5.2 Optimum number of Segmentation for initial iteration purpose.

Some methods of cluster subjects based on similarity on variables of interest (e.g., attribute importance), while other maximize the (within-segment) ability of the store images attributes to predict overall store image. The former type of method does not necessarily lead to cluster whose store –image attribute importance best explain the overall evaluation of the stores of each individual in the sample. One may obtain a good cluster solution (in terms of the homogeneity of estimated-image-attribute weights without any appreciable increase in predictive power over the unsegmented model, as was indeed found in the context of store image as previous studies indicates. It has been argued that predictive fit of the estimated store image functions should be maximized, as it is a key measure for evaluating market segmentation results and for developing a marketing strategy.

In this research it has been desired to utilize the following (Xie and Beni's function, the compactness and separation validity's,' function, the Partition Index SC, Dunn's Index (DI) and Alternative Dunn Index (ADI)) algorithms that can efficiently determine a reasonable number of clusters/segments to return from any non- hierarchical clustering/segmentation algorithm. In order to identify the correct number of clusters to return from a non-hierarchical clustering/segmentation algorithm, this research utilizes the above mentioned cluster validity function.

Suppose we don't have a clear idea about how many clusters there should be for a given set of data. Subtractive clustering proposed by S.L.Chiu is a fast, one-pass algorithm for estimating the number of clusters and the cluster centers in a set of data, in which data points are considered as the candidates for cluster centers. By using this method, the computation is simply proportional to the number of data points and independent of the dimensions of the problems under consideration.

Consider a collection of *n* data points $x_1, \ldots, x_n$ in a M-dimensional space. Without loss of generality, the data points are assumed to have been normalized within a hypercube. Since each data point is a candidate for cluster centers, a density measure at data point x_i is defined as

$$D_i = \sum_{j-1}^{n} \exp\left(- \frac{\text{II}x_i - x_j\text{II}^2}{(r_a / 2)^2} \right)$$

Where r_a is a positive constant. Hence, a data point will have a high density value if it has many neighboring data points. The radius r_a defines a neighborhood; data points outside this radius contribute only slightly to the density measure.

After the density measure of each data point has been calculated, the data point with the highest density measure is selected as the first cluster center. Let x_{c1} be the point selected and D_{C1} its density measure. Next, the density measure for each data point x_i is revised by the formula

$$D_i = D_i - D_{C1} \exp\left(- \frac{\text{II}x_i - x_{c1}\text{II}^2}{(r_b / 2)^2} \right)$$

Where r_b is a positive constant. Therefore, the data points near the first cluster center x_{c1} will have significantly reduced density measures, thereby making the points unlikely to be selected as the next cluster center. The constant r_b defines a neighborhood that has measurable reductions in density measure. The constant r_b is normally larger than r_a to prevent closely spaced cluster centers; generally r_b is equal to $1.5 r_a$. (S.L.Chiu 1994) After the density measure for each data point is revised, the next cluster center x_{c2} is selected and all of the density measures for data points are revised again. This process is repeated until a sufficient number of cluster center are generated. The cluster estimates obtained from the subtractive cluster function can be used to initialize iterative optimization-based clustering methods (FCM) and model identification methods (like ANFIS).

The "Subtractive clustering" - description method.

We performed Subtractive cluster in MatLab® by using the subclust function.

The Syntax is [C,S] = subclust(X,radii,xBounds,options)

The matrix X contains the data to be clustered; each row of X is a data point. The variable radii is a vector of entries between 0 and 1 that specifies a cluster center's range of influence in each of the data dimensions, assuming the data falls within a unit hyperbox. Small radii values generally result in finding a few large clusters. Good values for radii are usually between 0.2 and 0.5. "xBounds" is a 2-by-N matrix that specifies how to map the data in X into a unit hyperbox, where N is the data dimension. This argument is optional if X is already normalized. The first

row contains the minimum axis range values and the second row contains the maximum axis range values for scaling the data in each dimension.

If “xBounds” is an empty matrix or not provided, then “xBounds” defaults to the minimum and maximum data values found in each data dimension.

The options vector can be used for specifying clustering algorithm parameters to override the default values. These components of the vector options are specified as follows:

Options(1) = quashFactor: This is the factor used to multiply the radii values that determine the neighborhood of a cluster center, so as to quash the potential for outlying points to be considered as part of that cluster. (we have set the quash factor as 14.25).

Options(2) = acceptRatio: This sets the potential, as a fraction of the potential of the first cluster center, above which another data point will be accepted as a cluster center. (default: 0.5)

Options(3) = rejectRatio: This sets the potential, as a fraction of the potential of the first cluster center, below which a data point will be rejected as a cluster center. (default: 0.15)

Options(4) = verbose: If this term is not zero, then progress information will be printed as the clustering process proceeds. (default: 0)

The function returns the cluster centers in the matrix C; each row of C contains the position of a cluster center. The returned S vector contains the sigma values that specify the range of influence of a cluster center in each of the data dimensions. All cluster centers share the same set of sigma values.

We have implemented the subclust function using the following arguments

[C,S]=subclust(X,0.5,[],options);

options = [14.25 0.5 0.15 0];

Thus number of cluster specified is four.

Market researchers have discussed on store image based market segmentation in the perspective of various partitioning and clustering methods, but such studies have toiled with unsupervised clustering approaches. But, no studies have identified the integration of fuzzy c-means with subtractive clustering for obtaining store image based market segmentation. This research attempts to make use of the above mentioned cluster validity function for this research, The above mentioned cluster validity functions are usually used for image processing analysis where the data is images in nature. But such validity functions are rarely used in social science research. So, utilizing such validity function in social science research and finding its merit and value in social science research is reasonable value addition to the literature.

Most of the research often uses clustering analysis as a tool for market segmentation. In this research, the k-means partitioning method is used to segment the customers based on store image attributes. Such non-hierarchical method of clustering algorithm always tries to find the best fit for a fixed number of clusters and the parameterized cluster shapes. However this does not mean that even the best fit is meaningful at all. Either the number of clusters might be wrong or the cluster shapes might not correspond to the groups in the data. To overcome such

problems, this research utilises the (Xie and Beni's 1991) cluster validity function which is to solve the cluster membership,c', and to measure the effectiveness of cluster. Xie and Beni's validity index aims to quantify the ratio of the total variation within clusters and separation of clusters. The Xie and Beni's function is

$$XB(c) = \frac{\sum_{i=1}^{c}\sum_{j=1}^{N}(\mu_{ij})^{m}\|x_j - v_i\|^2}{N\ min_{i,k}\|x_j - v_i\|^2}$$

and $XB(c)$ is the optimal number of cluster which should minimize the value of the index. In continuation to further validate, the compactness and separation validity 's,' function is used. The validity function is

$$s = \frac{\sum_{i=1}^{c}\sum_{k=1}^{n}(\mu_{ik}{}^{2})d^2\,(v_i - x_i)}{n\ min_{ij}d^2(v_i - x_i)}$$

The smaller the value of s', the better the compactness and separation between the clustering groups of in-cluster samples. The goal should therefore be to minimize the value of s. At the same time, this approach also allows us to determine the minimal objective function of clustering algorithm. Till further to authenticate, the Partition Index SC is used, and this is the ratio of the sum of compactness and separation of the clusters.

$$SC = \sum_{i=1}^{c}\frac{\sum_{j=1}^{N}(\mu_{ij})^{m}\|x_j - v_i\|^2}{N\sum_{k=1}^{c}\|x_j - v_i\|^2}$$

SC is useful when comparing different partitions having equal number clusters. A lower value of SC indicates a better partition. Utilisation of mentioned validity function is well explored in soft computational problems and rarely used in

business and social science studies. The results of validity index are shown in the Figure 5.1. The analysis found the optimum number of cluster is four and thus the value obtained from such validity function can be used to initialize iterative optimisation-based clustering methods and model identification methods. Further to substantiate, the data set was validated with two more Indexes. They are Dunn's Index (DI) and Alternative Dunn Index (ADI). Dunn's Index (DI) is originally proposed to use at the identification of "compact and well separated clusters". So the result of the

clustering has to be recalculated as it was a hard partition algorithm.

$$\text{DI}\ (\text{C}) = min_{i\epsilon c}\{min_{j\epsilon c, i\neq j}\left\{\frac{min_{x\epsilon}c_i, y\epsilon c_j d(x,y)}{max_{k\epsilon c}\{max_{x,y\epsilon c} d(x.y)\}}\right\}\}$$

The main drawback of Dunn's index is computational since calculating becomes computationally very expansive as *c* and *N* increase. Where as the alternative Dunn Index (ADI), which the aim of modifying the original Dunn's index was that the calculation becomes more simple, when the dissimilarity function between two clusters $(min_{x\epsilon c_{i,y\epsilon c_j}} d(x.y)$ is rated in value from beneath by the triangle-nonequality:

$$d(x,y) \geq \mid d(y,v_j) - d(x,v_j) \mid$$

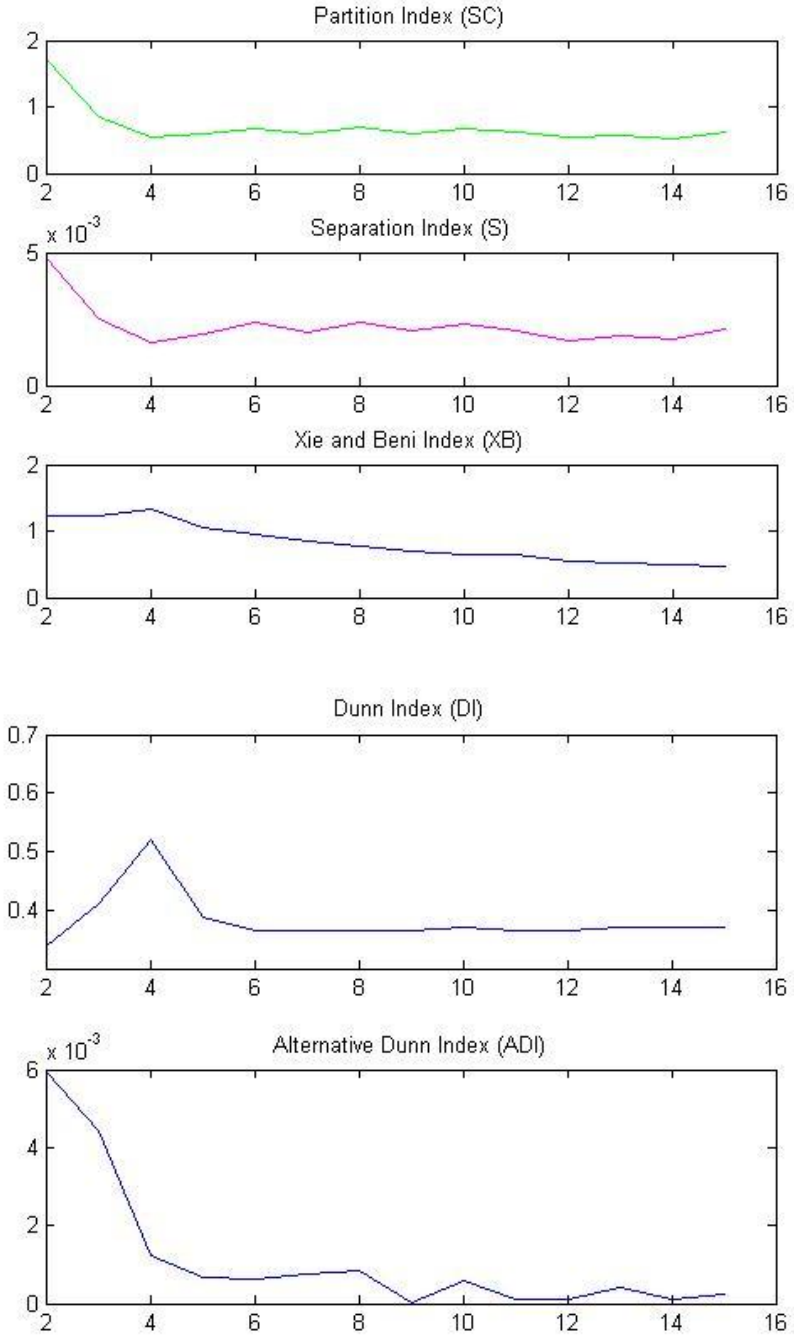
Where v_j is the cluster center of the j-th cluster.

$$ADI\ (C) = min\ _{i\epsilon c}\{min_{j\epsilon c, i\neq ji}\left\{\frac{min_{xi\epsilon Ci, xj\epsilon Cj} \mid d(y,v_j) - d(x_i,v_j) \mid}{max_{k\epsilon c}\{max_{x,y\epsilon c} d(x.y)\}}\right\}\}$$

Note, that the only difference of SC, S and XB is the approach of the separation of clusters. In the case of overlapped clusters the value of DI and ADI are not

really reliable because of re-partitioning the results with the hard partition method. By comparing all the cluster validity methods, the optimum number of cluster is to be four. All the indexes indicate that the line of validity measure cut and gets turn at the point of four (See Figure 5.1).

Figure: 5.1. Cluster Validity Index



Axis - X indicates Number of Cluster, Y indicates Validity Measure.
Source: Secondary Data Analysis.