

TwoStep Cluster Analysis¹

The SPSS TwoStep cluster method is a scalable cluster analysis algorithm designed to handle very large data sets. It can handle both continuous and categorical variables or attributes. It requires only one data pass. It has two steps 1) pre-cluster the cases (or records) into many small sub-clusters; 2) cluster the sub-clusters resulting from pre-cluster step into the desired number of clusters. It can also automatically select the number of clusters.

Notation

K^A	Total number of continuous variables used in the procedure.
K^B	Total number of categorical variables used in the procedure.
L_k	Number of categories for the k -th categorical variable.
R_k	The range of the k -th continuous variable.
N	Number of data records in total.
N_k	Number of data records in cluster k .
$\hat{\sigma}_k^2$	The estimated variance of the k -th continuous variable in whole data.
$\hat{\sigma}_{jk}^2$	The estimated variance of the k -th continuous variable in cluster j .
N_{jkl}	Number of data records in cluster j whose k -th categorical variable takes the l -th category.
$d(j, s)$	Distance between clusters j and s .
$< j, s >$	Index that represents the cluster formed by combining clusters j and s .

Two Step Clustering Procedure

Step 1: Pre-cluster

The pre-cluster step uses a sequential clustering approach. It scans the data records one by one and decides if the current record should be merged with the previously formed clusters or starts a new cluster based on the distance criterion.

The procedure is implemented by constructing a modified cluster feature (CF) tree. The CF-tree consists of levels of nodes, and each node contains a number of entries. A leaf entry (an entry in the leaf node) represents a final sub-cluster. The non-leaf nodes and their entries are used to guide a new record quickly into a correct leaf node. Each entry is characterized by its CF that consists of the entry's number of records, mean and variance of each continuous variable, and counts for each category of each categorical variable. For each successive

¹ This algorithm applies to SPSS 11.5 and later releases.

record, starting from the root node, it is recursively guided by the closest entry in the node to find the closest child node, and descends along the CF-tree. Upon reaching a leaf node, it finds the closest leaf entry in the leaf node. If the record is within a threshold distance of the closest leaf entry, it is absorbed into the leaf entry and the CF of that leaf entry is updated. Otherwise it starts its own leaf entry in the leaf node. If there is no space in the leaf node to create a new leaf entry, the leaf node is split into two. The entries in the original leaf node are divided into two groups using the farthest pair as seeds, and redistributing the remaining entries based on the closeness criterion. If the CF-tree grows beyond allowed maximum size, the CF-tree is rebuilt based on the existing CF-tree by increasing the threshold distance criterion. The rebuilt CF-tree is smaller and hence has space for new input records. This process continues until a complete data pass is finished. For details of CF-tree construction, see BIRCH by Zhang et al (1996).

All records falling in the same entry can be collectively represented by the entry's CF. When a new record is added to an entry, the new CF can be computed from this new record and the old CF without knowing the individual records in the entry. These properties of CF make it possible to maintain only the entry CFs, rather than the sets of individual records. Hence the CF-tree is much smaller and more able to be stored in main memory.

Note that the CF-tree may depend on the input order of the cases or records. To minimize the order effect, randomly order the cases.

Outlier-Handling Option

An optional outlier-handling step is implemented in the algorithm in the process of building the CF-tree. Outliers are considered as data records that do not fit well into any cluster. We consider data records in a leaf entry as outliers if the number of records in the entry is less than a certain fraction (25% by default) of the size of the largest leaf entry in the CF-tree. Before rebuilding the CF-tree, the procedure checks for potential outliers and sets them aside. After rebuilding the CF-tree, the procedure checks to see if these outliers can fit in without increasing the tree size. At the end of CF-tree building, small entries that cannot fit in are outliers.

Step 2: Cluster

The cluster step takes sub-clusters (non-outlier sub-clusters if outlier handling is used) resulting from the pre-cluster step as input and then groups them into the desired number of clusters. Since the number of sub-clusters is much less than the number of original records, the traditional clustering methods can be used effectively. SPSS uses the agglomerative hierarchical clustering method. A primary reason is that it works well with the auto-cluster method (see the section on auto-clustering below).

Accuracy

In general, the larger the number of sub-clusters produced by the pre-cluster step, the more accurate the final result is. However, too many sub-clusters will slow down the clustering during the second step. The maximum number of sub-clusters should be carefully chosen so that it is large enough to produce accurate results and small enough not to slow down the second step clustering.

Number of clusters: auto-cluster

How many clusters are there? The answer depends on the data set at hand. A characteristic of hierarchical clustering is that it produces a sequence of partitions in one run: 1, 2, 3, ... clusters. A K-means algorithm would need to run multiple times (one for each specified number of clusters) in order to generate the sequence. To determine the number of clusters automatically, SPSS developed a two-step procedure that works well with the hierarchical clustering method. In the first step, the BIC or AIC for each number of clusters within a specified range is calculated and used to find the initial estimate for the number of clusters. In the second step, the initial estimate is refined by finding the largest increase in distance between the two closest clusters in each hierarchical clustering stage.

The BIC and AIC for J clusters are defined as

$$BIC(J) = -2 \sum_{j=1}^J \xi_j + m_j \log(N),$$

$$AIC(J) = -2 \sum_{j=1}^J \xi_j + 2m_j.$$

Where

$$m_j = J \{ 2K^A + \sum_{k=1}^{K^B} (L_k - 1) \},$$

and is defined in equation (2) below.

Distance measure

A distance measure is needed in both the pre-cluster and cluster steps. Two distance measures are available.

Log-Likelihood distance

The log-likelihood distance measure can handle both continuous and categorical variables. It is a probability based distance. The distance between two clusters is related to the decrease in log-likelihood as they are combined into one cluster. In calculating log-likelihood, normal distributions for continuous variables and multinomial distributions for categorical variables are assumed. It is also assumed that the variables are independent of each other, and so are the cases. The distance between clusters j and s is defined as

$$d(j, s) = \xi_j + \xi_s - \xi_{\langle j, s \rangle}, \quad (1)$$

where

$$\xi_v = -N_v \left(\sum_{k=1}^{K^A} \frac{1}{2} \log(\hat{\sigma}_k^2 + \hat{\sigma}_{vk}^2) + \sum_{k=1}^{K^B} \hat{E}_{vk} \right), \quad (2)$$

$$\hat{E}_{vk} = - \sum_{l=1}^{L_k} \frac{N_{vkl}}{N_v} \log \frac{N_{vkl}}{N_v}. \quad (3)$$

If $\hat{\sigma}_k^2$ is ignored in equation (2), the distance between clusters j and s would be exactly the decrease in log-likelihood when the two clusters are combined. The $\hat{\sigma}_k^2$ term is added to solve the problem caused by $\hat{\sigma}_{vk}^2 = 0$, which would result in the natural logarithm being undefined (this would occur, for example, when a cluster only has one case).

Euclidean distance

This distance measure can only be applied if all variables are continuous. The Euclidean distance between two points is clearly defined. The distance between two clusters is here defined by the Euclidean distance between the two cluster centers. A cluster center is defined as the vector of cluster means of each variable.

Cluster Membership Assignment

Without outlier-handling

Assign a record to the closest cluster according to the distance measure.

With outlier-handling

Log-likelihood distance

Assume outliers or noises follow a uniform distribution. Calculate both the log-likelihood resulting from assigning a record to a noise cluster and that resulting from assigning it to the closest non-noise cluster. The record is then assigned to the cluster which leads to the larger log-likelihood. This is equivalent to assigning a record to its closest non-noise cluster if the distance between them is smaller than a critical value $C = \log(V)$, where

$$V = \prod_k R_k \prod_m L_m. \text{ Otherwise, designate it as an outlier.}$$

Euclidean distance

Assign a record to its closest non-noise cluster if the Euclidean distance between them is

smaller than a critical value $C = 2\sqrt{\sum_{k=1}^{K^A} \hat{\sigma}_{kl}^2 / K^A}$. Otherwise, designate it as an outlier.

Missing Values

No missing values are allowed. Cases with missing values are deleted on a LISTWISE basis.

References

Zhang, T., Ramakrishnan, R., and Livny M. (1996). BIRCH: An Efficient Data Clustering Method for Very Large Databases. *Proceedings of the ACM SIGMOD Conference on Management of Data*, p. 103-114, Montreal, Canada.

Chiu, T., Fang, D., Chen, J., Wang, Y., and Jeris, C. (2001). A Robust and Scalable Clustering Algorithm for Mixed Type Attributes in Large Database Environment. *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining*, 263.