ESSAYS IN RETAILING AND DISTRIBUTION

by

Tieshan Li

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate Studies

(Business Administration)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

February 2009

© Tieshan Li, 2009

ABSTRACT

Although aggregating retail outlets into retail districts is an important academic and practical issue in marketing and retailing, only limited academic work has been done on this problem. The growing availability of detailed location data through Geographic Information Systems makes this a particularly timely problem. Cluster analysis is a sound and well established approach for reducing data dimensionality. However, the existing clustering approaches do not handle the complicated geospatial structure that is typical of retailing data well, largely due to the high variation in observation density. One problem is that the "epsilon radius," a measure of how close stores need to be to each other in order to be classified as belonging to the same cluster, is assumed to be constant in methods such as density-based clustering. However, this turns out not to be a good assumption in practice. In addition existing methods of judging the quality of a clustering solution, so called cluster validation methods, do not provide sound guidance as to the best clustering solution for the type of retailing data we study. Consequently, we propose a new two-step clustering approach in which Variable Epsilon Spatial Density Clustering (VESDC) is developed, and a new clustering validation measure, the *CpSp* index, also is introduced.

VESDC effectively clusters data by systematically adjusting the epsilon radius to adapt to the local market environment. In particular, using the logistic transformation function, we propose a model in which the epsilon radius is determined by the population density in a small area. *CpSp*, which is scaled from 0 to 1, balances the compactness and separation of a proposed clustering solution. Extensive testing demonstrated that *CpSp* performed well as a cluster validation method.

We tested VESDC's performance on synthetic data. The underlying pre-specified data patterns were accurately recovered. Existing methods were not as successful in these tests. We then applied the two-step approach to Greater Victoria since Greater Victoria is a typical metropolitan city with large variation in store density.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

This thesis would not have been possible without the help, support and patience of my co-supervisors, Professor Daniel S. Putler and Professor Charles B. Weinberg. I am very grateful to both of them for their guidance and advice. Working with them during the period would be the most cherished experience in my life and would benefit me forever. Not only their broad knowledge but also their generous personalities have a big impact on me. I really appreciate what they have done for me from everlasting to everlasting. Thank you so much from my heart, Dan and Chuck!

I would also like to thank Professor Robert W. Helsey for serving on my thesis committee and for his great support.

Finally, I would like to especially thank my wife and my son for their love, supports, patience, and encouragement during the work. I want to dedicate this thesis to my mother, for having always believed in me in these years and for her love.

DEDICATION

This thesis is dedicated to my mother. Thank you for all the unconditional love, support, and encouragement that you have always given me, helping me to be the person I am today. Thank you for everything. I love you!

1. Introduction and Motivation

Recent advances in geodemographic coding have provided an impressive amount of data about the characteristics of individual commercial outlets (e.g., stores) and their locations. For example, location information for more than 16,000 retail stores in Vancouver is available. In addition, those stores can be easily and accurately located on maps using Geographic Information System (GIS) technology, making it easier to understand the marketing environment among the retailers. The distance between stores, for example, can be calculated after identifying the map locations using GIS. The availability of this considerable amount of data should potentially be useful for marketing, specifically for retailing. For example, when choosing sites for new stores and evaluating the performance of existing stores, retailers have access to extensive data on competitors, complementary stores that might build traffic in an area of interest, and other local geographic characteristics.

While local managers have always had some of this knowledge, the availability of such vast quantities of data allows for a much more systematic and comprehensive analysis than was previously possible. Recent research, such as that conducted by Ellickson and Misra (2008), demonstrate the potential usefulness of this data. However, for the data to be most useful, it is necessary to understand its structure. For example, rather than knowing the locations of each individual store when determining the attractiveness of an area for a new store location, it is more important to know whether there is an agglomeration of stores in a certain area, creating a retail center or retail district that might serve to draw traffic. In other words, when estimating the sales volume of a new store, the estimate is likely to depend on more than simply summing up the current sales of stores within a certain geographical distance of the new location. Obtaining such information from a large data set, especially location information, is challenging. For example, there are more than 16,000 retail outlets differentially distributed in the Greater Vancouver area, and reducing the data dimension quickly becomes an issue during the initial process of investigating the area further.

The challenge, then, is to provide ways to cluster data in ways that provide meaningful information to researchers and managers. In this thesis, we focus on the use of cluster analysis in marketing to understand the patterns in geographic data. As we will show, while a number of new techniques have been developed in computer science and other areas, these techniques have not yet been tested and developed for the type of geodemographic data commonly used in understanding marketing, distribution, and retailing issues. Moreover, as we will show, the existing techniques need substantial modification to be useful in such settings. In this thesis then, we will first review the techniques commonly used in marketing and then introduce the newer techniques from other literatures. Using both simulated and actual data from the Victoria, British Columbia metropolitan area, we will show why both the existing and newer techniques have difficulty when confronting geodemographic data typical of the retail sector of the economy. The greater Victoria area is a mid-size metropolitan area with a strong downtown core and a surrounding set of retail districts that vary widely in store density. We next will introduce modifications to these techniques and demonstrate their usefulness. By using actual data from Victoria, we will be able to test the robustness of our new approach to clustering data and testing the structure that our clustering approach produces.

1.1 Illustrative Settings

Ellickson and Misra (2008) provide a recent example of the use of clustering in a retail setting. They grouped supermarkets into “competing clusters” to investigate how firms react to the pricing strategies of their competitors. Since competitors who are relatively far away are unlikely to provoke strong reactions by the focal store in terms of pricing, the authors need to find a way to cluster the more than 28,000 supermarkets in the US into meaningful retail clusters. To accomplish this, Ellickson and Misra (2008) use a clustering method called K-means (described below) to group the stores into "competing" markets. This serves as the first step in identifying the actual competitors among US supermarkets, and then the pricing strategy of the focal store is studied within the relevant competition area. Reducing the dimensionality of the data by clustering the stores into different "competing" markets allows for effective investigation of pricing strategy.

The K-Means approach used by Ellickson and Misra (2008) is useful for handling large data sets. However, the number of clusters is an input parameter that must be determined before the clustering process can begin. Given the size of such a dataset, it is difficult to determine the appropriate number of clusters based on a priori reasoning. Thus the optimal number of clusters is determined by iterating through a large number of potential alternatives. As we show below, the choose of the optimal number of clusters can be highly sensitive to the criterion chosen (that is, the cluster validation method) and the structure of the clusters is very sensitive to the number of clusters that chosen to represent the data. A particular problem occurs when some stores are located in relatively isolated areas. Since K-means and other techniques used in marketing tend to force all stores into clusters, sometimes the clusters generated appear to be unreasonable. In other words, isolated supermarkets that should not belong to any competing market are identified within the competing market.

Understanding the geographic structure of an area is particularly important for store location decisions. In estimating sales potential of a proposed new outlet, it is often crucial to know whether the existing stores can be grouped together to form a retail district. By a "retail district", we mean a business area where a numbers of commercial outlets can be aggregated together to form a multifunctional center. Given GIS technology, the location of commercial outlets can be identified accurately, and the possibility of identifying retail districts has been improved significantly in recent years. As the first step toward investigating an area for a possible store location, we need to understand the local commercial structure in the area by reducing the data dimension and understanding the volume of traffic generated. For example, Vancouver's Robson Street (comprising approximately 200 stores) can attract around 80,000 visitors on a weekend, thus demonstrating the significant effect that agglomeration can have.

As noted above, the problem addressed here is how to accurately agglomerate individual stores into retail districts. Depending on the application, retail districts may consist of similar stores, as in the case of an auto mall, or different stores, as in the case of a shopping district such as Robson Street. The retail structure provides critical information on market demand and market competition that cannot be obtained otherwise. While an individual observer familiar with an area may be able to intuitively group together retail outlets into retail districts (for example, by being able to indicate which stores on Vancouver's west side fall into the Dunbar area), such an approach is neither feasible nor reliable on a wide scale basis. For example, Vancouver alone has more than 16,000 retail and service outlets, and developing reliable and consistent information on retail districts by using unaided intuition would clearly be infeasible. Therefore, a method needs to be developed that allows for structuring the data into retail districts accurately and efficiently in order to gain enough understanding about the retail structure. As discussed below, existing methods in marketing are unable to deal effectively with spatial clustering, particularly for large data sets. Methods developed in computer science and elsewhere appear to be promising, but make assumptions about the data structure that, as this thesis demonstrates, are unrealistic in the context of retail locations, and cannot handle retail district identification accurately.

1.2 Approaches to Clustering Data in Marketing

Although aggregating commercial outlets into retail districts is an important issue in marketing in general and retailing in particular for both marketing researchers and marketing managers, only limited academic work has been done on this problem. In general, there are three approaches for reducing the data dimension to form retail districts: predefined geographic areas such as those indicated by postal codes, human judgment, and formal methods of cluster analysis.

Postal codes cannot provide accurate information about retail districts. First, there are more commercial concentrations than postal areas. Second, postal areas are essentially random spatial units imposed across a complex spatial system. Therefore, they are only useful as an initial step toward understanding the general picture of an area. The use of human judgment takes advantage of GIS development and involves drawing retail districts on maps by hand (Simmons et al., 2000). The GIS can locate retail stores accurately and improve upon the innate human ability to identify patterns in visual displays. However, this is very labor intensive and cannot be updated easily if the data set is large. This labor-intensive work can be avoided by finding other computer-based or algorithmic approaches that can aggregate a large number of retail outlets into a manageable number of groups (retail districts) quickly and accurately. One reasonable solution for handling the issue is to reduce data dimensions by forming retail districts via cluster analysis. Cluster analysis, which covers a wide range of techniques, has emerged as a computer-based approach to grouping data on the basis of pre-defined criteria, and has been successful in a number of applications.

Cluster analysis is a statistical method for classification. It has been increasingly employed in the design and implementation of marketing strategy (see Table 1.1 for a summary of clustering studies in the marketing literature, based in part on Punj and Stewart, (1983)). The primary use of cluster analysis in marketing has been for market segmentation. All segmentation research, regardless of the method used, is designed to identify groups of entities, including consumers, markets, and firms, that share certain characteristics, such as attitudes, purchase behavior, geographical location, or business type. For example, based on characteristics such as demographics, transaction history, and benefits sought, consumers can be segmented by their similarity into different groups using a clustering method (Anderson et al., 1976; Schaninger et al., 1980; Sexton 1974; Lessig and Tollefson 1971; Landon 1974; Greeno et al., 1973), thereby helping marketing managers choose the proper targets. In another setting, relevant and competing markets are identified based on store features and geographic information (Sethi 1971; Day and Heeler 1971; Hooley et al., 1990; Ellickson and Misra 2008). For the firm, different types of firms are identified based on their business type, attitude towards innovation, and marketing practices (Moriarty and Venkatesan 1978; Kerin and Cron 1987; Bowen 1990; Coviello et al., 2002; Hollenstein 2003).

Cluster analysis has also been used to seek a better understanding of buyer behaviors by identifying homogeneous groups of buyers. It has been applied less frequently to this type of theory-building problem, possibly because of theorists' discomfort with a set of procedures that appear ad hoc (Punj and Stewart 1983).

Several studies use cluster analysis to improve predictions related to consumer buying decisions (Morrison and Sherman 1972; Kiel and Layton 1981; Kernan 1968; Claxton et al., 1974; Hagerty 1985; Krieger and Green 1996). Cluster analysis has also been used in the identification of potential new product opportunities. After determining the competing brands or products through cluster analysis, a firm can examine its current offerings and determine how the current or new products are positioned against other competing products (Srivastava et al., 1981; Srivastava et al., 1978).

Cluster analysis has also been employed in test market selection (Green et al., 1967). Using cluster analysis, marketers can choose markets that are similar to the larger geographic area to which the results of the test are to be applied, thereby reducing the number of test markets required.

Finally, cluster analysis has been used to develop aggregates of data that are well defined and managed more easily than individual observations. As discussed above, Ellickson and Misra (2008) clustered all the supermarkets in the United States into a manageable number of competing markets/areas to further study firms' pricing strategies.

Table 1.1 Cluster Analysis in Marketing Applications

Papers	**Purpose of research**	**Clustering method used**
Anderson, Cox, and Fulcher (1976)	To identify the determinant attributes in bank decisions and use them for segmenting commercial bank customers	Partitioning
Bowen (1990)	To identify different service types	Partitioning
Claxton, Fry, and Portis (1974)	To classify furniture and appliance buyers in terms of their information search behavior	Hierarchical
Coviello, Brodis, Danaher, and Johnston (2002)	To identify the relative emphasis on transactional and relational across firm type	Partitioning
Day and Heeler (1971)	To classify stores into similar strata	Partitioning & Hierarchical
Ellickson and Misra (2008)	To identify the competing market where the focal store is located	Partitioning
Green, Frank, and Robinson (1967)	To identify matched cities for test marketing	Hierarchical

Papers	**Purpose of research**	**Clustering method used**
Greeno, Sommers, and Kernan (1973)	To identify market segments with respect to personality variables and implicit behavior patterns	Hierarchical
Hagerty (1985)	To group customers with similar preferences to improve the predictive accuracy of conjoint analysis	Hierarchical
Hollenstein (2003)	To identify five clusters of firms with different innovation modes in Switzerland	Partitioning
Hooley, Lynch, and Shepherd (1990)	To group firms with different perspectives on the role of marketing	Partitioning
Kerin and Cron (1987)	To determine how marketing executives group their trade show programs on the selling and non-selling performance dimensions	Hierarchical
Kernan (1968)	To identify groups of people along several personality and decision behavior characteristics	Hierarchical
Kiel and Layton (1981)	To develop consumer taxonomies of search behavior by Australian new car buyers	Hierarchical
Krieger and Green (1996)	To enhance the prediction of an exogenous variable for the segmented customers	Partitioning (modified K-Means)
Landon (1974)	To identify similar groups of people using purchase intention and self-concept variables	Partitioning
Lessig and Tollefson (1971)	To identify similar groups of consumers along several buyer behavior variables	Hierarchical
Moriarty and Venkatesan (1978)	To segment educational institutions in terms of benefits sought when purchasing financial-aid Management Information System	Partitioning
Schaninger, Lessig, and Panton (1980)	To identify segments of consumers on the basis of product usage variables	Partitioning
Sethi (1971)	To classify world markets	Partitioning
Sexton (1974)	To identify homogeneous groups of families using product and brand usage data	Type not specified
This thesis	To identify retail districts based on retail store locations	Variable Epsilon Spatial Density Clustering

1.3 Summary and Outline of Thesis

While the marketing field has largely used traditional methods of clustering, new methods of cluster analysis have been developed (primarily in computer science) to address issues encountered in clustering spatial data. For example, as discussed in subsequent sections of this thesis, a density-based approach can identify outliers and is free from cluster shape restriction; a model-based approach can be used to choose the optimal multivariate model and number of clusters using BIC value (Ward's is one special case of the general model-based approach); and fuzzy clustering allows observations to belong to different clusters with different degrees of probability. These methods can partially overcome the problems with the K-Means and Ward's approaches to clustering. While the new methods provide a starting point for spatial point data clustering, they have largely been developed and tested using only simulated data. Attempts to apply these methods to actual marketing data demonstrate the need to extend the existing methods for use in practical applications. The extensions of the methods and the applications of those extensions form the basis of this thesis.

The remainder of this thesis is organized as follows. In Chapter 2, we provide a review of the existing clustering methods and cluster validation methods, and indicate potential problems in their applications to retail location data. In Chapter 3, we develop a new method for cluster validation and apply this approach to simulated data. In Chapter 4, we illustrate the performance of current approaches on the city of Victoria. In Chapter 5, we introduce a new two-step cluster approach, in which "Variable Epsilon Spatial Density Clustering (VESDC)" is developed. In Chapter 6, we show the result of the two-step approach on Greater Victoria in British Columbia. The summary and conclusions is the final chapter.

2 Literature Review

Since this thesis focuses on clustering methodology and cluster validation measurement, it will be helpful to first review the literature on cluster analysis, and then turn to a review of cluster validation measurements.

2.1 Cluster Analysis

Cluster analysis is one of the most useful tools in the data mining process for discovering groups and identifying similar observations in the underlying data. The main function of clustering is dividing a given data set into groups (clusters) such that the data points in a cluster are more similar to each other than to points in different clusters (Guha et al., 1998). Thus, the main concern in the clustering process is to reveal the organization of patterns into "sensible" groups, which allow us to discover similarities and differences, as well as to derive useful conclusions about them (Halkidi et al., 2001). In general, cluster algorithms can be broadly classified into the following types according to the method adopted to define clusters (Jain et al., 1999; Yeung et al., 2001):

- **Partitioning clustering** attempts to directly decompose the data set into a set of disjoint clusters. More specifically, it attempts to determine an integer number of partitions that optimize a certain criterion function. The criterion function may emphasize the local or global structure of the data, and its optimization is an iterative procedure.
- **Hierarchical clustering** proceeds successively by either merging smaller clusters into larger ones, or by splitting larger clusters into smaller ones. The result of the algorithm is a tree of clusters, called a "dendrogram", which shows how the clusters are related.

By cutting the dendrogram at a desired level depending on the clustering purpose, a clustering of the data items into disjointed groups is obtained.

- **Density-based clustering** groups neighboring objects within a data set into clusters based on density conditions. The objects with lower than required density are classified as outliers.
- **Grid-based clustering** is mainly proposed for spatial data mining. The space is divided into a finite number of cells and the clustering operation is conducted on a segmented data space grid structure.
- **Model-based clustering** assumes that the data are generated by a finite mixture of underlying probability distributions such as multivariate normal distributions. The issues of selecting a "good" clustering method and determining the "correct" number of clusters are reduced to model selection problems on the probability framework.
- **Fuzzy clustering** uses fuzzy techniques to cluster data and allows for an object to be classified into more than one clusters. These types of algorithms lead to clustering schemes that are compatible with everyday life experience, as they handle the uncertainty of real data.

In addition to applications in marketing, cluster analysis has been used in many fields, such as the life sciences, medicine, business, and engineering. The most widely used cluster analysis approaches in marketing are the partitioning approach and hierarchical approach. We will start with these two basic methods, providing only a brief review, as these approaches are well-known. See Theodoridis and Koutroubas (1999) for a more detailed review.

2.1.1 Partitioning Approach

Partitioning algorithms construct a partition of a database D of n objects into a set of k clusters. k is an input parameter that must be specified a priori, (i.e. some domain knowledge is required, which unfortunately is not available for many applications). The partitioning algorithm typically starts with an initial partition of D and then uses an iterative control strategy to optimize an objective function. Each cluster is represented by the gravity center of the cluster (k-means algorithms) or by one of the objects located near the center of the cluster (k-medoid algorithms). The objective function is typically the sum of the Euclidean distances between a point and the gravity center of a cluster (for K-Means) within each cluster for all the clusters. Consequently, partitioning algorithms use a two-step procedure. First, determine k representatives, minimizing the objective function initially. Second, assign each object to the cluster with its representative "closest" to the considered object, and re-compute the centers. The process continues until the centers of the clusters stop changing. It deals efficiently with large data sets. However, it also requires prior knowledge about the data set, since the number of clusters is a parameter in the method and must be determined before the clustering procedure. It is difficult to determine such information, especially for a large and unfamiliar data set. Sometimes, an inaccurate input can lead to unexpected clustering results. For example, if the number of clusters is set too small, some clusters will be very large, even including very isolated observations in order to accommodate the input parameter while ignoring the data structure. Although one can run K-Means clustering multiple times with different numbers of clusters, the returned clusters are not guaranteed to be structurally related. In addition, because the objective function is typically based on the Euclidean distance, the shape of all clusters found by a partitioning algorithm is convex, which is very restrictive. For example, if the data were based on locations of retail outlets, the stores may be located along streets, around an intersection, or in a shopping center; these configurations do not lend themselves well to convex shapes. Outliers in the dataset are not uncommon and they should not be classified into any cluster. However, the partitioning approach cannot identify outliers, and all data points are classified into clusters, which at times does not reflect the underlying data structure. The partitioning approach can handle large data sets well with computation cost of $O(n)$.

2.1.2 Hierarchical Approach

Hierarchical algorithms create a hierarchical decomposition of D. The hierarchical decomposition is represented by a "dendrogram", a tree that iteratively splits D into smaller subsets until each subset consists of only one object. A frequently used hierarchical approach is Ward's method. In such a hierarchy, each node of the tree represents a cluster of D. The dendrogram can either be created from the leaves up to the root (agglomerative approach) or from the root down to the leaves (divisive approach) by merging or dividing clusters at each step. It is common practice to begin with each observation in a cluster by itself, although the procedure could be initialized from a coarser partition if some groupings are known, a priori. In contrast to partitioning algorithms, hierarchical algorithms do not need k as an input. However, a termination condition must be defined, indicating when the merge or division process should be stopped. One example of a termination condition in the agglomerative approach is the critical distance $D_{\min}$ between all the clusters. In contrast to the partitioning method, the cluster shape can be arbitrary. However, the computational cost is very high due to the distance calculation for each pair of points. This is acceptable for applications such as character recognition with moderate values for n, but the computation burden is too high for a large database with computation cost of $O(n^2)$. As with the partitioning approach, outliers cannot be identified by the hierarchical approach.

Neither partitioning nor hierarchical methods directly address the issue of determining the number of groups within the data.

2.1.3 Density-Based Approach

Since the proposed new approach is developed from a density-based approach (DBSCAN is the basic algorithm of density-based approach), a more detailed explanation of density-based cluster analysis is provided here. Density-based cluster analysis was developed primarily by researchers in the Computer Science sub-discipline of Machine Learning. It was proposed first by Ester et al., (1996) for discovering clusters in large spatial databases with noise, and has been continually modified over time (e.g. Zaiane and Lee 2002). The density-based algorithm recognizes that within each cluster there is a typical density of points that is considerably higher than the density outside the cluster and, furthermore, the density within the areas of noise is lower than the density in any of the clusters. “Noise” refers to a point that is isolated and does not belong to any cluster. The key idea is that for each point of a cluster the neighborhood of a given radius has to contain at least a minimum number of points, (i.e. the population in the neighborhood must exceed some threshold). The shape of a neighborhood is determined by the choice of a maximum distance for two points. So, the density-based approach requires the specification of both MinPts and epsilon radius. MinPts refers to the minimum number of points that a qualified cluster can have in an epsilon radius neighborhood of that point. If the members of the groups are fewer than the MinPts, that point will be classified as noise. “Epsilon radius” refers to the maximum distance between two adjacent points that enables them to be considered reachable from one another.

Using the above principles, the density-based approach employs several constructs to describe the relationships within and among clusters; these concepts are based on the epsilon radius and MinPts. A point p is directly density reachable from a point q if p is within the epsilon radius neighborhood of the point q and there are more than MinPts within the epsilon radius neighborhood of the point q. Directly density reachable is not always symmetric. Figure 2.1 illustrates the concept. In Figure 2.1, the circles indicate the epsilon radius neighborhoods. MinPts is set to be 4. We find that B is directly density reachable from A, but A is not directly density reachable from B, since within the epsilon radius neighborhood of point B there are only 3 points, which does not meet the requirement of MinPts.

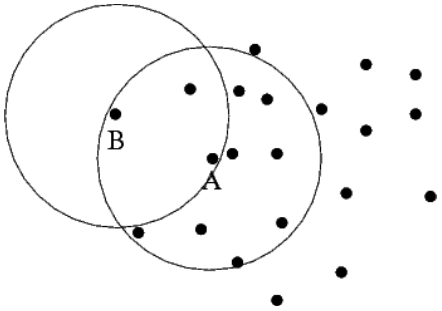


Figure 2.1 An Illustration of Directly Density Reachable Points

A point p is density reachable from a point q if there is a chain of points p_1, p_2, …, p_n, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density reachable from p_i. As with the directly density reachable concept, density reachable is not always symmetric. Figure 2.2 illustrates the concept. In Figure 2.2, the circles indicate the epsilon radius neighborhoods. MinPts is set to be 4. B is density reachable from C, but C is not density reachable from B, since within the epsilon radius neighborhood of point B there are only 3 points, which does not meet the requirement of MinPts.

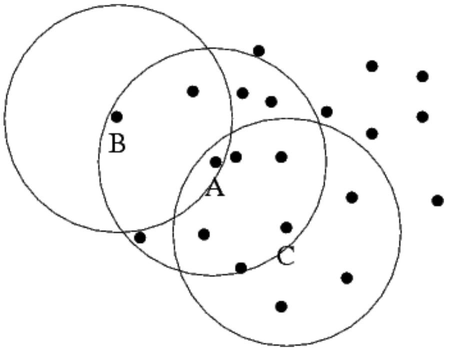


Figure 2.2 An Illustration of Density Reachable Points

A point p is density connected to a point q if there is a point o such that both p and q are density reachable from o. In contrast to the previous two concepts, density connectivity is a symmetric relation. Figure 2.3 illustrates the concept. In Figure 2.3, the circles indicate the epsilon radius neighborhoods. MinPts is set to be 4. We can say that points D and E are density connected.

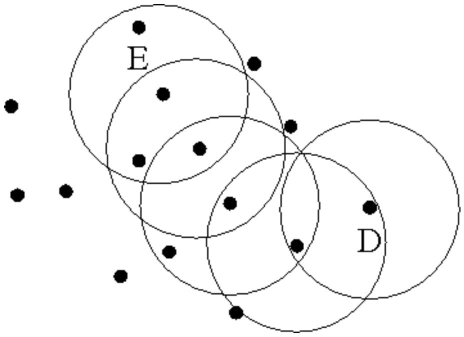


Figure 2.3 An Illustration of Density Connected Points

Based on these three concepts, the density-based cluster is defined to be a set of density connected points which is maximal with regard to density reachability. Noise is defined relative to a given set of clusters. It is simply the set of points in the database not belonging to any of its clusters. To find a cluster, the density-based approach begins with an arbitrary point p and retrieves all points density reachable from p with regard to epsilon radius and MinPts.

The density-based approach overcomes some of the problems associated with the traditional partitioning and hierarchical methods quite well. First, it can separate noise points from clustered points. So, it is possible but not necessary to cluster all the points. Second, it does not require domain knowledge in advance to determine the number of the clusters. Based on MinPts and epsilon radius setting, it can do the clustering automatically. Such advantages of the density-based approach are especially useful when the database is large and the domain is complicated, because it is difficult to know the number of clusters in advance. However, the choice of MinPts and epsilon radius is challenging. These issues are discussed more fully below.

The shape of a cluster is arbitrary, since determining whether a point can be clustered into a group is based on the density connectivity. This can overcome the convex shape restriction of the K-Means method, thus better reflecting reality in many settings. Finally, the density based approach can handle large databases very well with computation cost of $O(n\log(n))$ (Ester et al., 1996). These advantages are particularly important in relation to the problem studied in this thesis. For retail district identification, we know it is common to observe isolated commercial outlets (outliers); retail districts can appear in any shape; the number of commercial outlets can be extremely large; and outlet density can vary significantly. Therefore, the density-based approach can be helpful in accommodating the above special features associated with retail district identification.

2.1.4 Grid-Based Approach

The grid-based approach is another method for clustering spatial data. It divides the space into a finite number of cells and then does all operations within the cells. Since it considers cells rather than data points, the computation is generally more efficient. But, determining how to divide the space into a definite number of cells is a challenge at the start. The STING (Statistical information grid-based method) is representative of this category. It divides the spatial area into rectangular cells using a hierarchical structure. STING (Wang et al., 1997) goes through the data set and computes the statistical parameters (such as mean, variance, minimum, maximum and type of distribution) of each numerical feature of the objects within cells. Then it generates a hierarchical structure of the grid cells so as to represent the clustering information at different levels. Based on this structure, STING enables the usage of clustering information to search for queries or to conduct the efficient assignment of a new object to the clusters.

WaveCluster (Sheikholeslami et al., 1998) is a recent grid-based algorithm based on signal processing techniques (wavelet transformation) to convert spatial data into frequency domain. More specifically, it first summarizes the data by imposing a multidimensional grid structure onto data space (Han and Kamber 2001). Each grid cell summarizes the information of a group of points that map into the cell. The approach then uses a wavelet transformation to transform the original feature space. A prior knowledge about the exact number of clusters is not required in WaveCluster. The two grid-based approaches above deal well with low-dimensional data, Hinneburh and Keim (1999) and Pilevar and Sukumar (2005) proposed new grid-based approaches (OptiGrid and GCHL) to accommodate high-dimensional data.

A grid-based approach can handle arbitrarily shaped collections of points, for example, ellipsoidal, spiral, and cylindrical. As with the density-based approach, the grid-based approach can also identify the noise (outliers). Although grid-based techniques have some features that density-based approach lacks, a density-based procedure can better mimic the retail district intrinsic network. Since the way the density-based approach forms clusters shares similarities with customers walking in the retail district, the inherent network structure within a retail district can be captured to some degree by a density-based approach. In addition, our newly developed approach based on density-based approach can accommodate the special patterns. Therefore, we apply a density-based approach in retail district identification and develop advanced approach from the basic algorithm.

2.1.5 Model-Based Approach

The model-based approach, a relatively recent development (Banfield and Raftery 1993; Fraley and Raftery 2002), has demonstrated good performance in many applications on small- to moderate-sized data sets (MCLUST is the most common algorithm of model-based approach). However, it does not perform efficiently in terms of computer time and memory. In model-based clustering, it is assumed that the data are generated by a mixture of underlying probability distributions in which each component represents a different group or cluster. Given observations $(x_1, x_2, ..., x_n)$, the mixture model with density f is

$$\prod_{i=1}^{n}\sum_{k=1}^{G}\tau_k f_k(x_i \mid \theta_k),$$

where $f_k(x_i \mid \theta_k)$ is a probability distribution with parameters θ_k, and τ_k is the probability of belonging to the k^{th} component or cluster. G is the number of components in the mixture and will be determined by the BIC value. In most cases, $f_k(x_i \mid \theta_k)$ is taken to be multivariate normal distribution with the parameters of mean μ_k and covariance $\sum_k$. The parameters of the model are often estimated by maximum likelihood using the expectation-maximization (EM) algorithm (Mclachlan and Krishnan 1997). Each EM iteration consists of two steps, an E-step and an M-step. The E-step computes the conditional probability that object i belongs to cluster

k from the data with the current parameter estimates. In the M-step, parameters are estimated from the data given the conditional probabilities calculated in the E-step. The E-Step and M-Step are iterated until convergence, after which an observation can be assigned to the component or cluster corresponding to the highest conditional or posterior probability. Ward's method (Ward 1963) is the special case of general model-based approach in which the clusters are restricted to be spherical and identical in volume (size) (Fraley et al., 2005). Good initial values for EM can be obtained efficiently for small to moderate sized data sets via model-based hierarchical clustering (Banfield and Raftery 1993). Based on different parameterizations of the covariance matrix $\sum_k$, the approach can test different models in terms of distribution (Spherical, Diagonal, and Ellipsoidal), volume (Equal and Variable), shape (Equal and Variable), and orientation (Coordinate Axes, Equal, and Variable). A model-based approach is effective for determining the number of clusters, dealing with the outliers, and selecting the best clustering method in small- to moderate-sized data sets. But, this method cannot handle large data sets efficiently. Fraley et al., (2005) proposed an incremental approach to improve the computation efficiency by drawing a random sample of data, selecting and fitting a clustering model to the sample, and extending the model to the full data set by additional EM iterations.

2.1.6 Fuzzy Approach

The clustering approaches described above result in crisp clusters, meaning that a data point either belongs to a specific cluster or not. In other words, the clusters are non-overlapping. Unlike the crisp methods that force the points to belong exclusively to one cluster, fuzzy clustering allows points to belong to multiple clusters with varying degrees of membership. This approach allows additional flexibility. A typical fuzzy clustering algorithm is the Fuzzy

C-Means (FCM) (Bezdek et al., 1987). The FCM is an iterative optimization algorithm that minimizes the cost function

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ik}^{m} \left\| x_k - v_i \right\|^2 ,$$

where n is the number of data points, c is the number of clusters, x_k is the k^{th} data point, v_i is the i^{th} cluster center, μ_{ik} is the degree of membership of the k^{th} data in the i^{th} cluster, and m is the quantity controlling clustering fuzziness (typically $m = 2$). The degree of membership μ_{ik} is defined by $\mu_{ik} = \dfrac{1}{\sum_{j=1}^{c} \left(\dfrac{\left\| x_k - v_i \right\|}{\left\| x_k - v_j \right\|} \right)^{2/(m-1)}}$.

Starting with a desired number of clusters c and an initial guess for each cluster center $v_i, i = 1,2,...,c$, FCM will converge to a solution for v_i that represents either a local minimum or a saddle point of the cost function. The quality of the FCM solution, like that of most nonlinear optimization problems, depends highly on the choice of initial values (i.e. the number of c and the initial cluster centers). Based on classic FCM, some extensions have been done by researchers in modifying the objective functions (Pham and Prince 1999; Ahmed et al., 2002; Zhang and Chen 2003) and in improving initial value choosing efficiency (Chiu 1994; Kolen and Hutcheson 2002). Nevertheless, the sensitivity of this method to initial conditions makes it of limited value for the problem of establishing retail districts. Table 2.1 compares those cluster methods.

Table 2.1 Cluster Analysis Comparison

	Identification of outliers	**Cluster shape**	**Knowledge about cluster number in advance**	**Deals with large data sets**
Partitioning (K-Means)	No	Convex	Yes	Yes
Hierarchical (Ward's)	No	Polygon with no shape restrictions	No	No
Density-based (DBSCAN)	Yes	Polygon with no shape restrictions	No	Yes
Model-based (MCLUST)	Yes	Polygon with no shape restrictions	No	No
Grid-based (STING)	Yes	Polygon with no shape restrictions	No	Yes
Fuzzy (FCM)	No	Convex	Yes	Yes

2.2 Cluster Validation

Since clustering algorithms discover clusters and identify groups with similarities, neither of which are known a priori, the final clusters of a data set require evaluation (Rezaee et al., 1998). This evaluation serves at least two purposes. One is determining when the best set of clusters is obtained with a given algorithm or set of input parameters. The other is testing whether the clustering algorithm is a good representation of the underlying reality. For example, answering questions like “How many clusters are there in the data set?” and “Is there a better clustering result for the data set?” and “Does the resulting clustering match the data reality?” requires what we call a clustering validation technique. The quantitative evaluation of the results of the clustering algorithms is preferable to visual validation (applying to a maximum of two dimensions) and is called cluster validity methods. However, comparative studies on clustering algorithms are difficult to conduct in general due to the lack of universally agreed upon quantitative performance evaluation measures (Jain et al., 1999). Subjective (human) evaluation is often difficult and expensive, yet is still valuable in many real applications. There are three general approaches to investigating cluster validity (Theodoridis and Koutroubas 1999). The first is based on external criteria. This implies that the result of a clustering algorithm is compared with the pre-specified structure, which is imposed on a data set, then to evaluate the clustering performance. The second approach is based on internal criteria. The result of a cluster algorithm is evaluated in terms of quantities and features that involve the vectors of the data set themselves (e.g. proximity matrix). The third clustering validity approach is based on relative criteria. Clustering structure is evaluated by comparing it to other clustering schemes that result from the same algorithm but involve different parameter values.

The first two approaches are based on statistical tests, and their major drawback is their high computation cost. Moreover, the indices related to these approaches aim at measuring the degree to which a data set confirms a scheme specified *a priori*. On the other hand, the third approach aims at finding the best clustering scheme that a clustering algorithm can be defined under certain assumptions and parameters (Halkidi et al., 2001).

2.2.1 External Criteria

External criteria test whether or not the points of the data set are randomly structured. This can work in two ways. First, we can evaluate the resulting clustering structure C by comparing it to an independent partition of the data P built according to our intuition about the clustering structure of the data. Second, we can compare the proximity matrix to the partition P (Halkidi et al., 2001).

2.2.2 Internal Criteria

Internal criteria evaluate the clustering result of an algorithm using only quantities and features inherent to the data set (Halkidi et al., 2001). For low-dimensional vector data, the average (or summed) distance from cluster centers (e.g. the sum-squared error criteria used for the standard K-Means algorithm) is a common criterion (Zhong and Ghosh 2003).

2.2.3 Relative Criteria

The external and internal criteria are statistical testing and need high computation in general. In addition, the data structure needs to be specified by external and internal criteria. This can be done for the simulated data under a specified simulating mechanism. For the real data, for example retail outlet location data, we do not know the data structure in advance and must try to find the data pattern. Therefore, we cannot use external and internal criteria for a validation test using the real data. The relative criteria do not involve statistical tests (Halkidi et al., 2001). The idea is to choose the best clustering scheme of a set of defined schemes according to a pre-specified criterion. More specifically, the problem can be stated as follows (Halkidi et al., 2001):

> "Let $P_{a\lg}$ is the set of parameters associated with a specific clustering algorithm (e.g. the number of clusters nc). Among the clustering schemes $C_i, i = 1,2,3,...,nc$, defined by a specific algorithm, for different values of the parameters in $P_{a\lg}$, choose the one that best fits the data set."

There are two criteria proposed for clustering evaluation and selection of an optimal clustering scheme (Berry and Linoff 1996):

1. **Compactness** -- the members of each cluster should be as close to each other as possible. A common measure of compactness is the variance of the observations, which should be minimized.
2. **Separation** -- the clusters themselves should be widely spaced. There are three common approaches to measuring the distance between two different clusters:
   - Single linkage measures the distance between the closest members of the clusters.
   - Complete linkage measures the distance between the most distant members.
   - Comparison of centroids measures the distance between the centers of the clusters.

In general terms, we want clusters whose members have a high degree of similarity, while we want the clusters themselves to be widely spread. Several relative criteria have been proposed, for example, the modified Hubert T Statistic (Halkidi et al., 2001), Dunn (Dunn 1974) and Dunn-like indices (Pal and Biswas 1997), and the Davies-Bouldin (DB) index (Davies Bouldin 1979). The special measurements evaluating the fuzzy clustering approach have also been investigated (Bezdek et al., 1984; Pal and Bezdek 1995). Here we explain the SD Validity index (Halkidi et al., 2000) and the *Comp_Sepa* measure (Liu and Huang 2007), since we use the SD index to evaluate the K-Means performance and chose the optimal number of clusters for Victoria in the illustration section, and since our new cluster validation measurement is developed from *Comp_Sepa* measure.

2.2.3.1 SD Validity Index

A recent clustering validity measure – the SD validity index -- was proposed by Halkidi et al., (2000). Consistent with the general rule for cluster validation, the underlying principle of the SD index is that a good set of clusters should be compact within themselves (homogeneity within clusters) but distinct from each other (heterogeneity among clusters). The index is based on the concept of the average scattering (the "reverse" of compactness) of clusters and of the total separation among clusters. The average scattering of clusters is defined as

$$Scat(nc) = \frac{1}{nc}\sum_{i=1}^{nc} \left\|\sigma(v_i)\right\| / \left\|\sigma(X)\right\|.$$

Total separation between clusters is defined as

$$Dis(nc) = \frac{D_{\max}}{D_{\min}}\sum_{k=1}^{nc}\left(\sum_{z=1}^{nc}\left\|v_k - v_z\right\|\right)^{-1},$$

where $\sigma(v_i)$ is the variance of cluster *i*, $\sigma(X)$ is the variance of the whole data set, $D_{\max} = \max(\left\|v_i - v_j\right\|) \ \ \forall i, j \in \{1,2,3,...,nc\}$ is the maximum distance between cluster centroids, $D_{\min} = \min(\left\|v_i - v_j\right\|) \ \ \forall i, j \in \{1,2,3,...,nc\}$ is the minimum distance between cluster centroids. Then the SD index is defined as

$$SD(nc) = \alpha \bullet Scat(nc) + Dis(nc),$$

where α is a weighting factor.

The first term (i.e. $Scat(nc)$) indicates the average compactness of clusters (i.e. intra-cluster distance). A small value means good compactness, and as the scattering within clusters increases (i.e. they become less compact) the value of $Scat(nc)$ also increases. The second term $Dis(nc)$ indicates the total separation among the clusters (i.e. an indication of inter-cluster distance). The optimal number of clusters should be that which minimizes the SD index. However, the index cannot handle arbitrarily shaped clusters properly, because the method uses the diameter (or radius) of clusters to compute the evaluation function (Halkidi et al., 2001).

2.2.3.2 *Comp_Sepa* Index

The most recently published clustering validity measure, called the *Comp_Sepa* index, was proposed by Liu and Huang (2007). A traditional validity index, such as the SD index, does not properly address clustering algorithms that allow for the creation of non-convex clusters due to their use of compactness measures based on sum of squared differences between points within a cluster and the cluster centroid or related measures since these types of measures implicitly assume that "proper" clusters are circular (or at least convex). In contrast with other clustering validation metrics, the *Comp_Sepa* index can evaluate both convex and non-convex clusters. The widely accepted criteria for evaluating cluster performance are the separation of the clusters and their compactness. To take both measurements into consideration, the *Comp_Sepa* index is the ratio of compactness and separation, measuring the overall performance of clustering. *Comp* (the measure of compactness) is evaluated using the interior density of clusters (point density within the clusters). *Sepa* (the measure of separation) is computed using the outside density (the density of regions outside clusters) and the inter-cluster separation (the degree of separation between clusters). Both measurements are calculated using the minimum-cost spanning tree (MST).

To evaluate the compactness of a clustering scheme C, the MST is generated for each cluster and the edges of the MST are summed. The value of *Comp* for the solution is equal to the sum of the MST edges for the cluster with the greatest value of this sum for clustering scheme *C*. A smaller value indicates a more compact cluster. As a result, *Comp* represents a "worst case" measure of compactness. To assess separation, the MST for the set of points containing the centroids of the clusters (groupings that have more than a single member) and "noise points" (noise points are clusters that have only a single member, and thus have a compactness measure of zero) is calculated. The separation measure for the clustering scheme *C* is defined as the distance of the shortest edge of the MST. A large value indicates well-separated clusters, and again it represents a “worst case” measure. The validity index *Comp_Sepa* is the ratio between compactness and separation, namely, $Comp\_Sepa = \frac{Comp}{Sepa}$. Therefore, the cluster algorithm with the smallest value of *Comp_Sepa* is preferred.

The *Comp_Sepa* index is a simple but intuitive index evaluating clustering performance, and it can handle large data sets and arbitrarily shaped clusters (Liu and Huang 2007). However, we have found two serious limitations with the *Comp_Sepa* index in the application of the method to both simulated and real-world data. First, the *Comp_Sepa* index is biased towards having as many clusters as there are points in the data set under analysis. Second, the *Comp* and *Sepa* measures typically have very different potential ranges, with the potential values of *Comp* having a range that is often over an order of magnitude larger than the range of *Sepa*, which can exacerbate the first problem in some instances, or lead to the reverse problem (too highly aggregated clusters) in other situations. We explore the first problem in some detail in the remainder of this chapter, and discuss the second problem in the next chapter.

The bias in the *Comp_Sepa* index towards having as many clusters as points in the data can be seen using a mathematical argument. Specifically, the *Sepa* measure is always strictly greater than zero, while the *Comp* measure is always non-negative, and obtains a minimum, at zero, when all clusters contain a single member. Consequently, the *Comp_Sepa* measure has a degenerate “best” solution with a value of zero when every point is in its own cluster. While this particular case can be ruled out by requiring that the number of clusters in a solution must always be strictly less than the number of data points, one suspects that the basic problem still persists when there are a large number of clusters relative to the number of points (resulting in "under clustering"), which is indeed the case.

Two examples based on simulated data are used to illustrate the problem. The first example is based on the data set shown in Figure 2.4. A visual examination of the figure indicates that data contains 9 clusters arranged on a 3 by 3 grid. Each cluster consists of 20 points, with the coordinate values drawn from a unit uniform distribution, and then shifted so that the center of the ranges fell on the 3 by 3 grid. DBSCAN clustering solutions were then examined for values of the epsilon radius that varied from 0.01 to 1.20, incremented by 0.01. The results of this analysis are shown in Figure 2.5.

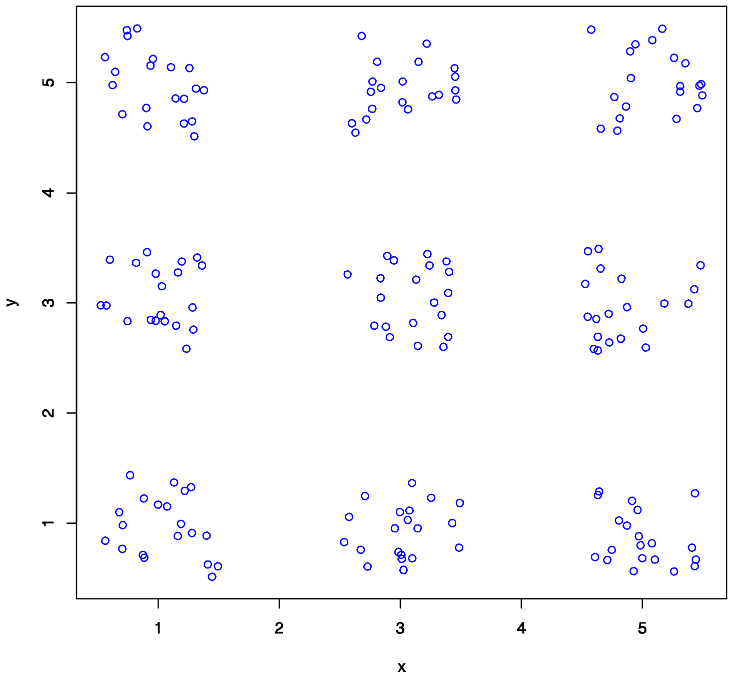


Figure 2.4 The First Simulated Data Set

The figure contains plots of the *Comp* (northwest panel), *Sepa* (northeast panel), and *Comp_Sepa* (the southwest panel) index, as well as the "best" solution clustering solution based on selecting the epsilon radius value that minimizes the *Comp_Sepa* index (southeast panel). As suggested above, an examination of Figure 2.5 indicates that the lowest (best) value of the *Comp_Sepa* index occurs at the smallest value of epsilon's range (0.01), which corresponds to each point being in its own cluster. After the value of the epsilon radius exceeds 0.02, the value of the *Comp_Sepa* index begins to climb, peaking at an epsilon radius value of 0.28, at which point it begins to fall into a trough that extends over a range of epsilon values from 0.50 to 1.06. This range of epsilon radius values corresponds to the nine-cluster solution that we would anticipate

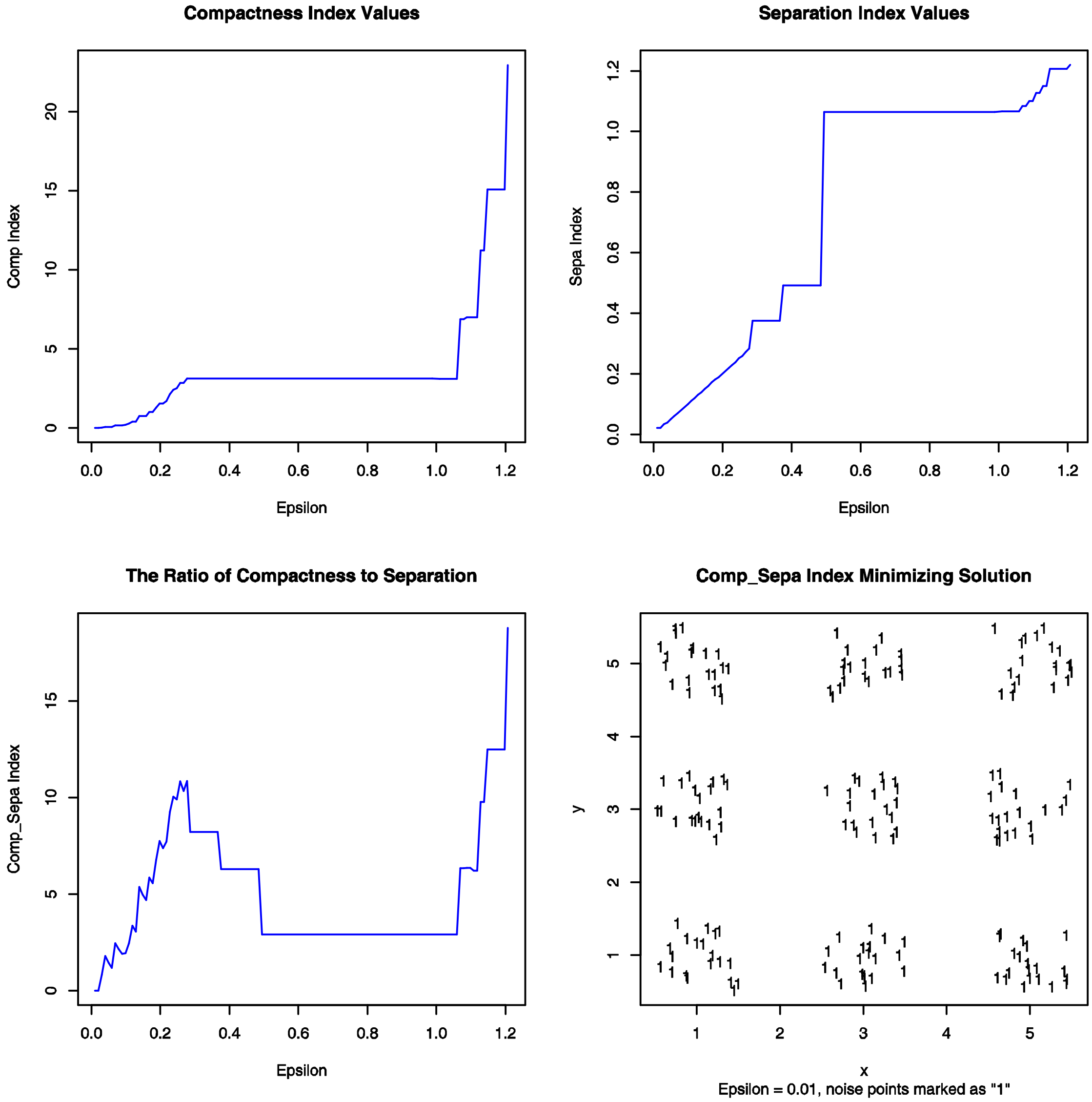


Figure 2.5 The *Comp_Sepa* Analysis of the First Data Set

finding by visually examining the data. However, there is a range of epsilon radius values (from 0.03 to 0.11) over which there are fewer clusters than points, and for which the value of *Comp_Sepa* is better (lower) than the range of epsilon radius values that corresponds to the nine-cluster solution. As a result, it is clear that the *Comp_Sepa* measure fails for what would appear to be a fairly straightforward example.

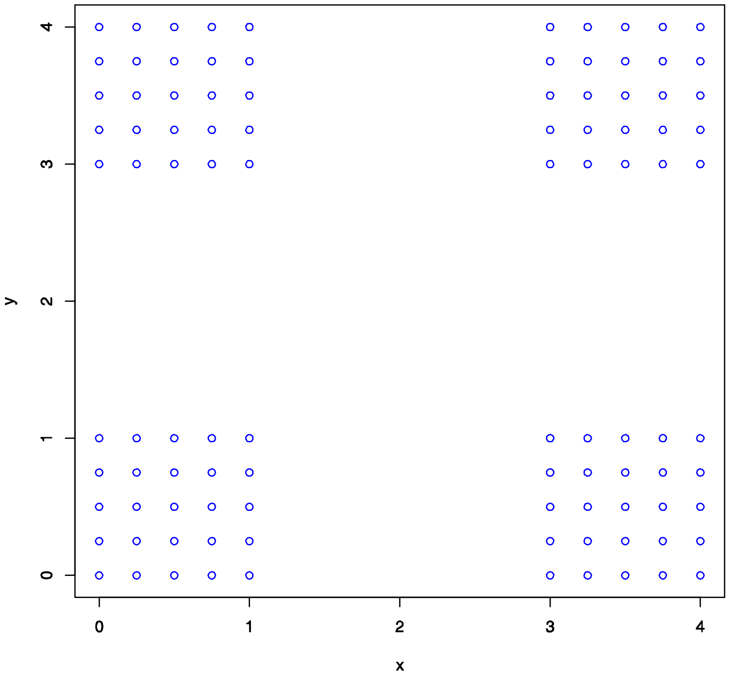


Figure 2.6 The Second Simulated Data Set

Figure 2.6 shows the data set for the second example. As the figure reveals, it corresponds to four grid pattern clusters, each in a corner of the overall space. In our opinion, this data set represents as clean an example of a four-cluster solution as is possible. Moreover, using DBSCAN as the clustering algorithm, there are only three possible solutions: (1) all clusters contain a single point; (2) the four-cluster solution; or (3) all points are in a single cluster. The third possible solution could not be found by the *Comp_Sepa* method since the separation between clusters is undefined when there is only a single cluster (which should not be a problem in this instance). As a result, the *Comp_Sepa* method should detect one of two possible clustering solutions. Figure 2.7 provides the results of applying the *Comp_Sepa* method to the second data set.

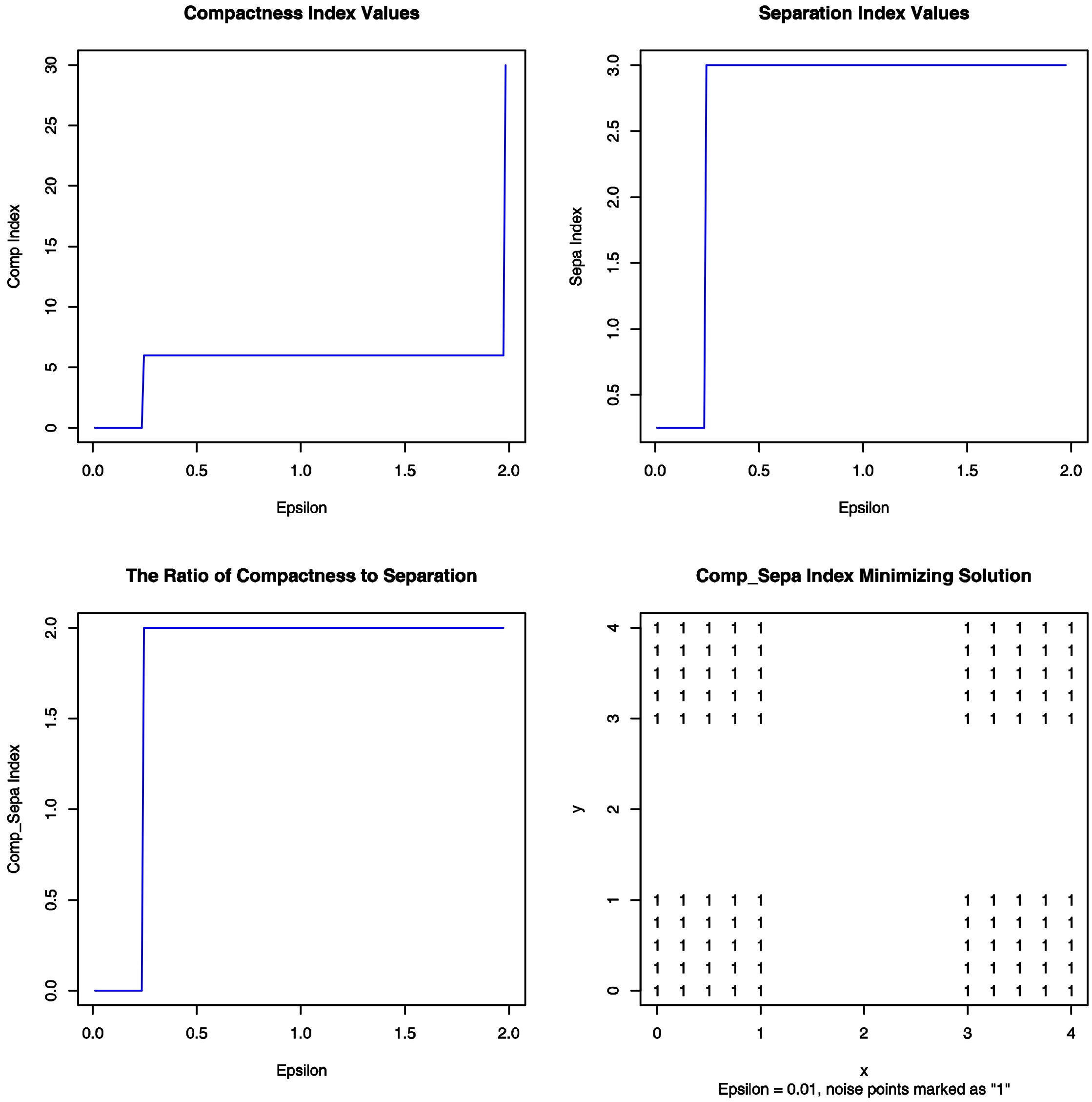


Figure 2.7 The *Comp_Sepa* Analysis of the Second Data Set

As with the first example, the "best" epsilon radius value based on the *Comp_Sepa* measure corresponds to values of the epsilon radius (0.01 to 0.24) that result in each cluster containing a single point. Epsilon radius values between 0.25 and 1.99 correspond to the four-cluster solution, while values of the epsilon radius of 2 and above correspond to the single cluster solution (which can be seen in the very large jump in the *Comp* index).

Both of these examples illustrate the bias in the *Comp_Sepa* index towards a large number of clusters relative to the number of data points. It can be reasonably argued that the method performs well within the neighborhood of the correct solution (as illustrated by the large trough between epsilon radius values of 0.50 and 1.06 for the first example data set). However, this assumes that a visual examination of the data allows for an assessment of the "likely" number of clusters in the data (something that appears to be the norm in the computer science literature in this area), allowing the range of clustering solutions examined to be within a "reasonable" neighborhood of the visual "truth".[1] Our experience with "real world" data suggests that the ability to determine the neighborhood of the correct number of clusters based on a visual examination of the data (even when the data falls in a two-dimensional plane as is the case of the focal example in our work) typically is not possible. As a result, there is a need to develop a relative cluster validation method with better "global" properties than the *Comp_Sepa* index, but that is still consistent with clustering methods that allow for non-convex clusters. In the next chapter we develop such a cluster validation measure.

[1] In Liu and Huang's (2007) defense, our suspicion is that they only examined clustering solutions in the neighborhood of the visually "correct" number of clusters in their simulation results.

3 An Alternative Approach to Cluster Validation

In this chapter we develop a new cluster validation measure that borrows the positive aspects of Liu and Huang's (2007) *Comp_Sepa* index, but eliminates some of the problems encountered with their measure in practical applications. The real innovation in their approach was the use of the minimum cost spanning tree to measure cluster compactness in a way that does not implicitly assume that the "true" shape of clusters is convex. We take advantage of this innovation, but attempt to address the problems associated with the interaction between the separation and compactness measures caused by scaling issues. In addition, we look at the implications of selecting different numbers of MinPts in DBSCAN clustering for cluster validation.

3.1 The *CpSp* Cluster Validation Index

As indicated in the previous chapter, there is a much greater range across potential values of compactness as compared to separation. What makes this problematic is that in some sense we should be interested in relative compactness and separation rather than absolute levels of these two measures. The problem with the *Comp_Sepa* approach is that it does not matter if separation is driven to its minimum possible level (resulting in the *Sepa* measure obtaining zero percent of the possible separation level that could be obtained in a data set), so long as the compactness measure can be driven to zero.

One way to solve this problem is to rescale the measures. Our approach is to place both measures on percentage terms with respect to both the upper and lower limits of the values could conceivably obtain. Based on this notion, we call our compactness measure the compactness percentage (which we label *Cp*). Mathematically, this value is given as

$$Cp = \frac{C_{max} - Comp}{C_{max} - C_{min}},$$

where C_{max} is the measure of the largest possible compactness value, *Comp* is Liu and Huang's (2007) compactness measure, and C_{min} is the measure of minimum possible compactness. While these measures may seem difficult to assess prior to performing any clustering, a great deal of information concerning the possible bounds of a clustering solution can be determined *a priori*.

The largest possible compactness value occurs when all the points in the data set are in a single cluster. Consequently, we measure C_{max} using the sum of the edges of the MST for all the points in the data set. The complication this introduces is how best to compute the MST. One approach that works for clustering in an arbitrary number of dimensions requires the calculation of the distance matrix for the entire data set (in order to determine the edge costs), an operation that is of order $O(n^2)$, and is infeasible for large data sets (largely due to computer memory limitations). An alternative approach is to base the MST on the Delauney triangulation of the points (De Berg, et al., 2008), which can be done in $O(n\log(n))$ time for two dimensions, but this limits us to the examination of clustering in two or at most three dimensions.[2] Given the nature of our objectives (clustering a large number of store locations in two dimensional space), we have opted to use the Delauney triangulation approach in our implementation.

There are two possible minimum compactness values, zero (corresponding to the case where each point is in a cluster by itself), and the minimum distance between any two points in the data set (which would consist of a single two-point cluster, with the remaining points being

[2] Practically, it limits us to two dimensions since we make use of the TRIPACK package (Renka, 1996), as implemented in the tripack R library, which only performs Delauney triangulations in two dimensions. However, there are open source libraries that can perform Delauney triangulations in three dimensions that could be used in the future by creating appropriate bindings to R.

in single point clusters). We have elected to use the latter option, however, this decision is likely to have minimal impact on the performance of the measure. Our reason for using the minimum distance between two points in a data set is motivated by the notion that for there truly to be clusters, at least two points in the data set should be agglomerated together.

Similar to compactness, we measure separation using the percentage of the highest possible level of separation that could possibly be obtained, which we call the separation percentage, or *Sp*. Mathematically, *Sp* is defined as

$$Sp = \frac{Sep - S_{min}}{S_{max} - S_{\min}},$$

where *Sep* is the minimum distance between any two points not in the same cluster (a single linkage measure), S_{min} is the *a priori* minimum possible separation between any two points not in the same cluster, and S_{max} is the *a priori* maximum possible separation between any two points not in the same cluster. We obtain the value of *Sep* using an efficient nearest neighbor search algorithm (via Mount and Arya's ANN library, Mount, 2006) in which the points within the cluster are taken to be the reference points, and those not in the cluster are taken to be the test points. We have moved from the centroid based average likage used by Liu and Huang (2007) to a single linkage measure for two reasons. First, it is conceptually more consistent with the measure of S_{max} we use (which is described below), second, because two large clusters (in terms of their lack of compactness) may be fairly distant from one another based on the distances between their centroids, but in fact may be very near to one another when this same distance is measured based on the distance between the closest points between nearest members in the two clusters.

The minimum possible separation between clusters (S_{min}) in a data set corresponds to either the shortest distance between any two points in a data set (corresponding to the case when each point is in its own cluster) or the second shortest distance between any two points in a data set (corresponding to the case where there is a single two member cluster, and the remaining points are in single point clusters). While not logically consistent with our definition of minimum compactness, but practically unimportant, we use the minimum distance between any points in the data set as the measure of minimum separation.

Probably the least obvious *a priori* bound on the clustering solution is S_{max}. We argue (and have empirically confirmed, but have yet to formally prove) that the maximum possible separation between two clusters in a point data set corresponds to the longest edge of the Gabriel graph (Gabriel and Sokal, 1969; Matula and Sokal, 1980) of that set of points. The Gabriel graph is a sub-graph of the Delauney triangulation of a set of points, with the edges of the graph linking the pairs of points that are Gabriel neighbors. In turn, two points are Gabriel neighbors if a sphere (or circle in two dimensions) centered at the mid-point between the two points (which also has the two points on the diameter of the sphere) has no other points from the data set within the sphere. Figure 3.1 illustrates the concept of Gabriel neighbors in two dimensions, while Figure 3.2 shows the Gabriel graph for the first example data set described in the previous chapter (see Figure 2.4). We use the length of the longest edge of the Gabriel graph as our measure of S_{max}.

We use as our composite measure of compactness and separation the product of the compactness percentage and the separation percentage (or *CpSp*, which corresponds to the index's name). This measure is bounded within the unit interval, and takes the value of 1 if a clustering solution is able to achieve both a compactness of C_{min} (causing *Cp* to equal 1) and a separation of S_{max} (causing Sp to equal 1), which is a situation that would never happen in practice. The index takes on a value 0 if a clustering solution has a maximum compactness that equals C_{max} (which occurs when all points are in a single cluster), or the minimum separation

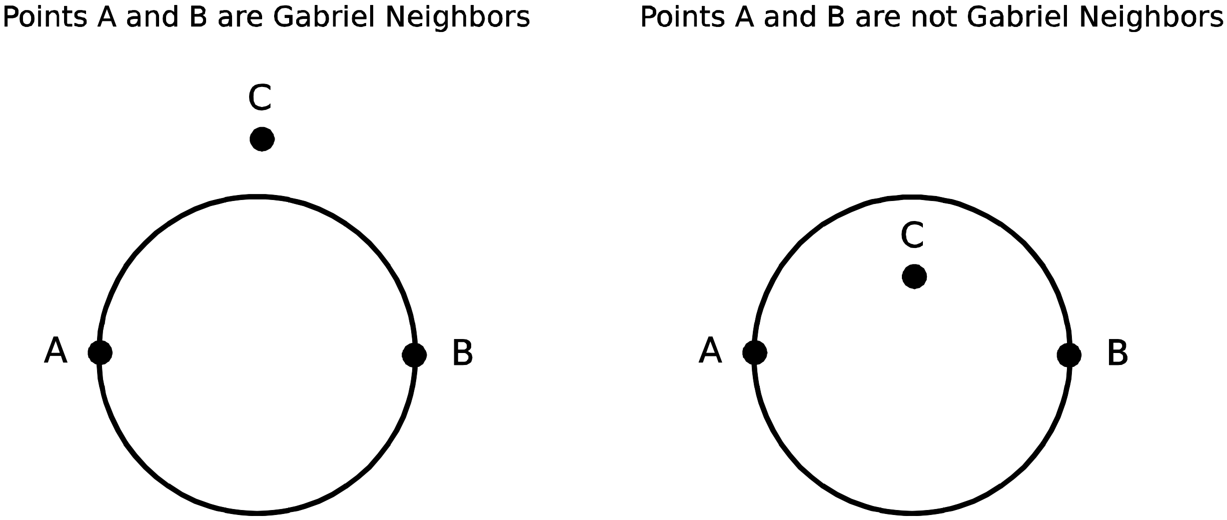

Points A and B are Gabriel Neighbors
C
A
B
Points A and B are not Gabriel Neighbors
C
A
B

Figure 3.1 An Illustration of Gabriel Neighbors

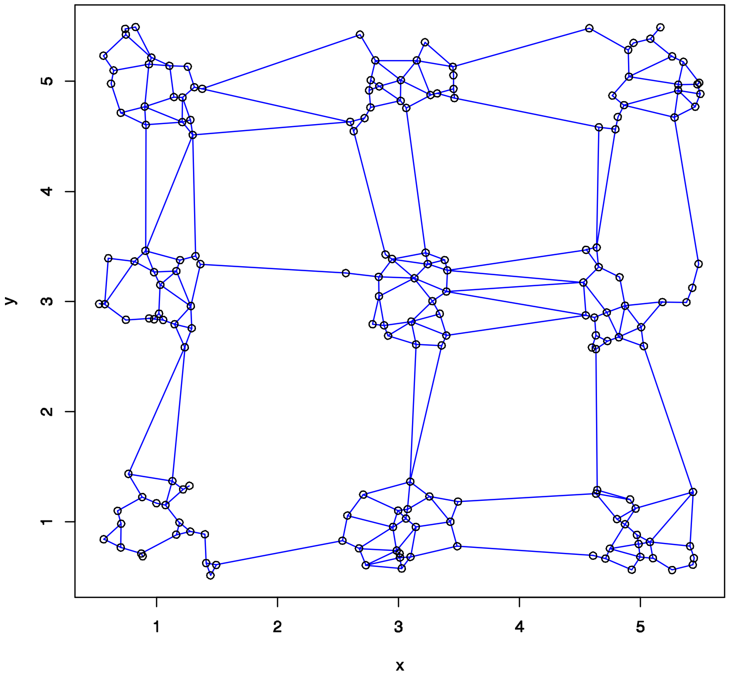


Figure 3.2 The Gabriel Graph of the First Example Data Set

between clusters corresponds to S_{min}. It is this second case that avoids the degenerate solution of selecting the clustering solution in which every cluster contains a single point as “best” which occurs with Liu and Huang’s (2007) *Comp_Sepa* method. Given the structure of the index, the clustering solution that performs best under this index is the one that maximizes its value.

Rather than a product of the *Cp* and *Sp* measures, we could have used the sum of the two measures. However, the use of the product is better able to maintain a balance between compactness and separation. To illustrate this point, consider three different clustering solutions of a point data set. The first solution has a *Cp* value 0.99 and an *Sp* value of 0.01, the second solution has a *Cp* value of 0.5 and an *Sp* value of 0.5, while the third solution has a *Cp* value of 0.01 and an *Sp* value of 0.99. If the sum of the two measures was used as the composite index, then all three solutions would yield a composite index value of 1 (for a measure that would be bounded between 0 and 2), but the first solution would have poor separation between clusters, while the third would have at least one very diffuse cluster. Conversely, the second solution strikes a much better balance between compactness and separation, and, in our opinion, should be preferred on these grounds. The product of the compactness and separation percentages is consistent with this, since the first and third solutions have a *CpSp* value of 0.099, while the *CpSp* value for the second solution is 0.25.

To determine if the *CpSp* measure overcomes the issues encountered with the *Comp_Sepa* method, we apply our new index to the same two example data sets and DBSCAN clustering solutions used in the previous chapter. The results of using the *CpSp* index with the first example data set is contained in Figure 3.3, while the application of the index to the second example data set is contained in Figure 3.4. In Figure 3.3, actually any value of epsilon radius along the best "flax max" region generates the same clustering solution (the southeast panel would look same), the value of *Sp* would have to change if the clustering structure changed (since the edge for the current *Sep* value would "go away" due to the merging of clusters), therefore changing the *CpSp* value.

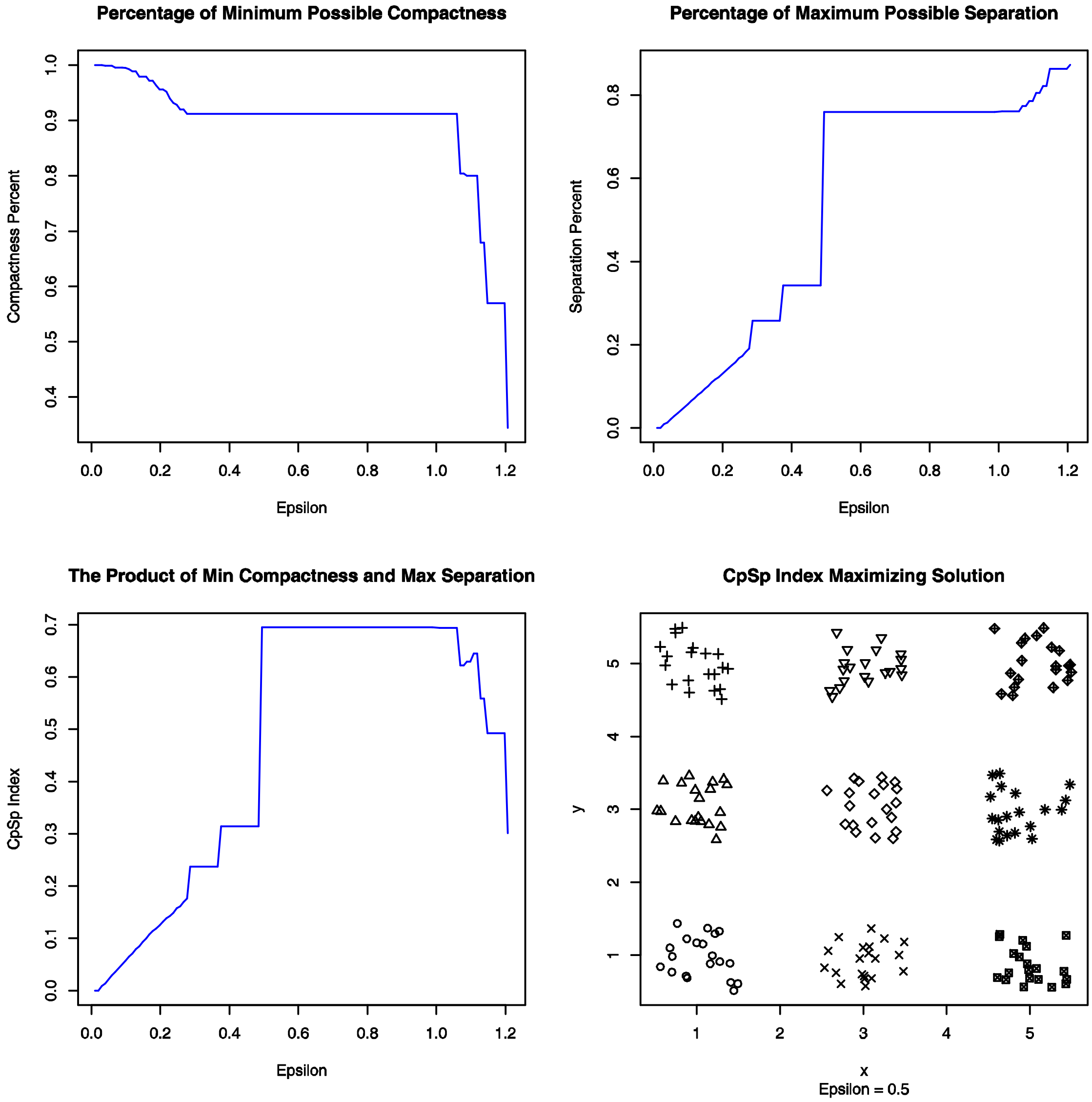


Figure 3.3 The *CpSp* Analysis of the First Example Data Set

An examination of both figures indicates that the *CpSp* index is able to successfully find both the nine-cluster solution for the first example data set, and the four-cluster solution of the second example data set. Perhaps what is most striking is a comparison of Figures 2.7 and 3.4. Specifically the plots of the *Comp_Sepa* index and the *CpSp* index over the range of epsilon radius values (which is contained in the southwest panel in both figures) are identical in shape,

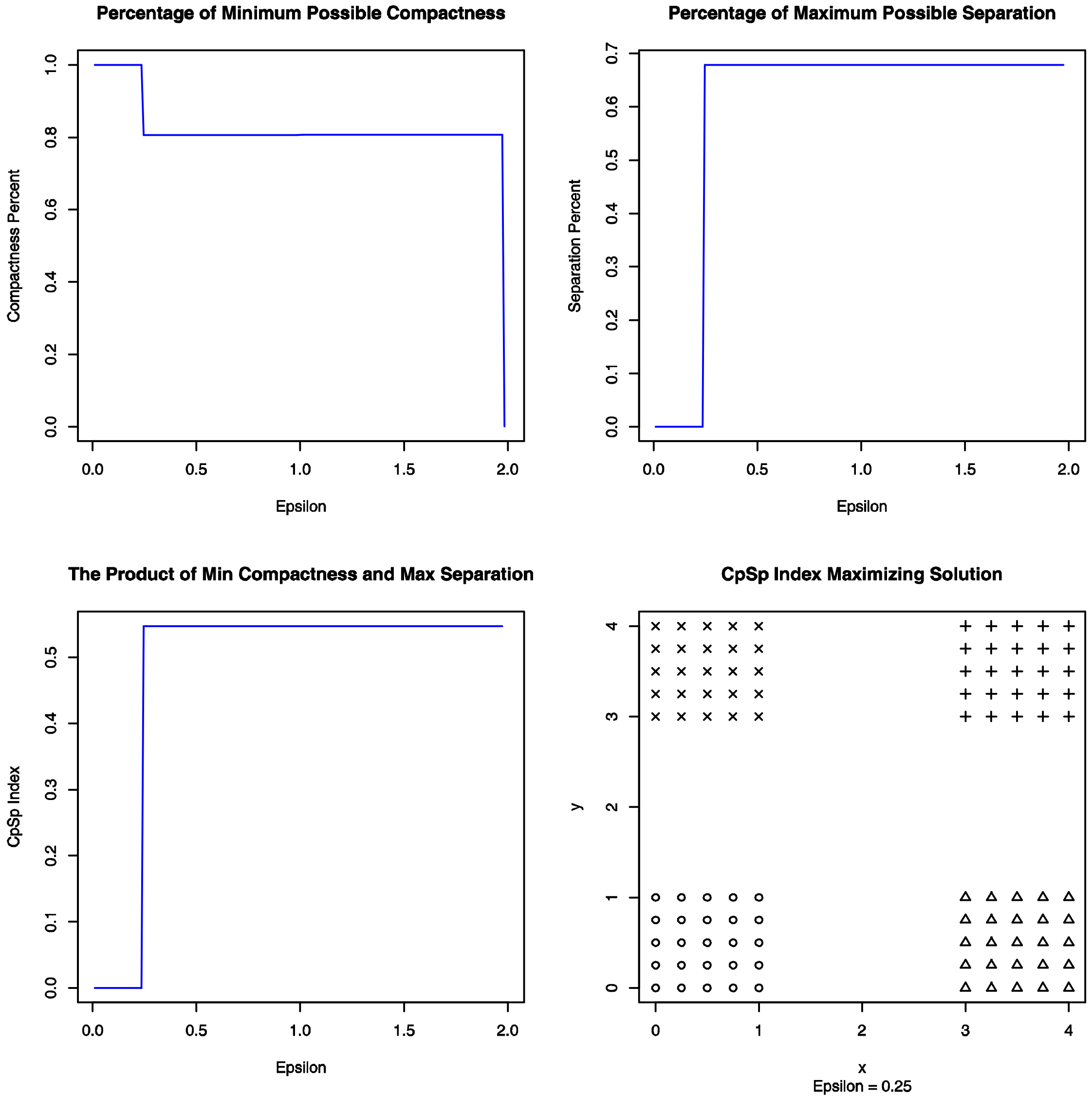


Figure 3.4 The *CpSp* Analysis of the Second Example Data Set

but are polar opposite in terms of interpretation since the "best" solution corresponds to the lowest value of the *Comp_Sepa* index, while the highest value of the index is "best" for *CpSp*.

Given the findings from this analysis, it appears the proposed *CpSp* measure is able to validate clustering solutions from methods capable of producing non-convex clusters (through the use of the MST to measure cluster compactness), but avoids the pitfalls of the *Comp_Sepa* method in terms of being biased towards a large number of clusters relative to the number of points. At this juncture, we turn to another issue we have uncovered in our efforts to determine the appropriate length of the epsilon radius for DBSCAN solutions, the interplay between the selection of the number of MinPts for the DBSCAN algorithm and cluster validation.

3.2 The Influence of the Number of MinPts on the “Best” Length of the Epsilon Radius

In performing the empirical application that is the basis of the sixth chapter of this thesis, we determined that the “best” numbers of clusters for a DBSCAN solution was very sensitive to the value of MinPts that was used, and this sensitivity seemed unpredictable to us.[3] Ultimately, we determined that this erratic behavior occurred when there was a group of two or more points that were distant from points not in the group, were fairly close to one another, but the size of the group was smaller than the value of MinPts. The “outlying” points act to halt the movement in the separation measure over some range of the epsilon radius values, once the size of the epsilon radius allowed one of the points to be reached by a point in a cluster (thereby allowing all the outlying points to enter that cluster)

An example (based on the data set shown in Figure 3.5) will help to make the issue more concrete. As can be seen in the example, there are two outlying points in the data that are in the southeast corner of the figure. These two points are fairly distant from the other points in the data, but are very close to one another. The two outlying points will induce a problem if MinPts is set above a size of two. To see this, Figures 3.6 and 3.7 provide a *CpSp* analysis of this data set with the two outlying points removed, with Figure 3.6 being based on DBSCAN

[3] We initially discovered the problem using the *Comp_Sepa* index for cluster validation (ruling out epsilon radius values we deemed too small, thereby avoiding degenerate solutions in an ad hoc way). However, the issue also arises when the *CpSp* index is used as well, although it is somewhat mitigated for *CpSp* relative to *Comp_Sepa*.

runs with MinPts set to two, and Figure 3.7 based on DBSCAN runs with MinPts set to 3. As can be seen

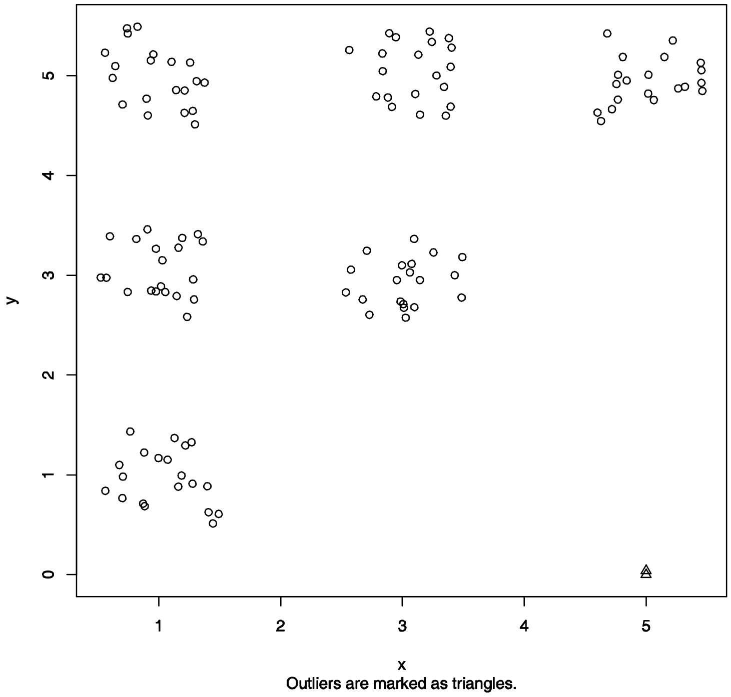


Figure 3.5 The Outlying Points Example Data Set

from the two figures, the solutions are identical, resulting in a “best” epsilon radius being within the range of 0.3 to 1.1 (we select the first value in this range, 0.3 in this case, as the “best” value, even though all values in the range correspond to the same solution).

As illustrated in Figures 3.8 (where the value of MinPts is set to two for the DBSCAN runs) and 3.9 (where MinEps is set to three), things become very different when the two outlying points are placed back into the data set. The cluster structure is nearly identical for the analysis done with MinPts set to two for the DBSCAN runs when the two outlying points are included compared to the analyses done when the two outlying points are removed. The only difference in the “best” solution from the inclusion of the two outlying points is that they are combined together into a seventh cluster that contains only those two points. However, this is not true for the analysis in which MinPts is set to three for the DBSCAN runs. In this case, the “best”

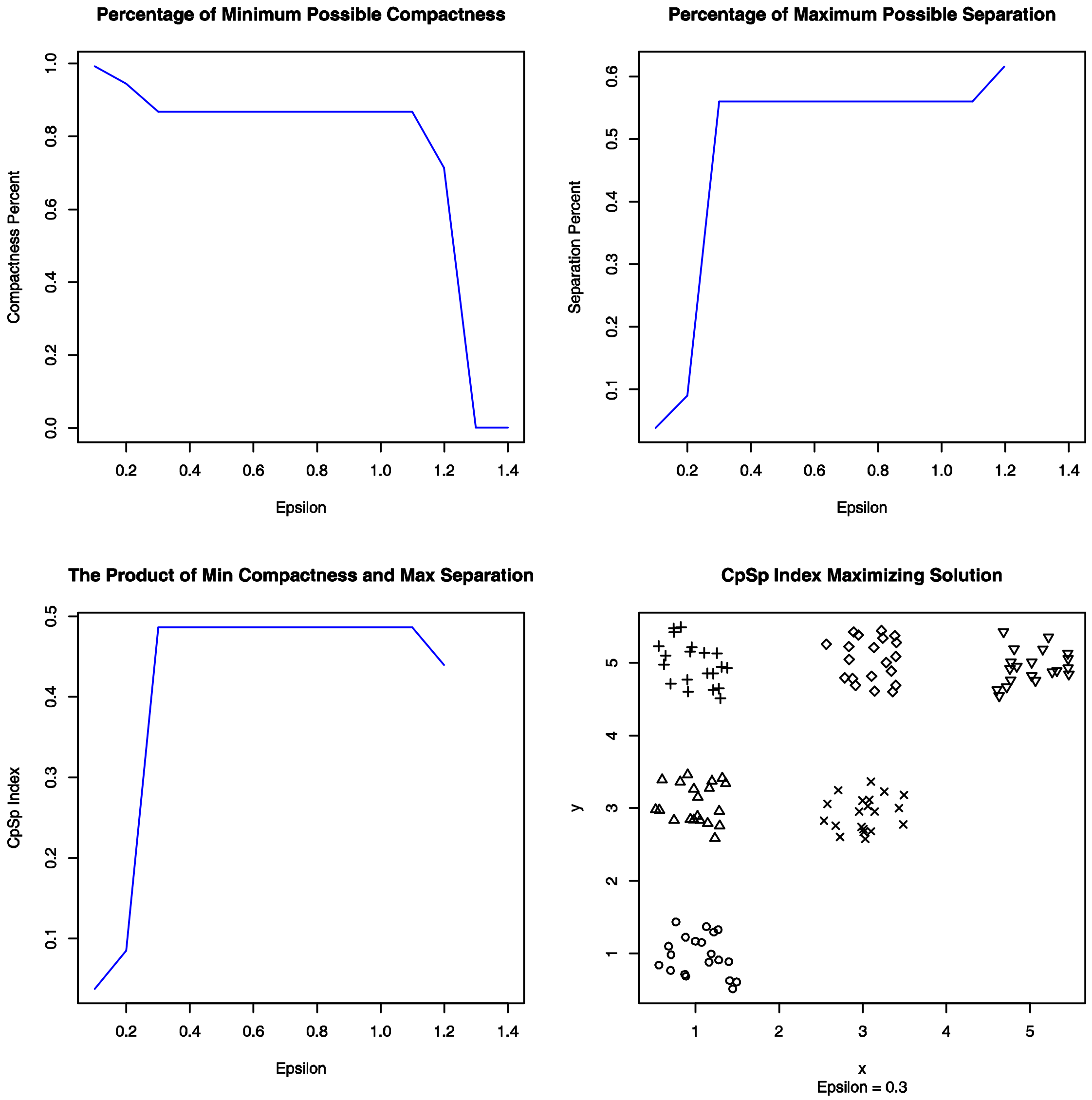


Figure 3.6 The *CpSp* Analysis with No Outlying points and MinPts = 2

solution corresponds to an epsilon radius value of 0.2, and contains 18 mutli-point clusters and 27 noise points (only two of which are the outlying points).

A close examination of Figures 3.8 and 3.9 indicates why this situation is occurring. Looking at the *Cp* plot (the northwest panel in both figures) reveals that the compactness measure is not influenced by the choice of MinPts for the DBSCAN runs. However, a similar examination of

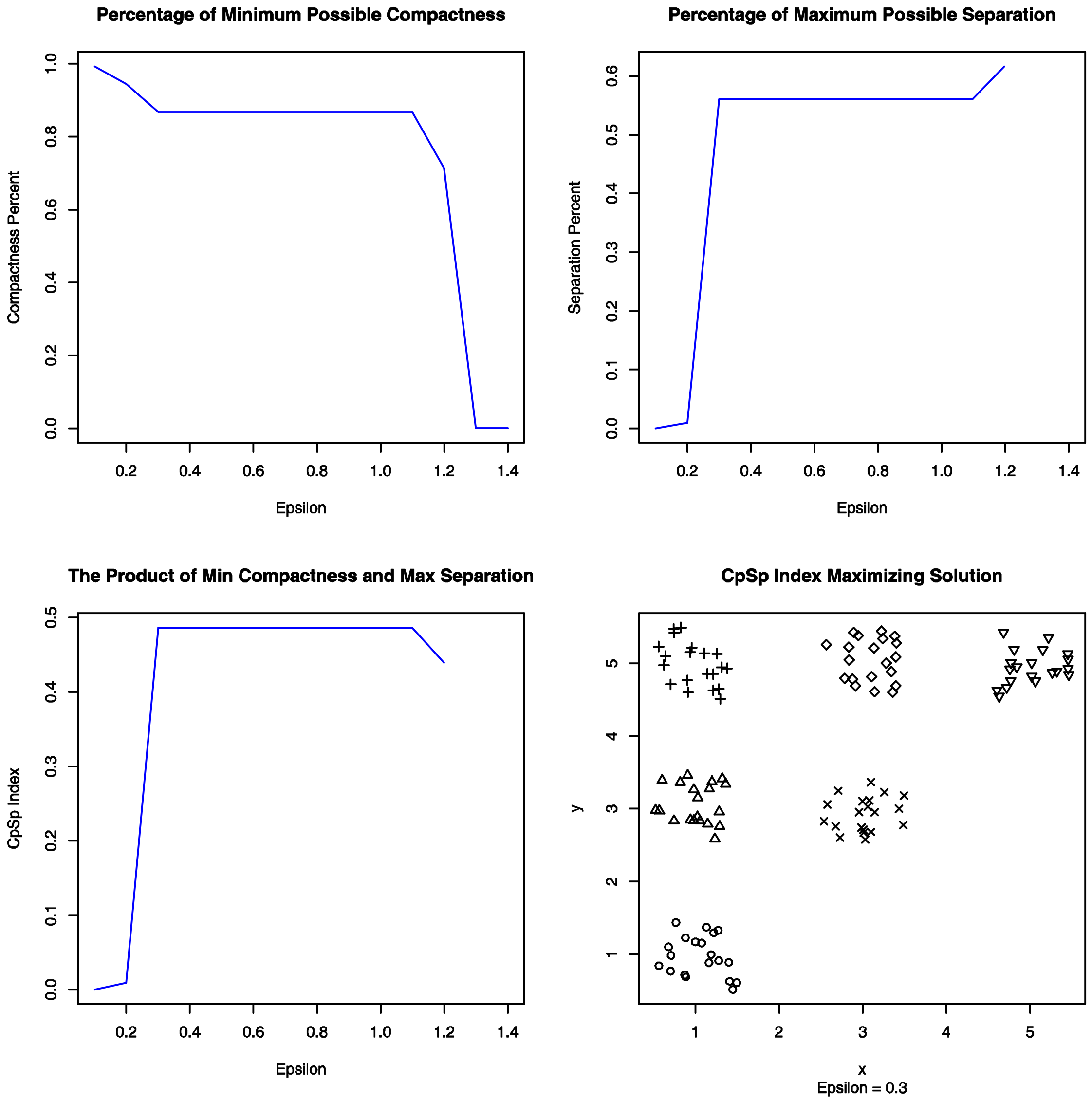


Figure 3.7 The *CpSp* Analysis with No Outlying Points and MinPts = 3

the *Sp* plot (the northeast panel in both figures) indicates that the outlying points have a profound effect on this measure. Specifically, the *Sp* value becomes “stuck” at a very low level (which can be seen by comparing the *y*-axis values between Figures 3.8 and 3.9 for the *Sp* plots in both figures) over an extended range. As a result of this, the comparatively small decrease in the *Cp* measure between 0.2 and 0.3 when MinPts is set to three for the DBSCAN runs drives the

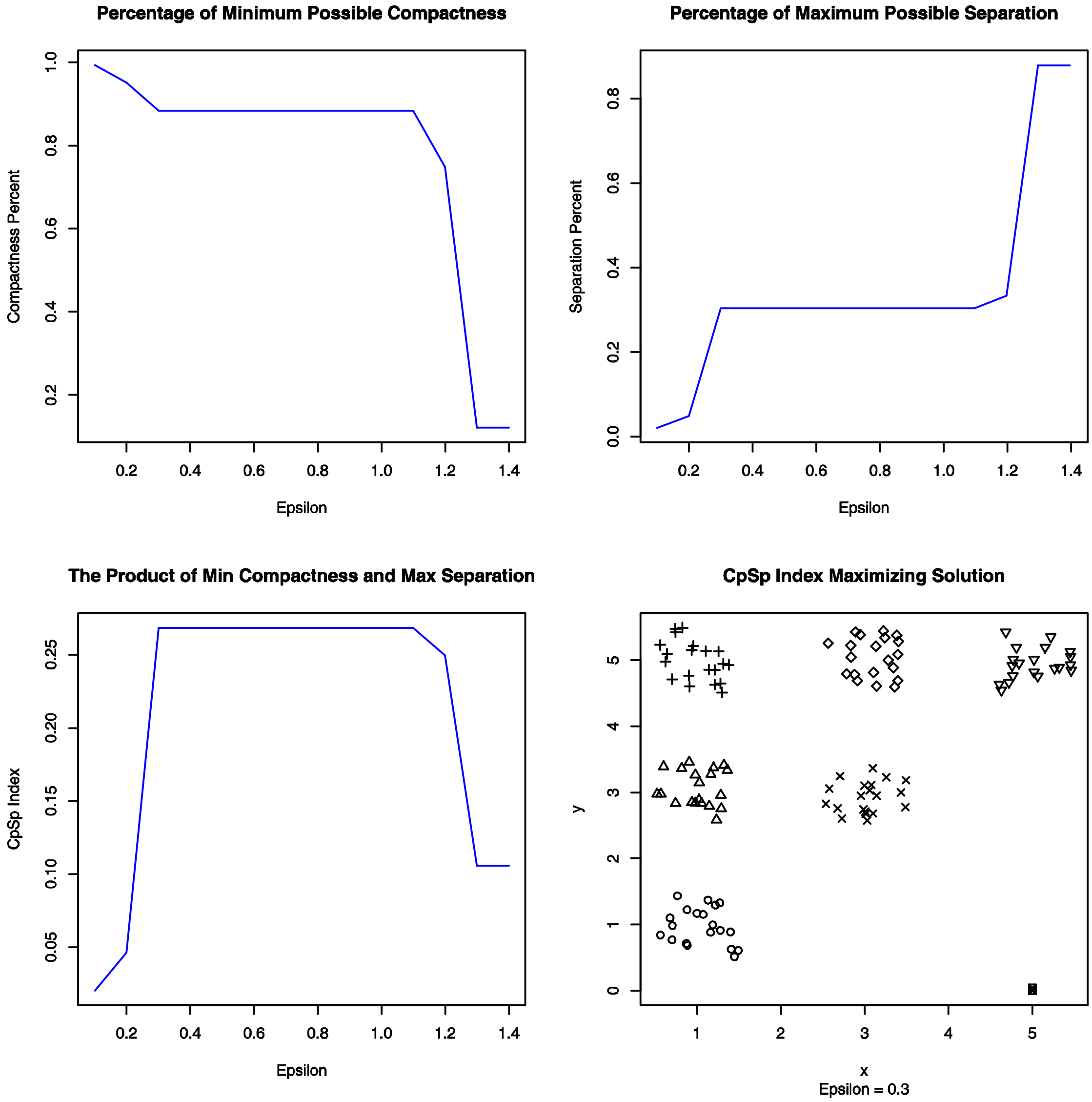


Figure 3.8 The *CpSp* Analysis with Outlying Points and MinPts = 2

solution since the outlying points prevent the *Sp* index from moving.

In their original paper introducing the DBSCAN algorithm, Ester et al. (1996) minimize the importance of selecting the value of MinPts, indicating that selecting a MinPts value of four should be satisfactory for most applications. A reason to downplay the importance of the

MinPts value is that when the length of the epsilon radius is held constant, the cluster structure for

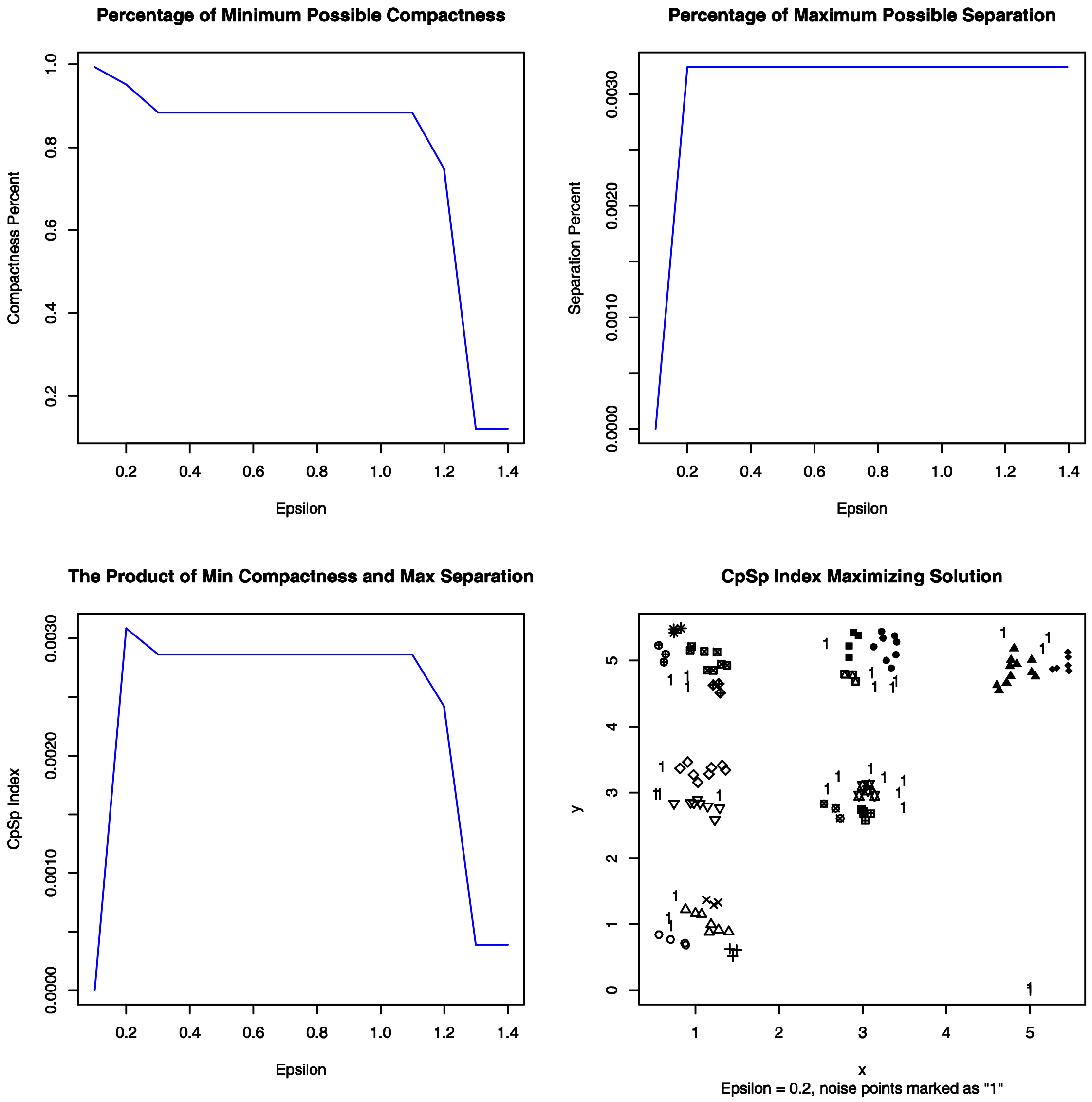


Figure 3.9 The *CpSp* Analysis with Outlying Points and MinPts = 3

clusters that have a number of clusters that exceeds the value of MinPts does not change with different values of MinPts. What occurs is that “marginal clusters”, those that had a number of members that equaled the MinPts limit, are broken into a set of noise points when the level of

MinPts increases (we illustrate this point in Figure 3.10 using our example data set when the epsilon radius is set to 0.3 and MinPts is set to 3). This, combined with our findings on how the selection of MinPts influences cluster validation measures, leads us to the conclusion that the appropriate length of the epsilon radius should be determined using a MinPts value of 2 for the DBSCAN run, and once this value is determined, the level of MinPts can be altered in an appropriate way for the application at hand.

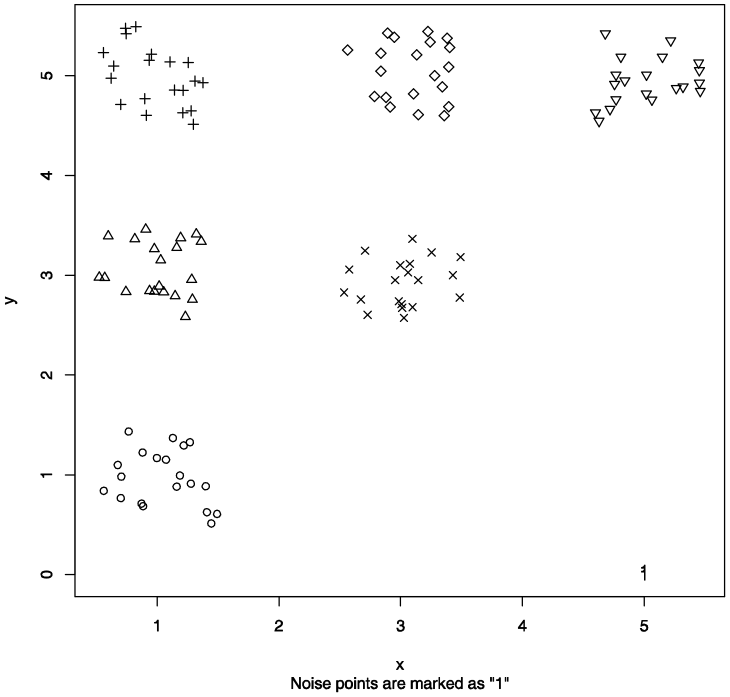


Figure 3.10 The Clustering Solution for the Example Data with Outlying Points when Epsilon = 0.3 and MinPts=3

Having both described different methods of clustering, and presented a new tool for validating cluster solutions that allows for cases where some or all of the clusters are non-convex, we next compare the performance of the different methods when applied to real world data. To foreshadow the results, it will become clear that even the improved tools are inadequate to address issues that arise in actual applications. The following chapter will introduce new tools that are up to the task.

4 An Illustrative Application of Current Approaches

As discussed previously, K-Means and Ward's method (a special case of the model-based approach) are the most widely used clustering algorithms in marketing, but the density-based approach is more flexible in terms of the non-convex clusters that can occur with spatial data, partially due to the need for objects (retail outlets in our case) to locate along a road network. It also appears to be better able to capture the network effect among retail outlets that leads to retail districts. Consequently, this chapter assesses the performance of K-means, Model-based clustering, and DBSCAN when applied to the problem of identifying retail districts in order to determine whether any of them are up to the task, and, if none are, examine the reasons why certain methods fail to assist in the development of more refined methods. As will be demonstrated, each of the three approaches is incapable of providing an acceptable representation of retail districts. Therefore, there is a need to develop a new approach for spatial cluster analysis, at least for retail district identification. Fortunately, our results suggest ways in which to improve the density-based methods for identifying retail districts in a particular geographic area.

For this application, we use retail outlet location data from the Capital Regional District of British Columbia, which corresponds to what is typically called the Greater Victoria Area.[4] Greater Victoria has a population of roughly 330,000 people, a land area of 695 km^2, and 2394 retail outlets. We have chosen Greater Victoria because it is large enough to have a dense urban core and surrounding areas of differing levels of retail outlet concentration, but at the same time it is small enough to be manageable. In particular, model-based clustering does not

[4] A regional district is administrative area that corresponds to what most jurisdictions in North America call a county. As a result, the data set consists of retail outlets both in the City of Victoria as well as a number of other outlying communities.

scale well (the underlying algorithm has $O(n^2)$ time) and, as a result, has limited usefulness for larger cities (such as Vancouver, which has over 16,000 retail outlets in its metropolitan area). Since Ward's method, which is a special case of model-based clustering, is one of the most widely used clustering methods in marketing, we decided to do our comparative testing on a city to which model-based clustering could be readily applied.

Data Conversion

The retail location data is from DMTI's "Enhanced Point of Interest" files, which are distributed on a province level basis in ASCII format, and are from the third quarter of 2005. Each file contains the name, a four digit SIC code that describes the location's primary activity, street address, city, postal code, and latitude/longitude coordinates.

From the British Columbia file the records associated with retail trade SIC codes were extracted, and were converted into shapefile format using the R maptools package.[5] The next step of the data preparation process was to attach a census subdivision identifier code to each location's attribute record, which was accomplished using a point-in-polygon method. Specifically, it was determined which census subdivision polygon each retail outlet fell into (through the use of Statistics Canada's census subdivision polygon shapefile for British Columbia), and the identifier code of that census subdivision was then attached to the attribute data for each retail outlet. The census subdivision identifier allowed us to easily assign a census metropolitan area code to each retail outlet. Following this, a new shapefile was created by extracting records only for those retail outlets located in the Greater Victoria census metropolitan area. This step was carried using the ogr2ogr utility in the GDAL library.

[5] The shapefile format was created by ESRI, and is the de facto standard format for vector GIS data layers. These files can be read and manipulated by a number of GIS programs and other software tools.

The last step in the process is to re-project the data from latitude/longitude coordinates (also called geographic coordinates) into UTM (Universal Transverse Mercator) zone 10N coordinates, which was also accomplished using the ogr2ogr utility. This step is necessary because the clustering tools we are using work with planar Euclidean distances between points, while the latitude and longitude coordinates give the location of a point on an ellipsoid representing the earth, and are measured in degrees.[6] The problem of converting geographic coordinates into planar coordinates is one that has been addressed by cartographers for literally centuries.[7] A commonly used planar coordinate system for geographic data is the UTM system. The reason for its popularity is that the projection from three dimensions to two dimensions is done with minimal distortion since it is based on dividing the earth into 60 zones running west to east, and providing a projection from geographic coordinates to planar coordinates for each zone and each hemisphere. Coastal British Columbia falls into the 10th UTM zone, and is located in the northern hemisphere; hence it is located in UTM zone 10N.

4.1 K-Means Cluster Analysis of the Data

When using the K-Means method, we must specify the number of clusters as an input parameter for the algorithm. Since we do not have any *a priori* information about the appropriate number of clusters for the 2394 retailers in Victoria, we tried every possible solution within a range of two to 150 clusters (assuming 150 clusters would be large enough to cover the optimal number of clusters). We then evaluated the performance for all of the possible cluster solutions using two clustering validation measures, the SD index of Halkidi et

[6] The earth is not perfectly ellipsoidal in shape. However, it is very close, and the datum our coordinates are based on (the North American Datum of 1983) assumes the earth is ellipsoidal (the underlying ellipsoid is the Geodetic Reference system of 1980).

[7] An introductory book on Geographic Information Systems (such as Clarke, 1997) provides much greater detail on the cartographic aspects of projections and coordinate systems.

al. (2001) and the *CpSp* index developed in the previous chapter. Figure 4.1 shows the SD index for the 149 solutions considered. An examinations of the figure reveals that the index takes an initial jump between two and four clusters, and then rapidly decreases until the eight-cluster solution is reached, and then begins to “chatter” over the remaining range of the data (from 9 to 150 clusters). The actual minimum point for the SD index (an index for which lower values are preferred) comes at the 141-cluster solution, with a value of 0.3094879 for the index. However, there are a number of values nearly as low (in particular, for the 58- and 150-cluster solutions) as the index chatters along.

Based on the SD index, there is no real clean solution for the data. Using the criteria that the “best” solution is the solution for which the SD index obtains its lowest value leads us to the 141-cluster solution. Figure 4.2 shows the clustering of the data for the downtown core of Victoria based on this solution, while Figure 4.3 shows the clustering of the data for an area with a very sparse concentration of retail outlets.

An examination of Figure 4.2 lets us clearly see the extent to which the K-means algorithm attempts to form circular clusters. However, it produces these circular clusters at the expense of creating essentially no separation between many of the clusters, while breaking up natural clusters that run along a street because the underlying “line clusters” are not circular in shape. At the same time, Figure 4.3 illustrates how the distances between cluster members become extremely large in areas with a sparse concentration of stores since the algorithm does not allow for noise points.

The *CpSp* based analysis of the K-means solutions can be found in Figure 4.4. This figure reveals that the implied “best” solution is very different than for the SD index based analysis.

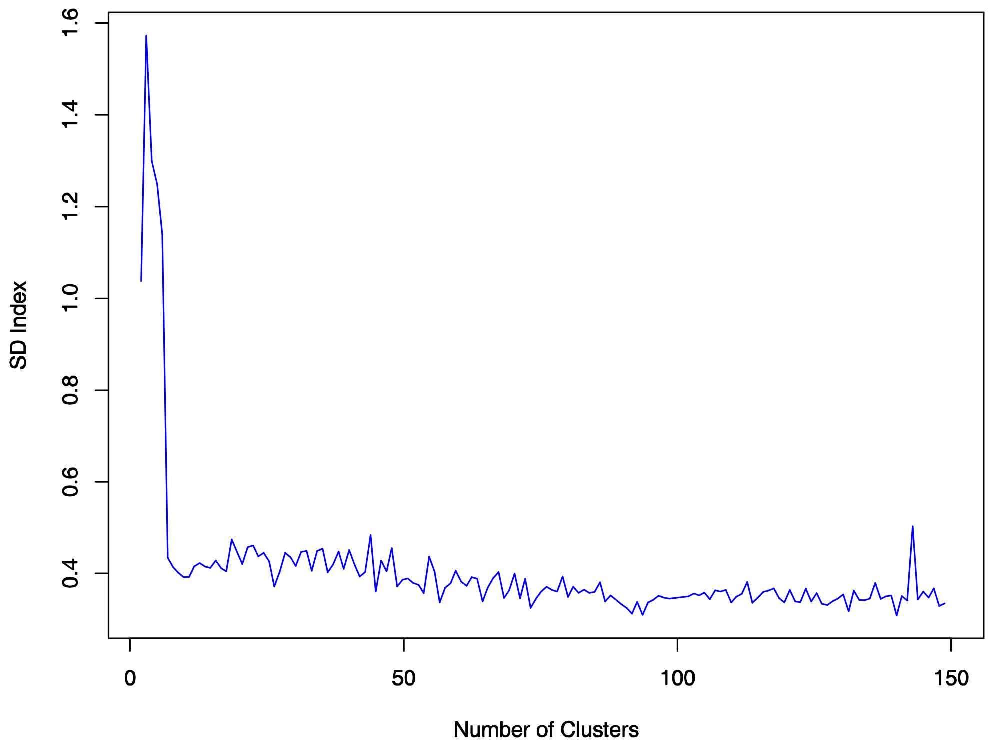


Figure 4.1 The SD Index Analysis of K-Means Solutions for Victoria

Instead of 141 clusters, *CpSp* indicates that the "best" solution corresponds to the three-cluster solution (which is shown in Figure 4.5), a solution consisting of three very large clusters, that is of little practical use given the extent of the agglomeration. What is interesting is that the *CpSp* index is much more definitive in its "judgment" with a fairly high peak for the three-cluster solution. A closer examination of the *Sp* index (northeast) panel of Figure 4.4 indicates, unsurprisingly, that it is the separation between clusters (or rather the lack thereof) that is driving this solution. Specifically, as we move from two to three clusters there is a drop in the *Sp* index to nearly zero percent of the maximum possible separation.

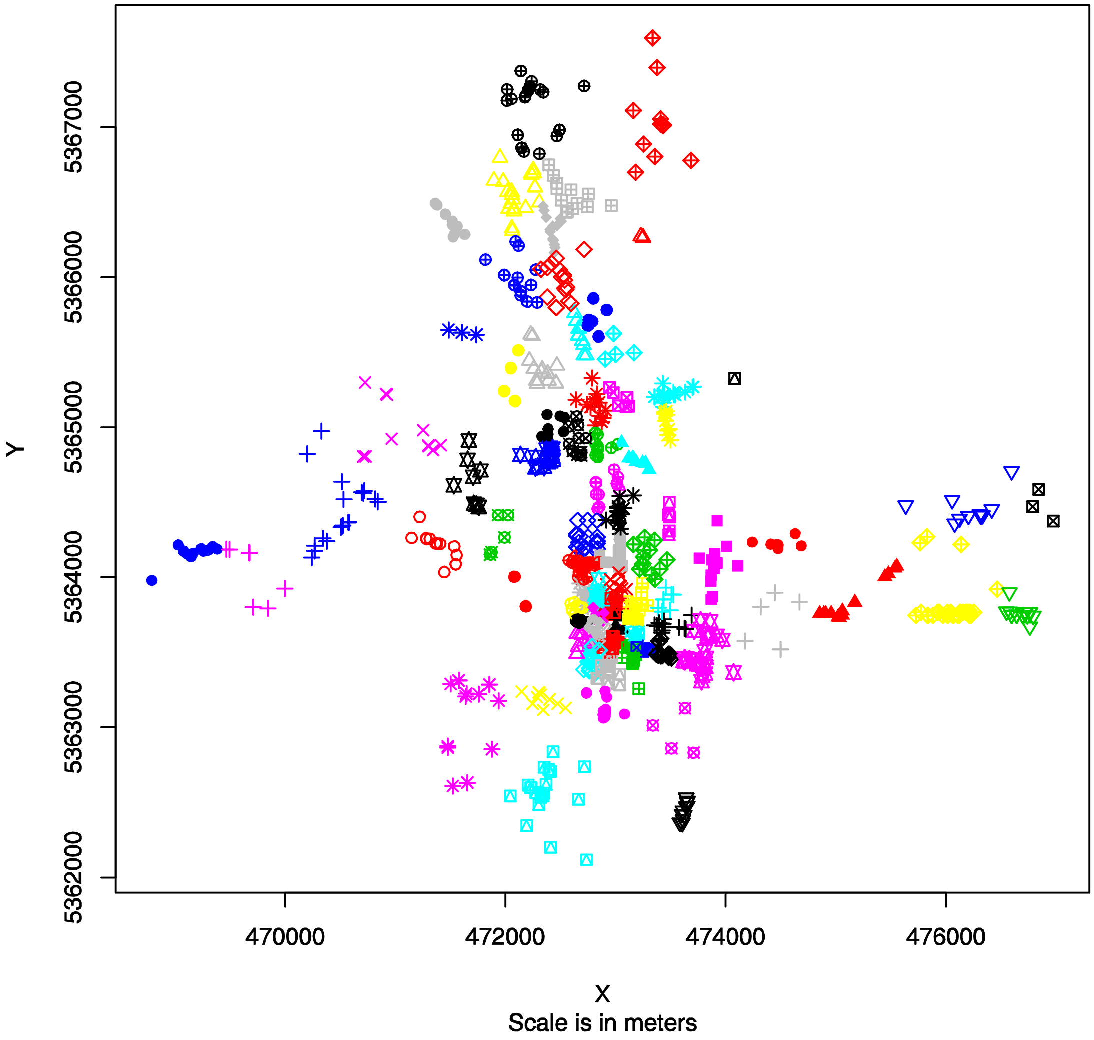


Figure 4.2[8] The 141-Cluster K-Means Solution for the Victoria Downtown Core

[8] In Figure 4.2 and elsewhere, X is the east-west coordinate with UTM zone 10N, and Y is the north-south coordinate, measured from the equator for locations north of the equator.

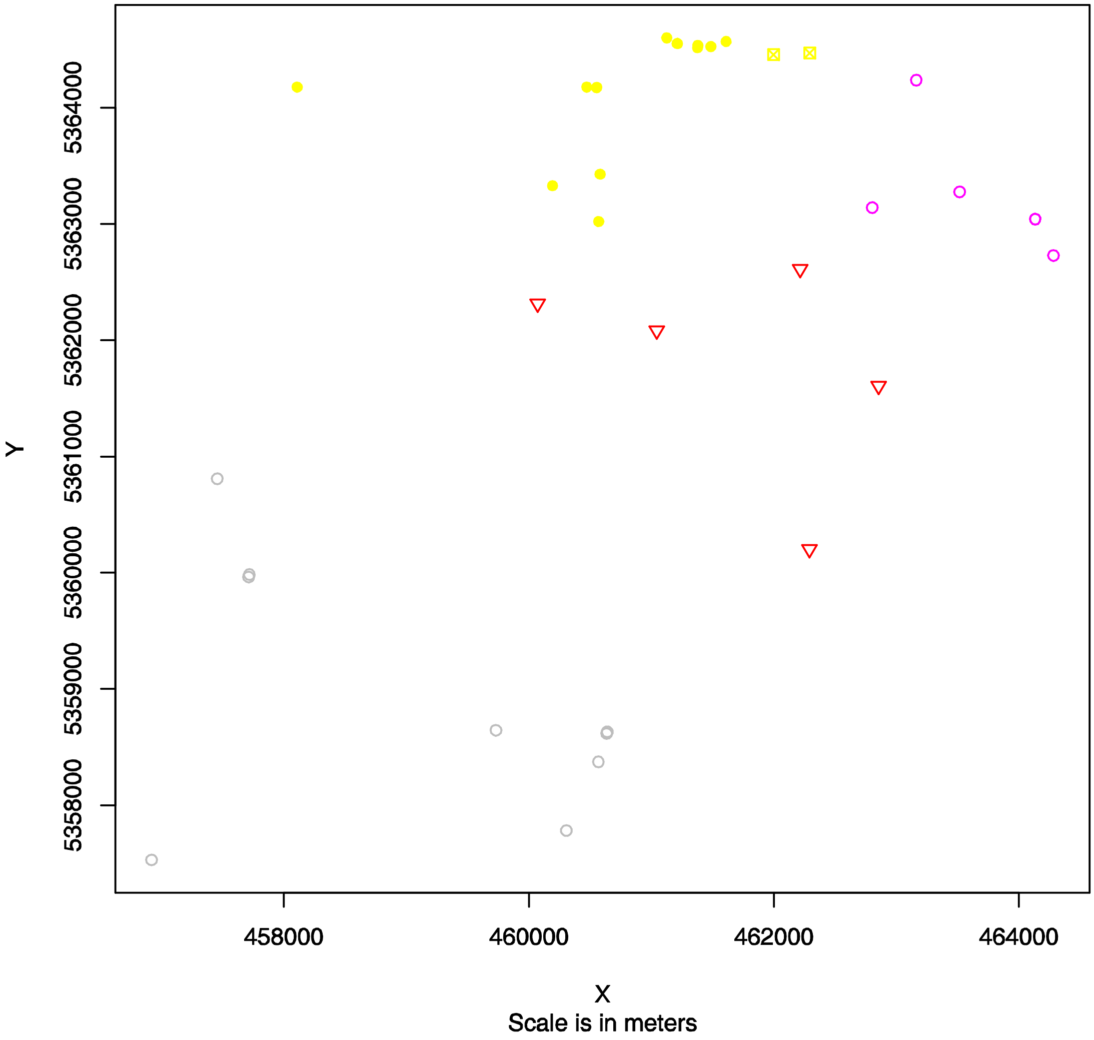
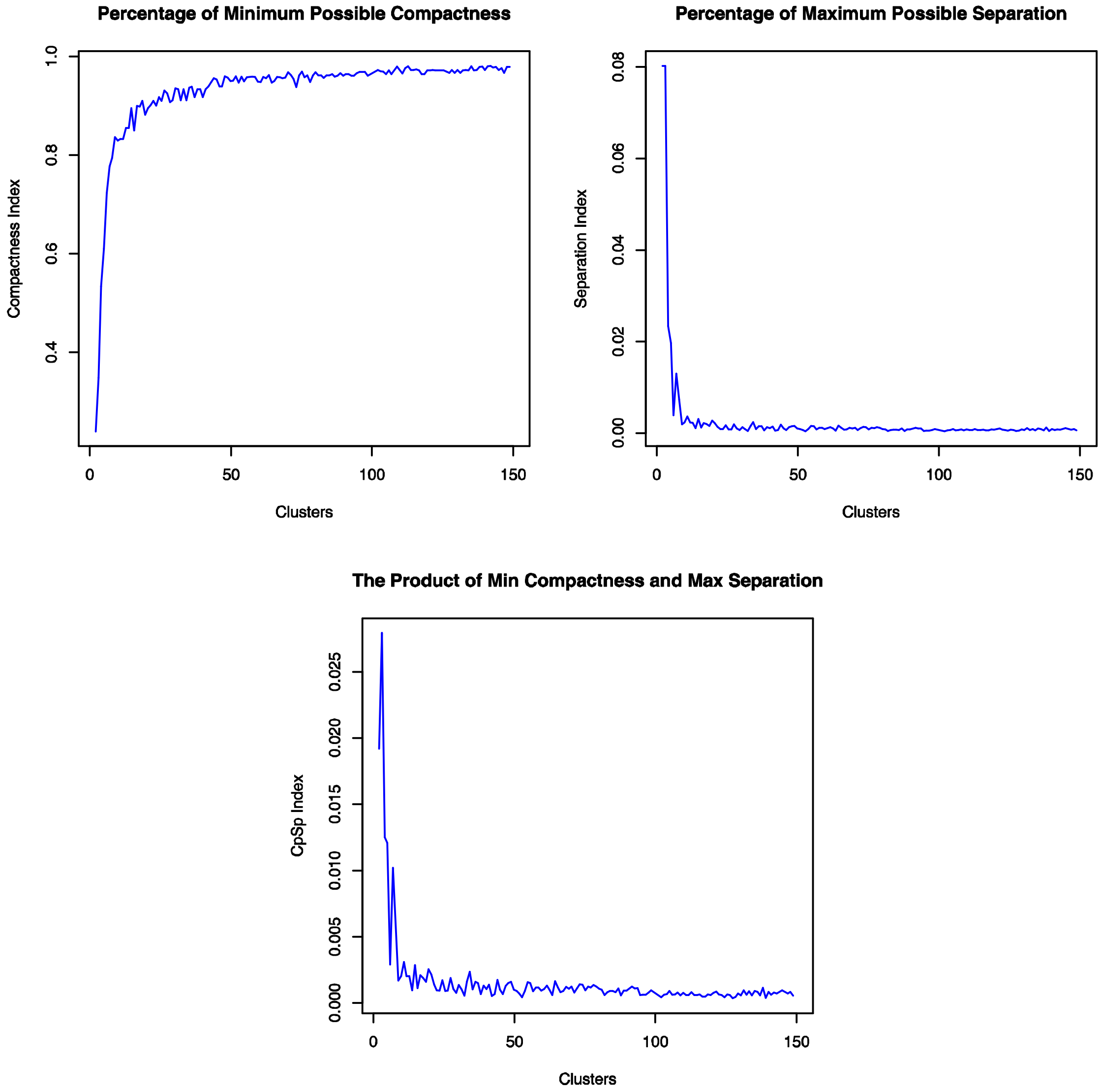


Figure 4.3 The 141-Cluster K-Means Solution for an Outlying Area of Greater Victoria


Percentage of Minimum Possible Compactness

Percentage of Maximum Possible Separation

The Product of Min Compactness and Max Separation

Figure 4.4 The *CpSp* Analysis of the K-Means Solutions for Greater Victoria

Overall, the analysis of the possible K-means solutions reveals that the use of this method is not at all appropriate in this type of setting. This can be seen in the widely varying values of the number of clusters that appear to optimal or near optimal, according to the criterion measure used. However, the problem for retail district data is more fundamental. As mentioned above, these results are primarily due to the maintained hypothesis of the algorithm that clusters are circular in nature, when in this instance this assumption is simply incorrect.

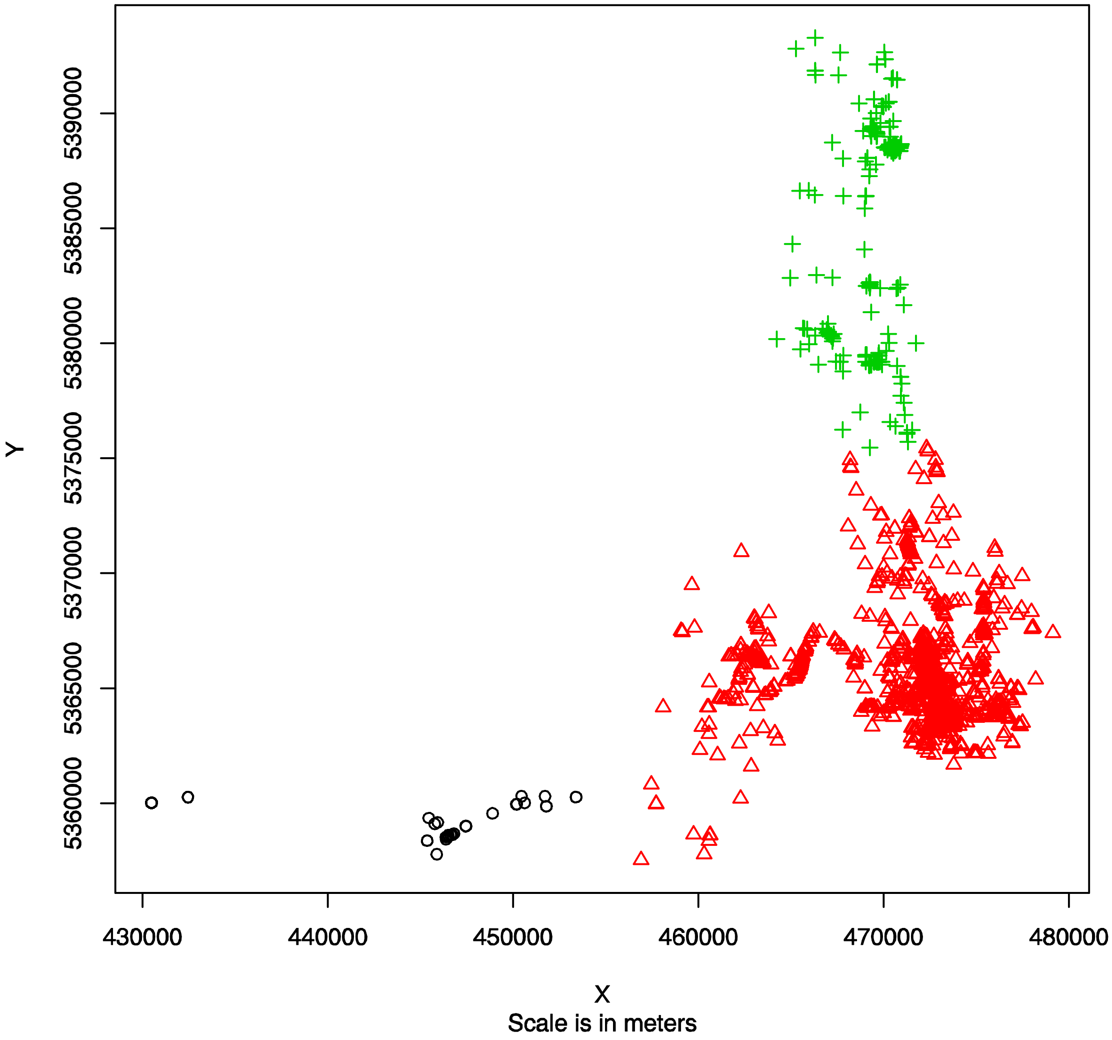


Figure 4.5 The 3-Cluster K-Means Solution for Greater Victoria

4.2 Model-Based Clustering of the Data

Ward's method is one of the most widely used clustering approaches in marketing. As noted above, Ward's method is a special case of model-based approach, an approach that can determine the optimal cluster solution (in terms of both the number of clusters and the model it is based on) using of the Bayesian Information Criterion, or BIC (Fraley and Raftery 2003). The models differ in their assumptions concerning the volume, shape, and orientation of the clusters. As a result, the method has the potential to provide a better clustering solution than K-Means (albeit, it still attempts to form convex clusters, just much more flexible ones), but is closely related to methods traditionally used in marketing applications.

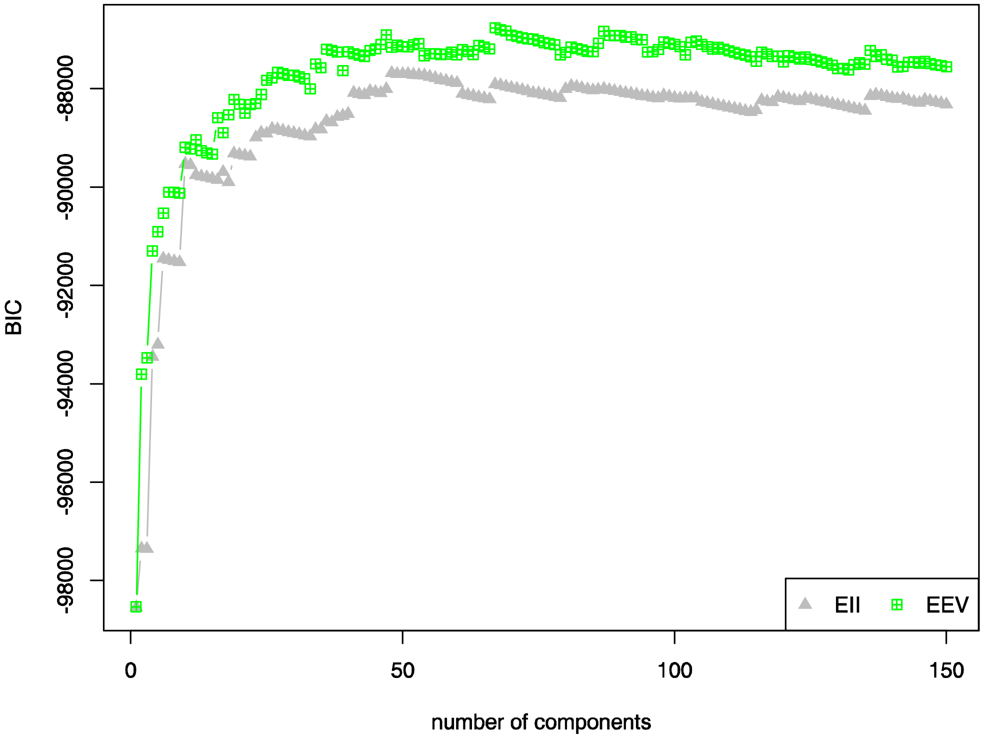


Figure 4.6 The EEV and EII Model-Based Clustering Results for Victoria

As with our K-Means analysis, we set the possible number of clusters in a solution from two to 150, and the algorithm examines ten different underlying models. Based on the BIC, the "best" model for the data is the EEV[9] (equal volume, equal shape, and variable orientation) model for a solution with 67 clusters. Figure 4.6 shows the BIC value for this model across the possible different number of clusters, as well as the same information for the EII model (which corresponds to Ward's method). For model-based clustering, larger values of the BIC are preferred to smaller values. An examination of this figure reveals that the BIC for both models

[9] The first letter stands for volume of the clusters, the second for shape of the clusters, and the third for orientation. E stands for equal, V stands for variable, and I stands for identity. A complete discussion of the different possible models that can be generated in model-based clustering can be found in Fraley et al., (2003). The statistical function used for this analysis (the R mclust library) tests ten different model structures.

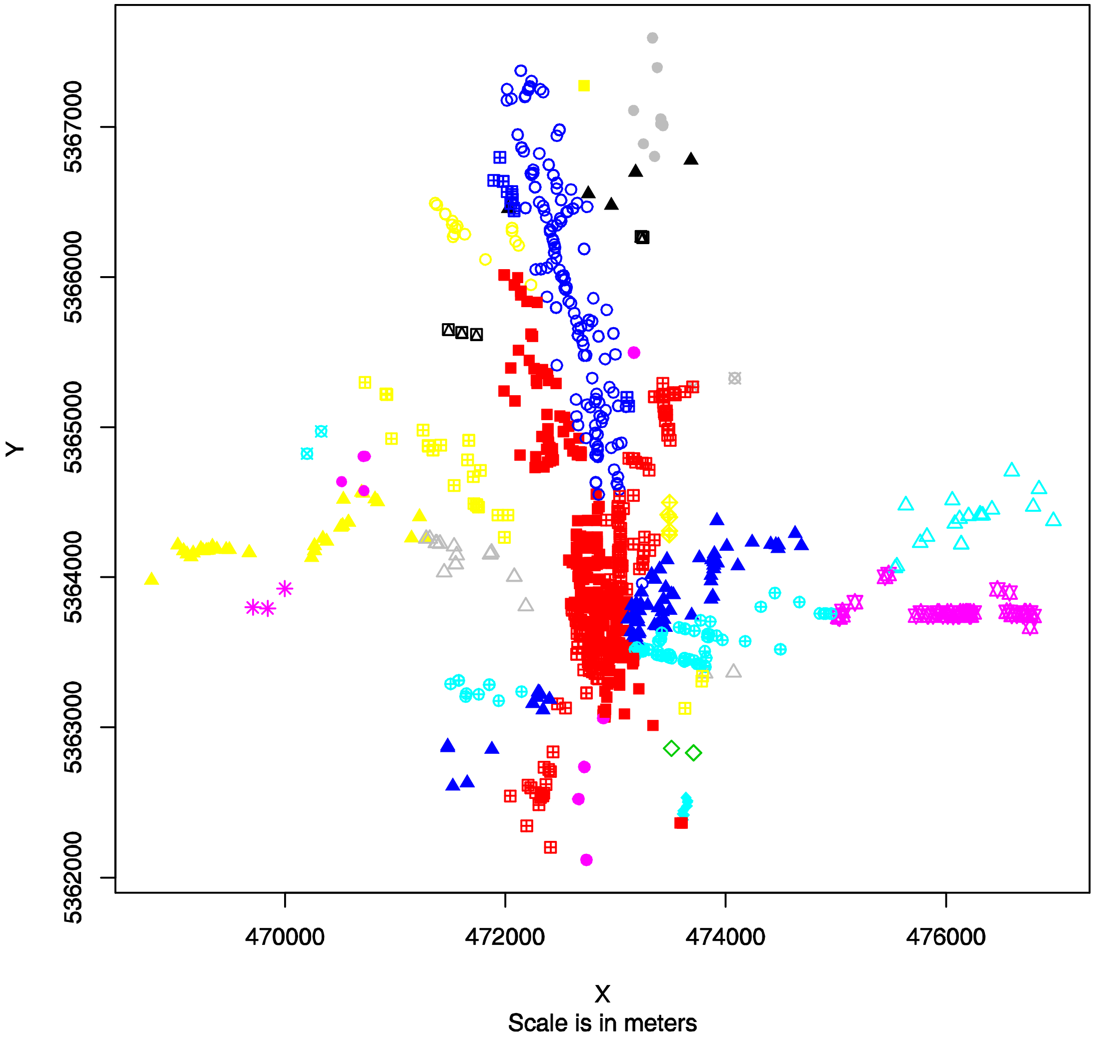


Figure 4.7 The 67-Cluster EEV Solution for the Victoria Downtown Core

rises fairly rapidly as the number of clusters in the solution increases, quickly reach a plateau (with the EEV having a higher value for the BIC within the plateau), and then begins to “chatter,” much as we found for the SD index for the K-means solution. As a result, the figure reveals that the EEV model is preferred over the EII model, but there is not a definitive indication of the appropriate number of clusters to use in the solution.

Based on the largest BIC value, we select the EEV model's 67-cluster solution as the one to examine in more detail. Figure 4.7 shows the clustering pattern produced by this solution for Victoria's downtown core. An examination of the figure shows that most of the clusters are fairly ellipsoidal in nature. However, they are still incapable of adequately capturing "string clusters" that run along a street in a road network, and, therefore, do not match the "ground reality." As with the K-Means results, the cluster separation seems extremely minimal for larger numbers of clusters. Also as with the K-Means methods, the outliers cannot be identified in this application. In addition to the above two weaknesses, and consistent with it being an algorithm that runs in $O(n^2)$ time, it also took a relatively long time to perform the model-based analysis for Victoria (more than two hours on a machine with an Intel Pentium Dual Core Processor with a clock speed of 2Ghz and 4GB of 667 MHz memory), when the maximum number of clusters was set at 150. If the number of points becomes much larger, such the over 16,000 retail outlet points for Vancouver, the use of the model-based approach becomes impractical.

4.3 The DBSCAN Clustering of the Data

Some of the advantages of DBSCAN (Ester, et al., 1996) are that outliers (or "noise points") can be identified, there is no shape restriction on the clusters (such as a convexity), and the computational efficiency is fairly good (being on the order of $O(n\log(n))$). In addition, the way the method determines the cluster solution through density connectivity (as discussed in Chapter 2) is similar to the way a customer walks or drives in a retail district. Consequently, the way the retail districts are formed using DBSCAN actually mimics the underlying network effect among retailers within a district. Therefore, we expected that DBSCAN would outperform both the K-Means and model-based approaches.

We ran the DBSCAN algorithm on the Greater Victoria data with epsilon radius values that varied from 20 to 10,000 meters, in increments of 20 meters (a total of 500 separate solutions). Given our discussion in Chapter 3, MinPts for the runs were set at two. It took roughly an hour of computer time to obtain all 500 solutions.

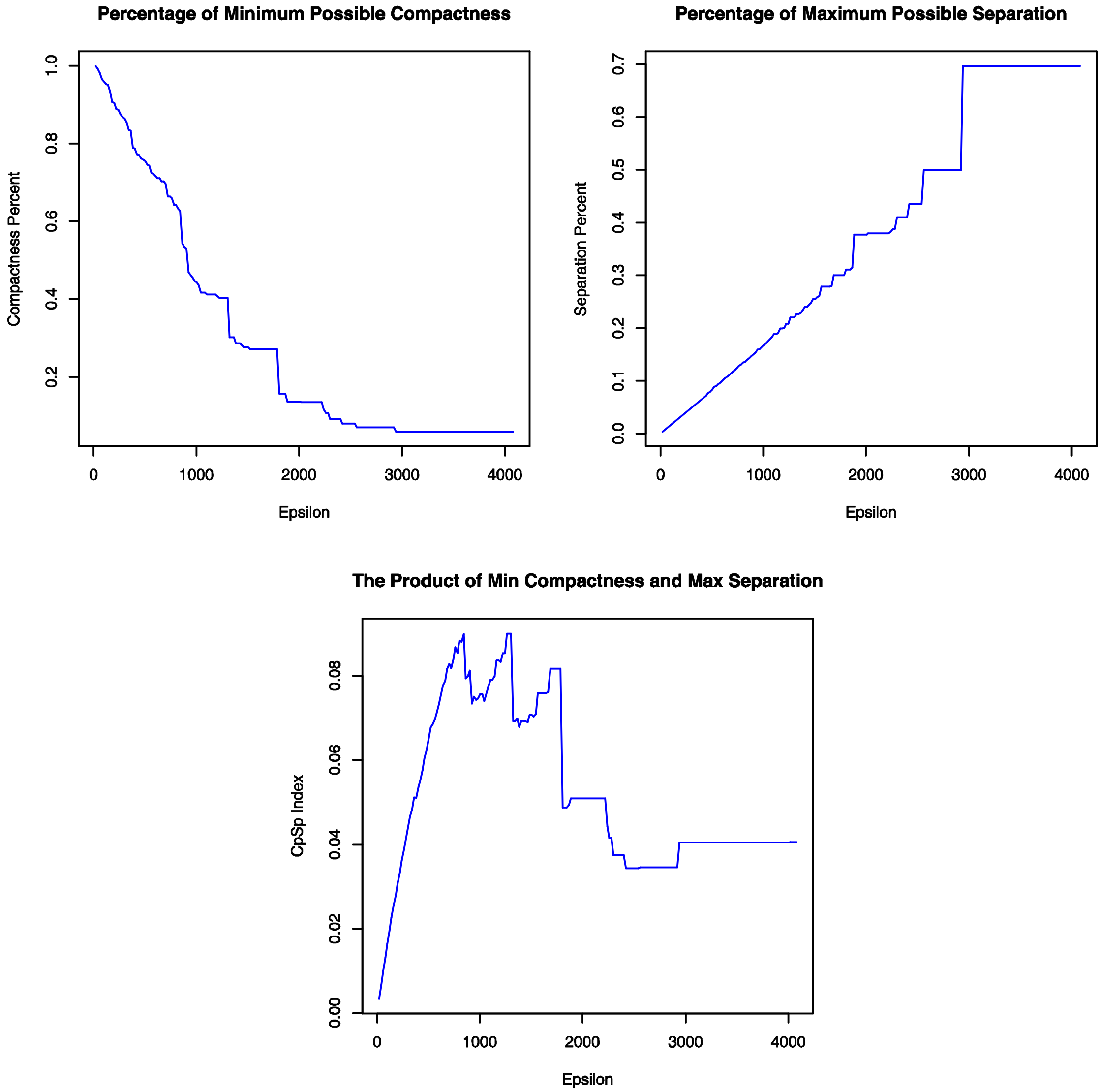


Figure 4.8 The CpSp Analysis of the DBSCAN Clustering of Greater Victoria

We used the *CpSp* index to determine the length of the epsilon radius since it balances both compactness (inside density) and separation (outside density) to evaluate the performance of the possible cluster solutions. The *CpSp* index indicates that the “best” solution corresponds to an epsilon radius of 4100 meters, resulting in just two clusters, one with three points, and the other with all the remaining points. It is readily apparent that with 2394 retail outlets distributed

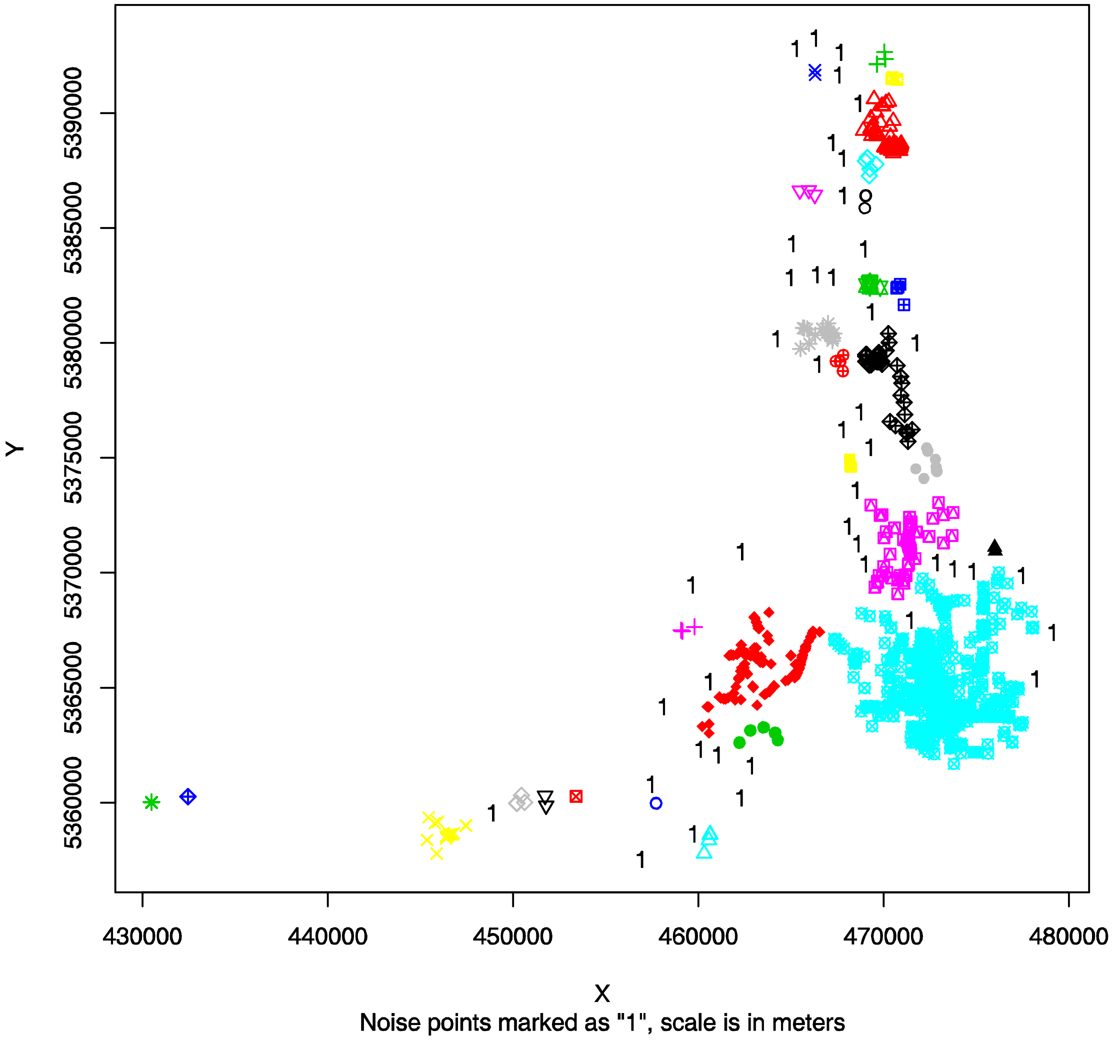


Figure 4.9 The 26-Cluster DBSCAN Solution for Greater Victoria

unevenly in the Victoria area, neither the number of clusters nor the epsilon radius of 4100 meters is reasonable. Therefore, we conducted a second analysis within the one large cluster, and obtained a more reasonable (but still imperfect) result with an epsilon radius of 840 meters, resulting in 26 clusters.

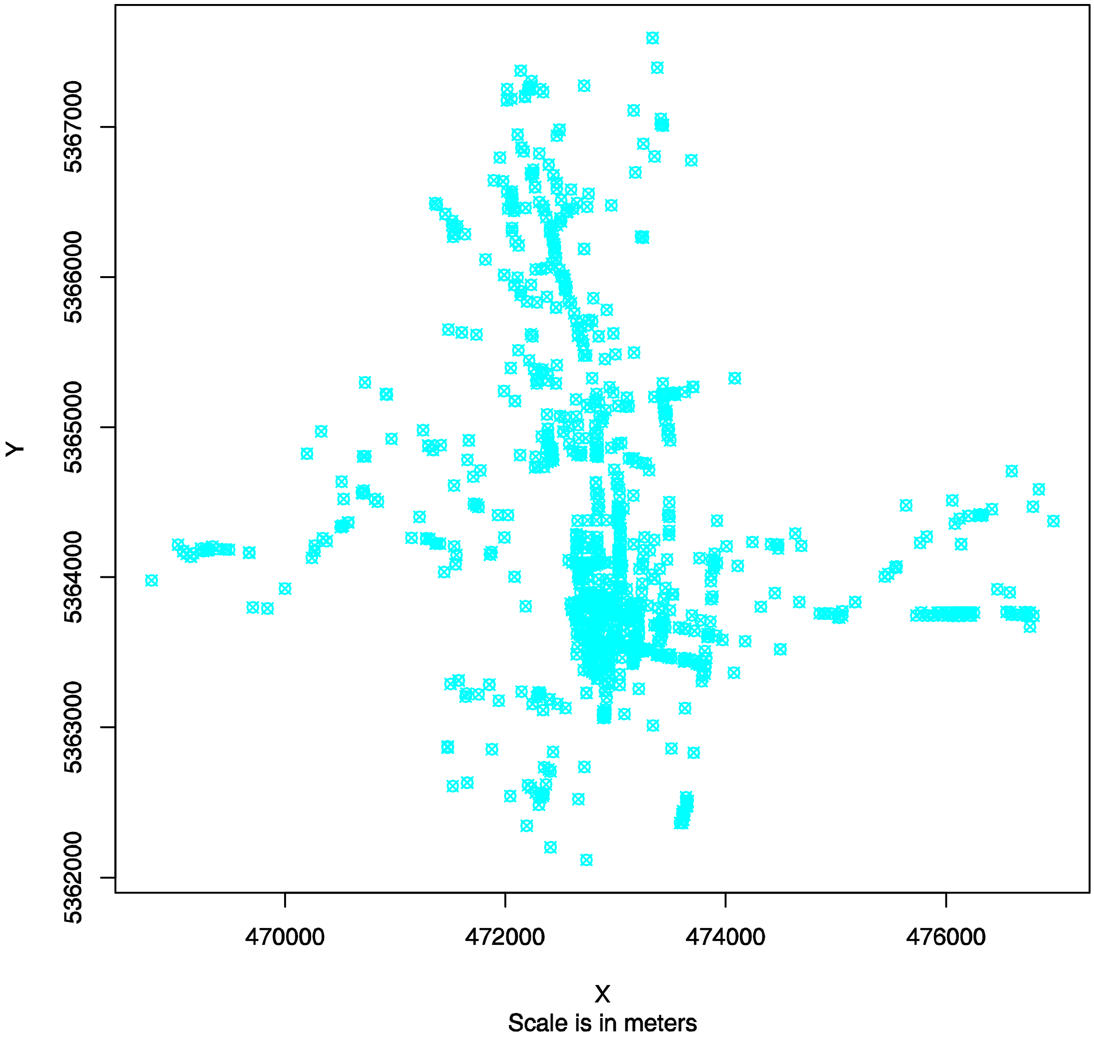


Figure 4.10 The 26-Cluster DBSCAN Solution for the Victoria Downtown Core

Figure 4.8 shows the *CpSp* analysis after the outlying three stores were deleted from the data, Figure 4.9 illustrates the clustering solution obtained with an epsilon radius of 840 meters for all of Greater Victoria, while in Figure 4.10 we show (in more detail) the same solution for the clustering for the Victoria downtown core. Figure 4.9 reveals that this solution generates clusters, as well as a number of noise points. However, the clusters that are created tend to be over agglomerated. In particular, as Figure 4.10 reveals, the downtown core of Victoria has been agglomerated into a single cluster, something that is clearly undesirable.

The key reason, we believe, for our disappointing result is that the value of 840 meters is very large, especially for the downtown area. As we examine the geographic distribution of stores in the greater Victoria area visually, it is apparent that there is a large variation in the density of stores over the area. However, the fixed epsilon radius in the density-based approach cannot reflect the retailer density variation, and appears to take on a high value to accommodate the data structure. In addition, as was indicated by the drastic change in the clustering solution when three points were removed from the data set, the existence of distant outlying points has an enormous impact on the clustering solution. While disappointing, this latter effect is not surprising. What is driving this is the ability to obtain a large minimum separation (driving the value of *Sp* upwards) when there are large distances between outlying points. Taken together, these two factors result in some overly large clusters in terms of the number of stores and geographic area covered in some areas.

We initially expected these problems to be avoided by the density-based approach. In this case, however, the density-based approach results in an epsilon radius that is too large for the more densely distributed stores in the downtown area. For reasons that are different, but with results similar, to those found using the K-Means and model-based approaches, the density-based approach cannot successfully handle the spatial clustering for this complicated data set with large variations in observation density. As discussed next, we look to improve the density-based approach in order to overcome the problems identified here.

4.4 Existing Problems and Potential Solutions

As the clustering results for Victoria indicate, none of the three approaches (K-Means, model-based, and density-based) performed well on a fairly large data set of store locations in a mid-size metropolitan area. The K-Means approach simply seems inappropriate given its underlying assumption to capture a pattern in which retail districts form fairly thinlines that run along a road network. As a result, it does not have a stable clustering solution across validation methods. The model-based approach, despite allowing for much more flexible cluster shapes, also seems incapable of capturing the "strings" of stores that tend to characterize many retail districts, at least for Greater Victoria. More importantly, its computational requirements limit the usefulness of the approach since it cannot be used in larger metropolitan areas.

The current density-based approach can mimic the thin, string-like structure of the data, and allows for some points to remain isolated (becoming a noise point). However, it assumes a fixed epsilon radius value, and the *CpSp* validation index is very sensitive to distant outlying points, which leads to unacceptable results in this case.

We believe that the density-based approach can be improved upon in a way that makes it suitable for the problem at hand. As we have already implicitly suggested, there are two underlying issues. First, while we have so far treated Greater Victoria as a monolithic entity, it is really a region that contains a number of different "communities." Communities could be municipalities such as Esquimalt or Saanich in Greater Victoria, they could be well defined neighborhoods within a municipality (such downtown Victoria), or they could even consist of abutting "neighborhoods" of different municipalities.[10] Downtown and urban communities often have a higher population density, and hence a very different "scale," than the areas

[10] While we do not know of a cross-municipality "community" in Greater Victoria, the Kingsway corridor that lies partially in Vancouver and partially in Burnaby in the Greater Vancouver metropolitan area.

surrounding them. As a result, it is likely to be more appropriate to create clusters at the level of the community rather than at the level of the metropolitan area. In addition, there is likely to be heterogeneity even within communities, likely due to factors such as local population density, which will have a within community effect on cluster structure.

To illustrate these two points, Figure 4.11 provides a plot of the retail outlets coded by the total population density (based on the sum of both the local residential and workforce population) of the census designation area in which each outlet is located, while Figure 4.12 provides the same plot, but is focused on the downtown core of Victoria. An examination of Figure 4.11 reveals that, unsurprisingly, there is a very strong relationship between the total population density of an area and the concentration of retailers in that area. Moreover, it strongly suggests that the concept of densely populated communities separated by comparatively less populated areas is consistent with “ground truth” (at least in Greater Victoria). The notion that within communities the “micro-level” retail density (and hence cluster structure) is likely to vary with population density can be clearly seen in Figure 4.12. In the next chapter we will present an approach that addresses these two issues.

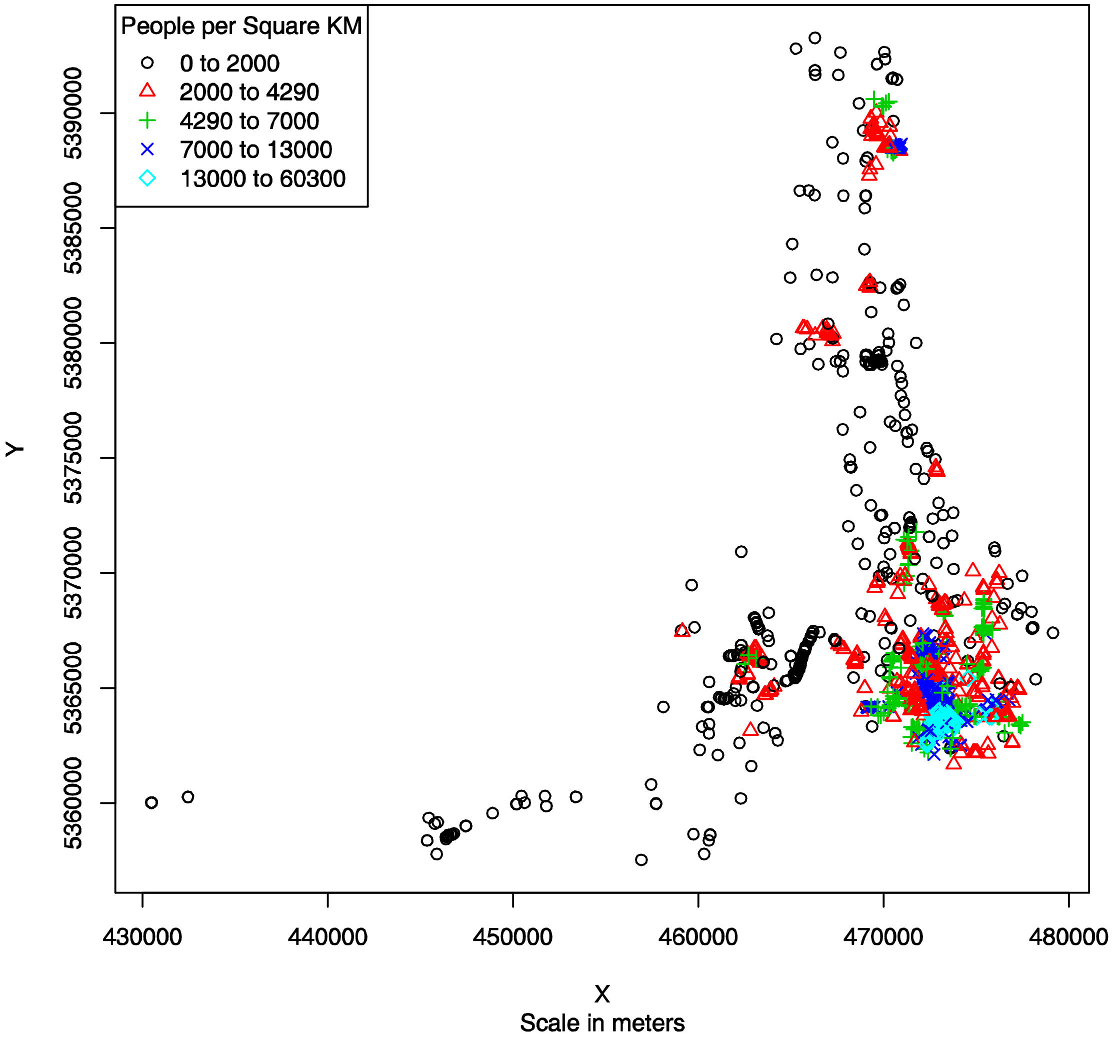


Figure 4.11 The Population Density for Retail Locations in Greater Victoria

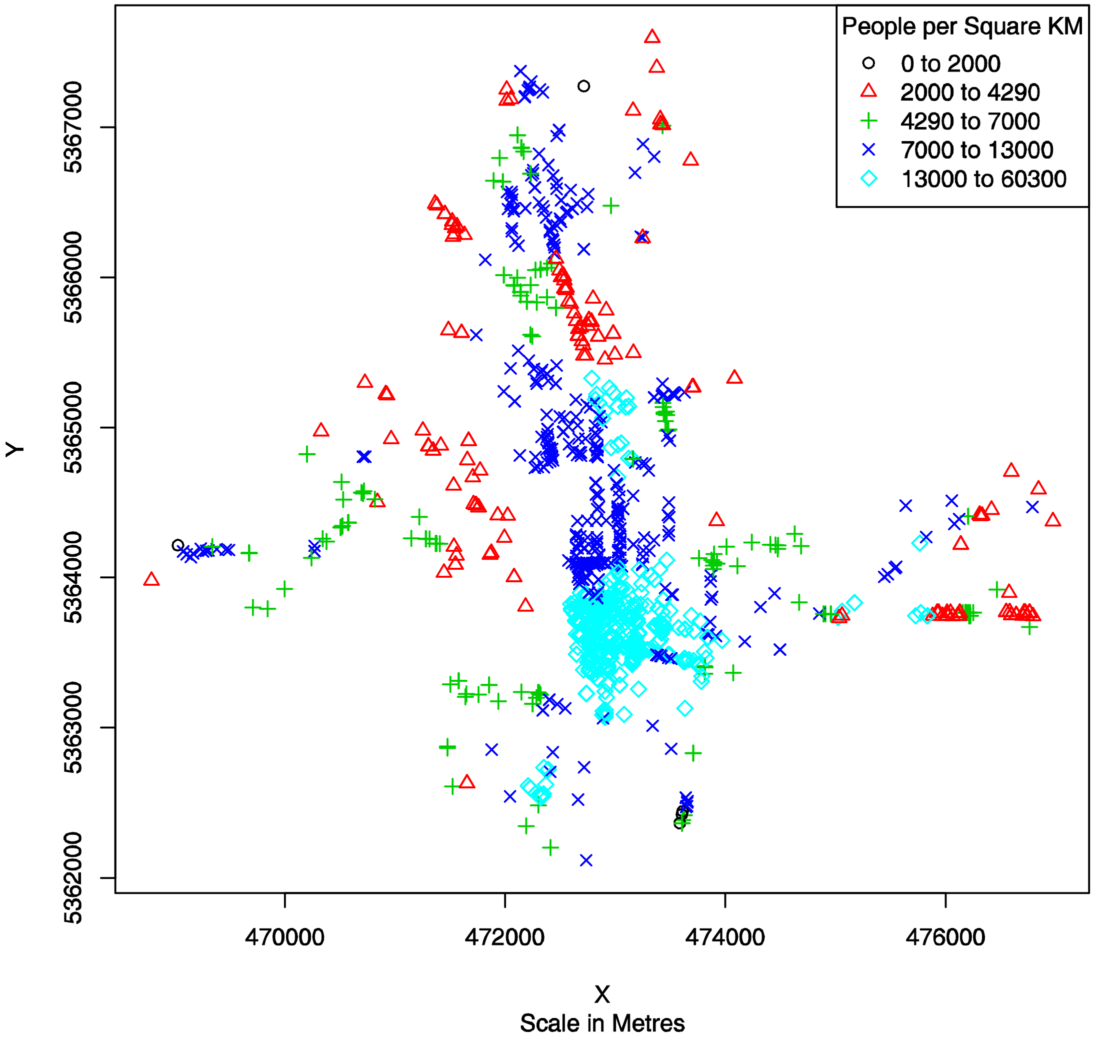


Figure 4.12 The Population Density for Retail Locations in the Victoria Downtown Core

5 A Two-Step Approach to Spatial Density Clustering

At the end of the previous chapter we discussed the issues of the importance of “communities” in determining the appropriate “scale” of an area to be clustered and the potential need to account for heterogeneity within communities (often induced by factors such as “micro-level” population density differences) on cluster structure. In this chapter we present a two-step approach to spatial density clustering that addresses both of these issues. In the first step a set of “communities” are determined via a graph theory based method. The second step consists of clustering the communities identified in a way that allows for intra-community heterogeneity to have an influence on cluster formation. To do this, we generalize the DBSCAN method of spatial density clustering by allowing the epsilon radius to vary in size with a micro-level factor, such as total population density, in a systematic way. We call this generalization of DBSCAN, VESDC, which is an acronym that stands for variable epsilon spatial density clustering.

While in application we would create “communities” first, and then cluster within communities, in this chapter we will reverse this order and first present the VESDC method along with a number of surrounding issues, and then describe our proposed method for determining communities in the first step. In addition to the development of the methods, we also demonstrate the use of the methods with simulated data to allow the reader to gain some understanding of the performance capabilities of the proposed methods.

5.1 Clustering Within Communities (Step 2)

In this section we present the VESDC method and then present an extension of the *CpSp* cluster validation method to assist in determining the settings of any underlying parameters needed to link the size of the epsilon radius to some underlying factor. An extension to the *CpSp* method is needed since in some sense, when we allow the epsilon radius to vary, we are locally rescaling the space. In more concrete language, we are effectively assuming that a meter of distance is in some sense "longer" in a densely populated area than it is in a sparsely populated area. For the case of population density, this notion has a great deal of intuitive appeal since we would expect that driving times per unit distance are longer in densely populated areas compared to less densely populated areas, and we would also expect that the use of walking (as opposed to driving) as a mode of transportation would be higher in densely populated areas. In turn, these differences should have an influence on the structure of retail districts.

The need to extend the *CpSp* method arises because the measures of compactness and separation are unique to a particular assumed scale, so when we compare two different clustering solutions (that vary slightly in the scaling implied by the different set of epsilon values used for the two solutions) we could potentially be making an apples and oranges comparison. As a result, there is a need to make one solution "conform" to the scale of the solution it is being compared against, and obtaining this conformity in the underlying scale is what the extension attempts to accomplish. Conversely, and perhaps surprisingly, it turns out that extending the basic DBSCAN method to allow different points to have different epsilon radius values is fairly straightforward. In addition to developing these new methods, we also provide three applications (to simulated data sets) in ordering to gain some sense of how well the methods perform before applying them to "real world" data in the next chapter.

5.1.1 Variable Epsilon Spatial Density Clustering

As indicated above, the VESDC method represents a simple generalization of the DBSCAN method of clustering. In the paper that introduces DBSCAN, Ester et al., (1996) call the length of the radius around a point used to define an epsilon neighborhood "Eps" (while we have called it "epsilon"), in this sub-section we will use their notation for consistency and ease of comparison. The other important aspect of notation in Ester, et al., (1996) that needs to be discussed is $N_{Eps}(p)$, which gives the set of points that are within the epsilon neighborhood of point p. The definition of this set is

$$N_{Eps}(p) = \{q \in D \,|\, dist(p,q) \le Eps\}, \tag{5.1}$$

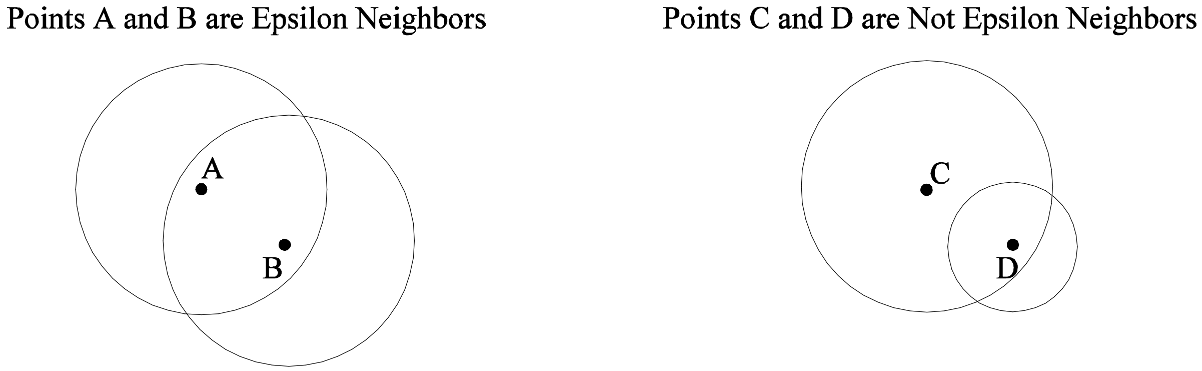
where D is the set of points in the data set, and $dist(\bullet)$ is a distance function (e.g., Euclidean, Manhattan, etc.).

An extension to allow for a variable epsilon radius requires that we make epsilon a vector for which all elements are strictly positive (how this is done is an application implementation detail), and redefine the set of points that are in the epsilon radius of point p ($N_{Eps}(p)$) as

$$N_{Eps}(p) = \left\{q \in D \,|\, dist(p,q) \le \min\left(Eps_p, Eps_q\right)\right\} \tag{5.2}$$

where Eps_p is the epsilon radius for point p and Eps_q is the epsilon radius for point q. In words, what this means is that for points p and q to be epsilon neighbors, it must be the case that point p is no further away from q than Eps_q, and point q is no further away from p than Eps_p. Figure 5.1 illustrates two situations, one where A and B are epsilon neighbors, and one where C and D are not. After making these two changes, the remaining five definitions and the two lemmas that validate the correctness of their algorithm follow directly. In addition, if $Eps_p = Eps_q = Eps$, then (5.1) and (5.2) return identical results. Consequently, we can view DBSCAN as a special case of VESDC.

Determining if this revised definition of epsilon neighbors is met algorithmically is also straightforward, and requires no additional near neighbor searches beyond what is needed for the original DBSCAN method.[11] Specifically, we first find all the points within the epsilon radius of each point. Next, if point *q* was found to be within Eps_p of *p*, we determine if *p* was within Eps_q of *q*, if it is then *q* remains an epsilon neighbor of *p* and vice versa. On the other hand, if *p* is not within Eps_q of point *q*, then point *q* is removed from the epsilon neighbor list of point *p*.


Points A and B are Epsilon Neighbors

Points C and D are Not Epsilon Neighbors

Figure 5.1 An Illustration of the New Definition of Epsilon Neighbors

5.1.2 *CpSp* Index Modified for Variable Epsilon Radius Sizes

Fortunately, there is a direct (inverse) relationship between the size of the epsilon radius and the local “scale” of distance. As an example, if the epsilon radius for point A is 10 meters, while the epsilon radius for point B is 20 meters, then a distance of one meter is re-scaled to be twice as long at point A as it is at point B. Since distances are ratio scaled, we are actually only interested in the relative distances between objects. Consequently, we can simply divide a

[11] The near neighbor search is implemented with the ANN library of Mount and Arya (Mount, 2006) using a kd-tree and an exact search.

distance at a point by the epsilon radius of that point. Going back to our example, the scaled distance for 1 meter at point A becomes 0.1 (1/10), while the scaled distance becomes 0.05 (1/20) at point B.

The problematic concept in the last paragraph is "a distance at a point," since points have zero area. What we are really interested in is the distance between points. In what follows we will rely on what we call a "weighted distance" between two points, which we define as the Euclidean distance between two points divided by the average of the epsilon radius values of those points. The example of the last paragraph can help to make this concept more concrete. Assume that points A and B are 10 meters apart, and the epsilon radius value for each point are as they were before (i.e., 10 meters for point A and 20 meters for point B), then the average of the epsilon radius values is (10+20)/2 = 15, and the weighted distance between the points is 10/15 ≈ 0.67. Obviously, this measure of weighted distance can only be viewed as an approximation to the "actual" weighted distance. Fortunately, we are typically measuring the distances between neighboring points, so the errors introduced by the approximation should be fairly small.[12]

Given this definition of the weighted distance, we can now describe the scaled versions of the measures underlying the *CpSp* index, beginning with the scaled version of the *Comp* measure. In Chapter 3 we defined *Comp* as the maximum value across clusters of the sum of the edge Euclidean distances for the MST of each cluster. For the scaled version of *Comp* we use the weighted distances (rather than the standard Euclidean distances) to construct the MST, and use the sum of the weighted edge distances for the measure. Similarly, the maximum possible compactness (C_{max}) is equal to the sum of the weighted edge distances for the MST of

[12] Developing a more exact measure may be possible, but would likely be computationally expensive.

the entire data set. Both the minimum possible compactness (C_{min}) and minimum possible separation (S_{min}) is taken to be the minimum weighted distance between any two points in the data set. The maximum possible separation (S_{max}) is equal to the longest weighted distance associated with the edges of the Gabriel graph of the points. Finally, the *Sep* measure is equal to the shortest weighted distance between any two points not in the same cluster for the entire data set.

The above measures are appropriate for a given vector of epsilon radius values. If the elements of that vector change differentially, then the relative scaling of the underlying measures (e.g., C_{max}, S_{min}, etc.) can change relative to one another, resulting in a comparison across solutions that is non-comparable.[13] Consequently, in comparing two cluster solutions (each based on a different vector of epsilon neighborhoods), there is a need to hold the vector of epsilon radius values constant for the purposes of the comparison.

The use of an example here may help to fix ideas. The way we will use the modified *CpSp* measure is to determine how to set the parameters of a function that links an underlying variable to the length of the epsilon for a point. As a simple example of this consider the case (which we empirically examine later in the chapter via synthetic data) where there are two types of areas, high-density areas and low-density areas, and we believe that points in high-density areas should have a smaller epsilon neighborhood than points in low-density areas. Assume that we already know what the correct value of the epsilon neighborhood for points in low-density areas, and are interested in determining the appropriate epsilon neighborhood value for points in high-density areas (we label this value *MinEps* because it is the smaller of

[13] If the value of the epsilon radius is the same for all points, then a change in the epsilon radius will leave the relative distances across points unchanged. It is the possibility of a differential change in the epsilon radius values that causes the problem.

the two epsilon neighborhood values). Further assume that we are determining the correct value by incrementing the value of *MinEps* along a range of values. Let there be two adjacent candidate *MinEps* values, $MinEps_{i\text{-}1}$ and $MinEps_i$, with $MinEps_{i\text{-}1}$ < MinEps_i. Define $Eps_{i\text{-}1}$ as the vector of epsilon radius values when *MinEps* is at the level $MinEps_{i\text{-}1}$, and Eps_i as the vector of epsilon radius values when *MinEps* equals $MinEps_i$. In addition, let $A_{i\text{-}1}$ be the assignment of points to clusters when $Eps_{i\text{-}1}$ is the vector of epsilon neighborhoods, and let A_i be the cluster assignments of points when Eps_i is the vector of epsilon radius values.

Having set out the basic notation, we define $Comp(A_{i\text{-}1} \mid Eps_i)$ as the compactness value of the clustering solution when the epsilon radius vector corresponds to $\mathrm{Eps}_{i\text{-}1}$, but is evaluated using the weights for the edge distances that correspond to Eps_i, while $\mathrm{Comp}(A_i \mid Eps_i)$ is the compactness level when both the clustering solution and the distance weights when the epsilon radius vector is Eps_i. One measure of interest is the change in the *Cp* index when we go from $MinEps_{i\text{-}1}$ to $MinEps_i$, which we can measure as

$$\Delta Cp_i = \frac{C_{max}(Eps_i) - Comp(A_i \mid Eps_i)}{C_{max}(Eps_i) - C_{min}(Eps_i)} - \frac{C_{max}(Eps_i) - Comp(A_{i-1} \mid Eps_i)}{C_{max}(Eps_i) - C_{min}(Eps_i)},$$

where $C_{max}(Eps_i)$ is the maximum possible compactness when the distance weights correspond to the epsilon radius vector Eps_i, and $C_{min}(Eps_i)$ is the minimum possible compactness when the weights are given by Eps_i. In a similar fashion we can measure the change in the *Sp* index as

$$\Delta Sp_i = \frac{Sep(A_i \mid Eps_i) - S_{min}(Eps_i)}{S_{max}(Eps_i) - S_{min}(Eps_i)} - \frac{Sep(A_{i-1} \mid Eps_i) - S_{min}(Eps_i)}{S_{max}(Eps_i) - S_{min}(Eps_i)}.$$

Finally, we can measure the change in the *CpSp* index as

$$\Delta CpSp_i = \frac{\left(\frac{C_{max}(Eps_i) - Comp(A_i \mid Eps_i)}{C_{max}(Eps_i) - C_{min}(Eps_i)}\right)\left(\frac{Sep(A_i \mid Eps_i) - S_{min}(Eps_i)}{S_{max}(Eps_i) - S_{min}(Eps_i)}\right) -}{\left(\frac{C_{max}(Eps_i) - Comp(A_{i-1} \mid Eps_i)}{C_{max}(Eps_i) - C_{min}(Eps_i)}\right)\left(\frac{Sep(A_{i-1} \mid Eps_i) - S_{min}(Eps_i)}{S_{max}(Eps_i) - S_{min}(Eps_i)}\right)}.$$

While the changes in these three measures are of great importance, what is more closely related to the various *CpSp* measures (and the plots of those measures) presented in Chapter 3 is the cumulative sum of those changes up to a point. We can define the cumulative sum for the *Cp* measure up to and including the *j*th incremental value of *MinEps* (*MinEps$_j$*) as

$$Cp_j = \sum_{i=2}^{j} \Delta Cp_i.$$

Similarly, we can define the cumulative sum of the *Sp* measure as

$$Sp_j = \sum_{i=2}^{j} \Delta Sp_i,$$

and the cumulative sum of the *CpSp* index itself as

$$CpSp_j = \sum_{i=2}^{j} \Delta CpSp_i.$$

Up to this point we have motivated the derivations relative to an example where we are searching over the possible values of the parameter *MinEps*. However, any parameter that alters the elements of the epsilon radius vector can be substituted into the equations above. As a result, these correspond to the general *CpSp* measures modified to account for variable epsilon radius sizes.

5.1.3 Validating the Modified Methods against Simulated Data

5.1.3.1 Creating the Simulated Test Data

For this validation exercise, we wanted to create a set of test data sets that more closely mimicked the patterns we actually observe for retail outlet locations than was the case for the simulated data sets that were used for illustrative purposes in Chapters 2 and 3. As we saw in Chapter 4 for the case of Greater Victoria, retail outlets often cluster along a street. As a result, many of the clusters resemble thin strings of outlets when plotted, hence the term “string clusters” that was used in the previous chapter. We also decided to mimic a grid layout of streets that is common in many North American cities. The two considerations led us to the creation of both horizontally and vertically oriented clusters that produce a grid pattern (allowing string clusters to intersect one another). Finally, we wanted to create both high-density and low-density areas.

To accomplish all our data creation objectives, each synthetic test data set contains two different unique “panes”, one of high-density, and one of low-density. The panes are then laid out in a matrix, with the high-density pane in the northwest cell of the matrix, and the low-density pane being replicated in the remaining cells of the table (Figure 5.2 illustrates this layout for the first simulated data set). The panes differ in the number of points they contain, with high-density panes having 440 points and low-density planes having 66 points. Each pane has the

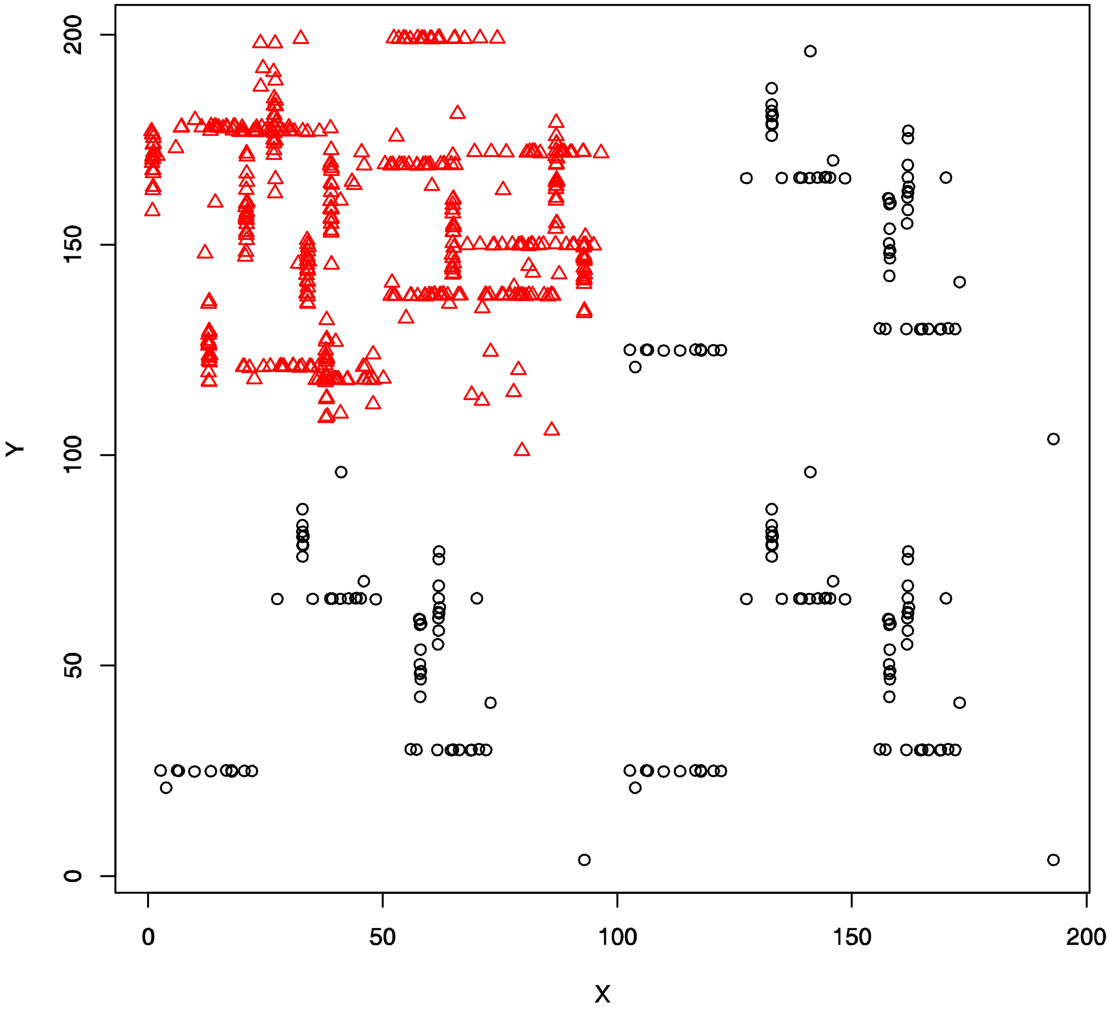


Figure 5.2 The First Simulated Data Set

same area of 10,000 square map units. High-density panes were created by drawing the mean coordinate values from a uniform distribution that was bounded between 12 and 88 map units for 10 horizontal and 10 vertical oriented clusters. The x (horizontal) mean coordinate for a horizontal cluster was rounded to be an integer value to obtain the desired grid effect for the clusters. An identical approach was taken for the y (vertical) mean coordinate for vertical clusters. In the case of horizontal clusters, the x-coordinate values for the 20 members of a cluster were drawn from normal distribution with the mean corresponding to the x-coordinate mean for the cluster, and a standard deviation equal to 6, while the y-coordinate values equaled the cluster y-coordinate mean plus a very small amount of normal noise (with mean of zero and a standard deviation of 0.1) to avoid potential problems with creating the Delauney triangulation graph of the data. If one of the selected x-coordinate values resulted in either a negative x-coordinate value, or an x-coordinate value that was greater than 100, rejection sampling was used to avoid this from occurring. Consequently, all points fell within 0 and 100 map units. The creation of the vertical oriented clusters was done using a directly analogous procedure. Finally, 40 randomly chosen points from a uniform distribution that was bounded between 0 and 100 was added to the data in order to mimic noise points.

Essentially identical procedures were used to create the low-density panels. The exceptions were that only 3 vertical and 3 horizontal oriented clusters were constructed, each cluster had only 10 members as opposed to 20, and only six randomly chosen points were added to the data.

5.1.3.2 The Simulated Data Set Analysis

We begin with a detailed discussion associated with clustering the first simulated data set using the methods developed in this chapter, and then briefly summarize the results from clustering two other data sets using these methods. The first example was not selected at random from the set of simulated data testing we have conducted. Rather, it was chosen because it exhibited some challenging aspects in clustering the clustering process.

To provide a benchmark by which to compare the results from the VESDC clustering of the full data set, we begin by showing the DBSCAN results for each of the two unique panes.

Figure 5.3 provides the DBSCAN results for the high-density panel, while Figure 5.4 provides the DBSCAN results for the low-density panel. An examination of Figure 5.3 reveals a very clean clustering of the high-density panel, with a clear peak in the *CpSp* index at 5 map units, indicating that the "best" epsilon radius for the high-density area is 5 map units. A similar

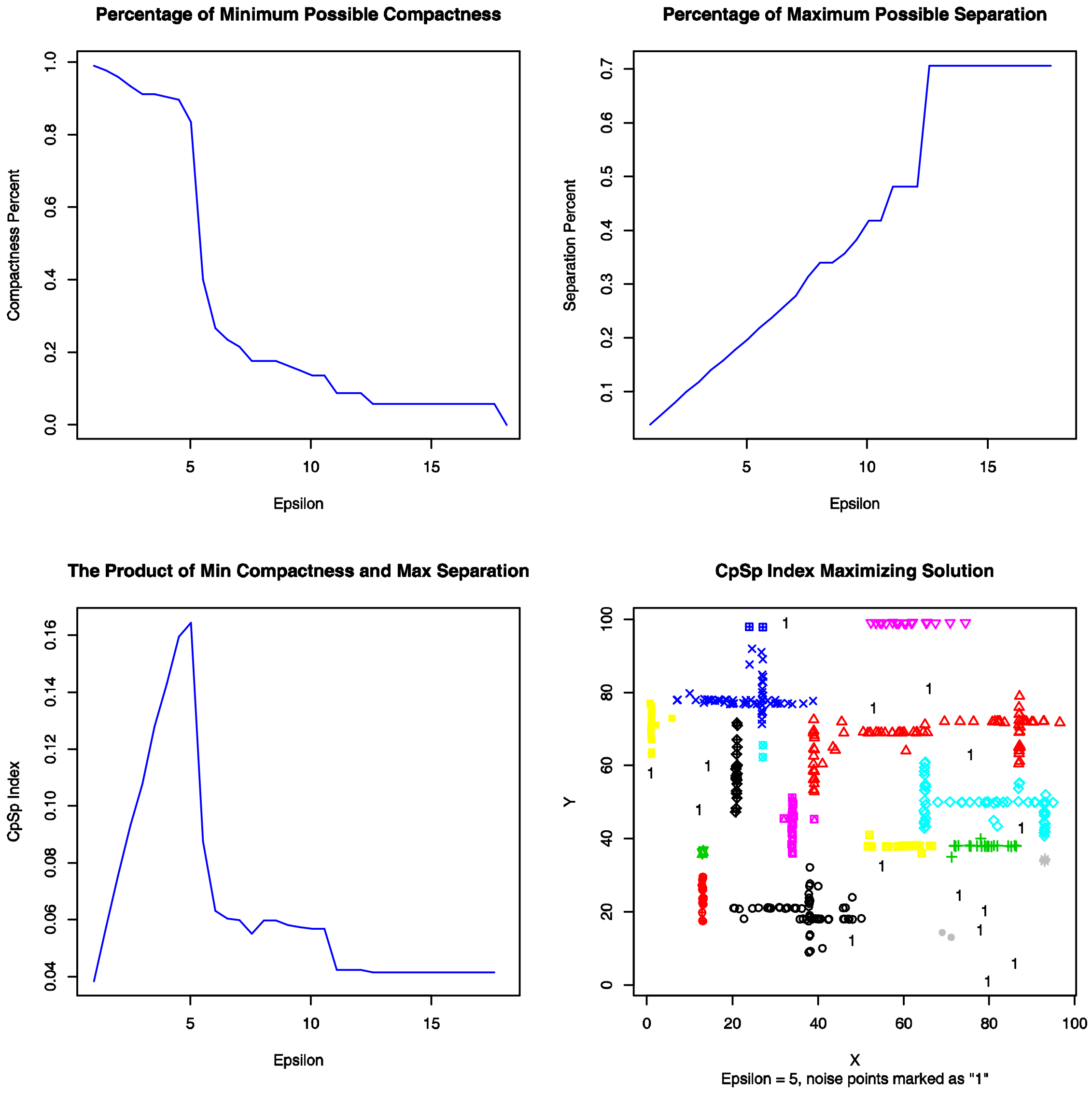


Figure 5.3 The DBSCAN Results for the High Density Panel of the First Simulated Data Set

examination of Figure 5.4 also indicates a fairly clean clustering solution. Albeit, there is a fairly large range of the epsilon radius values (from 13 to 32 map units) that captures the same clustering solution. However, given the distance between the outlying group in the lower left-hand corner of the figure's southeast panel, this result isn't surprising. What is encouraging is that the "best" epsilon radius value for the high-density area is less than one-half the size of the

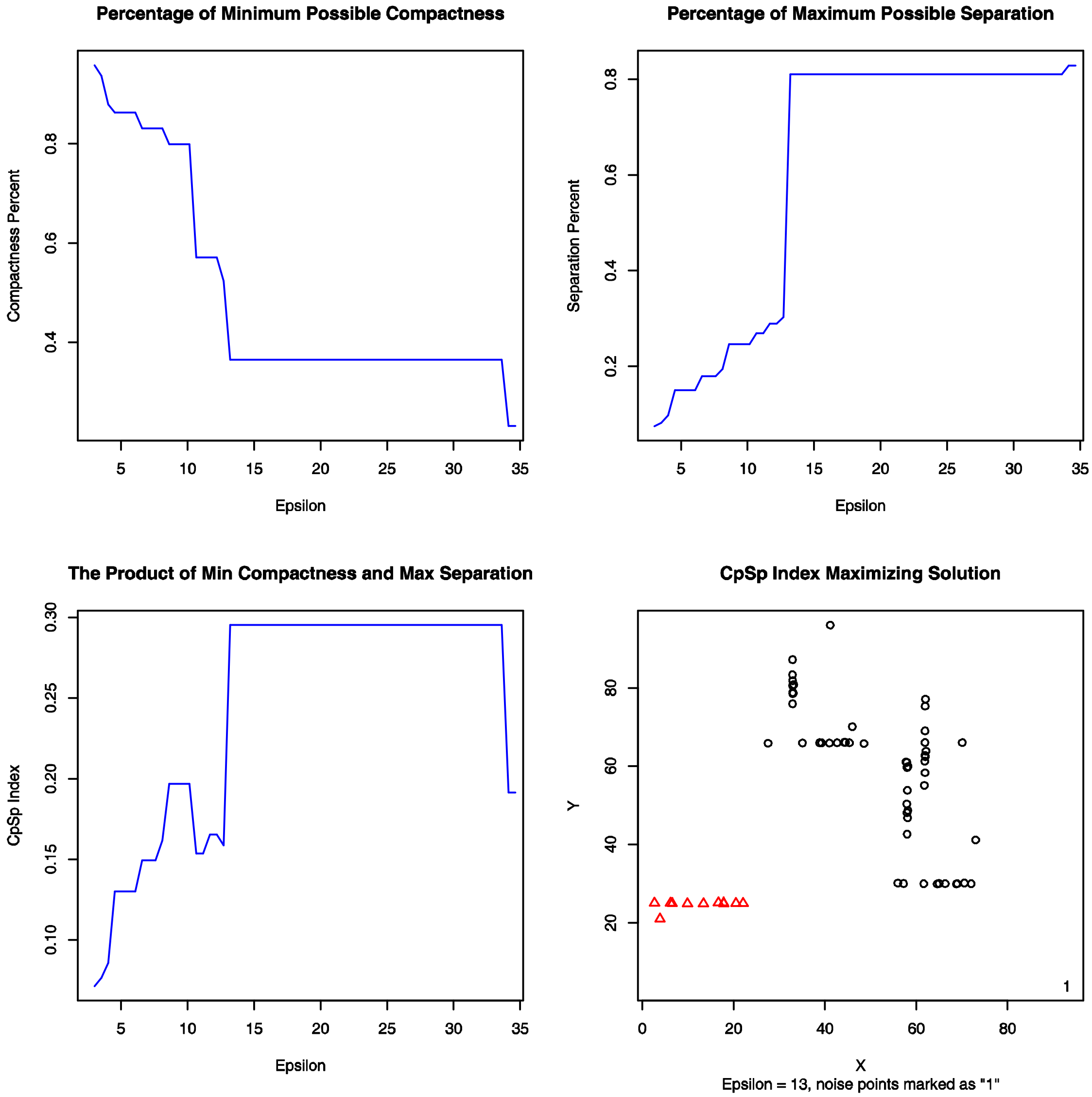


Figure 5.4 The DBSCAN Results for the Low Density Panel of the First Simulated Data Set

epsilon radius value for the low-density area (5 map units versus 13 map units), which is consistent with our expectations.

Figure 5.5 presents the DBSCAN results for the entire (four pane) data set. The results reveal that the best epsilon radius based on *CpSp* index is between 20.5 and 31.5. What is interesting is that start of the “best” range begins at a much larger value of the epsilon radius than was the case

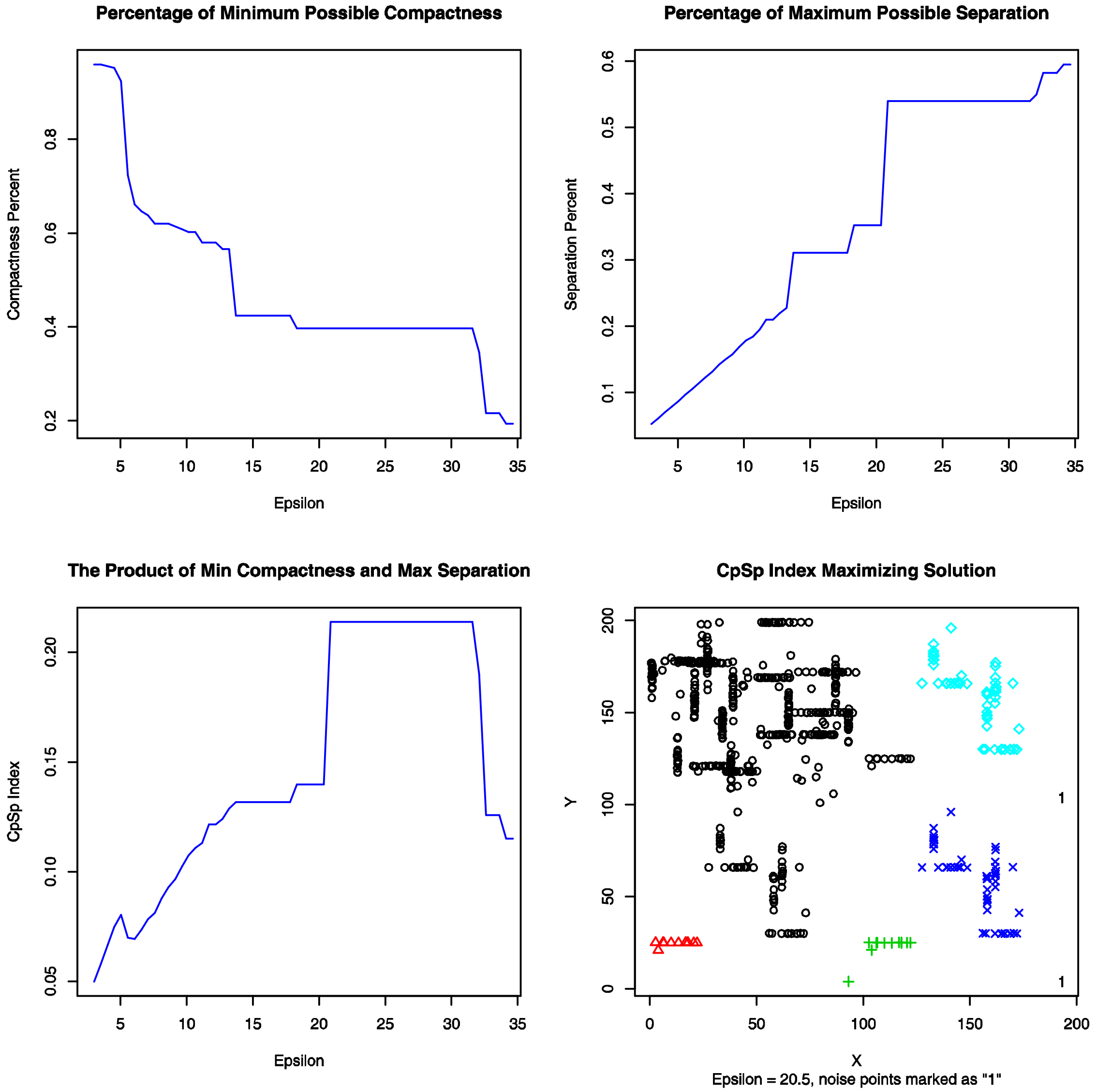


Figure 5.5 The DBSCAN Results of the First Simulated Data Set

when the low-density panel was analyzed in isolation. In any event, this high value of the epsilon radius results in all the points in the high-density pane (along with some of the points belonging to the low-density areas) to be in a single cluster. This over agglomeration of points in dense areas is similar in nature to the problems that were seen in applying DBSCAN to the Greater Victoria data.

As discussed earlier in this chapter, given the structure of the data, where a point is either in a low or high-density area, the appropriate way to parameterize the function that links an underlying factor to the epsilon radius is a simple indicator function of the form

$$Eps_i = \begin{cases} MinEps, & density = high \\ MaxEps, & density = low \end{cases},$$

where Eps_i is the epsilon radius for a particular point in the data set. In this formulation, there are two parameters that need to be determined, *MinEps* and *MaxEps*.

Based on the DBSCAN clustering results for the data set as a whole (Figure 5.5), it would appear to the analyst that the best epsilon radius value for that solution would be more likely to be near the value of *MaxEps* than *MinEps*. As a result, the natural thing to do is to set *MaxEps* to 20.5 (the start of the best range), and then search over possible values of *MinEps* to find one that improved the cumulative change in the modified *CpSp* index. Figure 5.6 provides the *CpSp* analysis associated with searching for the value of *MinEps* over a range of values from 0.5 to 20.5, incremented by 0.5. An examination of this figure reveals that it is strongly bipolar, with a sharp peak at a *MinEps* value of 5, peaking again in the range from 18.5 to 20.5. Addressing this situation, which is actually fairly uncommon in the simulated data examples we have run, is what makes this data set comparatively challenging. The first peak has a cumulative *CpSp* index value that is very slightly below that of the second peak, while the value of *MinEps* at the second peak replicates the unsatisfactory solution found using DBSCAN. What should the analyst do in this situation? We argue that analyst should explore both potential solutions.

Assuming the analyst realized that the second peak simply replicates the DBSCAN solution, and opted to start with a *MinEps* value of 5, the natural thing to do, in EM algorithm-like fashion, is to set *MinEps* to 5, and grid search over new potential values of *MaxEps* to see if the cumulative *CpSp* measure could be improved. The results of doing this over a range of 5 to 34 is shown in

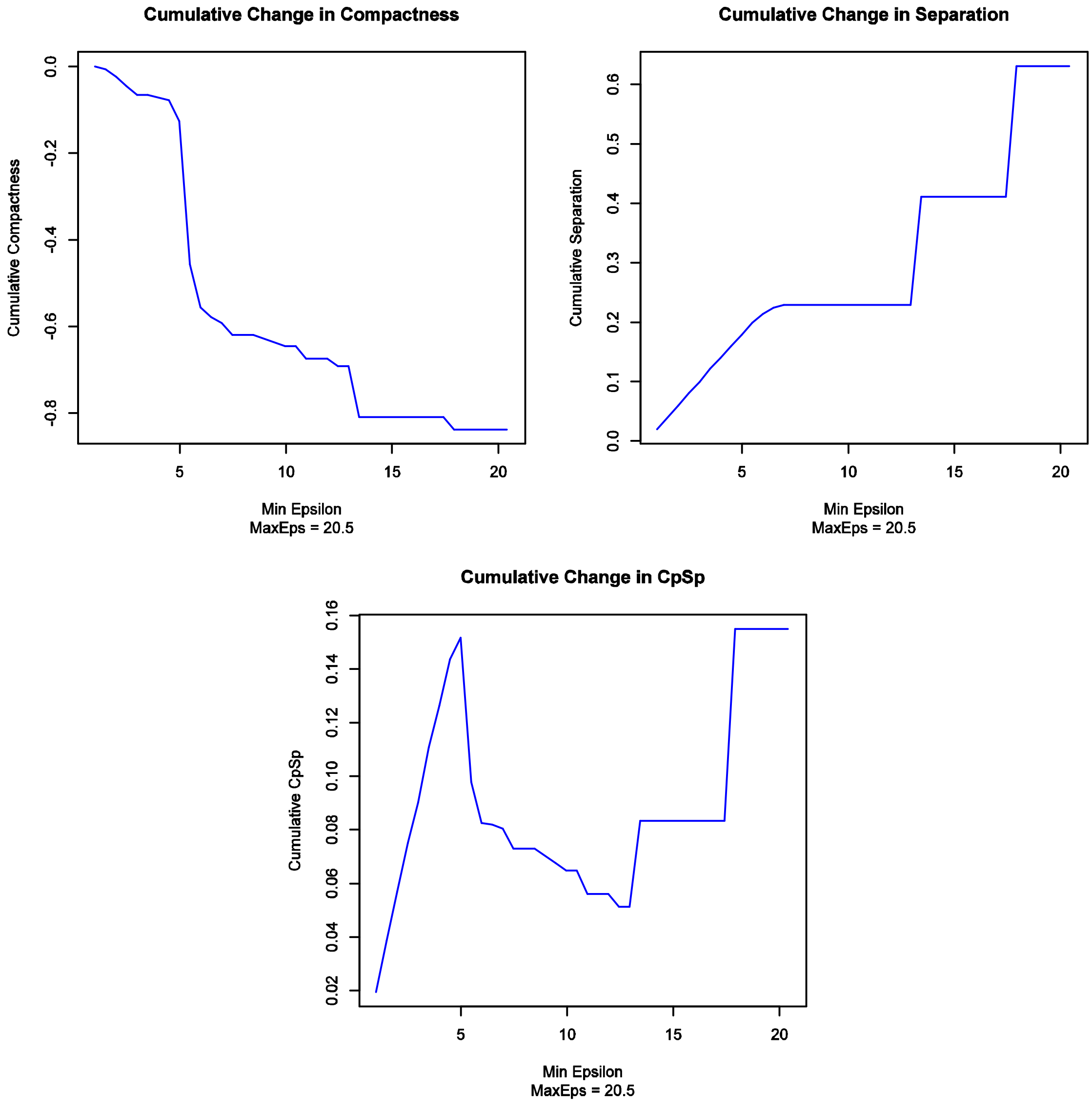


Figure 5.6 The Variable Epsilon *CpSp* Analysis of MinEps for the First Simulated Data Set

Figure 5.7. An examination of this figure reveals that the start of the “best” range is 20.5, indicating that we have converged with a MinEps of 5, and a MaxEps of 20.5. However, an examination of the cumulative *Cp* measure (the upper left panel in the figure) and cumulative *Sp* measure (the upper right panel in the figure) indicates that changes in the cumulative *CpSp* are

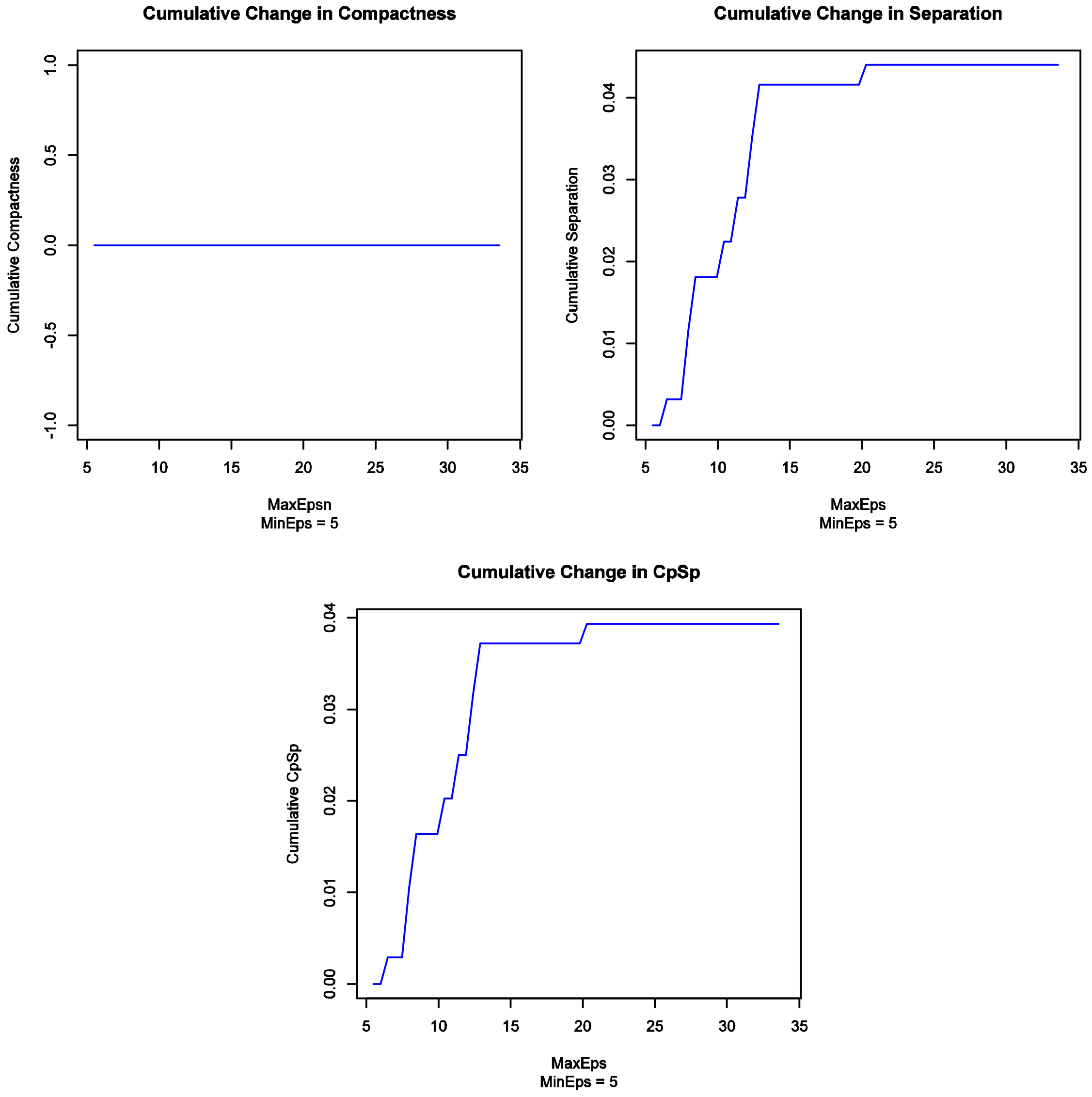
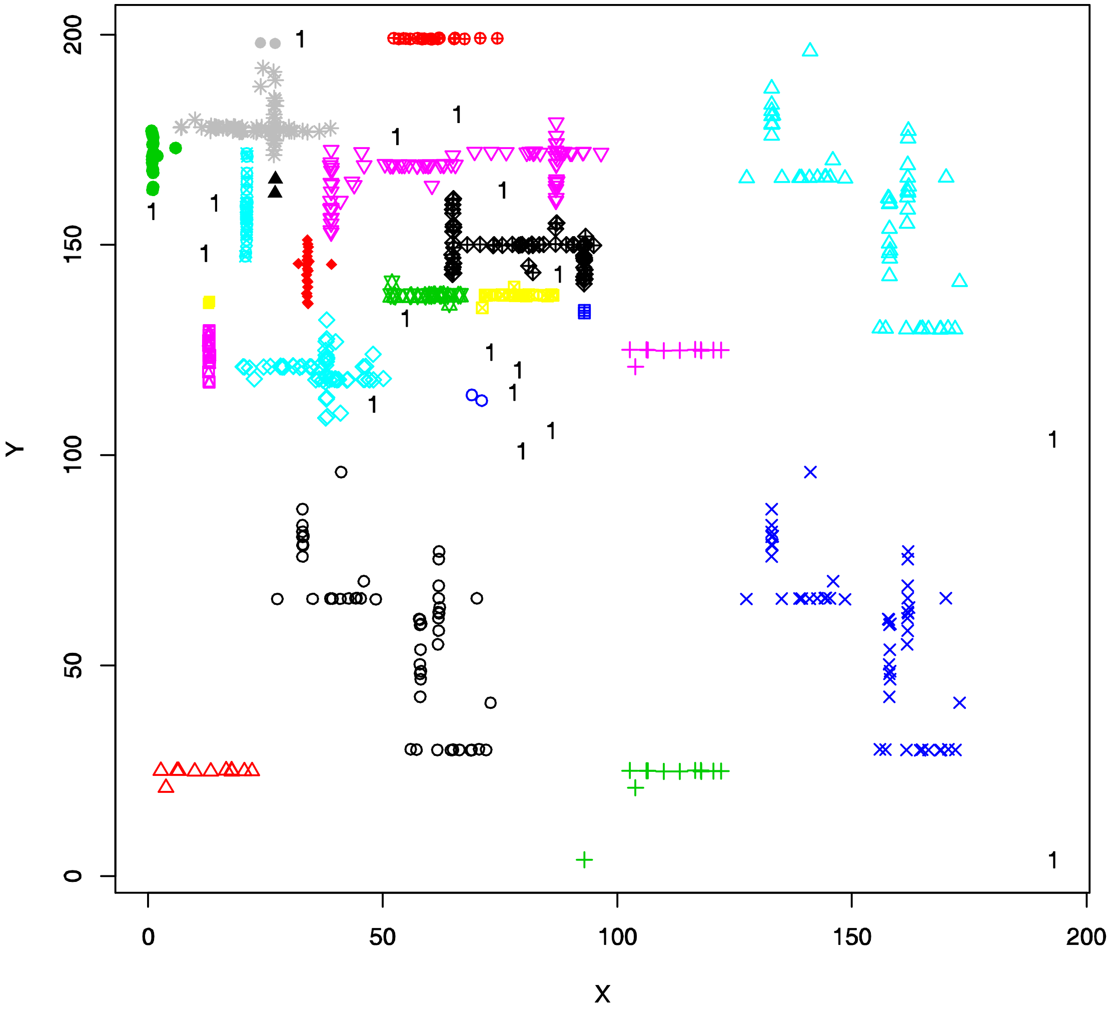


Figure 5.7 The Variable Epsilon *CpSp* Analysis of MaxEps for the First Simulated Data Set

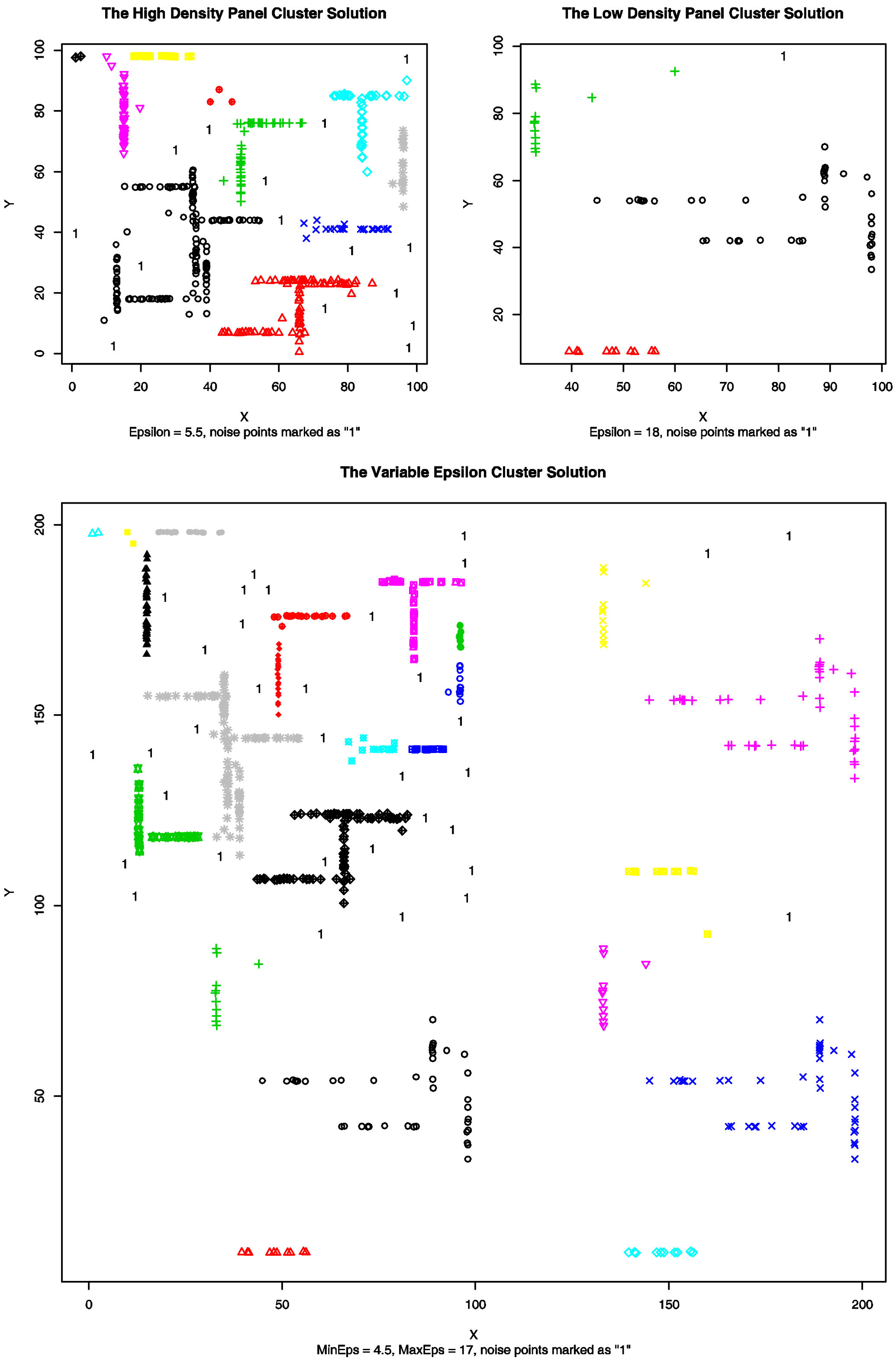
entirely due to changes in the *Sp* measure, with no movement whatsoever in cumulative *Cp*. This suggests to us that there is less information available for setting parameter values when two parameters vary as to opposed to only a single parameter, and represents a limitation of our proposed approach. Having said this, it does remain a very useful, albeit imperfect, tool for determining parameter values. To illustrate this point, Figure 5.8 provides the final VESDC



MinEps = 5, MaxEps = 20.5, noise points marked as "1"

Figure 5.8 The VESDC Solution for the First Simulated Data Set

clustering solution. A comparison of this figure with Figures 5.3 and 5.4 reveals that this solution corresponds to the exact solutions obtained when the two panels are analyzed individually. As additional evidence of the efficacy of the proposed methods, Figures 5.9 and 5.10 provide both the DBSCAN solutions for the separate high- and low-density panes, as well as the final VESDC



The High Density Panel Cluster Solution

Epsilon = 5.5, noise points marked as "1"

The Low Density Panel Cluster Solution

Epsilon = 18, noise points marked as "1"

The Variable Epsilon Cluster Solution

MinEps = 4.5, MaxEps = 17, noise points marked as "1"

Figure 5.9 The VESDC and DBSCAN Results for the Second Simulated Data Set

solutions for the pooled data sets for two different simulated data sets. An examination of these figures reveals that while the VESDC solution does not perfectly correspond to the separate DBSCAN solutions for the high- and low-density panes, the solutions are very close. Moreover, a visual inspection of the clustering solutions suggests that the DBSCAN solutions are no better, and perhaps worse, than the pooled VESDC solution. Overall, the VESDC approach, along with the modified *CpSp* measures to assess clustering parameters, appear to be able to address aspects of the intra-community complexity that current clustering methods seem incapable of addressing. However, as will be illustrated shortly, even these methods breakdown if there is not a reasonable definition of "community" to base detailed clustering on. It is to the issue of defining "community" to which we now turn.

5.2 Creating "Communities" (Step 1)

Our definition of "community" is very different than the typical definition of this concept. Specifically, what we mean by community is an area that is sufficiently compact that detailed VESDC clustering can work in a reasonable manner. In an effort to illustrate what we mean by this, we use a modified version of the first simulated data set as an example, which is shown in Figure 5.11. In this data set, the panes, and the placement of those panes, is identical to the original data set shown in Figure 5.2. What differs is that instead of the panes just touching one another, a buffer of 150 map units in both the vertical and horizontal dimensions now separates them.[14]

When we do a DBSCAN analysis on this data set we obtain a "best" epsilon radius of 34.5 meters. If we then set *MaxEps* to this value, and search over potential values of MinEps we

[14] We have also conducted the analysis we are about to present with smaller 50 map unit buffers, and obtained identical results. We use the 150 map unit buffers here since it makes visual imagery more striking.

obtained the analysis contained in Figure 5.12. A comparison between this figure and Figure

5.6

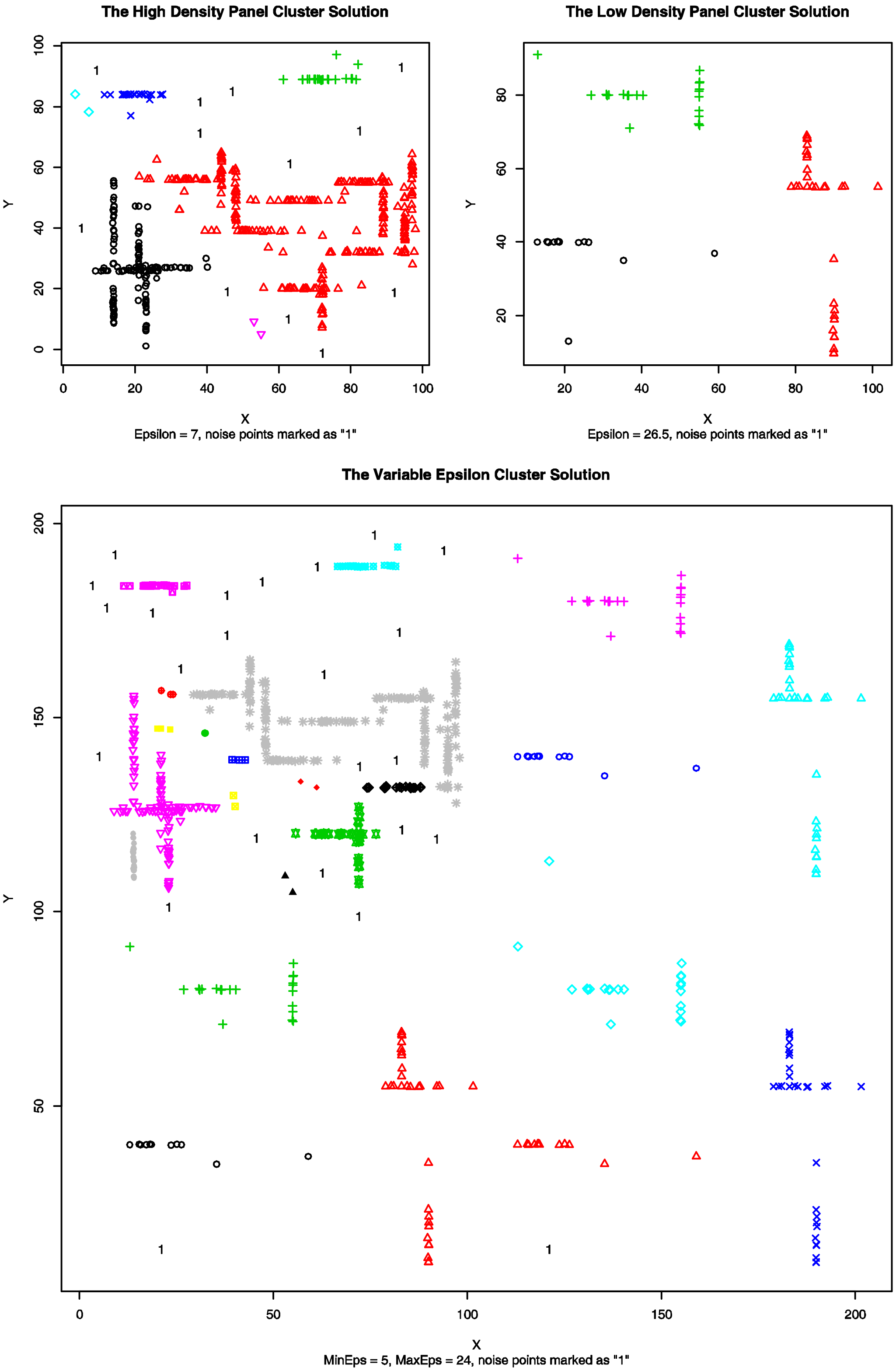


Figure 5.10 The VESDC and DBSCAN Results for the Third Simulated Data Set

reveals a number of similarities. However, the critical difference between the two figures is the height of the first mode (at a *MinEps* value of 5 map units). In Figure 5.12 this mode is comparatively much shorter than it is in Figure 5.6. As a result, an analyst would likely select a *MinEps* value along the upper plateau in the cumulative *CpSp* plot (the lower panel in the figure). This value of *MinEps*, along with the value of *MaxEps* results in all the points in each pane to be agglomerated together into a single cluster, resulting in four clusters overall, one for each pane. To avoid this problem, each pane must be treated as a “community,” with the clustering done at the community level.

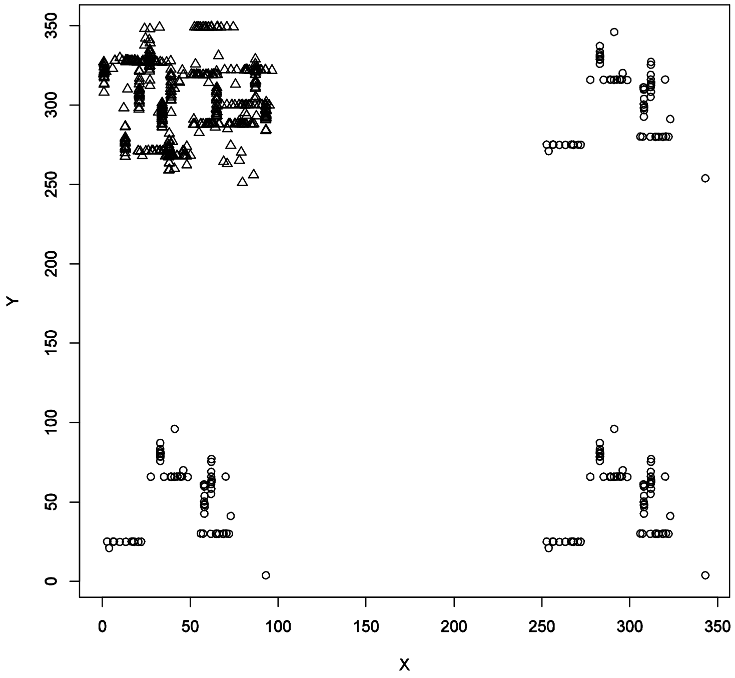


Figure 5.11 The Separated Panel Version of the First Simulated Data Set

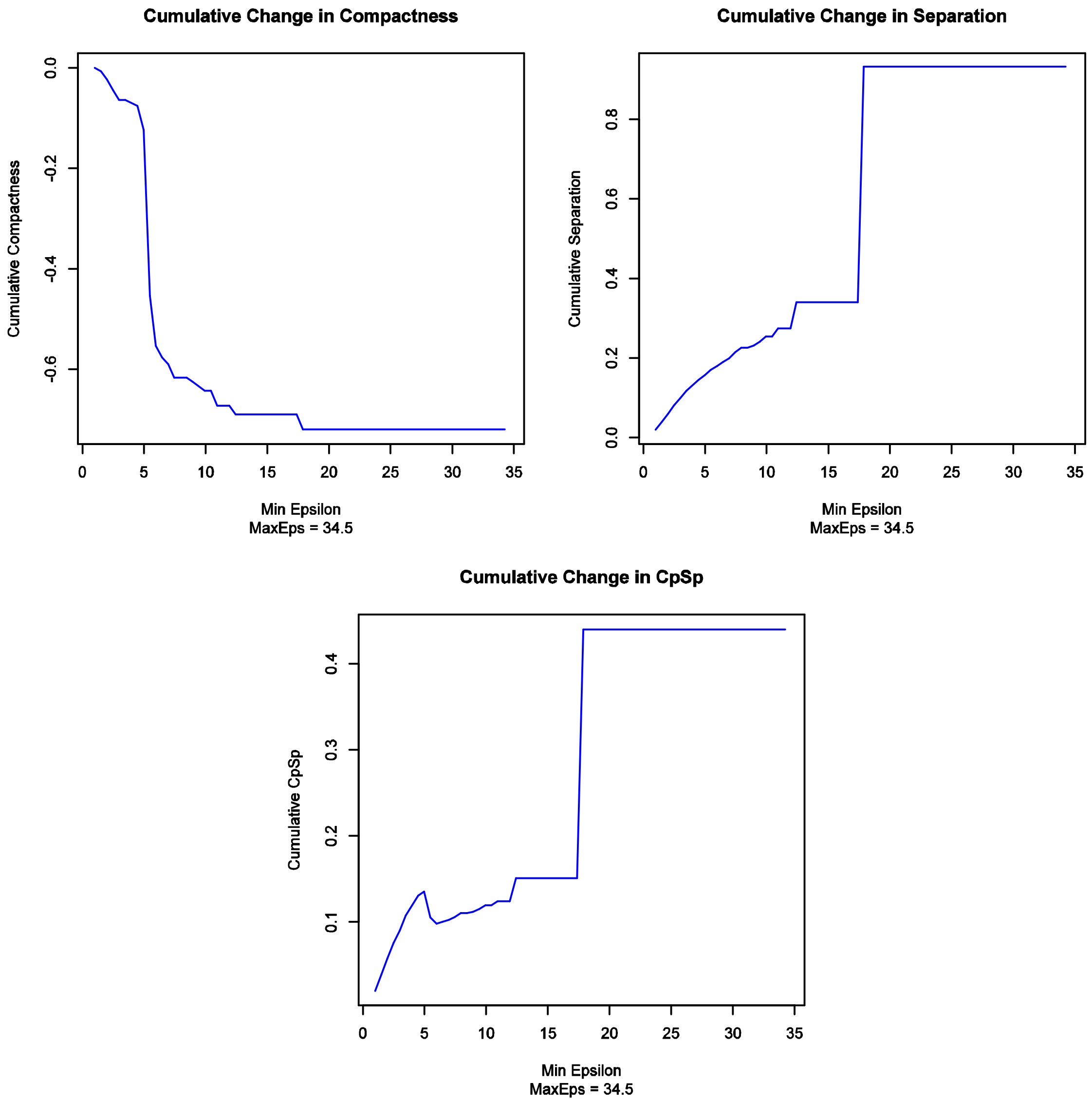


Figure 5.12 The Variable Epsilon *CpSp* Analysis of MinEps for the Separated Panel Data Set

The method we propose for creating "communities" relies on yet one more sub-graph of the Delauney triangulation of the full set of points. In this instance we propose the use of the relative neighbor graph of the points (Toussaint 1980; de Berg et al., 2008, p. 217). Specifically, this process involves creating the relative neighbor graph of the full set of points and then removing from the graph edges between relative neighbors that are deemed too far from one another to allow them to fall into the same community. Doing this results in a set of disconnected sub-graphs. Both the relative neighbor graph and the “trimmed” set of sub-graphs (based on removing edges from the relative neighbor graph that exceeded 40 map units in length) for the first simulated data set whose panes have been separated by 150 meters buffers is shown in Figure 5.13. The final step is then to apply VESDC clustering (or in this specific example DBSCAN clustering since the sub-graphs are the original panes, each of which has a constant density) to the points in each sub-graph (community) that has a sufficient number of points to cluster (we will see in the next chapter that a large percentage of the identified sub-graphs in real world data have a very small number of members).

Determining what distance to use for trimming the relative neighbor graph does require some domain specific knowledge. However, the level of domain specific knowledge is fairly minimal, and often amounts to having a minimally informed opinion. In the case of developing retail districts, we have decided that the appropriate distance between relative neighbors that is deemed too far is 400 meters. This conclusion was based on the notion that having to travel roughly a quarter mile to get from one store to the next in some direction was simply too far to consider these stores neighbors. While this assessment is a bit “off the cuff,” the 400 meters rule seems to work well in empirical application.

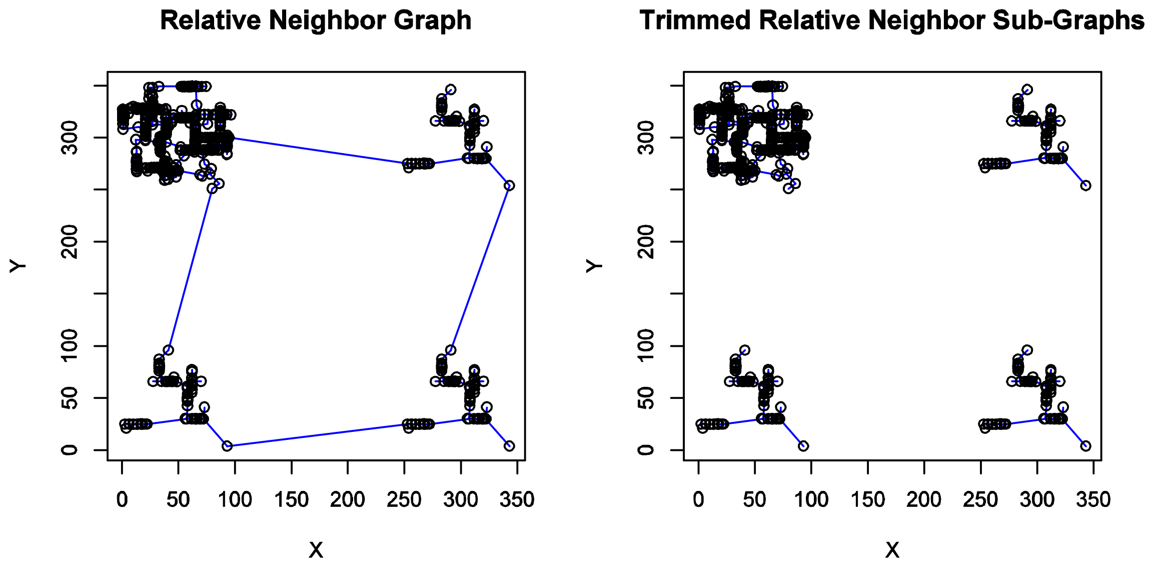


Figure 5.13 The Relative Neighbor Graph and Trimmed Sub-Graphs of the Separated Panel Data Set

5.3 Summary

In this chapter we have presented a set of a two-step clustering approach and companion cluster validation tools that appear to be able to address complex spatial point data. While the tools are not perfect (we expect that a number of improvements will be made overtime, particularly to the modified *CpSp* validation tools), they provided much improved solutions as compared to currently available approaches. With these new tools in hand, we now return to Greater Victoria.

6 Greater Victoria Revisited

In this chapter we illustrate the two-step clustering approach developed in the last chapter to Greater Victoria. We will provide the results of each step in turn, as well as the assumptions and choices we made along the way.

6.1 Step 1

As was indicated in our discussion of the first step in the last chapter, we assume that two relative neighbors that are separated by more than 400 meters cannot fall into the same "community." When we apply this rule to trim the relative neighbor graph of the Greater Victoria retail outlet locations, the result is a set of 189 disconnected sub-graphs. Figure 6.1 provides a plot of these Greater Victoria "communities." An examination of this plot reveals that there is one extremely large community in the southeast of the metropolitan area, which corresponds to the Victoria downtown core (something we saw in Chapter 4 when examining the relationship between total population density and retail outlet density). In addition, there are a number of smaller "communities" along the eastern edge of Greater Victoria as you go north from the downtown core. In addition there are a number of communities immediately to the east of the downtown core as well. However, one thing that is striking is that many of what we have been calling "communities" consist of one or two relatively isolated points, a pattern particularly prevalent in the southwest region of the metropolitan area, but which can be seen elsewhere.

To gain a better sense of the distribution of stores across communities, Figure 6.2 presents a bar blot of the number of stores in each community, while Figure 6.3 provides a distribution of communities by the number of stores they contains. An examination of Figure 6.2 is striking since it vividly illustrates the importance of the downtown core in Greater Victoria's overall

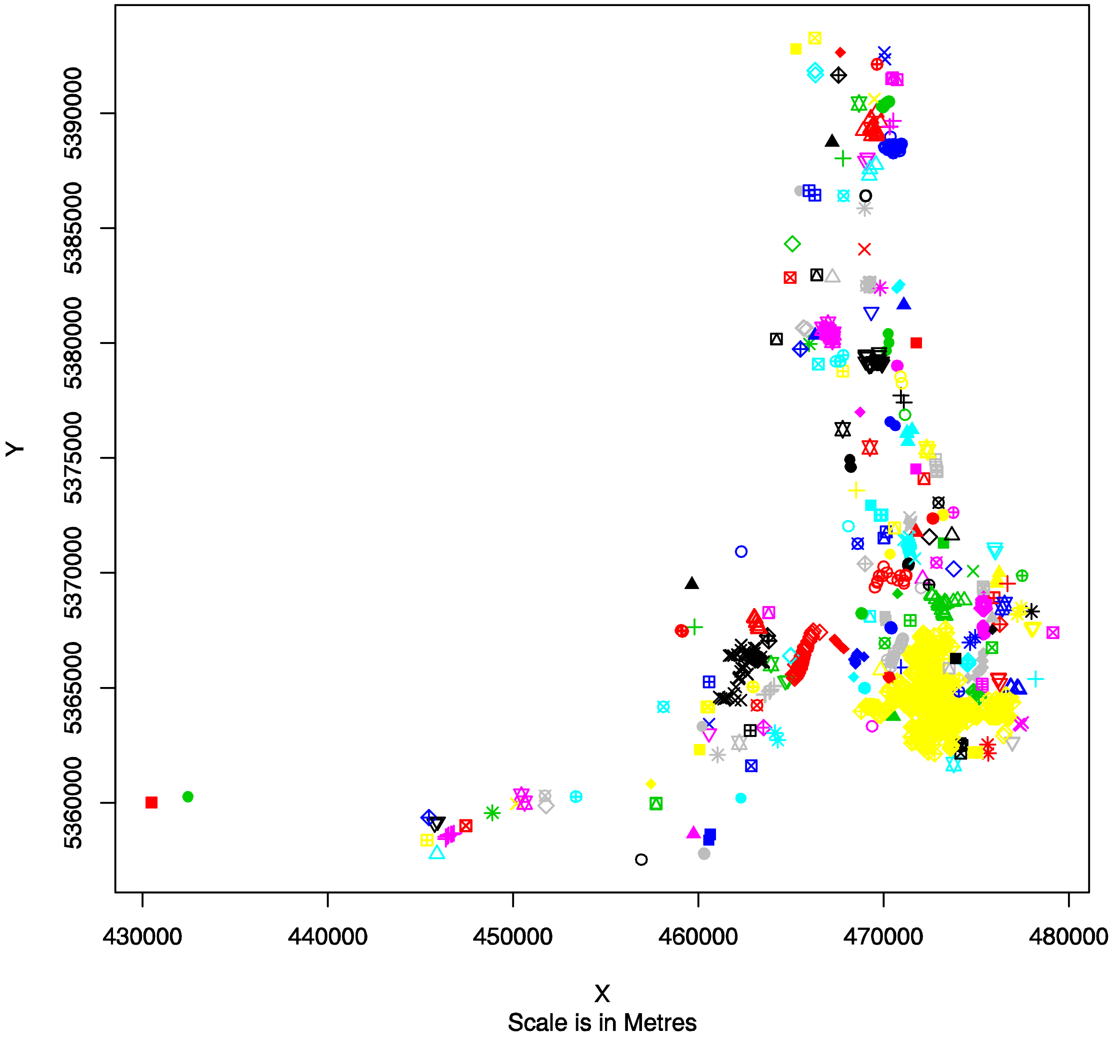


Figure 6.1 The "Communities" of Greater Victoria

retail structure. In fact, approximately half of the retail outlets in Greater Victoria are located in the downtown core. While the downtown core has 1296 retail outlets, the next two largest communities have only 109 and 101 retail outlets, respectively. The distribution of communities by the number of retail outlets (Figure 6.3) is extremely skewed, with a majority of "communities" containing a single cluster.

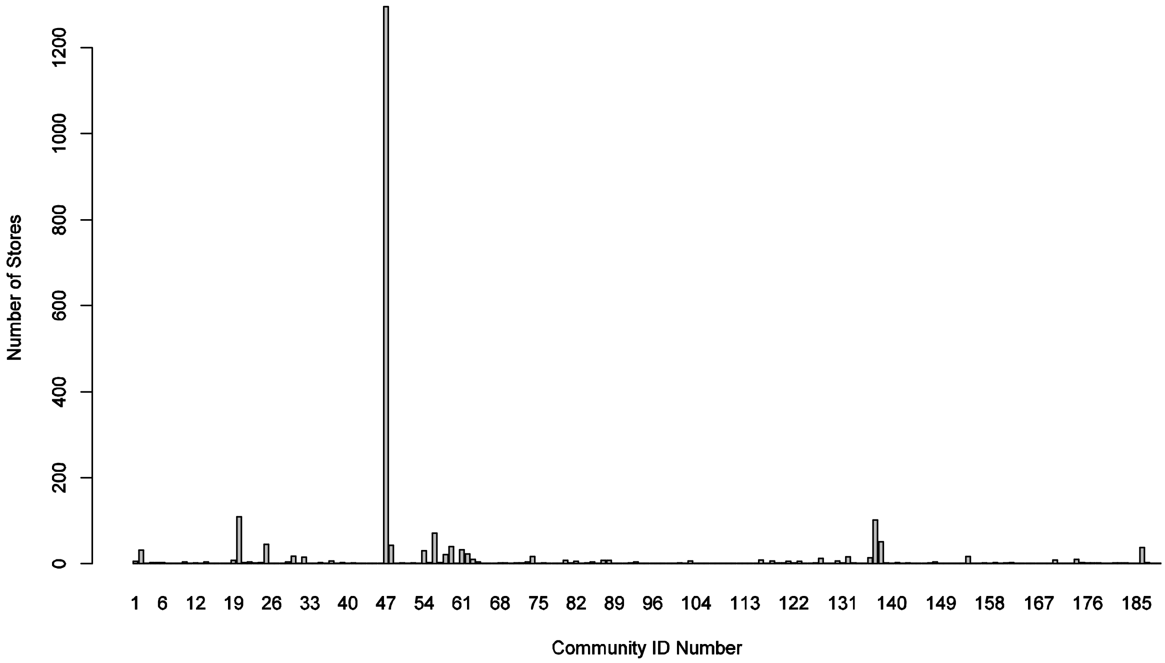


Figure 6.2 The Number of Stores in Each Community

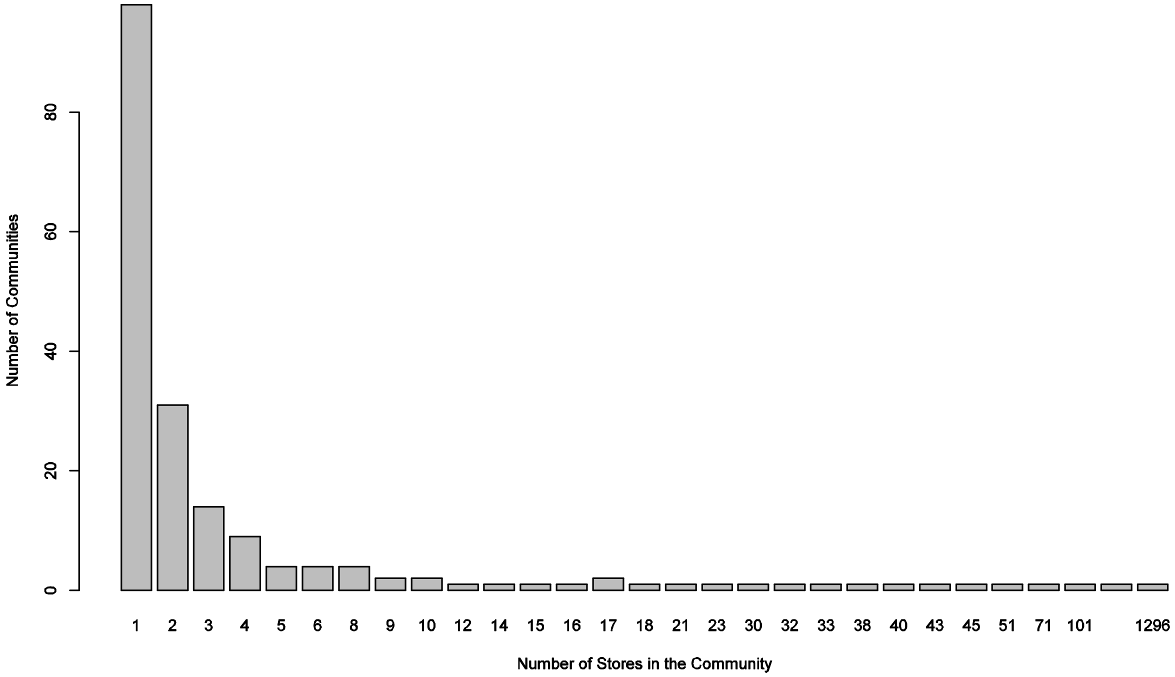


Figure 6.3 The Number of Communities with a Given Number of Stores

6.2 Step 2

Based on the Step 1 results, it is clear that gaining a better understanding of the retail structure of the Victoria downtown core is an extremely relevant undertaking. To do this, we make use of VESDC clustering. However, instead of using the simple high density / low density indicator of the last chapter, given the heterogeneity in total population within this community (from under 1000 people per square kilometer to over 60000 people per square kilometer) it makes sense to use a more refined measure. The relationship we make use of is a four parameter growth curve (actually a shrinkage curve in this case) of the form

$$Epsilon = MaxEps - \frac{MaxEps - MinEps}{1 + e^{-r(Dens-M)}},$$

where *MaxEps* is the upper asymptote on the size of the epsilon radius, *MinEps* is the lower asymptote on the size of the epsilon radius, *r* is the “shrinkage” rate on the size of the epsilon neighborhood as the total population density increases, *M* is the inflection point for the curve, and Dens is the total population density (the sum of residential and workforce population divided by area) for the census designation area the store in located within.

To find good parameter values for this curve we did an iterative grid search over each parameter in the order *MinEps*, *MaxEps*, *M*, and *r*. The initial value of *r* was set to unity, which causes the growth curve to discretely jump from one asymptote to the other at the inflection point. We were surprise how well behaved the function was through EM like iteration process. We fairly quickly converged to the values *MinEps* = 140, *MaxEps* = 170, *M* = 12000, and *r* = 0.00015. Unlike a regression model, there is no way to obtain standard errors for these parameters. To gain a sense of what this relationship looks like, Figure 6.4 provides a plot of the implied relationship between total population density and the value of the epsilon radius.

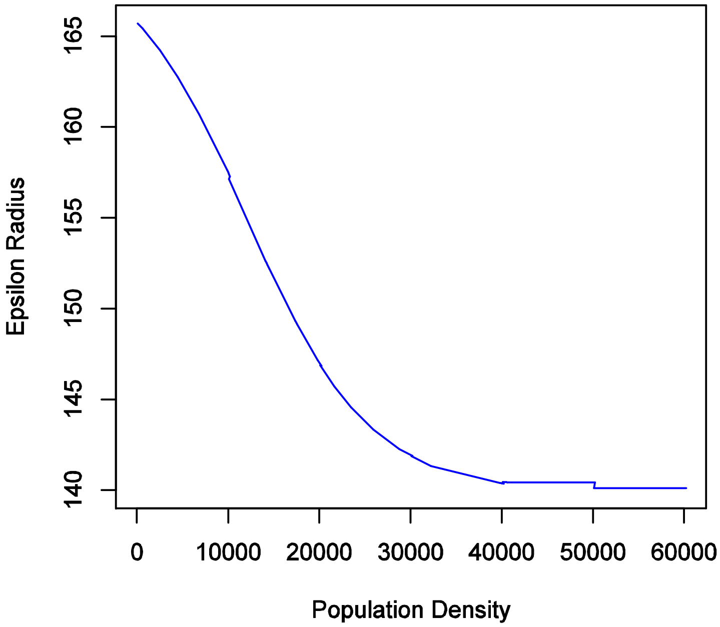


Figure 6.4 The Implied Relationship Between Population and the Epsilon Radius

Based on the parameters of the shrinkage curve given above, Figure 6.5 provides the retail outlet cluster structure for the Victoria downtown core. An examination of this figure appears to be very reasonable, with clusters being fairly well defined. There is still one large cluster with over 350 retail outlets (which can be seen in the central part of the figure towards the bottom). However, these 350 stores are in an area that is between 2 km^2 and 3 km^2 so is extremely densely packed. It would likely be difficult to further partition it in a reasonable way.

The results for the two-step approach for clustering retail outlet locations into retail districts seems very promising based on this application. However, additional applications will need to be undertaken to confirm that the promising results seem here hold more generally.

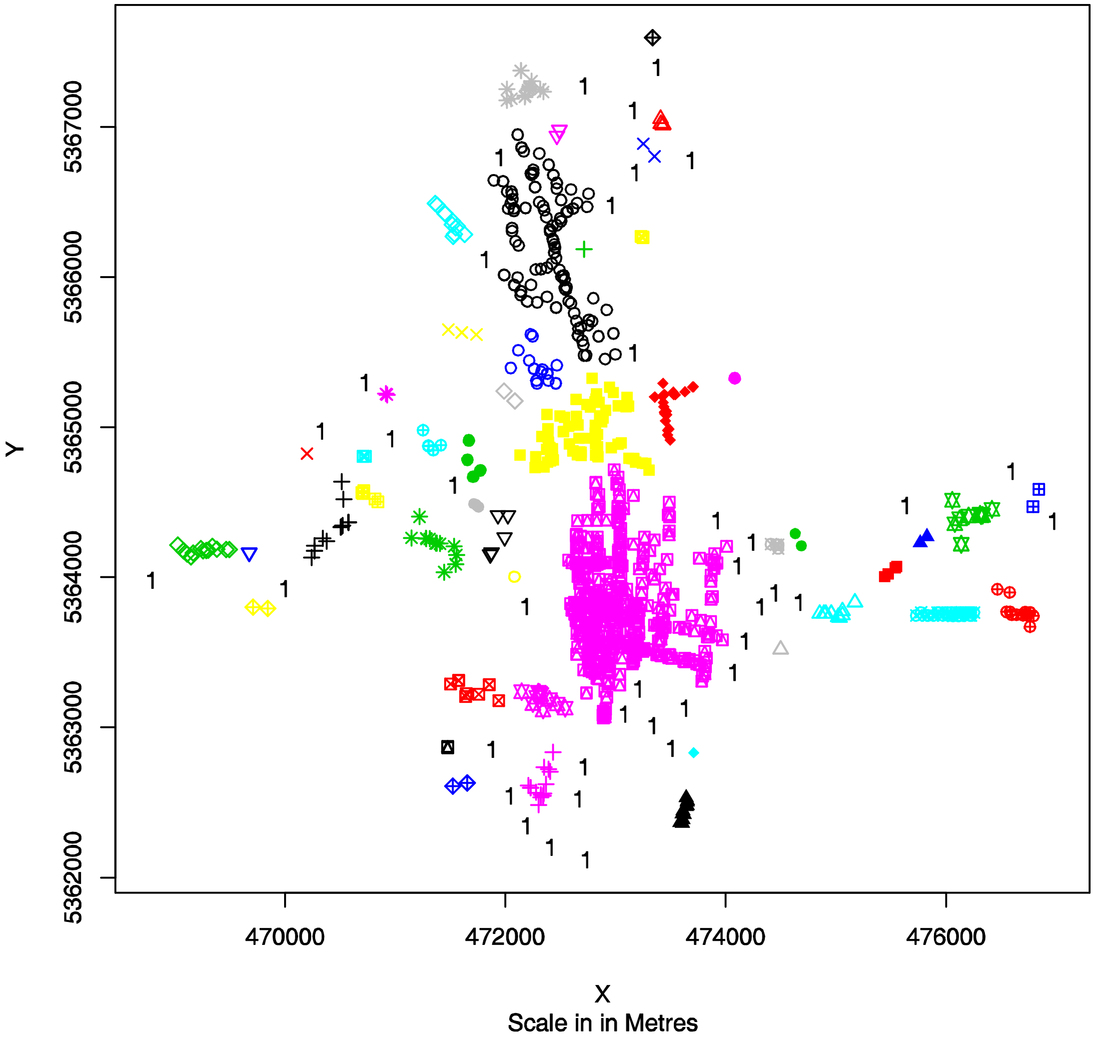


Figure 6.5 The VESDC Cluster Solution for the Victoria Downtown Core

7 Summary and Conclusions

Although aggregating retail outlets into retail districts is an important academic and practical issue in marketing and retailing, only limited academic work has been done on this problem. The growing availability of detailed location data through Geographic Information Systems (GIS) makes this a particularly timely problem. In some cases, analysts use simple geographic designations such as postal codes to form retail districts. While grouping data by postal codes can be a useful first step, postal codes and other geographic boundaries were developed for other purposes and do not summarize retail data well. For example, the Kingsway shopping area in Vancouver includes stores in both Burnaby and Vancouver. In applied situations, experienced managers and analysts sometimes rely on their own background and insights to form retail districts. However, human judgment, even when aided by GIS methods, is time consuming, labor intensive, and generally not replicable.

7.1 Cluster Analysis

Cluster analysis is a sound and well established approach for reducing data dimensionality. However, the existing cluster approaches do not handle the complicated geospatial structure that is typical of retailing data well, primarily due to their high variation in observation density. Newer methods, such as density based clustering, developed in computer science, appear to have promise for marketing settings. One problem is that the "epsilon radius," a measure of how close stores need to be to each other in order to be classified as belonging to the same cluster, is assumed to be constant in such methods as density-based clustering. However, this turns out not to be a good assumption in practice. In addition existing methods of judging the quality of a clustering solution, so called cluster validation methods, do not provide sound guidance as to the best clustering solution for the type of retailing data we study. Consequently, we propose a new two-step clustering approach in which Variable Epsilon Spatial Density Clustering (VESDC) is developed, and a new clustering validation measure, the *CpSp* index, also is introduced.

The *CpSp* index evaluates the overall performance by taking both compactness and separation of a proposed clustering solution into consideration. It is formed from the product of these two measures and is normalized so that it is scaled in the range of 0 to 1. The optimal solution is the one with the highest *CpSp* score. Extensive testing demonstrated that *CpSp* performed well as a cluster validation method. The modified *CpSp* index can accommodate varied epsilon radius in evaluating the performance of clustering solutions.

We applied clustering methods commonly used in marketing such as K-Means to data from the Greater Victoria metropolitan area of British Columbia, and did not get good results. Newly developed density-based cluster methods were also unable to capture the retail district structure in the Greater Victoria area. One critical underlying reason for these results is the variation in the density of retail outlets in downtown, suburban, and other areas included in the Greater Victoria metropolitan area. VESDC effectively clusters data by adjusting the epsilon radius systematically to adapt to the local market environment. In particular, using the exponential "shrinkage" transformation function with four parameters ($MaxEps$, $MinEps$, r, and M), we developed a model in which the epsilon radius is determined by the population density in a small area.

We tested VESDC's performance on synthetic data. The underlying pre-specified data patterns were accurately recovered. We then applied the two-step approach to Greater Victoria. The approach clearly outperformed the existing clustering approaches. VESDEC effectively reduced the data dimensionality to a manageable number by forming retail districts; In addition, isolated retail outlets, which do not properly belong to a retail district, were identified. Also, the VESDEC method is not restricted to convex shapes so that different shapes of retail districts (circle, rectangular, linear, etc) were identified, which can match the actual retail district appearance. The exponential "shrinkage" function used in the model performed well and the four parameters could be identified from the data. One critical reason for the relative success of the VESDC approach is that the varied epsilon radius is consistent with the underlying data structure in which there is high variation in retail store density.

7.2 Future Research

Based on our encouraging results for the VESDC approach and the *CpSp* validation method, we see future research proceeding in at least three directions. The first is to apply the method in different settings, the second is to make further improvements to the methodology, and the third is to apply the method to retail site location problems. We briefly review each of these areas.

7.2.1 Additional Applications

We have tested VESDC over a wide range of simulated data and for Greater Victoria in British Columbia. A logical next step is to test the method over a broader range of geographic areas to test its limitations. Although Victoria is a strong real test of the method, such testing would provide further insight into the robustness of the approach and the value of the *CpSp* measure in identifying cluster structures. Beyond the retailing area, there appear to be many other areas in marketing in which the technique can be applied. Marketing applications have relied very much on traditional approaches such as K-Means and have not, at least in the published literature, incorporated the use of newer methods such as density based clustering.

Segmentation has been a very important applied area for clustering, but existing approaches have a number of limitations. One is not to allow for outliers in the solution. We believe that not to do so is unrealistic in many settings. We believe that VESDC and other methods of density based clustering have much promise for segmentation research and to other marketing problems.

More broadly, our approach can be applied in a wide variety of areas. While different functional forms and relationships than the ones used here to develop the variable epsilon radius will likely be required, the overall concept of adjusting the epsilon radius to different local conditions appears to be widely applicable. Moreover, our *CpSp* measure overcomes weaknesses with the existing *Comp_Sepa* measure that should be relevant in many areas.

7.2.2 Improvement on Methodology

In the current approach, the epsilon radius is related to the underlying population density through a four parameter exponential “shrinkage” transformation function. This response function was developed by logical analysis of the nature of the relationship between these two variables and after extensive testing of alternative functions. Nevertheless, further testing of this relationship would be worthwhile. One extension would be to have additional explanatory variables beyond population density, which might be needed in some settings. On the other hand, additional complexity is not always a good modeling strategy and a simpler approach might be more valuable. In that framework, our analysis suggested that we needed four parameters in our response function, but the possibility of employing a simpler response function should also be investigated.

Our empirical analyses indicated a number of systematic problems with existing cluster validation methods and led to the development of the *CpSp* measure. The *CpSp* and Modified

CpSp measures exhibited very good performance, but given the crucial importance of cluster validation, we believe that continued investigation in this area is worthwhile. We would not be surprised if several cluster validation methods emerged, each providing different insights on the quality of a proposed clustering solution.

7.2.3 Retail Site Location

While the formation of retail districts is interesting in itself, it is often the first step in both academic and applied research. For example, as discussed in the first chapter, Ellickson and Misra (2008) clustered supermarkets to investigate the nature of price competition among stores.

In applied settings, managers frequently want to evaluate the performance of their stores both historically across time for an individual store and comparatively. Store performance, however, relies highly on the surrounding market environment, which in turn, influences the sales potential of that store. Retail district identification can provide such information. Consequently we believe that our approaches can lead to improved estimates of retail performance. Depending upon the application, the retail districts may be formed by only examining stores in the same industry (e.g. having the same SIC) or across industries, to obtain an indicator of market potential and/or store traffic.

For companies, the new store location decision is a challenging task. Estimation of market potential for each possible store location is critical (Hansen and Weinberg 1979). While the surrounding residential and work place population is an important component of demand generation, many approaches do not fully consider the ability of some areas to draw additional traffic from people attracted to the area. Taking as an example the fast food industry, the demand for a fast food store comes from four types of customers, residential customers (who live in the area), daytime population (who work in the area), transitory customers (who pass by the area), and ancillary customers (who visit the area for other purposes). Depending on the store location, the importance of each type of customer may be different. In the fast food industry, ancillary and transitory customers can be particularly important. In practice, estimation of the transitory and ancillary customers without on-site observation is extremely difficult and costly, especially when many possible sites are being considered. However, knowledge of the retail district structure of an area can significantly improve our ability to estimate transitory and ancillary demand. Consequently, we anticipate future research on site location to incorporate the type of retail district information we have developed in this thesis. Through the co-operation of a major fast food retailer, we have already begun work in this direction.

In summary, our new clustering approach, VESDC, and a new clustering validation method, *CpSp* index, have demonstrated good and robust performance in both synthetic and real data. These new approaches appear to be applicable to a wide variety of settings and lead to a number of interesting future research opportunities.

Bibliography

Ahmed, M.N., Yamany, S.M., Mohamed, N., farag, A.A., and Moriarty, T. (2002). A Modified Fuzzy C-Means Algorithm for Bias Field estimation and Segmentation of MRI Data. *IEEE Transactions on Medical Imaging*, Vol. 21. 193-199

Anderson, W.T., Cox, E.P., and Fulcher, D.G. (1976). Bank Selection Decisions and Market Segmentation. *Journal of Marketing*, 40, 40-45

Banfield, J.D. and Raftery, A.E. (1993). Model-based Gaussian and Non-Gaussian Clustering. *Biometrics*, 49, 803-821

Berry, M.J.A. and Linoff, G. (1996). *Data Mining Techniques for Marketing, Sales and Customer Support*. John Wiley & Sons, Inc., USA

Bezdek, J.C., Ehrlich, R., and Full, W. (1984). FCM: Fuzzy C-Means Algorithm. *Computers and Geoscience*, 10, 191-203

Bezdek, J., Hathaway R., Sabin, M., and Tucker, W. (1987). Convergence Theory for Fuzzy C-Means: Counter-Examples and Repairs. *The Analysis of Fuzzy Information*, CRC Press, Vol. 3, Chap. 8

Bowen, J. (1990). Development of a Taxonomy of Services to Gain Strategic Marketing Insights. *Journal of the Academy of Marketing Science*, 18, 43-49

Chiu, S.L. (1994). Fuzzy Model Identification Bases on Cluster Estimation. *Journal of Intelligent and Fuzzy system*. Vol. 2. 267-278

Clarke, K. C. (1997). *Getting Started with Geographic Information Systems*, Upper Saddle River, NJ, Prentice-Hall

Claxton, J.D., Fry, J.N., and Portis, B. (1974). A Taxonomy of Pre-purchase Information Gathering Patterns. *Journal of Consumer Research*, 1, 35-42

Coviello, N.E., Brodie, R.J., Danaher, P.J., and Johnston, W.J. (2002). How Firms Relate to Their Markets: An Empirical Examination of Contemporary Marketing Practices. *Journal of Marketing*, 66, 33-46

Davies, D.L. and Bouldin, D.W. (1979). A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2), 224-227

Day, G.S. and Heeler, R.M. (1971). Using Cluster Analysis to Improve Marketing Experiments. *Journal of Marketing Research*, 8, 340-347

De Berg, M., Cheong, O., Kreveld, M.V., and Overmars, M. (2008). *Computational Geometry: Algorithms and Applications*, 3rd ed., Berlin, Spring-Verlag

Dunn, J.C. (1974). Well Separated Clusters and Optimal Fuzzy Partitions. *Journal of Cybernetics*, 4, 95-104

Ellickson, P.B. and Misra, S. (2008). Supermarket Pricing Strategies. *Marketing Science*, 27 (5), 811-828

Ester, M., Kriegel, H-P., Sander, J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Database with Noise. *In Proceedings of 2nd Int. Conf. On Knowledge Discovery and Data Mining*, Portland, 226-233

Fraley, C. and Raftery, A.E. (2002). Model-based Clustering, Discriminant Analysis and Density Estimation. *Journal of the American Statistical Association*, 97, 611-631

Fraley, C. and Raftery, A.E. (2003). Enhanced Model-based Clustering, Density Estimation, and Discriminant Analysis Software: MCLUST. *Journal of Classification*, 20, 263-286

Fraley, C., Raftery, A.E. and Wehrens, R. (2005). Incremental Model-based Clustering for Large Datasets with Small Clusters. *Journal of Computational and Graphical statistics*. 14, 529-546

Gabriel, K.R. and Sokal, R.R. (1969). A New Statistical Approach to Geographic Variation Analysis. *Systematic Zoology*, 18(3), 259-278

Green, P.E., Frank, R.E., and Robinson, P.J. (1967). Cluster Analysis in Test Market Selection. *Management Science*, 13, 387-400

Greeno, D.W., Sommers, M.S., and Kernan, J.B. (1973). Personality and Implicit Behavior Patterns. *Journal of Marketing Research*, 10, 63-69

Guha, S., Rastogi, R., and Shim K. (1998) CURE: An Efficient Clustering Algorithm for Large Databases. *In Proceedings of the ACM SIGMOD Conference*

Hagerty, M.R. (1985). Improving the Predictive Power of Conjoint Analysis: The Use of Factor Analysis and Cluster Analysis. *Journal of Marketing Research*, 22, 168-184

Halkidi, Maria, Y. Batistakis, and M. Vazirgiannis (2001) On Clustering Validation Techniques. *Journal of Intelligent Information System*, 107-145

Halkidi, M., vazirgiannis, M., and Batistakis, Y. (2000). Quality Scheme Assessment in the Clustering Process. *In Proceedings of PKDD*, Lyon, France.

Han, J. and Kamber, M. (2001). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, USA

Hanson, M.M., and Weinberg, C.B. (1979). Retail Market Share in a Competitive Market. *Journal of Retailing*, 55, 37-46

Hinneburg, A. and Keim, D.A. (1999). Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-dimensional Clustering. *Proceedings of 25th VLDB Conference*, Edinburgh, Scotland, 1999

Hollenstein, H. (2003). Innovation Modes in the Swiss Service Sector: a Cluster Analysis based on Firm-level data. *Research Policy*, 32, 845-863

Hooley, G.J., Lynch, J.E., and Shepherd, J. (1990). The Marketing Concept: Putting the Theory into Practice. *European Journal of Marketing*, 24, 7-24

Jain, A.K., Murty, M.N., and Flyn, P.J. (1999). Data Clustering: A Review. *ACM Computing Surveys*, 31(3), 264-323

Kerin, R.A. and Cron W.L. (1987). Assessing Trade Show Functions and Performance: An Exploratory Study. *Journal of Marketing*, 51, 87-94

Kernan, J.B. (1968). Choice Criteria, Decision Behavior, and Personality. *Journal of Marketing Research*, 5, 155-169

Kiel, G.C. and Layton, R.A. (1981). Dimensions of Consumer Information Seeking Behavior. *Journal of Marketing Research*, 18, 233-239

Kolen, J.F. and Hutcheson, T. (2002). Reducing the Time Complexity of the Fuzzy C-Means Algorithm. *IEEE Transactions on Fuzzy Systems*. Vol. 10. 263-367

Krieger, A.M. and Green, P.E. (1996). Modifying Cluster-Based Segments to Enhance Agreement with an Exogenous Response Variable. *Journal of Marketing Research*, 33, 351-363

Landon, E.L. (1974). Self Concept, Ideal Self Concept, and Consumer Purchase Intentions. *Journal of Consumer Research*, 1, 44-51

Lessig, V.P. and Tollefson, J.D. (1971). Market Segmentation Through Numerical Taxonomy. *Journal of Marketing Research*, 8, 480-487

Liu, S. and Huang, Y. (2007). A New Clustering Validity Index For Evaluating Arbitrary Shape Clusters. *In Proceedings of Sixth International Conference on Machine Learning and Cybernetics*.

MacQueen, J. (1967). Some methods for Classification and Analysis of Multivariate Observations. *In Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, 1, 281-197

Matula, D.W. and Sokal, R.R. (1980). Properties of Gabriel Graphs Relevant to Geographic Variation and the Clustering of Points in the Plane. *Geographic Analysis*, 12(3), 205-222

McLachlan, G.J., and Krishnan, T. (1997). *The EM Algorithm and Extension*, New York: Wiley

Moriarty, M. and Venkatesan, M. (1978). Concept Evaluation and Market Segmentation. *Journal of Marketing*, 42, 82-86

Morrison, B.J. and Sherman, R.C. (1972). Who Responds to Sex in Advertising? *Journal of Advertising Research*, 12, 15-19

Pal, N.R. and Bezdek, J.C. (1995). On Cluster Validity for the Fuzzy C-Means Models. *IEEE Transactions on Fuzzy Systems*. Vol.3. 370-379

Pal, N.R. and Biswas, J. (1997). Cluster Validation Using Graph Theoretic Concepts. *Pattern Recognition*. 30(6). 847-857

Pham, D.L. and prince J.L. (1999). An Adaptive Fuzzy C-Means Algorithm for Image Segmentation in the presence of Intensity Inhomogeneities. *Pattern Recognition Letters*. 20, 57-68

Pilevar, A.H. and Sukumar, M. (2005). GCHL: A Grid-based Algorithm for High-dimensional Very Large Spatial Data bases. *Pattern Recognition Letters,* 26, 999-1110

Punj G. and Stewart, D.W. (1983). Cluster Analysis in Marketing Research: Review and Suggestions for Application. *Journal of Marketing Research*, Vol. 20, 134-148

Razaee, R., Lelieveldt, B.P.F., and Reiber, J.H.C. (1998). A New Cluster Validity Index for the Fuzzy C-Mean. *Pattern Recognition Letters*, 19, 237-246

Schaninger, C.M., Lessig, V.P., and Panton, D.B. (1980). The Complementary Use of Multivariate Procedures to Investigate Nonlinear and Interactive Relationships Between Personality and Product Usage. *Journal of Marketing Research*, 17, 119-124

Sethi, S.P. (1971). Comparative Cluster Analysis for World Market. *Journal of Marketing Research*, 8, 348-354

Sexton, D.E. (1974). A Cluster Analytic Approach to Market Response Functions. *Journal of Marketing Research*, 11, 109-114

Sheikholeslami, G., Chatterjee, S., and Zhang, A. (1998). WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial databases. In *Proceedings of the 24th VLDB Conference*, New York, USA

Simmons, J., Barbiero, P., Bylov, G., Kamikihara, S., andZsigovics, G. (2000). Exploring a National Database of Commercial Activity. *Center For the Study of Commercial Activity*, Ryerson Polytechnic University, Toronto

Srivastava, R.K., Leone, R.P., and Shocker, A.D. (1981). Market Structure Analysis: Hierarchical Clustering of Products Based on Substitution-in-Use. *Journal of Marketing,* 45, 38-48

Srivastava, R.K., Shocker, A.D., and Day, G.S. (1978). An Exploratory Study of Usage-Situational Influence on the Composition of Product-Markets in Advances in Consumer Research, Ann Arbor: *Association for Consumer Research,* 32-38

Theodoridis, S. and Koutroubas, K. (1999). *Pattern Recognition*. Academic Press

Toussaint, G.T. (1980). The Relative Neighbourhood Graph of a Finite Planar Set. *Pattern Recognition,* 12 (4), 261-268

Wang, W., Yang, J., and Muntz, R. (1997). STING: A Statistical Information Grid Approach to Spatial Data Mining. *In Proceedings of 23rd VLDB Conference*

Ward, J.H. (1963). Hierarchical Groupings to Optimize an Objective Function. *Journal of the American Statistical Association*, 58, 234-244

Yeung, K.Y., Fraley, C., Murua, A., Raftery, A.E., and Ruzzo, W.L. (2001). Model-based Clustering and data Transformations for Gene Expression Data. *Bioinformatics*, 17 (10), 977-987

Zaiane, O. and Lee, C. (2002). Clustering Spatial Data When Facing Physical Constraints. *In Proceedings of the 2002 IEEE International Conference on Data Mining*

Zhang, D and Chen, S. (2003). Clustering Incomplete Data using Kernel-Based Fuzzy C-Means Algorithm. *Neural Processing Letters*, 18, 155-162

Zhong, S. and Ghosh, J. (2003). A Unified Framework for Model-based Clustering. *Journal of Machine Learning Research*, 4, 1001-1037