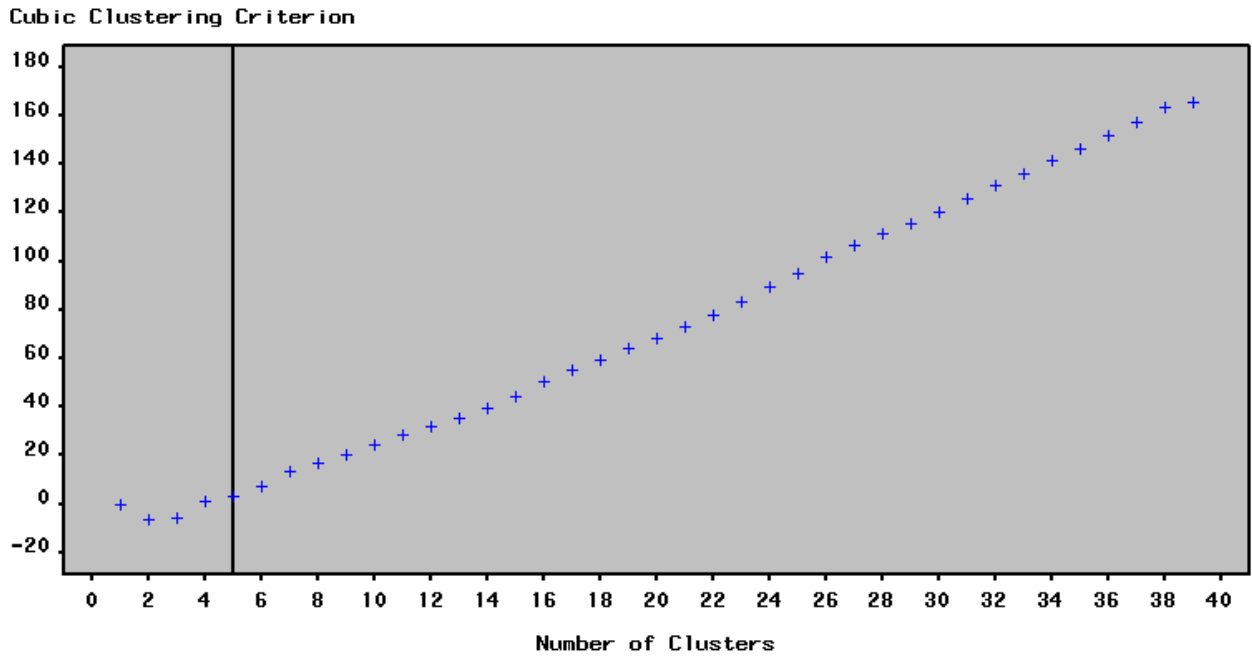
CLUSTER	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
1	0	21.07545	4.22362	20.70423	15.86699
2	21.07545	0	20.10392	41.19896	31.23802
3	4.22362	20.10392	0	22.08002	19.00479
4	20.70423	41.19896	22.08002	0	24.16746
5	15.86699	31.23802	19.00479	24.16746	0

- Profile Tree and Profile Plot for categorical variables and continuous variables:

The following profile tree shows that the customers in cluster 5 are much younger than the other clusters.



- CCC (Cubic Clustering Criterion Plot):



INSIGHT node: We can use INSIGHT node to look at the relationship among discrete variables. MASAIC is designed to study the relationship among categorical variables and box plot can be used to within cluster distribution for each categorical variable.

After study these graphs and statistics, we can conclude that

- Cluster 1: People live in climate zone 10
- Cluster 2: Married male in climate zone 20

- Cluster 3: Homeowners in climate zone 30
- Cluster 4: Female home owners in climate zone 20
- Cluster 5: Unmarried male in climate zone 20

The usefulness of these cluster definition depends on whether the business can develop business strategies to deal with customers in each cluster effectively.

Section 3.7 Case Study 3: Correlations among Physical Variables

Variable Clustering (PROC VARCLUS)

The following **data** are correlations among eight physical variables as given by Harman (1976). The first PROC VARCLUS run clusters on the basis of principal components, the second run clusters on the basis of centroid components. The third analysis is hierarchical, and the TREE procedure is used to display a tree diagram. The results of the analyses follow.

```

data phys8 (type=corr);
  title 'Eight Physical Measurements on 305 School Girls';
  title2 'Harman: Modern Factor Analysis, 3rd Ed, p22';
  label height='Height'
        arm_span='Arm Span'
        forearm='Length of Forearm'
        low_leg='Length of Lower Leg'
        weight='Weight'
        bit_diam='Bitrochanteric Diameter'
        girth='Chest Girth'
        width='Chest Width';
  input _name_ $ height arm_span forearm low_leg weight bit_diam
girth width ;
  _type_='corr';
  datalines;
height      1.0      .846      .805      .859      .473      .398      .301      .382
arm_span    .846     1.0      .881      .826      .376      .326      .277      .415
forearm     .805     .881     1.0      .801      .380      .319      .237      .345
low_leg     .859     .826     .801     1.0      .436      .329      .327      .365
weight      .473     .376     .380     .436     1.0      .762      .730      .629
bit_diam    .398     .326     .319     .329     .762     1.0      .583      .577
girth       .301     .277     .237     .327     .730     .583     1.0      .539
width       .382     .415     .345     .365     .629     .577     .539     1.0
;

proc print data=phys8; run;
proc varclus data=phys8; run;
proc varclus data=phys8 centroid; run;

```

```

proc varclus data=phys8 maxc=8 summary outtree=tree;
run;

goptions ftext=swiss;
axis2 label=(justify=left);
axis1 order=(0.5 to 1.0 by 0.1);
proc tree horizontal vaxis=axis2 haxis=axis1 lines=(width=2);
    height _propor_;
    id _label_;
run;

```

The PROC VARCLUS statement invokes the procedure. By default, PROC VARCLUS clusters on the basis of principal components.

Section 3.8 Case Study 3: Variable Clustering for Logistic Regression

The following SAS code can be used to find variable clusters as one data preparation steps for logistic regression analysis. Interested students can consultant data preparation course to see the details.

```

%let inputs=ACCTAGE dda DDABAL DEP DEPAMT CASHBK CHECKS
DIRDEP NSF NSFAMT PHONE TELLER ATM ATMAMT POS POSAMT CD
CDBAL IRA IRABAL LOC LOCBAL INV INVBAL ILS ILSBAL MM
MMBAL MMCRED MTG MTGBAL SAV SAVBAL CC CCBAL CCPURC SDB INCOME HMOWN
LORES HMVAL AGE CRSCORE MOVED INAREA;
%let misind=miacctage miphone mipos miposamt miinv miinval micc miccbal
miccpurc miincome mihmown milores mihmval miage micrscore;
ods listing close;
ods output clusterquality=summary rsquare(match all)=clusters simple;
proc varclus data=lecture.develop maxeigen=.7 outtree=fortree short hi;
var &inputs &misind;
run;

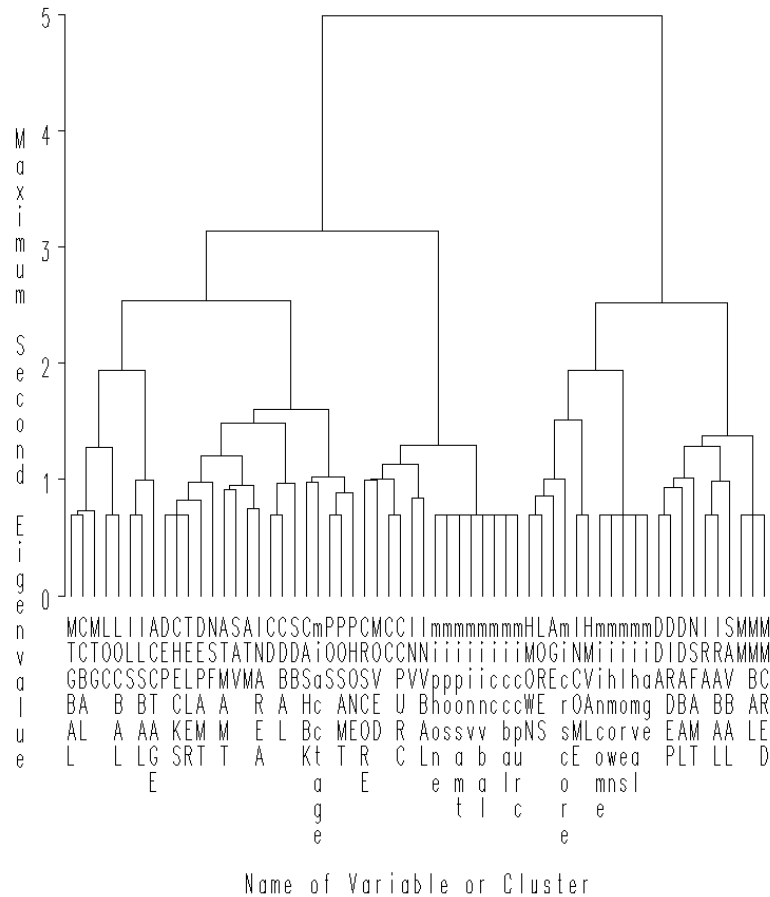
goptions ftext=swiss;
axis2 label=(justify=left);
axis1 order=(0.05 to 1 by 0.1);
proc tree data=fortree horizontal vaxis=axis2 haxis=axis1
lines=(width=2);
    height _propor_;
run;

ods trace off;

```


41	-	Cluster 16	INVBAL	1.0000	0.0259	0.0000	Investment Balance
42	-	Cluster 17	CASHBK	1.0000	0.0058	0.0000	Number Cash Back
43	-	Cluster 18	NSFAMT	1.0000	0.2667	0.0000	Amount NSF
44	-	Cluster 19	CRSCORE	1.0000	0.1985	0.0000	Credit Score
45	-	Cluster 20	microscore	1.0000	0.0206	0.0000	
46	-	Cluster 21	ACCTAGE	1.0000	0.0219	0.0000	Age of Oldest Account
47	-	Cluster 22	MOVED	1.0000	0.0016	0.0000	Recent Address Change
48	-	Cluster 23	SAVBAL	1.0000	0.0640	0.0000	Saving Balance
49	-	Cluster 24	NSF	1.0000	0.2667	0.0000	Number Insufficient Fund
50	-	Cluster 25	miacctage	1.0000	0.0060	0.0000	
51	-	Cluster 26	SDB	1.0000	0.0123	0.0000	Safety Deposit Box
52	-	Cluster 27	INAREA	1.0000	0.0851	0.0000	Local Address
53	-	Cluster 28	DDABAL	1.0000	0.1238	0.0000	Checking Balance
54	-	Cluster 29	ATMAMT	1.0000	0.0490	0.0000	ATM Withdrawal Amount
55	-	Cluster 30	PHONE	1.0000	0.0860	0.0000	Number Telephone Banking
56	-	Cluster 31	AGE	1.0000	0.1985	0.0000	Age
57	-	Cluster 32	INV	1.0000	0.0259	0.0000	Investment
58	-	Cluster 33	DEPAMT	1.0000	0.1238	0.0000	Checking Amount Deposited
59	-	Cluster 34	ATM	1.0000	0.1050	0.0000	ATM
60	-	Cluster 35	MTG	1.0000	0.1692	0.0000	Mortgage

Cluster Tree:



Appendix 3.1 Data Used in Lecture 3 Section 3.6

The data set, CLUSEXAM, has 10,000 observations and 9 variables from a catalog company. They periodically purchase demographical information from outside source. They want to use this data set to segment their potential customers into several segments. For each segment, they will conduct a testing campaign to learn the potential profit for their new products. The output from PROC CONTENTS is as follows:

```

Data Set Name: LECDATA.CLUSEXAM              Observations:      10000
Member Type:  DATA                          Variables:         9
Engine:       V8                              Indexes:          0
Created:      4:33 Saturday, July 21, 2001   Observation Length: 1064
Last Modified: 4:33 Saturday, July 21, 2001 Deleted Observations: 0
Protection:                                     Compressed:       NO
Data Set Type:                               Sorted:          NO
Label:

-----Engine/Host Dependent Information-----

Data Set Page Size:      16384
Number of Data Set Pages: 667
First Data Page:        1
Max Obs per Page:       15
Obs in First Data Page: 13
Number of Data Set Repairs: 0
File Name:               C:\Morgan\LectureData\clusexam.sas7bdat
Release Created:         8.0101M0
Host Created:            WIN_PRO

-----Alphabetic List of Variables and Attributes-----

#  Variable  Type  Len  Pos  Format  Informat  Label
-----
2  AGE       Num   8    0
9  CLIMATE   Char 255  805  $255.  $255.    Climate Code: 10, 20, 30
6  FICO      Num   8    24
1  ID        Char 255  40   $255.  $255.    Identification Number
3  INCOME    Num   8     8
8  LOC       Char 255  550  $255.  $255.    Location of Residence (A-H)
5  MARRIED   Num   8    16
7  OWNHOME   Num   8    32
4  SEX       Char 255  295  $255.  $255.    F=Female, M=Male
    
```

Appendix 3.2 Data Used in Lecture 3 Section 3.7

```

Data Set Page Size:      16384
Number of Data Set Pages: 1009
First Data Page:       1
Max Obs per Page:      32
Obs in First Data Page: 16
Number of Data Set Repairs: 0
File Name:             C:\Morgan\LectureData\imputed.sas7bdat
Release Created:       8.0101M0
Host Created:         WIN_PRO

```

-----Alphabetic List of Variables and Attributes-----

#	Variable	Type	Len	Pos	Format	Label
1	ACCTAGE	Num	8	0		Age of Oldest Account
42	AGE	Num	8	328		Age
15	ATM	Num	8	112	YESNOFMT.	ATM
16	ATMAMT	Num	8	120		ATM Withdrawal Amount
47	BRANCH	Char	8	488		Branch of Bank
6	CASHBK	Num	8	40		Number Cash Back
34	CC	Num	8	264	YESNOFMT.	Credit Card
35	CCBAL	Num	8	272		Credit Card Balance
36	CCPURC	Num	8	280		Credit Card Purchases
19	CD	Num	8	144	YESNOFMT.	Certificate of Deposit
20	CDBAL	Num	8	152		CD Balance
7	CHECKS	Num	8	48		Number of Checks
43	CRSCORE	Num	8	336		Credit Score
2	DDA	Num	8	8	YESNOFMT.	Checking Account
3	DDABAL	Num	8	16		Checking Balance
4	DEP	Num	8	24		Checking Deposits
5	DEPAMT	Num	8	32		Checking Amount Deposited
8	DIRDEP	Num	8	56	YESNOFMT.	Direct Deposit
39	HMOWN	Num	8	304	YESNOFMT.	Owens Home
41	HMVAL	Num	8	320		Home Value
27	ILS	Num	8	208	YESNOFMT.	Installment Loan
28	ILSBAL	Num	8	216		Loan Balance
45	INAREA	Num	8	352	YESNOFMT.	Local Address
38	INCOME	Num	8	296		Income
46	INS	Num	8	360	YESNOFMT.	Insurance Product
25	INV	Num	8	192	YESNOFMT.	Investment
26	INVBAL	Num	8	200		Investment Balance
21	IRA	Num	8	160	YESNOFMT.	Retirement Account
22	IRABAL	Num	8	168		IRA Balance
23	LOC	Num	8	176	YESNOFMT.	Line of Credit
24	LOCBAL	Num	8	184		Line of Credit Balance
40	LORES	Num	8	312		Length of Residence
29	MM	Num	8	224	YESNOFMT.	Money Market
30	MMBAL	Num	8	232		Money Market Balance
31	MMCRED	Num	8	240		Money Market Credits
44	MOVED	Num	8	344	YESNOFMT.	Recent Address Change
32	MTG	Num	8	248	YESNOFMT.	Mortgage

33	MTGBAL	Num	8	256		Mortgage Balance
9	NSF	Num	8	64	YESNOFMT.	Number Insufficient Fund
10	NSFAMT	Num	8	72		Amount NSF
11	PHONE	Num	8	80		Number Telephone Banking
17	POS	Num	8	128		Number Point of Sale
18	POSAMT	Num	8	136		Amount Point of Sale
48	RES	Char	8	496	\$RESFMT.	Area Classification
13	SAV	Num	8	96	YESNOFMT.	Saving Account
14	SAVBAL	Num	8	104		Saving Balance
37	SDB	Num	8	288	YESNOFMT.	Safety Deposit Box
12	TELLER	Num	8	88		Teller Visits
49	miacctage	Num	8	368		
58	miage	Num	8	440		
61	micc	Num	8	464		
54	miccbal	Num	8	408		
62	miccpurc	Num	8	472		
59	micrscore	Num	8	448		
63	mihmown	Num	8	480		
57	mihmval	Num	8	432		
55	miincome	Num	8	416		
60	miinv	Num	8	456		
53	miinvbal	Num	8	400		
56	milores	Num	8	424		
50	miphone	Num	8	376		
51	mipos	Num	8	384		
52	miposamt	Num	8	392		

Appendix 3.3 Cubic Cluster Criterion

The best way to use the CCC is to plot its value against the number of clusters, ranging from one cluster up to about one-tenth the number of observations. The CCC may not behave well if the average number of observations per cluster is less than ten. The following guidelines should be used for interpreting the CCC:

- Peaks on the plot with the CCC greater than 2 or 3 indicate good clusterings.
- Peaks with the CCC between 0 and 2 indicate possible clusters but should be interpreted cautiously.
- There may be several peaks if the data has a hierarchical structure.
- Very distinct nonhierarchical spherical clusters usually show a sharp rise before the peak followed by a gradual decline.
- Very distinct nonhierarchical elliptical clusters often show a sharp rise to the correct number of clusters followed by a further gradual increase and eventually a gradual decline.

- If all values of the CCC are negative and decreasing for two or more clusters, the distribution is probably unimodal or long-tailed.
- Very negative values of the CCC, say, -30, may be due to outliers. Outliers generally should be removed before clustering.
- If the CCC increases continually as the number of clusters increases, the distribution may be grainy or the data may have been excessively rounded or recorded with just a few digits.

A final and very important warning: neither the CCC nor R^2 is an appropriate criterion for clusters that are highly elongated or irregularly shaped. If you do not have prior substantive reasons for expecting compact clusters, use a nonparametric clustering method such as Wong and Lane's (1983) rather than Ward's method or k-means.

Appendix 3.4 References for Lecture 3

David Hand, Heikki Mannila, and Smyth (2001) Chapter 9 of Principles of Data Mining, Massachusetts Institute of Technology.

Michael J. A. Berry and Linoff S. Gordon (2000) Chapter 5 of Mastering Data Mining, John Wiley & Sons, Inc.: New York, New York.

Richard A. Jognson and Dean W. Wichern (1982) Chapter 11 of Applied Multivariate Statistical Analysis, Prentice-Hall, Inc.: Englewood Cliffs, New Jersey.

Rud, O. P. (2001), "Data Mining Cook Book", John Wiley & Sons, Inc.: New York, N.Y.

SAS Institute (2000) Enterprise Miner: Applying Data Mining Techniques Course Notes, SAS Institute: Cary, N.C.

SAS Institute (2000) Predictive Modeling Using Logistic Regression Course Notes, SAS Institute: Cary, N.C.