

VALIDATION OF CLUSTERING METHODS FOR POSITRON EMISSION TOMOGRAPHY
DATA

by

Prasanna K Velamuru

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
MASTER OF SCIENCE in COMPUTATIONAL BIOSCIENCES

REPORT NUMBER : 05-02

ARIZONA STATE UNIVERSITY

May 2005

VALIDATION OF CLUSTERING METHODS FOR POSITRON EMISSION TOMOGRAPHY
DATA

by

Prasanna K Velamuru

has been approved

May 2005

APPROVED:

, Chair

Supervisory Committee

ABSTRACT

Positron Emission Tomography (PET) data, in general, is difficult to segment due to low signal to noise ratio. However, in recent years, it has been demonstrated that clustering can be used as an important preprocessing step prior to parametric estimation from dynamic PET data. Classical clustering methods such as hierarchical clustering and k-means have been used to improve the accuracy of voxel level quantification in PET images. To obtain meaningful cluster groupings, it is necessary to perform clustering using an appropriate weighting technique, for each slice, based on the different time instants at which the data is sampled. Traditional hierarchical clustering methods require one to maintain a connectivity matrix showing the (dis)similarities between voxels. This requirement places severe computer memory constraints due to the high dimension of PET data. Prior work done by our group has shown that an application of a two-stage clustering process that combines a preclustering scheme along with classical hierarchical analysis can be used as a fast clustering alternative for dynamic PET data. In spite of the significant advances made in the application of various clustering strategies to dynamic PET data, very little work has been done related to validation of cluster results and methods in this domain. In this study, we compute and analyze several known intra-cluster measures, inter-cluster measures and indices that are a combination of these measures. Quantification and evaluation of the clustering results with the help of these measures is important in getting an estimate of the optimal number of clusters within the data. This information is crucial since most of the commonly used clustering methods are unsupervised and it is difficult to know where to cut the hierarchical tree or how many partitions are desired. Cluster validation is also applied to compare different clustering methods with respect to efficiency and accuracy, in which accuracy is measured with respect to whether or not a clustering method that is less computationally expensive still maintains the characteristics of an expensive clustering method. The results of this study show that it makes only little difference in the results of clustering integral data (scalar values) as compared to the entire time course data (TACs), when measured with respect to most inter, intra and inter-intra cluster measures, except that the relevant measures are generally higher for the multidimensional clustering meaning, not surprisingly, that the

TAC clustering produces clusters not only of greater width but also better separated. When using inter-intra cluster measures, such as the average silhouette width and modified Dunn's ratios, for estimating the optimal number of clusters for a given data set, integral data give more pronounced maxima. Moreover, it is not appropriate to a-priori set the number of clusters in any case, because it is clear that dependent on slice and subject, the optimal number of clusters varies. In general, it can be concluded that for a fast clustering, use of the integral data is sufficient, but that for more separated clusters TTAC data should be utilized, and that it is insignificant whether fast linkage algorithms are adopted as compared to their standard forms. On the other hand, K-means is very fast but the average dissimilarity within clusters is relatively high and separation relatively low compared to the linkage methods.

To My Beloved Parents

For your perennial Love, Encouragement and Magnanimity

ACKNOWLEDGMENTS

I am extremely grateful to my mentor and internship advisor, Dr. Rosemary Renaut, for her valuable guidance and support. I would like to thank Dr. Hongbin Guo for all his help and valuable suggestions. I would like to thank Dr. Huan Liu for kindly consenting to be on my committee and for his feedback.

This work was partially supported by the Arizona Center for Alzheimer's Disease Research which is funded by the Arizona Department of Health Services and by NIH grant EB 2553301

TABLE OF CONTENTS

CHAPTER 1	Introduction	1
1.	Positron Emission Tomography- A brief overview	1
1.1.	Labeling and FDG	2
2.	FDG-PET tracer kinetic model	3
3.	Clustering as a preprocessing tool for PET data - related literature	5
4.	General Problem description and Goals of this study	5
CHAPTER 2	Review of Clustering Concepts and Methods	11
1.	Hierarchical Clustering [10, 13]	13
1.1.	Linkage Metrics	14
2.	Partitional Clustering [14, 13]	16
2.1.	Squared Error Clustering Criteria [11]	18
2.2.	Other Partitional Clustering Methods	19
CHAPTER 3	Cluster Validation	22
1.	Indices of Cluster Validity	22
1.1.	Indices Used for External Criteria	23
1.2.	Indices for Internal Criteria	25
1.3.	Indices for validating Hierarchical structures using Internal Criteria	25
1.4.	Indices used for Relative Criteria	26
2.	Other criteria	28
CHAPTER 4	Clustering Methods and Validation Indices Used in the Study	29
1.	Distance Measures	29
2.	Histogram-based Thresholding	30
3.	Preclustering	31
4.	Clustering Methods	32
4.1.	Classical Hierarchical Linkage Algorithms	32

4.2.	Parameter-Dependent Algorithm [3]	33
4.3.	Modified Parameter Dependent Algorithm [3]	37
4.4.	K-Means Algorithm [11, 36, 37]	37
5.	Clustering Validation Indices used in this Study	38
5.1.	Intra-Cluster measures	39
5.2.	Inter-Cluster Measures	40
5.3.	Inter-Intra Cluster Measures	41
6.	Statistical Testing	43
CHAPTER 5 Results		45
1.	Clinical Data	45
2.	Mean Time Activity Curves	46
3.	Representative Slice Results Obtained using Integrals and TACs	47
CHAPTER 6 Results-Observations and Conclusions		60
1.	Data: Integrals	60
2.	Data: Time Activity Curves	64
3.	Significant Differences between TACs and Integrals	68
4.	Optimal Number of Clusters	68
4.1.	Conclusions	69
5.	Conclusions	70
6.	Future Work	71
Bibliography		72

CHAPTER 1

Introduction

This report describes in detail the internship work done that concerns clustering and validation of clusters obtained from sequences of PET images. The quantitative information obtained as a result of clustering PET images is used to analyze and validate the efficiency of different clustering algorithms with respect to PET data. Clustering and cluster validation is considered to be an important preprocessing tool in developing more efficient methods for parameter estimation in human brain FDG-PET studies that are based on a three-compartment model. The ultimate goal is to use the appropriate cluster method in an automated tool for parametric estimation of brain function for Alzheimer's Disease studies.

1. Positron Emission Tomography- A brief overview

Positron Emission Tomography (PET) is a non-invasive, diagnostic imaging technique for measuring the metabolic activity of cells in the human body. PET allows physicians and scientists to determine the function of various organs and tissues. Unlike other scanning techniques such as CT scans and MRI scans that help in looking at the anatomy, PET allows us to look at the function of the human body. Functional information is very useful in the sense that it enhances our knowledge of the biochemical basis of both normal as well as abnormal functions that occur within the body. Biochemical examination of patients through PET scans is now an integral part of their clinical care. It is important to have these capabilities because of the following reasons [1]:

- The basis of all tissue function is chemical in nature.
- Diseases result from errors introduced into chemical systems by viruses, bacteria, genetic abnormalities, drugs, environmental factors, aging and behavior.
- The most selective, specific, and appropriate therapy is one chosen from a diagnostic measure of the basic chemical abnormality.
- Detection of chemical abnormalities provides the earliest identification of disease, even in the presymptomatic stages before the disease process has exhausted the chemical reserves or overridden the compensatory mechanisms of the brain.
- Assessment of restoration of chemical function provides an objective means for determining the efficacy of therapeutic interventions in the individual patient.
- The best way to judge whether tissue is normal is by determining its biochemical function.

The relevance of examining the biochemical processes with an imaging technology such as PET can be attributed to the fact that in most cases the location and the extent of a disease is unknown and cannot be detected through anatomy scanning techniques. PET imaging provides an efficient means of searching throughout the body to determine exact location. The visual system is widely regarded as the most efficient human sensory system for search identification and interpretation. Hence, the pictorial output generated by the PET imaging technology makes it a lot easier for physicians and researchers to develop/suggest appropriate diagnostic measures for the specific disease condition.

1.1. Labeling and FDG. Chemical compounds that need to be followed through the body are labeled with radioactive atoms that decay by emitting positrons (β^+). Such positron-emitting isotopes for PET are produced with charged particle accelerators such as *cyclotrons*. Labeling is a process of attaching some kind of identifying tag, using the radioactive isotope, to the compound that would help us to identify where the compound has gone. In PET, the compounds that can be labeled are limited only by the imagination of the investigators and the physical half-life of the

positron emitting label. Nearly 1400 different references of labeled compounds have been listed recently [2]. Due to the short half-life of the positron-emitting radioisotopes, the chemical synthesis of the labeled compounds has to be rapid. The *specific activity* of a labeled compound is defined as the ratio of radioactivity to the mass of the compound. This measure is useful to determine the mass of a compound that needs to be injected in a patient to achieve the desired radioactivity.

A tracer is a substance that follows a process under study. It usually refers to the use of compounds labeled with a positron-emitting radioisotope. A tracer must satisfy the following criteria [1]:

- The behavior of the tracer should, in a known and predictable manner, be identical or related to that of the natural compound or process being traced.
- The mass of the tracer used should not alter the process being studied.
- Any difference between the tracer and the natural compound should be negligible, or a correction should be applied to the effect.

[^{18}F]-labeled 2-deoxyglucose (FDG) is a widely used tracer with several applications in neurology, cardiology and oncology for the study of glucose metabolism. PET studies performed using FDG are usually referred to as FDG-PET studies. Since [^{18}F]-labeled FDG measures glucose metabolism, it is also useful for tumor localization and quantification. FDG is potentially useful in differentiating benign from malignant forms of stimulated osteoblastic activity because of the high metabolic activity of many types of aggressive tumors.

2. FDG-PET tracer kinetic model

In our study, we consider a three-compartment FDG tracer kinetic model, shown in Figure 1, that is used for the estimation of cerebral metabolic rate of glucose. The functional activity in the brain can be related to the cerebral metabolic rate of glucose. Compartment models are the most common form of tracer kinetic models used in PET [1]. A compartment is a volume or space

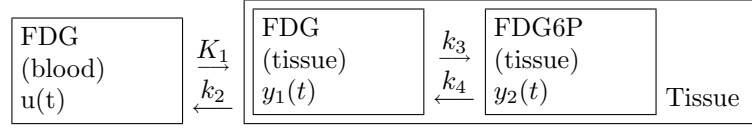


Figure 1. FDG tracer kinetic model

within which the tracer gets uniformly distributed. The time behavior of a compartment is derived through the kinetic rate constants which connect the compartments. The separate compartments within the model represent the different stages of the FDG sequence from blood to tissue, including phosphorylation in tissue.

The measurement of local cerebral metabolic rate for glucose metabolism within the model is based on the principle of *competitive substrate kinetics*. The FDG tracer competes with glucose for the carrier-mediated transport sites in the Blood-Brain Barrier (BBB) region. After entering the cerebral tissue, FDG and glucose then compete for hexokinase in order to get phosphorylated to FDG-6-Phosphate and glucose-6-phosphate. After phosphorylation, glucose-6-phosphate either proceeds down the glycolytic pathway or is converted to glycogen. Unlike glucose-6-phosphate, FDG-6-phosphate is not a significant substrate in the glycolytic pathway and does not get converted to glycogen. Hence, FDG-6-phosphate does not diffuse through the membranes and is instead trapped and accumulated in cells. The amount of FDG-6-phosphate in the cells is in proportion to the local cerebral metabolic rate for glucose metabolism. The first compartment in the model represents the FDG concentration in blood plasma. This is denoted as $u(t)$. The second compartment represents FDG concentration in tissue that is denoted as $y_1(t)$ and the third compartment refers to the concentration of phosphorylated FDG (FDG-6-Phosphate) within the tissue denoted by $y_2(t)$. The dynamics of the tracer within the model can be described by the following initial value problem [3]

$$\frac{dy_1}{dt} = k_1 u(t) - (k_2 + k_3)y_1(t) + k_4 y_2(t) \quad (1.1)$$

$$\frac{dy_2}{dt} = k_3 y_1(t) - k_4 y_2(t) \quad \text{with} \quad y_1(0) = 0 \quad \text{and} \quad y_2(0) = 0. \quad (1.2)$$

k_1, k_2, k_3 and k_4 are the rate constants that describe the relationships between the concentrations and fluxes of a substance between two compartments. k_1 represents the transport rate from the blood plasma to the extra-vascular space. k_2 is the rate constant that describes the transport rate from the extra-vascular space to the blood. k_3 represents the phosphorylation rate of the intra-cellular FDG by the hexokinase enzymes to FDG-6-Phosphate and k_4 represents the dephosphorylation rate of the intra-cellular FDG-6-phosphate.

3. Clustering as a preprocessing tool for PET data - related literature

Clustering could be viewed as a process of segmentation of data. Unlike in the field of Magnetic Resonance Imaging (MRI), where clustering has been extensively researched [4] and used as a form of segmentation, this process has not been explored to such great depth with respect to dynamic PET imaging. This is mainly attributed to the low signal-to-noise ratios (SNR) in dynamic PET images. The results presented by a few research groups in the recent past indicate that clustering could be used as an effective preprocessing step prior to parametric estimation, thus improving the accuracy of voxel level quantification of the parameters [5, 6]. *Kimura et al.* [5] used Principal Component Analysis for clustering by using simulated data with and without noise while *Zhou et al.* [6] used the classical hierarchical average linkage (HAL) method. *Acton et al.* [7] used Fuzzy *c*-means clustering (FCM) for segmentation of PET images. *Liptrot et al.* [8] use K-means clustering for cluster analysis in kinetic modelling of the brain.

4. General Problem description and Goals of this study

In parametric PET studies, the tissue concentration $y(t)$ is the model output and is taken to be the image intensity of a reconstructed PET image. The FDG plasma concentration i.e., the input function $u(t)$ can be determined either non-invasively by PET imaging of the blood supply

through the carotid artery in the lower portion of the brain or invasively by taking arterial or venous samples at specific intervals. The main goal of an FDG-PET study is to accurately determine the values of the rate constants at the voxel level. The ultimate goal of our research group's endeavor is to develop an automated tool for determination of these values. Such a tool would be extremely useful for Alzheimer's disease studies in the future.

The PET data used for purpose of our study, was generated with the help of a 951/31 ECAT scanner that generates a sequence of volumetric PET images of the brain that consist of 31 different slices and that are taken at different time instants $t_i, i = 1, \dots, n$, where n is typically a small number, precisely 22, that is used in this study and the time intervals are computed as $\Delta t_i = t_{i+1} - t_i$. The 21 time intervals need not be constant across the range t_i, \dots, t_n . Figure 3 illustrates the brain volume at the last time frame for a given subject. Hence, the output data is in the form of a vector with 21 different values representing the intensities at different time instants for each voxel in the parametric PET image. Figure 4 illustrates the 21 different time frames of slice 16 for a given subject. Each of these vectors are known as the output *Time Activity Curves(TACs)*. Figure 2 gives examples of the measured output for random choices of voxels.

For the purpose of study, we also consider output data obtained by integrating the TACs over the time intervals. This converts the data from a vector to a scalar. There is an obvious loss in the resolution when we go from the TACs to the integrals but we expect an improvement in the computational cost. One goal of the study is to evaluate the difference in the cluster results obtained using these two different kinds of output data and to test if there is any clear benefit in making this tradeoff. It should be noted that every method and calculation used in this study that are introduced in the following chapters can be applied to integral data too unless otherwise stated.

The main area of focus of the study described in this report is the output data that is used to obtain the parametric values. There are certain important characteristics of this data that require attention [3]:

- The output data is very noisy. There are a quite a few negative values in the output data

even though the output data represents the intensity values that correspond to densities that should be positive. This is not a major issue of concern since these values can be thresholded and set to zero. We are more concerned about the noise in the non-zero portion wherein we see a lot of oscillations in the data instead of monotonically increasing data points.

- The data points indicate the unequal time intervals over which PET data is obtained.
- The initial time intervals are very short in order to see the gradient in the output. There is usually a steep increase in the output at early time intervals.
- The last time interval is long, and the output changes little over this interval but this time interval data is useful for determining the long term decay term in the output, and hence contributes significantly in the estimation of the kinetic rates.

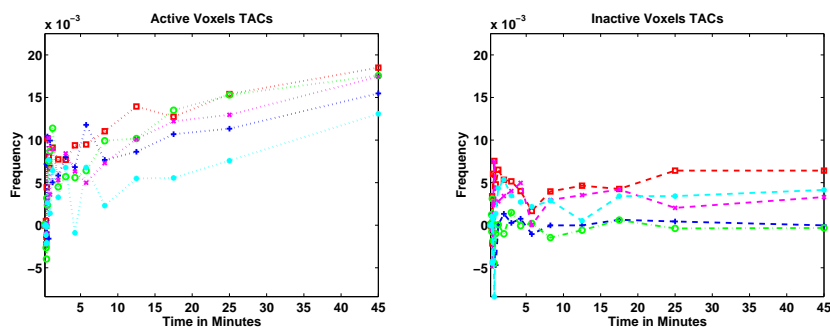


Figure 2. Representative output data $y(t)$ for a random selection of 5 active and inactive voxels. Note that the data are measured at times 0.1, 0.2167, 0.25, 0.2833, 0.3167, 0.35, 0.3833, 0.4167, 0.45, 0.55, 0.7167, 0.9, 1.25, 2, 3, 4.25, 5.75, 8.25, 12.5, 17.5, 25 and 45, measured in minutes.

The above characteristics of parametric PET data that describe the FDG tracer dynamics in the brain provide us with an insight into the difficulties associated with the problem of parametric estimation from this data. To obtain statistically reliable estimates of kinetic rate constants at the voxel level for any given region in the brain, it is absolutely crucial that this data is adequately pre-processed to reduce the influence of the high percentage of noise in this data on the parametric estimation procedures.

As was cited in the previous section, clustering has been demonstrated to be an important

preprocessing step performed prior to parametric estimation from dynamic PET data. Clustering, as a form of segmentation, is useful in providing better information that helps to partially overcome the effect of noise in the data and thereby improve the accuracy of voxel level quantification in PET images [3].

The primary goal of the study elucidated in this report is to validate the clusters obtained from applying different clustering techniques. This is done on the basis of the quantitative data obtained using intra, inter and intra-inter cluster metrics that are described in detail in the later sections of this report. It is important to validate the clusters and the clustering methods because most commonly used clustering methods are unsupervised and it is difficult to know apriori where to cut the hierarchical tree or how many partitions are desired. Once the methods are adequately validated, they can be used as a standard pre-processing procedure before estimation of the parametric values.

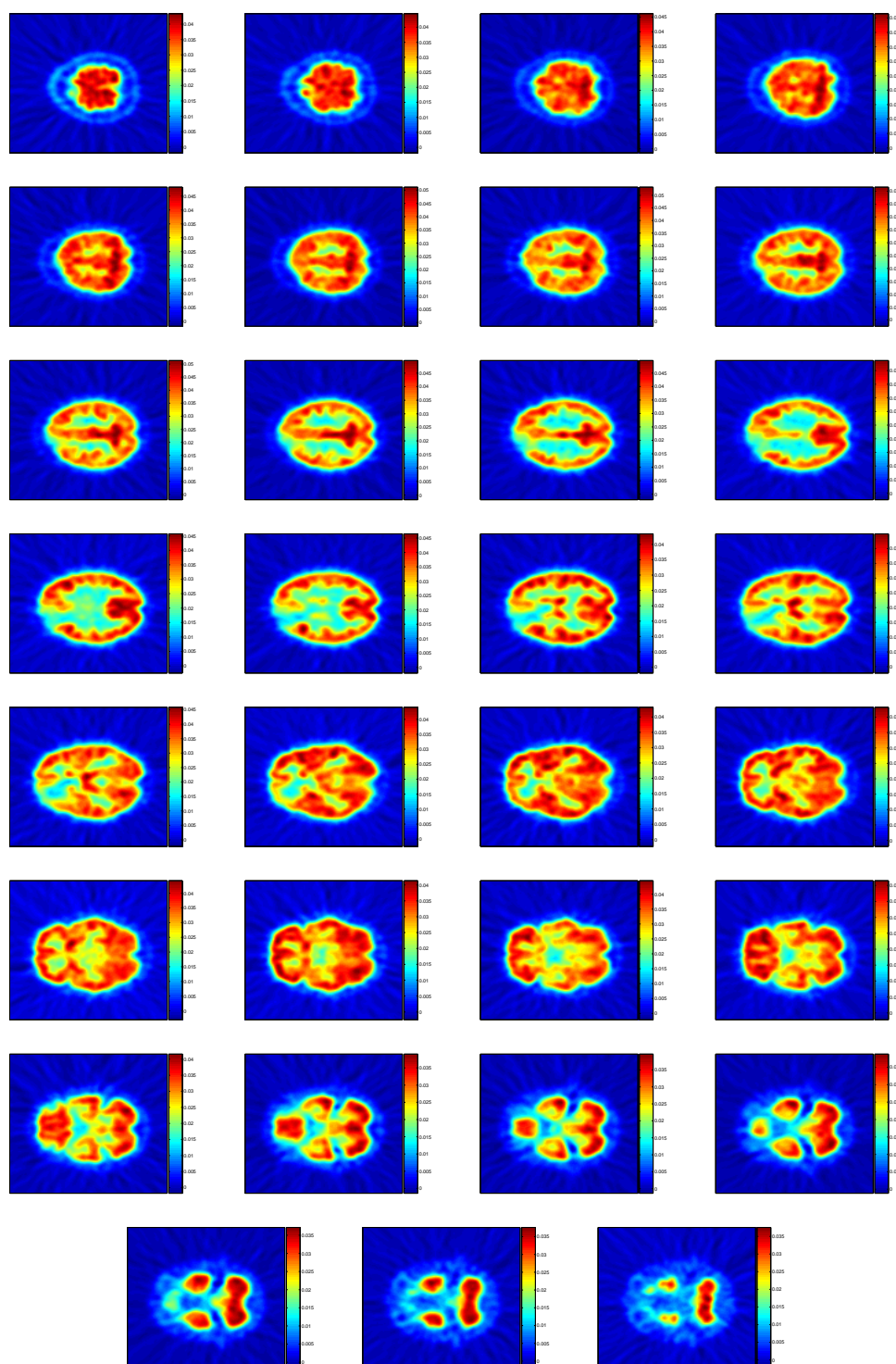


Figure 3. Entire Brain Volume at the last time frame ($t=45$ minutes)

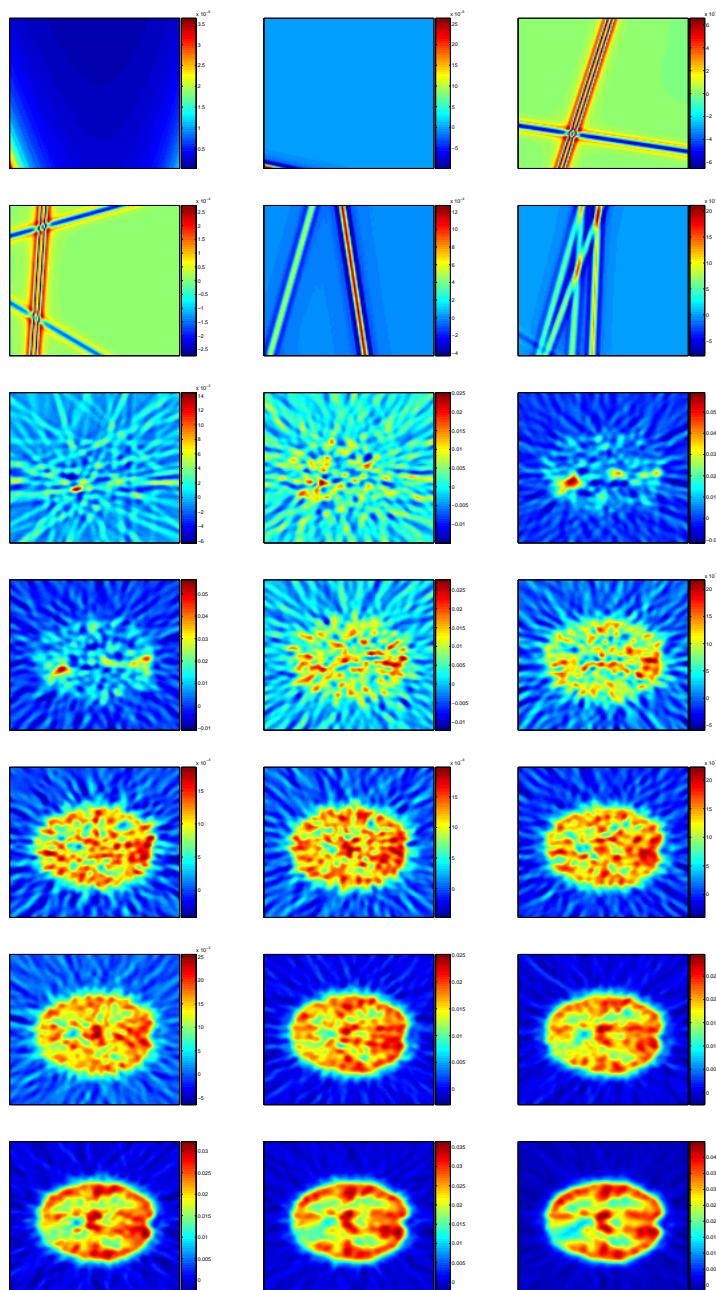


Figure 4. Brain Slice 16 data over 21 time intervals

CHAPTER 2

Review of Clustering Concepts and Methods

Clustering refers to the process of classifying data into groups of similar objects. Each group, called a *cluster*, consists of objects each of which are similar to each other but dissimilar to objects of other groups. Each cluster is presumed to have a meaning in the context of a particular problem.

The central concept behind forming clusters is to have the knowledge of how “close” or “far apart” are different data points with respect to each other. The general term used to refer to the degree of closeness or dissimilarity between cluster components is *proximity*. Different proximity measures exist depending on the data at hand. For example, in case of categorical data with only two levels, two individual data constituents would have a similarity coefficient of unity if both have identical values for all variables and they will have a similarity coefficient of zero in other cases. Likewise, different proximity measures exist for binary data, categorical data with more than two levels, continuous data and also for data comprised of both continuous and categorical variables. Different measures of proximity can be calculated on the same set of data and each of them may lead to different solutions. For example, in the case of continuous data, proximity measures can further be classified into *distance measures* and *correlation measures*. Different distance measures such as the Euclidean distance, City block or Manhattan distance, Minkowski distance, Canberra distance etc., and different correlation measures based on Pearson correlation etc., have been formulated and extensively used for clustering continuous data.

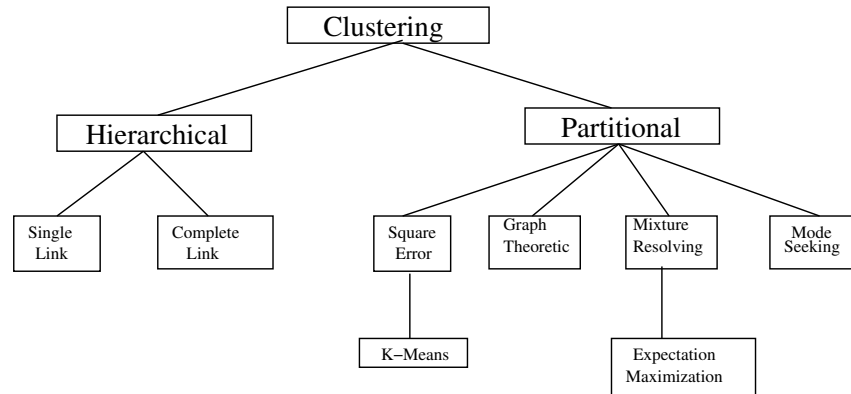


Figure 5. Types of Clustering Methods

Consequently, it would be helpful to know which particular measure(s) can be considered as being *optimal*. Unfortunately, despite quite a few comparative studies [17, 18, 19] that were performed to compare different proximity measures, it has not been conclusively determined as to what measure can be considered to be “optimal” in an absolute sense. The choice of a proximity measure is largely determined by the type of variables being used for clustering and the intuition of the researcher.

Categorization of clustering methods is neither straightforward nor canonical [12]. One way to describe some of the popular methods would be by considering a broad classification such as that shown in the following figure [13]. Hierarchical clustering methods produce a nested series of partitions whereas partitional methods produce only one. It is often the case that one uses an algorithm to express a clustering method. There could be several options for implementing these algorithms such as those described below [13, 14].

- *Agglomerative vs. Divisive*: Agglomerative approach refers to having each distinct pattern in a distinct cluster and successively merging clusters until a specific stopping condition is satisfied. Unlike the agglomerative approach, the divisive method starts off with having all the patterns in a single cluster following by successive splitting until a stopping criterion is satisfied.
- *Monothetic vs. Polythetic*: This option refers to the sequential or simultaneous use of features

in the clustering process. A monothetic clustering algorithm uses the features one by one, while a polythetic algorithm uses all of the features together. Most of the commonly used clustering algorithms are polythetic.

- *Hard vs. Fuzzy:* A hard clustering method would associate each pattern to a single cluster during its operation and in its output whereas a fuzzy clustering method would assign degrees of membership in different clusters to each input pattern. Thus, a fuzzy clustering result can be converted into a hard clustering by assigning each pattern to a cluster with the largest measure of membership.
- *Deterministic vs. Stochastic:* In partitional methods that are designed to optimize a squared error function, the optimization can be achieved using traditional (deterministic) search techniques or through a random (stochastic) search of the state space.
- *Incremental vs. Non-incremental:* This option is especially relevant when dealing with large pattern sets and when memory and time constraints affect the algorithm. Non-incremental algorithms aim at either minimizing number of scans through the pattern set, or reducing the number of patterns examined during execution or reducing the size of the data structures used to represent the patterns.

1. Hierarchical Clustering [10, 13]

In hierarchical clustering, the objects are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to N clusters each containing a single object. Hierarchical Clustering is subdivided into *agglomerative methods*, which proceed by series of fusions of the N objects into groups, and *divisive methods*, which separate N objects successively into finer groupings [16]. Agglomerative methods are the most widely used of the hierarchical methods. Conceptually, hierarchical clustering methods build a tree of cluster, also known as a *dendrogram*. This allows for assessing data on different levels

of granularity. An important feature in hierarchical clustering methods is the construction of an $N \times N$ matrix of distances that represents the (dis)similarities between the data points. This matrix is usually referred to as the *connectivity matrix*. It is often the case that N is a large number for PET data. It would not be practical to store such a large matrix in the computer memory. To get around this limitation, some of the methods usually adopted are to introduce sparsity in the matrix by omitting certain values that are smaller than a certain threshold, by using only a subset of data points, or by keeping with each point, only a certain set of its nearest neighbors. Different variants of hierarchical methods are based upon the different linkage metrics that are constructed from this matrix.

1.1. Linkage Metrics. In order to merge or split subsets of objects rather than individual objects, the distance between individual objects has to be generalized to the distance between the subsets. This derived proximity measure is called a *linkage metric*. This measure reflects the level of closeness and connectivity of the clusters. The commonly used inter-cluster linkage metrics are *single link*, *average link* and *complete link*. The dissimilarity measure d is usually represented by the distance between each pair of objects with one object in one set and the other object in another set. The different linkage metrics differ in the mathematical operation that is applied to a pair-wise dissimilarity measure. For example, single linkage uses “minimum”, average linkage uses “average” and complete linkage uses “maximum” as the *operation*. The general formula is:

$$d(C_1, C_2) = operation\{d(x, y) | x \in C_1, y \in C_2\}$$

Generalized Agglomerative Hierarchical Clustering Algorithm [13]

- | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"> 1 Compute the proximity matrix containing the distance between each pair of patterns consisting of pixels/voxels. Each pattern is treated as a cluster.
 2 Find the most similar pair of clusters using the proximity matrix. Merge these two clusters into one cluster. Update the proximity matrix to reflect this merge operation. Different similarity measures could be used to compute the similarity between the pairs.
 3 If all the patterns are in one cluster, then stop, otherwise repeat go back to step 2. |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Most of the early hierarchical clustering algorithms defined the inter-cluster distances in terms of pairs with objects in two respective clusters. These are generally referred to as the *graph* methods since they do not use any other cluster representation other than a set of points. Some of the later methods known as *geometric* methods represent an individual cluster by its central point. This results in different linkage metrics such as *centroid*, *median* and *minimum variance* that are used to determine the proximity between clusters. The different linkage metrics that are commonly used for agglomerative methods could be derived as instances of the following recurrence formula that was formulated by Lance and Williams [22]. It represents the distance between a group k and a group (i, j) that is formed by the fusion of two groups i and j as

$$d_{k(ij)} = \alpha_i d_{ki} + \alpha_j d_{kj} + \beta d_{ij} + \gamma |d_{ki} - d_{kj}|,$$

where d_{ij} is the distance between groups i and j and α, β and γ are the coefficients corresponding to a particular linkage.

Hierarchical clustering methods based on linkage metrics result in clusters of proper shape, that are flexible and are easy to handle. Hierarchical methods suffer from the inability to repair what was done in the previous steps[15]. Essentially, when an agglomerative algorithm has joined two objects, they cannot be separated. The same is the case with divisive algorithms, wherein it would not be possible to reunite objects that have already been split. Hierarchical methods are also considered to *impose* structure on the data rather than *reveal* structure in the data. It has been rightly pointed out in [15] that this rigidity of hierarchical methods is both the key to their success and their main disadvantage. Nevertheless, hierarchical clustering methods are widely used in different domains by incorporating appropriate heuristics to deal with their limitations. Hierarchical methods have been found to be very useful for biological applications particularly in the classification of animals and plants. Biologists have been credited with being the most instrumental in the development of hierarchical methods, especially in the framework of numerical taxonomy [15]. Hierarchical clustering methods have also been successfully applied to several other areas where there is not necessarily an underlying hierarchical structure.

2. Partitional Clustering [14, 13]

A partitional clustering method, unlike a hierarchical clustering method, obtains a single partition of the data instead of a clustering structure such as the dendrogram. Partitional methods have advantages in applications involving large data sets for which the construction of a dendrogram is computationally very expensive [13]. The use of a partitional clustering algorithm is associated with the constraint of specifying the choice of the desired number of output clusters. Partitional clustering techniques produce clusters by optimizing a criterion/objective function. This function is either defined *locally* on a certain subset of the clusters or *globally* on all the clusters. Criteria are highly dependent on problem parameters and they must be simple for computational purposes but should be complex enough to distinguish between different data structures. Let $C(n, P)$ denote the

number of clustering of n objects into P clusters. The order of the objects in each cluster and the order of the clusters are not important and empty clusters are not counted. If all the clusterings of $n - 1$ objects have been listed, then the clustering of n objects can be formed from this list by two ways:

1. The n^{th} object can be added as a singleton cluster to each member of the list with exactly $(P - 1)$ clusters.
2. The n^{th} object can be added to each cluster of any member of the list with exactly P clusters.

A partial difference equation [14] can be written for $C(n, P)$ as

$$C(n, P) = C(n - 1, P - 1) + PC(n - 1, P).$$

The boundary conditions for the above equation are

$$C(n, 1) = 1, C(n, n) = 1, C(n, P) = 0. \text{ if } P > n.$$

The solution of this difference equation requires that we know the values for $\{C(j, l)\}$ where $1 \leq j \leq n - 2$ and $1 \leq l \leq P$. Solutions are called Stirling numbers of the second kind [26]:

$$C(n, p) = \frac{1}{P!} \sum_{i=1}^P (-1)^{P-i} \binom{P}{i} (i)^n.$$

The above formula can be used to determine the number of distinct partitions. For example, there are just 966 distinct partitions of 8 objects into 3 clusters, but the number increases to being approximately equal to 11,259,666,000 if we were to partition 19 objects into 4 clusters. Hence, it can clearly be seen that it is not computationally feasible to list all possible partitions even for a small number of clusters. To avoid enumerating all the possible partitions, the criterion function is evaluated only on a small set of partitions. A criterion function differs depending on the approach adopted in identifying a small subset of partitions that represent an optimal partition. A commonly used approach is to optimize the criterion function using an iterative, hill climbing technique. The algorithms that are based on this approach are computationally less expensive but run the risk of converging to a local minimum of the criterion function. Another approach is based on dynamic

programming and relies on its ability to identify and reject a large number of partitions that are not of potential importance. There is no criterion function that can be considered as the “single best” for obtaining an optimal partition because clusters can be of arbitrary shapes and sizes in a multidimensional space [14].

2.1. Squared Error Clustering Criteria [11]. This is the most widely used criterion function for partitional clustering techniques. The objective of this function is to obtain a partition for a fixed number of clusters that minimizes the square-error. This strategy normally works well for isolated and compact clusters. Suppose a given dataset is partitioned into P clusters $\{C_1, C_2, \dots, C_P\}$, then the squared error for cluster C_P is the sum of the squared distances between each pattern in C_P and its cluster center.

$$e_P^2 = \sum_{j=1}^P \sum_{i=1}^{n_j} \|x_i^{(j)} - m_j\|^2$$

where x_i^j is the i^{th} pattern belonging to the j^{th} cluster and m_j is the centroid of the j^{th} cluster. This function is used as an intra cluster validity measure known as *total energy* in analyzing the different clustering algorithms used in this study. See chapter 4 for more details.

2.1.1. *K-Means Clustering Algorithm [11].* The K-means algorithm is the simplest and most widely used algorithm that employs a squared error criterion/objective function. Basically, the algorithm starts with a random initial partition and keeps reassigning the patterns to the clusters based on the similarity between the pattern and the cluster center. This is repeated until the convergence criterion is satisfied such as when there is no other reassignment of any pattern from one cluster to the other, or when the squared error no longer decreases significantly after a given number of iterations.

K-Means Clustering Algorithm
<ol style="list-style-type: none"> 1 Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids. 2 Assign each object to the group that has the closest centroid. 3 When all objects have been assigned, recalculate the positions of the K centroids using the current cluster memberships. 4 Repeat Steps 2 and 3 until the convergence criterion is met.

The K-means algorithm has a time complexity of $O(n)$ where n is the number of patterns. It suffers from a few drawbacks pertaining to its sensitivity to the selection of the initial random partition and its tendency to converge at a local minimum of the objective function value. Several variants of the K-Means algorithm exist. The difference between each being the approach adopted to address the drawbacks mentioned earlier. Some try to differ in their attempt to select a good initial partition so as to find the global minimum value and others differ in the sense that they try to permit splitting and merging of resulting clusters. Other variants of the K-means algorithm use a different objective function [20, 21].

2.2. Other Partitional Clustering Methods. There are several different types of divisive algorithms. Some use *Graph-theoretic* concepts for clustering, an example being constructing clusters by deleting the edges with the largest lengths from a minimal spanning tree of the data to generate clusters. Other algorithms such as the *density-based* partitioning techniques assume that the patterns are drawn from different distributions that are identified by certain parameters and involve obtaining a maximum likelihood estimate of the parameter vectors of the component densities. An important advantage of density-based methods is that they are capable of discovering

clusters of arbitrary shapes. *Nearest neighbor* clustering methods are based on the iterative procedure of assigning unlabeled patterns to the cluster of its nearest neighbor pattern whose distance is less than a given threshold. *Fuzzy clustering* is a technique that associates each pattern within a cluster with a membership function. Unlike traditional clustering techniques in which each object belongs to one and only one cluster (“hard clusters”), fuzzy clustering techniques do not present the output as disjoint clusters but instead each fuzzy cluster consists of ordered pairs (i, ω_i) where i denotes the i^{th} object or pattern and ω_i denotes the membership value. Higher membership values indicate a greater confidence of the object being in a given cluster. By setting appropriate threshold on the membership values, hard clusters could be easily obtained from fuzzy clusters. The design of efficient membership functions is the most important problem in fuzzy clustering. *Constraint-based clustering* methods have been recently developed to deal with constraints in certain real-world applications. Certain constraints such as those on individual objects and parameter constraints can be addressed through pre-processing or by specifying certain external cluster parameters. Other constraints such as those that specify bounds on aggregate functions on the number of objects within each cluster require a new clustering methodology. Constraint-based clustering is extremely useful in clustering 2D spatial data in the presence of *obstacles*. Euclidean or any such traditional proximity measures cannot be used in such a case. Instead, the shortest path between two objects needs to be determined that takes the obstacles into consideration. *Artificial Neural Networks* (ANNs) are also widely used for clustering. Some popular examples of ANN techniques used for clustering are learning vector quantization (LVQ) and self-organizing map (SOM) [27]. The architecture of ANNs is simple. Patterns or objects are presented at the input and are in turn associated with the output nodes. The weights between the input and the output nodes are progressively changed in each iteration until a specific termination criterion is reached. SOM is popular in vector quantization applications and it generates a two dimensional map of the multi-dimensional input data set that is easier to interpret. If the weights are not properly chosen, it is highly likely that ANN techniques would result in sub-optimal clusters. The convergence of such techniques is also dependent on pa-

parameters such as the learning rate of the ANN. *Evolutionary methods* are based on the notion of using evolutionary operators and a family of solutions to obtain an optimal global partition. The solutions are encoded as *chromosomes* and the most commonly used evolutionary operators are selection, recombination and mutation. The evolutionary operators transform one or more input chromosomes into one or more output chromosomes. Fitness functions are used to determine the survival rates of a chromosome in the next generation. *Genetic algorithms*(GA) are the most widely used evolutionary techniques for clustering [28]. GAs represent points in the search space in the form of binary strings. A recombination operator known as *crossover* that takes as its input a pair of chromosomes and outputs a new pair of chromosomes is used to explore the search space in GAs. The main advantage of GAs is that they perform a globalized search for finding optimal solutions but they suffer from their high sensitivity to control parameter selection hence, requiring one to tune the parameter values for every application. *Search-based* methods that employ either deterministic or stochastic search strategies can also be used for clustering. Most of the commonly used methods are deterministic. *Simulated Annealing*(SA) [29] is an example of a sequential stochastic search technique that uses a perturbation operator to avoid solutions that correspond to local optima in objective functions. *Tabu Search* [30] is another stochastic search technique that is designed to avoid local optima by systematically using certain constraints that would allow certain otherwise forbidden regions to be explored.

The clustering methods used in the current study are based on only the hierarchical and K-means clustering techniques because these are the two most widely used methods that have been extensively studied in various domains. As mentioned in the previous chapter, these methods have been successfully used in clustering of PET data [6, 3, 8].

CHAPTER 3

Cluster Validation

The various clustering algorithms that were mentioned in the previous section are *unsupervised* clustering methods that help us to obtain clusters. These cluster groups are not known *a priori*. Hence, the final cluster groupings of a data set requires some sort of evaluation in most applications. One would typically like to address questions like

- What are the optimal number of clusters in the data set?
- How appropriate is the clustering method for the data set at hand?
- Would it possible to have a better set of clusters for our data set?

To answer and address issues raised through such questions requires one to validate the clustering results. The procedure of evaluating the results of a clustering algorithm in a quantitative and objective manner is referred to as *cluster validation*.

1. Indices of Cluster Validity

An *index* of cluster validity is used to measure the quality and appropriateness of a clustering structure. Indices of cluster validity gives us an idea as to what extent the clustering structure provides true information about the data and to what extent the retrieved structure reflects the intrinsic character of the data. A clustering structure evaluated by an index of cluster validation could either refer to a hierarchy obtained from hierarchical clustering methods or a partition obtained

from partitioning clustering algorithms or could be applied to clusters that are defined by other means. Traditionally, there are three types of criteria used to investigate cluster validity [14, 38]. The first is based on *external criteria*. This involves matching and evaluating the results of a clustering algorithm to a pre-specified structure or *a priori* information that reflects our intuition about the clustering structure of the data set. The second approach is based on *internal criteria*. In this approach, the results of a clustering algorithm are evaluated in terms of quantities that involve the vectors of the data set themselves (e.g. proximity matrix). The third approach of clustering validity is based on *relative criteria*. Relative criteria helps us to compare two clustering structures and determine which is more stable and appropriate for the given data of the two. An important difference between a criterion and an index should be noted. A criterion refers to the strategy by which a clustering structure is to be validated whereas an index is a statistic in terms of which validity of the clusters is tested.

1.1. Indices Used for External Criteria. The basic idea in the external criteria approach is to test whether the points of the data set denoted by X are randomly structured or not. The Null Hypothesis, $H_0 =$ The data set X is randomly structured. To test this hypothesis we need to perform certain statistical tests which can be computationally complex.

1.1.1. *Monte Carlo Analysis* [32]. Monte Carlo analysis is a method for estimating parameters and probabilities with the help of computer simulation. This is particularly useful when the quantities are difficult to calculate directly. For example, suppose if sr_1 is the value of an index such as the scatter ratio [33]. Let sr_2, \dots, sr_l denote the $l - 1$ values obtained through Monte-Carlo sampling under some null hypothesis. It would be possible to determine if sr_1 is a desired value for the index by checking if sr_1 is significantly large or small. In case of the scatter ratio, low values are desired. Hence, it would be possible to check if sr_1 is significantly small by selecting an integer k such that $k/m = \alpha$, where α is the level of significance such as 0.05 or 0.01. If sr_1 is among the k smallest of the l values, it can be assumed that the index is significantly small and the null

hypothesis could be rejected.

Based on the external criteria we can either evaluate the resulting clustering structure C , by comparing it to an independent partition of the data P that is built according to our intuition about the clustering structure, or we can compare the proximity matrix D to the partition P .

1.1.2. *Comparison of C with partition P (for partitional structures)*. Consider $C = C_1 \dots C_p$, a clustering structure of a data set X and $P = P_1 \dots P_q$ is a defined partition of the data. We refer to a pair of points (x_i, x_j) from the data set using the following terms:

a : Number of pairs of objects that are in the same group in C and P . d : Number of pairs that are in different groups in C and P . b : Number of pairs of objects that are in the same group in C but in different groups in P . c : Number of pairs of objects that are in different groups in C but in the same group in P .

If g_1 denotes the number of pairs of objects that are in the same group in C and g_2 denotes the pairs of objects that are in the same group in P . Hence, $g_1 = a + b$ and $g_2 = a + c$. The total number of pairs of objects is denoted by $M = a + b + c + d = \frac{n(n-1)}{2}$ where n is the total number of points in the data set.

The following indices that measure the similarity between C and P can be defined based on the above terms.

- Rand Statistic [34]: $R = \frac{a+d}{M}$,
- Jaccard Coefficient: $J = \frac{a}{a+b+c}$.

The above two indices take values between 0 and 1, and are maximized when the number of groups m in the clustering structure is equal to the number of groups n in the partition set.

- Folkes and Mallows index [35]: $FM = \frac{a}{\sqrt{g_1 g_2}} = \sqrt{\frac{a}{a+b} \frac{a}{a+c}}$
- Huberts Γ statistic [23]: $\Gamma = \frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n X(i, j)Y(i, j)$, where X, Y are the matrices that are compared, each representing groups in C and P .

- Normalized Γ statistic:

$$\bar{\Gamma} = \left[\frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n (X(i, j) - \mu_X)(Y(i, j) - \mu_Y) \right] / \sigma_X \sigma_Y,$$

where $\mu_X, \mu_Y, \sigma_X, \sigma_Y$ are the respective means and variances of X, Y matrices.

High values of the above indices indicate greater similarity between two partitions or between a clustering structure and a defined partition.

1.1.3. *Comparison of proximity matrix D with partition P.* Partition P can be considered as a mapping $f : X \rightarrow \{1 \dots n\}$. Assuming matrix

$$Y : Y(i, j) = \{1, \text{ if } f(x_i) \neq f(x_j) \text{ and } 0, \text{ otherwise}\}, \quad i, j = 1 \dots n,$$

we can compute Γ or normalized Γ statistic using the proximity matrix D and the matrix Y as defined above. Based on the statistic index value, we may get an idea of the similarity between the two matrices.

1.2. Indices for Internal Criteria. Using this approach of cluster validity, one could evaluate the clustering result of an algorithm using only quantities and features inherent to the data set. Depending on the clustering structure used, there are two cases in which we can apply internal criteria of cluster validity : a) hierarchical structures scheme, and b) single clustering scheme. In case of validating a single clustering scheme, one could use Hubert's Γ statistic or Normalized Γ statistic in determining the degree of similarity between the proximity matrix and the clustering scheme.

1.3. Indices for validating Hierarchical structures using Internal Criteria. A matrix called cophenetic matrix, denoted by D_c , can be used to represent the dendrogram produced by a hierarchical algorithm. The $d_c(i, j)$ element of the cophenetic matrix represents the proximity level at which the two vectors x_i and x_j are found in the same cluster for the first time. The statistical

index to measure the degree of similarity between D_c and D (proximity matrix) matrices is called Cophenetic Correlation Coefficient (CPCC) and defined as:

$$CPCC = \frac{\frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n d(i, j) d_c(i, j) - \mu_d \mu_c}{\sqrt{[\frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n d^2(i, j) - \mu_d^2][\frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n d_c^2(i, j) - \mu_c^2]}}$$

μ_d and μ_c are the means of matrices D and D_c respectively, and are given by

$$\mu_d = \frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n d(i, j),$$

$$\mu_c = \frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n d_c(i, j).$$

$d(i, j)$, $d_c(i, j)$ are the (i, j) elements of D and D_c matrices respectively. If the value of the CPCC index is close to 0, it is an indication of a significant similarity between the two matrices.

1.4. Indices used for Relative Criteria. The basis of the external and internal criteria validation methods is statistical testing. Thus, a major drawback of techniques based on internal or external criteria is their high computational demands. Validation through relative criteria provides an alternative to this problem since they do not involve statistical tests. The fundamental idea of this approach is to choose the best clustering scheme of a set of defined schemes according to a pre-specified criterion such as with an optimal number of clusters.

The procedure of identifying the best clustering scheme is based on a validity index. Selecting a suitable performance index, q , the following steps are performed:

- Run the clustering algorithm for all values of p between a minimum p_{min} , say 3 and a maximum p_{max} . The minimum and maximum values are defined a-priori by the user.
- For each of the values of p , run the algorithm with different initial conditions and parameters.
- Plot the best values of the index q obtained for $p_{min} \leq p \leq p_{max}$.

Based on the plot it is possible to identify the best clustering scheme. If the relative criteria validity index does not exhibit an increasing or decreasing trend as n increases, the maximum or minimum

of the plot is selected based on the property of the index and user preference. On the other hand, for indices that increase or decrease as the number of clusters increase it would be appropriate to search for the values of p at which a *significant local change* in value of the index occurs. This change gives us an indication of the number of clusters underlying the data set. An absence of a such a change indicates the data set does not possess any clustering structure.

The following are some of the indices that can be classified under the relative criterion and that are used to choose the best clustering scheme.

- *Modified Hubert Γ statistic:* This index [23] is defined as

$$\Gamma = \frac{1}{M} \sum_{i=1}^{n-1} \sum_{j=i+1}^n D(i, j) Q(i, j)$$

where D is the proximity matrix and Q is a matrix of dimension $n \times n$ whose (i, j) element is equal to the distance between the representative points of the clusters that contain the objects x_i and x_j .

- *Dunn's Index:* This index [24] is defined by the following formula

$$DI_n = \min_{i=1, \dots, n} \left\{ \min_{j=i+1, \dots, n} \left(\frac{d(C_i, C_j)}{\max_{k=1, \dots, n} \text{diam}(C_k)} \right) \right\}$$

where $d(C_i, C_j)$ is the dissimilarity between two clusters C_i and C_j defined as $d(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y)$ and $\text{diam}(C)$ denotes the diameter of a cluster. It is a measure of dispersion of the clusters. The diameter of a cluster C is defined as

$$\text{diam}(C) = \max_{x, y \in C} d(x, y).$$

If a dataset contains clusters that are compact and well-separated, the distance between the clusters is expected to be large and the diameter of the clusters is expected to be small. From the definition of Dunn's index, it can be inferred that large values of the index indicate the presence of compact and well-separated clusters.

2. Other criteria

Berry and Linoff [31] proposed the following two criteria for clustering evaluation and selection of an optimal clustering scheme:

Compactness: This connotes that the members of each cluster should be as close to each other as possible. A common measure of compactness is the variance, which should be minimized.

Separation: This criteria suggests that the clusters themselves should be widely spaced. There are three common approaches measuring the distance between two different clusters:

- Single linkage: It measures the distance between the closest members of the clusters.
- Complete linkage: It measures the distance between the most distant members.
- Comparison of centroids: It measures the distance between the centers of the clusters.

The various indices and validation strategies described in this chapter provide a basic overview of those that were reviewed before choosing specific validation indices that were used in this study.

The validation indices used in this study are described in detail in the following chapter.

CHAPTER 4

Clustering Methods and Validation Indices Used in the Study

This chapter describes the specific distance measures, pre-processing operations and clustering algorithms that are used in this study and the specific reasons for choosing each of them. The later part of the section is devoted to the description of the validation indices used to evaluate the clusters obtained from the different clustering algorithms.

1. Distance Measures

The measurement of the dissimilarity between the multivariable vectors or integrals x and y is denoted by $d(x, y)$. For example a typical choice would be the usage of $d(x, y) = (\sum_{l=1}^n |(x_l - y_l)w_l|^p)^{1/p}$ a weighted Minkowski p -norm which satisfies the usual mathematical requirements for a norm, and where the choice $w_l = 1$ for all l would suppose that each feature x_l has equal significance. For $p = 2$ this is the weighted sum of squares, or Euclidean distance, and for $p = 1$ the weighted Manhattan norm. Different distance measures, to be used within the clustering methods were considered during the research stage. These were simple “L1” Manhattan distance, simple “L2” Euclidean distance, “*timeL1*” Manhattan distance with time duration weight and “*timeL2*” Euclidean distance with time duration weight. “*timeL1*” and “*timeL2*” were observed to be good distance measures for PET data [3]. Hence, it is imperative that for PET data, weighting is included in order to account for the difference in the SNR of the different time frames due to the increasing

time windows used in functional PET. We thus weight elements x_l with $w_l = \Delta t_l$, where Δt_l is the width of the l^{th} time window. Distance may be calculated elementwise, i.e. for all elements in a given cluster as compared to all elements in another cluster, or with respect to a *representative* element of a cluster, such as the cluster centroid. The hierarchical centroid linkage (HCL), [10], uses the centroid μ_I for group I as the representative group vector and the distance between clusters is given by the distance between the group vectors, $D_{IJ} = d(\mu_I, \mu_J)$. The distance of a single point to a single cluster is $d(\mu_I, x_j)$. *Zhou et al.* [6], uses Hierarchical Average Linkage (HAL) with an unweighted distance measure, where the average distance between clusters I and J , of sizes n_I and n_J , respectively, given by $D_{IJ} = \frac{1}{n_I n_J} \sum_{i \in I} \sum_{j \in J} d(x_i, y_j)$ is used to measure an average distance with respect to all pairs of elements, [10].

2. Histogram-based Thresholding

Clustering of PET data is prohibitive in time and memory. Clustering of this high-dimensional data can be made feasible by adopting the strategy outlined in *Guo et al.* [3]. The first step involves identifying the *active voxels*, those in which the strongest kinetic activity is occurring as indicated by the voxels with highest intensity, and second by *preclustering* these active voxels. The rationale of the first step is illustrated in Figure 6 that shows histograms for the density frequencies of multidimensional data at different time frames. It is observed that the final frame most clearly shows that there are differences in voxel activity over time. In the early frames, there is significant evidence of noise from the reconstruction process. Noise is typically characterized by negative values of density due to the small sampling interval. Automatic separation into active and inactive voxels using histogram-based thresholding for the final frame is an effective way of reducing noise and computational costs. The thresholding process relies on the identification of a separation between the two distributions using a mixture model.

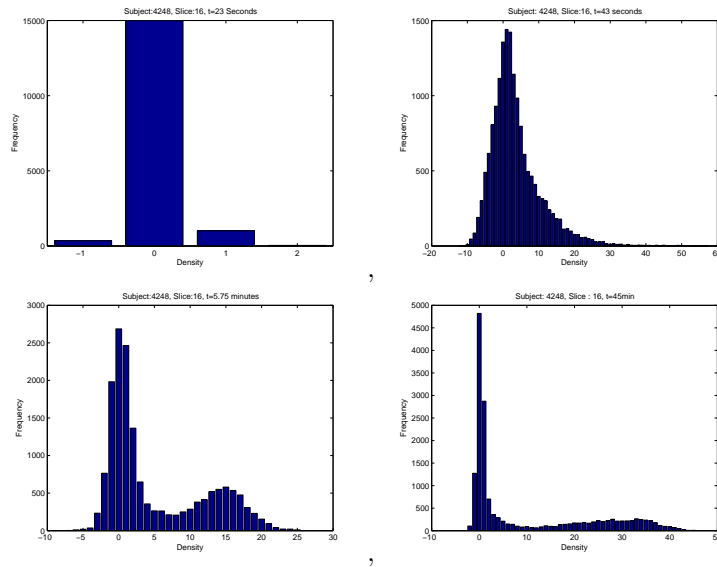


Figure 6. Histograms of densities summed over all three spatial dimensions for time frames with $t=23$ seconds, 43 seconds, 5.75 minutes and the last frame at time 45 min.

3. Preclustering

The initial thresholding into active and inactive voxels does not provide enough information to cluster the data. This is because clustering should include features over the entire TAC and not just the final frame. The clustering methodology used in this study is based on the strategy adopted by *Guo et al.* [3] that uses a fast approach for the initial clustering of the data followed by standard and more accurate clustering at the later stages. The procedure involves calculating a mean TAC from the set of TACs, \mathcal{N}_0 , containing voxels with the highest density. This mean TAC can then be used to find an initial cluster, \mathcal{G}_1 , of voxels with TACs near this mean. The search is performed over all voxels, i.e., not only those in \mathcal{N}_0 , and distances are measured with respect to the entire TAC and not only the last frame. Removing voxels in \mathcal{G}_1 from the histogram of the final frame, we can obtain a histogram which now has two peaks on two separated intervals, each representing a different cluster. We can repeat this process on each subinterval to find several preclusters for initialization of the cluster algorithm. Preclustering forms the basis of the modified hierarchical linkage algorithms HCL1, HCL2, HAL1 and HAL2 that are described in the following section.

4. Clustering Methods

The kinds of clustering methods that were considered in this study fall into the traditional broad classification of clustering methods as described in Chapter 2. Six different variants of hierarchical clustering methods and K-means partitional clustering method were compared and evaluated in this study. All of these methods are unsupervised clustering methods and it is not known *a-priori* where to cut the hierarchical tree or how many partitions are desired using K-means. For the purpose of our validation study, we have chosen to test the different algorithms by setting the number of clusters desired to be a number between 3 and 7. These numbers were set with the help of domain experts. The algorithms for each of the clustering methods used in this study are described under this section.

4.1. Classical Hierarchical Linkage Algorithms.

Hierarchical Average Linkage (HAL) Algorithm

- 1 Find the similarity or dissimilarity between every pair of voxels/pixels in the data set by calculating the distance between TACs or integrals.
- 2 Group the objects into a binary, hierarchical cluster tree by linking together pairs of voxels/pixels that are in close proximity using the *Average Linkage* function. Average linkage uses the average distance between all pairs of objects in cluster C_1 and cluster C_2 of sizes n_{C_1}, n_{C_2} respectively.

$$d(C_1, C_2) = \frac{1}{n_{C_1} n_{C_2}} \sum_{i=1}^{n_{C_1}} \sum_{j=1}^{n_{C_2}} d(x_{C_1(i)}, x_{C_2(j)})$$

- 3 Determine where to divide the hierarchical tree into clusters based on the number of cluster groups desired by the user.

Hierarchical Centroid Linkage (HCL) Algorithm

- 1 Find the similarity or dissimilarity between every pair of voxels/pixels in the data set by calculating the distance between TACs or integrals.
- 2 Group the objects into a binary, hierarchical cluster tree by linking together pairs of voxels/pixels that are in close proximity using the *Centroid Linkage* function. Centroid linkage uses the distance between the centroids of the two groups.

$$d(C_1, C_2) = d(\mu_{C_1}, \mu_{C_2})$$

- 3 Determine where to divide the hierarchical tree into clusters based on the number of cluster groups desired by the user.

4.2. Parameter-Dependent Algorithm [3]. The density of voxels q in the chosen frame is denoted by $f(q)$ and the frequency of the occurrence of $f(q)$ by $F(f(q))$. The average frequency over density interval $[a, b]$ is denoted by $F_{mean}[a, b]$. An integer labeling is associated with each voxel. This helps to not only identify the voxels under consideration, but also the cluster to which the voxel is assigned. The labelings are stored in a labeling array $B(q)$ with $B(q) = 0$ for inactive voxels and $B(q) = 999$, otherwise. The choice 999 is arbitrary. It is meant to represent a significant overestimate on the maximum number of clusters anticipated. The initial interval for preclustering, the current active interval $[a, b]$, is defined such that $a \leq f(q) \leq b$ for $B(q) = 999$. The basic parameter-dependent algorithm, as the name suggests, requires the specification of parameters, $\kappa, \beta, \gamma_{max}$ and γ_{nk} . These parameters are used to set the tolerances for extraction of the preclusters. The density corresponding to the maximum frequency on the current interval $[a, b]$ is defined as

$f_{max}[a, b] = \arg \max_{f(q) \in [a, b]} F(f(q))$. A set of voxels with density near f_{max} is defined by the set

$$\mathcal{N}_0 = \{q : (f(q) \in [a, b]) \quad \& \quad (\frac{|f(q) - f_{max}[a, b]|}{f_{max}[a, b]} \leq \kappa) \quad \& \quad (\frac{F(f(q))}{F_{mean}[a, b]} \geq \beta) \quad \& \quad B(q) = 999\}.$$

The first condition insures that only the voxels with density in the chosen interval are looked at on the histogram. The second condition, dependent on κ , seeks voxels with densities near the density with maximum frequency, measured relatively. Finally, the third condition, dependent on β seeks only voxels whose density has high frequency.

Associated with each entry q of \mathcal{N}_0 is a TAC x_q for voxel q . The next step would involve calculation of the average $TAC(\mathcal{N}_0)$ over all $q \in \mathcal{N}_0$. At the next step all voxels with TACs which are close to $TAC(\mathcal{N}_0) = \bar{\mathcal{N}}_0$, measured with respect to a tolerance given by γ_{max} , which is an estimation of the maximum relative radii of the preclusters are found. The initial cluster at stage nk of the preclustering process is then defined by the set

$$\mathcal{G}_{n_k} = \{q : \frac{d(x_q, \bar{\mathcal{N}}_0)}{d(0, \bar{\mathcal{N}}_0)} < \gamma_{max} \quad \& \quad B(q) = 999\}.$$

The estimate of \mathcal{G}_{n_k} is refined by calculating its average TAC, updating its center to this new average, and thus determining its new radius. This is done recursively until the center becomes stable and such that \mathcal{G}_{n_k} has a radius γ_{n_k} which makes it sufficiently dense with respect to the frequency of occurrences of its elements. Finally, the labeling of the voxels is updated according to $B(q) = nk$ for voxels $q \in \mathcal{G}_{n_k}$, which removes these voxels from further process during the preclustering phase. We also calculate new intervals for further preclustering, the intervals $[a, c]$ and $[c, b]$, where c is the average for $f(q) : q \in \mathcal{G}_{n_k}$.

4.2.1. *Parameter Selection [3]*. The final clustering is most sensitive, in terms of effectiveness and efficiency, to the parameter γ_{n_k} which determines the effective width and density of the G_{n_k} . Consider the case when all γ_{n_k} are taken large: the voxels are quickly clustered, the preclustering

produces fewer but larger clusters, and there is little remaining work for the final clustering. The limit of large γ_{n_k} corresponds to an initial cluster which is the whole brain but the limit as all $\gamma_{n_k} \rightarrow 0$, creates empty initial clusters. There is clearly a trade-off in accuracy and efficiency between smaller and larger choices for γ_{n_k} and their choice is purpose dependent. If the goal is individual voxel clustering, then they should be really small so that the clustering is as accurate as possible. But if the goal is to generate mean TACs for subsequent refinement at the tissue level, then γ_{n_k} can be chosen to be relatively large. In the latter case it has been found that an appropriate choice can significantly reduce the size of the data set for the final clustering, while maintaining good accuracy in the calculation of the tissue TACs.

**Hierarchical Centroid/Average Linkage with Preclustering
(HCL1 & HAL1) Algorithm**

1 Initialization Phase

- Filter out voxels with insignificant values using thresholding to identify active voxels in the last frame.
- Set the value of the labeling array $B(q) = 0$ for all inactive voxels and $B(q) = 999$ for active voxels.

2 Find the first active interval $[a, b]$ such that $0 \leq a \leq f(q) \leq b$ where $f(q)$ is the density q voxels/pixels in the given frame.

HCL1 & HAL1 Algorithm Contd...

- 3 For each active voxels interval,
 - Calculate $f_{max} = \arg \max_{f(q)} F(f(q))$, for $f(q) \in [a, b]$.
 $F(f(q))$ denotes the frequency of occurrence of the voxels/pixels.
 - Find the current initial cluster \mathcal{N}_0 .
 - If $|\mathcal{N}_0| < 2$ then the current interval is inactive.
 - Otherwise,
 - * Calculate the mean TAC for \mathcal{N}_0 , denoted by $\bar{\mathcal{N}}_0$.
 - * Find the n_k^{th} precluster \mathcal{G}_{n_k} corresponding to relative radius γ_{n_k} .
 - * If $|\mathcal{G}_{n_k}| < 2$ then current interval is inactive.
 - * Else
 - Set $B(q) = n_k$ for $q \in \mathcal{G}_{n_k}$.
 - Form new active intervals $[a, c], [c, b]$,
 $c = \text{average of } f(q) : q \in \mathcal{G}_{n_k}$.
 - Update cluster number $n_k = n_k + 1$.
- 4 Repeat step 3 until either no more active intervals are available or a maximum iteration number has been reached.
- 5 Perform Hierarchical (average/centroid) Linkage clustering on the reduced data set, i.e. \mathcal{G}_{n_k} and all voxels/pixels for which $B(q) = 999$.

4.3. Modified Parameter Dependent Algorithm [3]. This algorithm contains few additional steps compared to the previous algorithm. Before performing the last HL clustering step, a mean TAC \bar{G}_j is calculated for all preclusters G_j . The distance to the mean TAC is calculated for all active voxels with $B(q) = 999$. Each voxel is merged with a cluster such that the distance of the voxel to the mean TAC of the cluster is the same as the distance between the voxel and the mean TAC of the precluster. The updated clusters are finally joined by using HL clustering method until a given number of clusters are obtained.

Hierarchical Centroid/Average Linkage with Preclustering and Classification (HCL2 & HAL2) Algorithm

1-4 The first four steps of this algorithm are the same as those in the case of HCL1/HAL1 algorithm described earlier.

5 For all preclusters G_j calculate the mean TAC, \bar{G}_j , $j = 1, \dots, nk$.

6 For all voxels $q : B(q) = 999$ calculate $d(x_q, \bar{G}_j)$ $j = 1, \dots, nk$.

7 Merge voxel q with cluster G_{j^*} , such that $d(x_q, \bar{G}_{j^*}) = \min_j d(x_q, \bar{G}_j)$.

8 Join updated clusters by Hierarchical (average/centroid) Linkage clustering clustering until we reach the given number of clusters.

4.4. K-Means Algorithm [11, 36, 37]. The K-Means algorithm that is used in this study is based on a two-phase iterative procedure that is used to minimize the sum of point-to-centroid distances, summed over all k clusters. The first phase provides a fast but potentially approximate solution that is used as a starting point for the second phase.

K Means Algorithm

1 Phase I: Each iteration consists of the following

- Reassign points, all at once, to their nearest cluster centroid
- Recalculate cluster centroids for each of these clusters.

2 Phase II: Each iteration during this second phase consists of one pass through all the points.

- Reassign points individually by checking if the reassignments reduce the sum of distances between the points in each group
- Recompute cluster centroids after each reassignment

5. Clustering Validation Indices used in this Study

Clustering of PET data, in general, can be perceived as an unsupervised process since there are no predefined classes and no specific examples that would show what kind of important relationships within the data are of biological significance. The different clustering algorithms that have been mentioned in the previous sections are each based on certain intrinsic assumptions to define cluster groups from the data set. Hence, it is of known consequence that each of them would behave in a different way based on their input parameters and the features of the data set. A related problem stemming from the unsupervised aspect of clustering is the problem of determining the optimal number of clusters that fit the data. For a high-dimensional data set such as PET data, it would be difficult to visually verify the validity of clustering algorithms. In the context of PET data, it thus makes a lot of sense to evaluate the results of clustering algorithms with the help of cluster validity measures. The three approaches [14, 38] usually adopted to investigate cluster validity have been described in Section 3 of this report. The use of the *external* and *internal* indices

approach is not too appealing due to their high computational cost and because their basic aim is to measure the degree to which a data set confirms an a-priori specified scheme which is not of relevance in the context of this study. Relative indices such as the *silhouette* measure[15] and *modified Dunn's Indices*[39] are used in this study. Certain fundamental measures are also used, for validation purposes, that quantify the within (intra) and between (inter) cluster characteristics.

5.1. Intra-Cluster measures.

Average Distance from Mean At the element level, the average distance to the mean is $d_i^j = d(x_i, \mu_j)$, where $\mu_j = \frac{1}{n_j} \sum_m x_m^j$, the distance of TAC x_i^j from the mean of cluster j . Here without loss of generality, the sum is taken only over the elements x_i^j which are members of cluster j . At the cluster level, the measure is computed as $d^j = \frac{1}{n_j} \sum_i d_i^j$. For all the clusters (global level), the average distance to the mean is denoted by

$$d = \frac{1}{p} \sum_j d^j. \quad (4.1)$$

This is one of the measures used to test for the compactness of each cluster. Generally, a low average distance to the mean value is a desirable property for a cluster.

Maximum distance from Mean This measure gives us an idea about the radius of a given cluster grouping. At the cluster level, the maximum distance from the mean is represented as

$$r^j = \max_{1 \leq i \leq n_j} d(x_i, \mu_j)$$

The global value of this intra-cluster measure is computed as

$$r = \max_{1 \leq j \leq p} r^j. \quad (4.2)$$

A low value of this measure suggests that the clusters are compact.

Maximum Diameter The maximum diameter for a given cluster j is given by

$$\Delta(X_j) = \max_{x, y \in X_j} d(x, y). \quad (4.3)$$

This intra cluster measure gives an indication of how wide a given cluster is. This measure is a part of the generalized Dunn's Index [24] mentioned in Chapter 3. A low value of this measure is indicative of a compact cluster. As is the case with most intra cluster measures, this measure is sensitive to noisy points. But, this seems to be mitigated for its use in the modified Dunn's indices.

Average Spread This measure represents the average distance of elements within a cluster to all other elements of the same cluster. This measure is also computed at three different levels. At the element level, $a_i^j = \frac{1}{n_j} \sum_m d(x_i^j, x_m^j)$.

At the cluster level, $a^j = \frac{1}{n_j} \sum_i a_i^j$. At the global level,

$$a = \frac{1}{p} \sum_j a^j. \quad (4.4)$$

The average spread is a measure of homogeneity of a cluster. The average spread values should be low for compact cluster groupings.

Total Energy This measure is defined as the sum of squares of the distances to the mean for all cluster points. It is defined as

$$e_i^j = d^2(x_i, \mu_j). \quad (4.5)$$

Energy forms the basis of the objective function in the case of the K-means function as was described in Section 2. The energy values are summed up at the cluster and global level. This measure is expected to be the least for the results obtained from the K-Means algorithm since it selects cluster groupings with the least energy value.

5.2. Inter-Cluster Measures.

Separation Separation represents the average distance of element i in cluster j to elements of cluster $k \neq j$. It can be represented as $b_i^{jk} = \frac{1}{n_k} \sum_m d(x_i^j, x_m^k)$. At the cluster level, the separation is calculated as $b^j = \frac{1}{n_j} \sum_i b_i^{jk}$. The overall average separation for a group of

clusters is computed as

$$b = \frac{1}{p} \sum_j b^j. \quad (4.6)$$

Minimum Separation The minimum separation is found by calculating the minimum of the separation values at the element level as $m_i^j = \min_{k \neq j} b_i^{jk}$. At the cluster level, the minimum separation $m^j = \min_i m_i^j$. The minimum separation at the global level is computed as

$$m = \min_j m^j. \quad (4.7)$$

This measure gives us an understanding as to how close are two neighboring clusters. The minimum separation should be high for well separated clusters.

Average Split Split is the closest point in cluster k to point i in cluster j . The average split is computed by computing the split for each element. It is denoted as $s_i^{jk} = \min_{x_i^j \in X_j, x_m^k \in X_k} d(x_i^j, x_m^k)$. At the cluster level, the mean of the split values are computed as $s^j = \frac{1}{n_j} \sum_i s_i^j$ and at the global level, the average split is found by the following equation

$$s = \frac{1}{p} \sum_j s^j. \quad (4.8)$$

The average split is generally high for well separated clusters.

5.3. Inter-Intra Cluster Measures.

Silhouette This measure is proposed by Kaufman et.al. [15] and is a popular method used to evaluate clusters. For each object i in a cluster j , the index is defined as $c(i) \in [-1, 1]$. This index measures the *standardized* difference between $b(i)$ and $a(i)$, where $a(i)$ is the average dissimilarity of object i to all other objects in its own cluster and $b(i)$ is the average dissimilarity of object i to all other objects in its nearest cluster.

$$c(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}. \quad (4.9)$$

When $c(i)$ is close to 1, it means that object i is nearer to its own cluster than a neighboring cluster and is assumed to be a *well-classified object*. Conversely, when $c(i)$ is close to the value

-1 , it means that the object is nearer to its neighboring cluster as compared to its own cluster. This gives us reason to assume that the object may be *mis-classified*. When $c(i)$ is close to 0, it is not clear whether the object should have been assigned to its current cluster or to its neighboring cluster.

The silhouette measure can be averaged over the entire set of clusters i.e., at the global level, to provide us with the *average silhouette width*. This value can be analyzed to help in determining the number of cluster groups. The maximum value corresponds to a suitable number of cluster groups. An average silhouette width greater than 0.5 indicates a good classification and an average silhouette width less than 0.2 indicates a lack of substantial cluster structure [15].

Bezdek et.al. [39] propose several new indices of cluster validity. Some of these were based on extensions of the Dunn's index [24] that was discussed in Chapter 3. The results presented in the paper suggest the use of some new measures of cluster validity which can be applied in the context of this study. In the discussion of the results presented in their paper, the authors conclude that intraset distances should use all the data points.

As was mentioned in the previous chapter, the general form of the Dunn's Index is given by

$$DI_n = \min_{1 \leq i \leq n} \left\{ \min_{1 \leq j \neq i \leq n} \left\{ \frac{\delta(X_i, X_j)}{\max_{1 \leq k \leq n} \{\Delta(X_k)\}} \right\} \right\} \quad (4.10)$$

It is observed that the numerator and the denominator of v_{GD} could be sensitive to changes in the cluster structure. This is normally the case with the denominator, being an intra-cluster measure called maximum diameter that has already discussed earlier. This measure could be greatly influenced by a few noisy points. But appropriate definitions of $\delta(X_i, X_j)$ could lead to suitable validity indices.

The three new measures of inter-cluster quality based on the Dunn's Index are given below.

Modified Dunn's Index (Average Linkage) This measure is based on the set distance func-

tion used in the Average Linkage(AL) clustering algorithms [14].

$$\delta_{AL}(X_j, X_k) = \frac{1}{n_j n_k} \sum_{x \in X_j, y \in X_k} d(x, y). \quad (4.11)$$

The $\delta_{AL}(X_j, X_k)$ set distance measure is substituted in Equation 4.10 to get a modified Dunn's Index corresponding to the Average Linkage distance.

Modified Dunn's Index (Complete Linkage) As the name suggests, this measure is based on the set distance function used in the Complete Linkage(CL) clustering algorithms.

$$\delta_{CL}(X_j, X_k) = \max_{x \in X_j, y \in X_k} \{d(x, y)\}. \quad (4.12)$$

It has been observed that when this measure is used in the Dunn's Index, it may be affected by the noisy points in the dataset because it does not use all the points in X_j and X_k . However, it has been asserted [14] that complete linkage produces quite useful hierarchies in most applications and that it usually produces clusters that are tight and hyperspherical.

Modified Dunn's Index (Combined Average) This measure combines the averaging concept of δ_{AL} with the idea of computing the distance between the TACs within a cluster to the mean TAC of a different cluster. The set distance function is represented as

$$\delta_{CAL}(X_j, X_k) = \frac{1}{n_j + n_k} \left\{ \sum_{x \in X_j} d(x, \mu_k) + \sum_{y \in X_k} d(y, \mu_j) \right\} \quad (4.13)$$

The above inter-intra cluster measures are useful in finding an optimal number of clusters. A global maximum value of these indices for a set of different number of cluster groupings help in determining the optimal number of clusters. It is also deduced that the intercluster separation measures are more important in assessing cluster reliability than the actual individual cluster characteristics.

6. Statistical Testing

We performed simple statistical tests using one-way analysis of variance(ANOVA). We hypothesized that the *p-value* obtained by performing such a test would be high for comparable

clusters obtained from the parameter dependent algorithms and traditional hierarchical methods. We did not observe any particularly high p-values between clusters of large size between the different methods. In the case of certain small-sized clusters, a high correlation was observed between different hierarchical methods. A thorough multivariate statistical framework to validate different clusters obtained using different methods for dynamic PET data needs to be developed in the future. Representative results are given in *Appendix I* for a slice of a subject for integral and TAC data.

CHAPTER 5

Results

1. Clinical Data

PET data from 5 healthy subjects who participated in a previous study were used to test the proposed technique. Subjects in this study provided their informed consent, and were studied under guidelines approved by Human-Subjects committees at Banner Good Samaritan Regional Medical Center (Phoenix, AZ) and the Mayo Clinic (Scottsdale, AZ.) PET data was collected using a 951/31 ECAT scanner (Siemens, Knoxville, TN). For each subject, an initial 20 min transmission scan was acquired for attenuation correction. An intravenous bolus of 10 mCi FDG was administered under resting condition. A filtered back projection algorithm with the Hanning filter of 0.40 cycles per voxel was used to reconstruct PET images. The scanner produced 31 transaxial slices of 128×128 voxels, each voxel of size $0.18776 \times 0.18776 \text{ cm}^2$, with a center-to-center slice separation of 3.375 mm and a 10.8 cm axial field of view. The final reconstructed PET images had an in-plane resolution of 9.5 mm full-width at half maximum (FWHM) in the center of the field of view and an axial resolution of 5.0-7.1 mm FWHM. The scanning time durations, given in minutes, for the reconstructed frames are 0.2, 8×0.0333 , 2×0.1667 , 0.2, 0.5, 2×1 , 2×1.5 , 3.5, 2×5 , 10 and 30. The results comparing the clustering methods for 2 subjects using integrals and TACs are presented in the figures that follow in this chapter.

Figure 7 indicates the mean TACs for the clusters obtained using the different methods. A pairwise comparison between them indicates high similarity between different methods. It is also

observed that the order of the clusters that need to be compared may vary depending upon the method. For example, it is observed from Figure 7 that the order of clusters produced using K-means is different from that of the hierarchical methods. Hence, it is important to correctly order the groups prior to any comparisons across clustering methods.

2. Mean Time Activity Curves

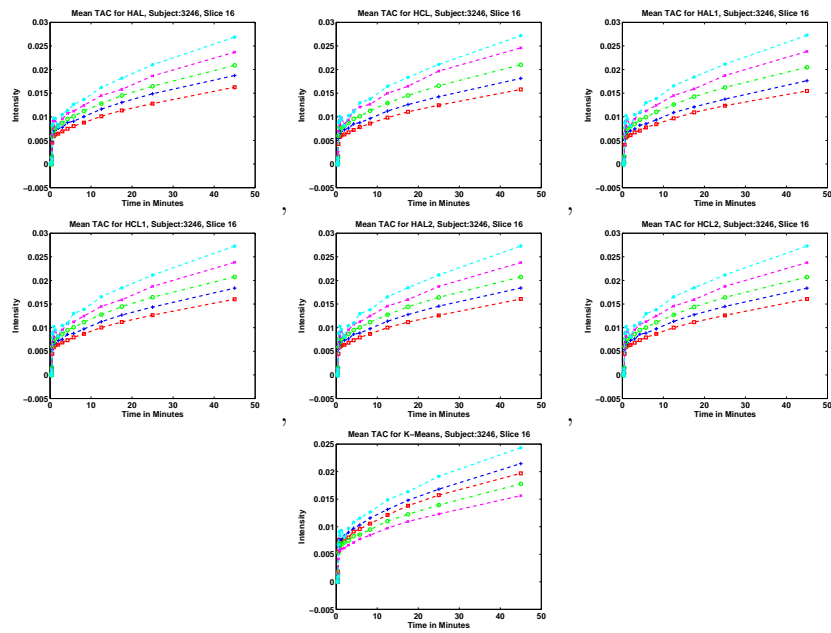


Figure 7. Mean TACs, All 7 Algorithms for subject 3246, slice=16, Number of Clusters=5

3. Representative Slice Results Obtained using Integrals and TACs

The following results indicate the plots derived from the quantitative results obtained for each cluster measure that was used in this study. Results obtained from each cluster measure using Integral and TAC data are presented on the same page for comparison purposes. It should be noted in the following plots that the same scale is used on the yaxis (different for each measure) in order to allow comparison between slices and subjects. Where the given number of lines cannot be distinguished in the figures, the results are such that results are overlaid on one another. This is clear from the numeric values presented in all cases in Appendix II.

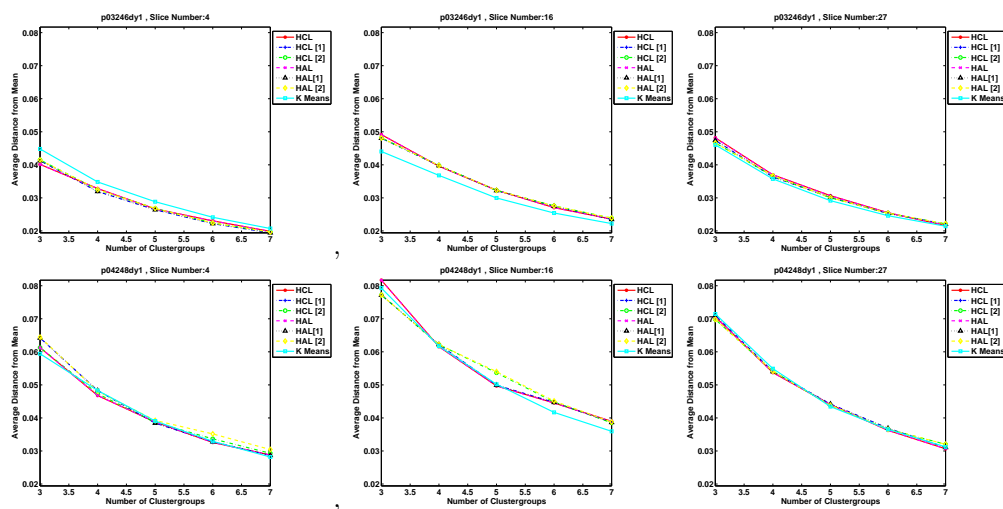


Figure 8. Average Distance to the mean Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

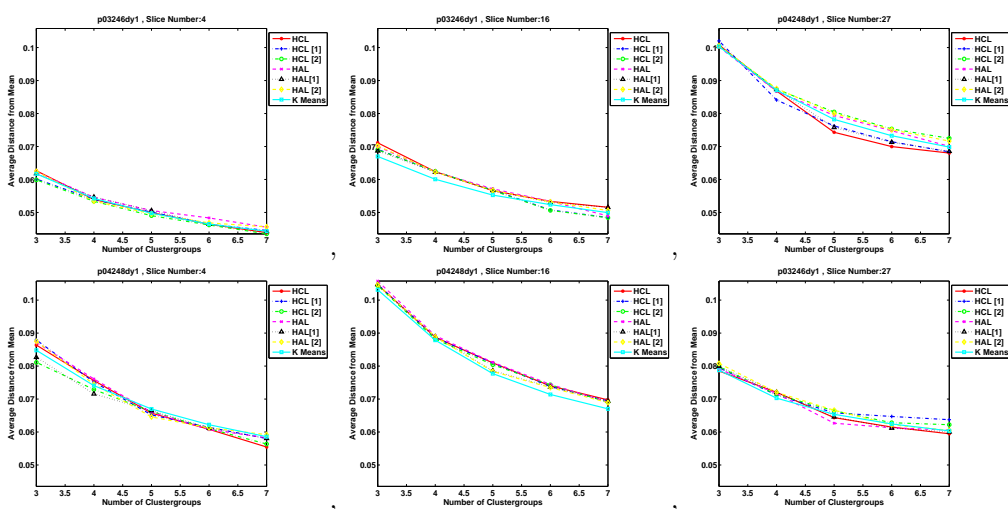


Figure 9. Average Distance to the mean Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

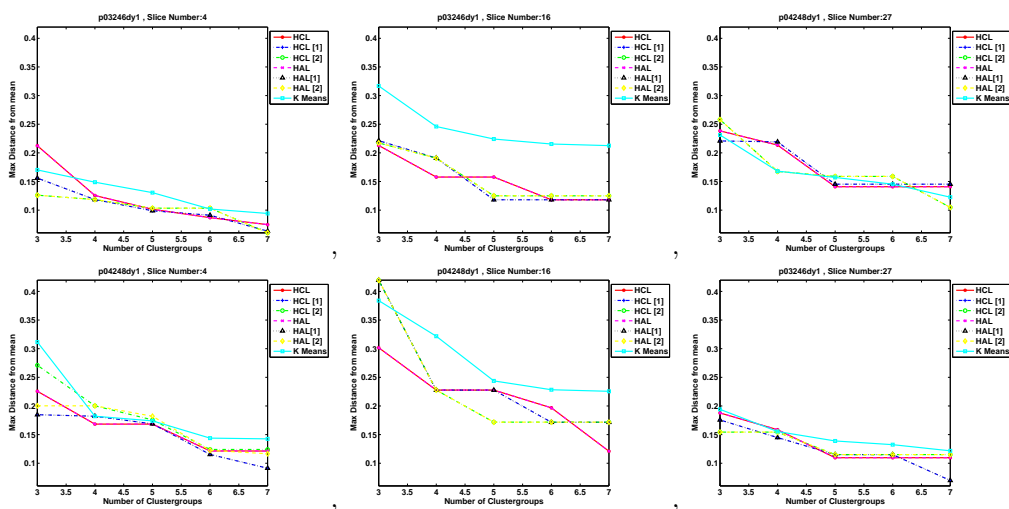


Figure 10. Maximum Distance to the Mean Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

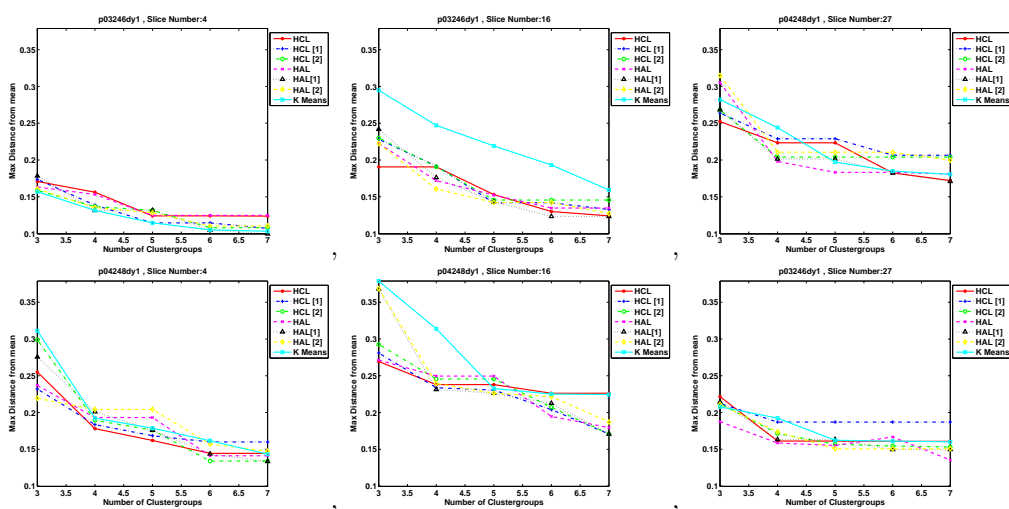


Figure 11. Maximum Distance to the Mean Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

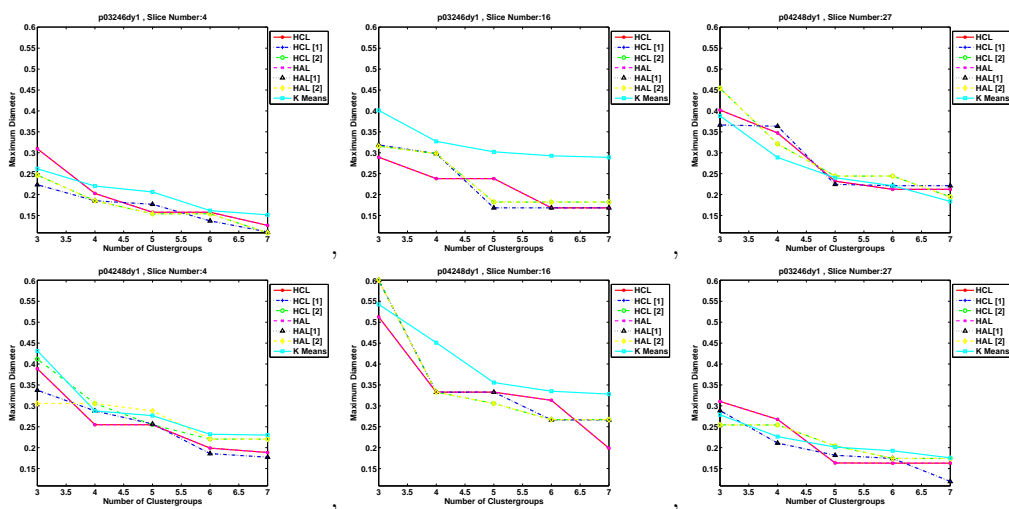


Figure 12. Maximum Diameter Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

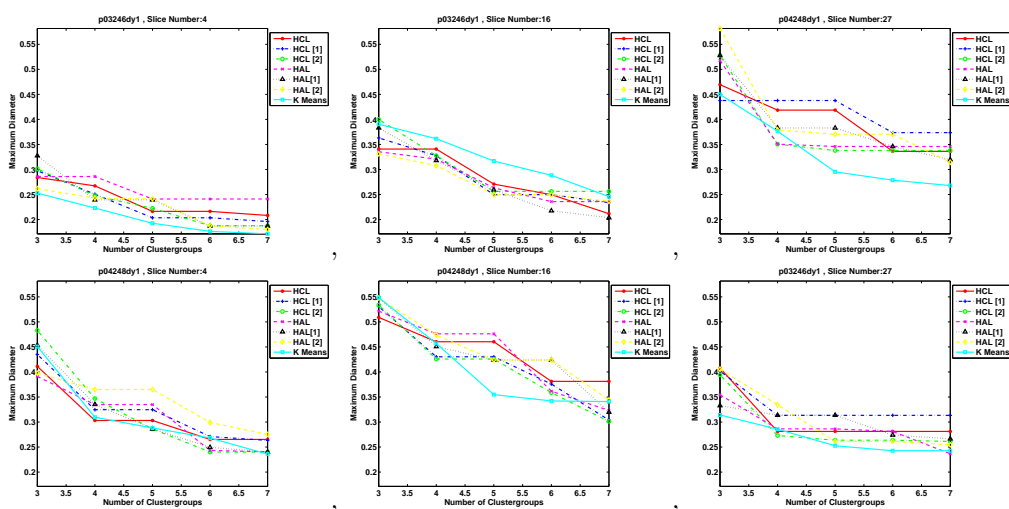


Figure 13. Maximum Diameter Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

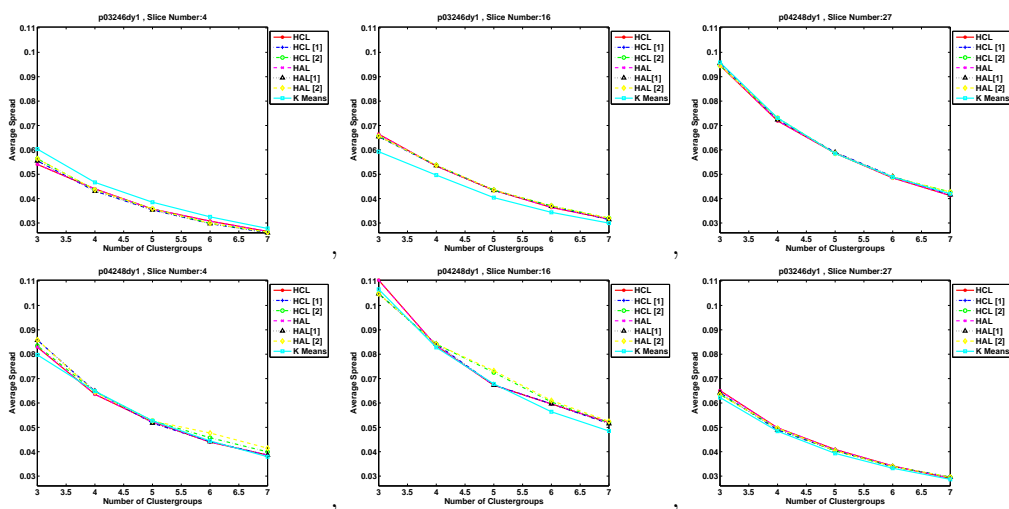


Figure 14. Average Spread Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

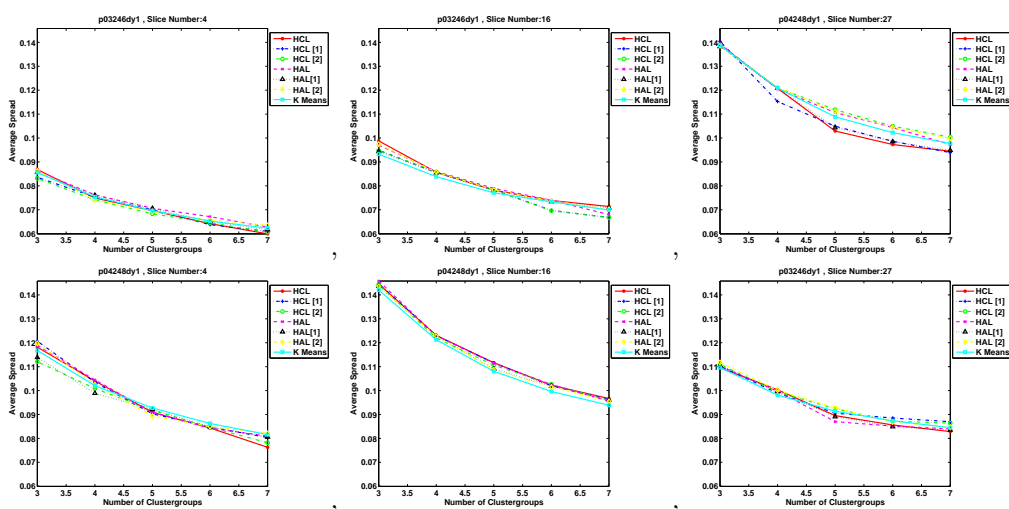


Figure 15. Average Spread Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

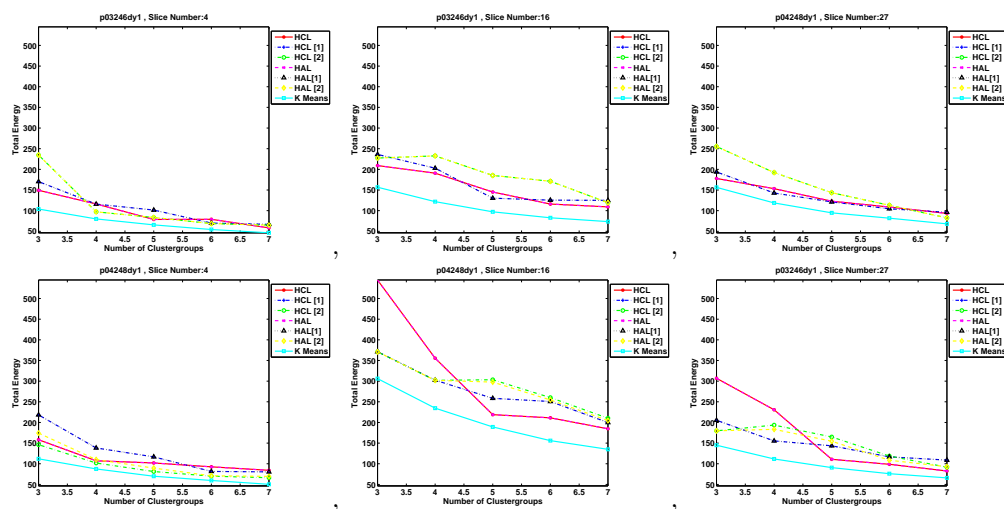


Figure 16. Total Energy Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

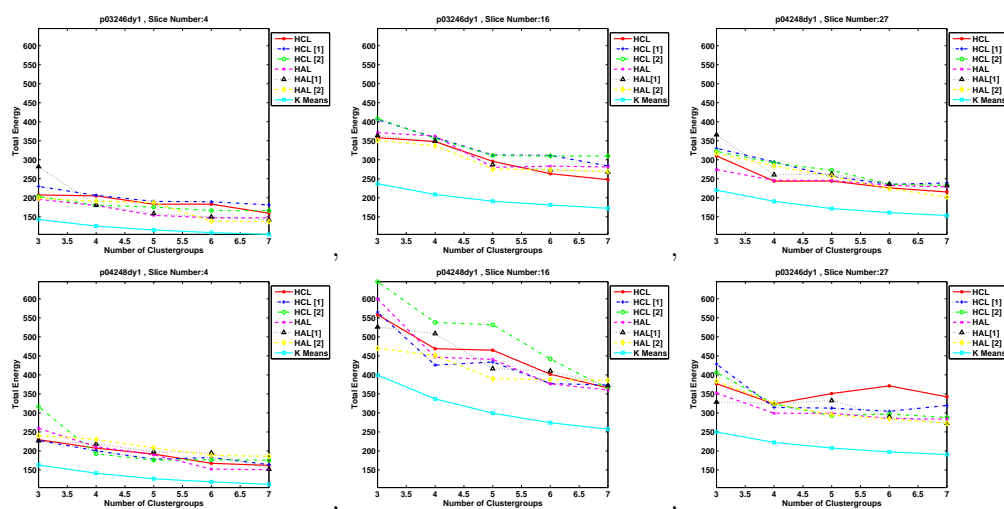


Figure 17. Total Energy Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

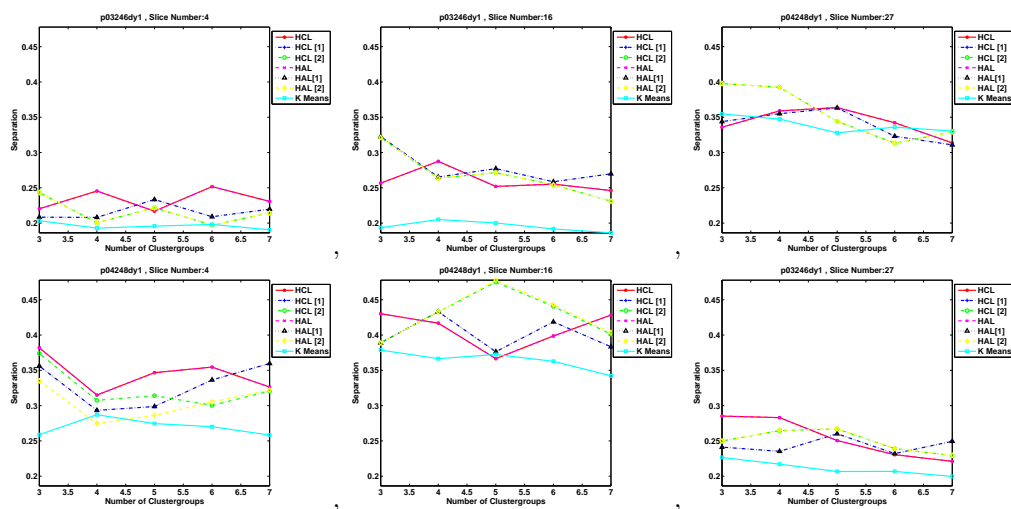


Figure 18. Separation Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

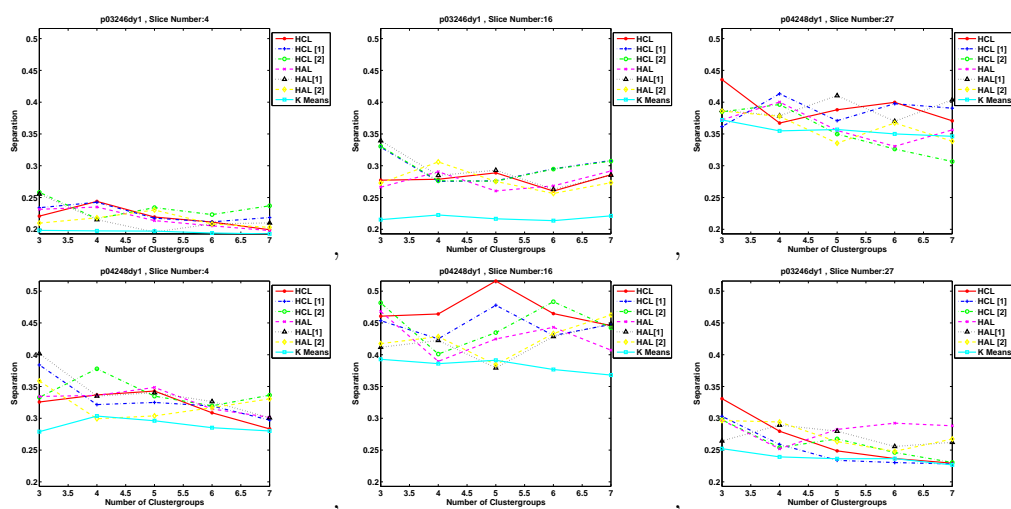


Figure 19. Separation Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

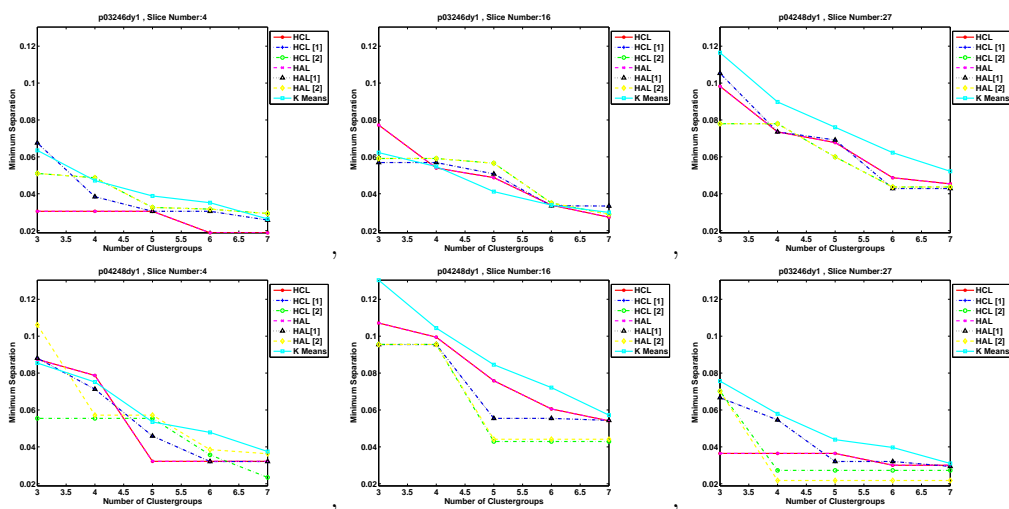


Figure 20. Minimum Separation Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

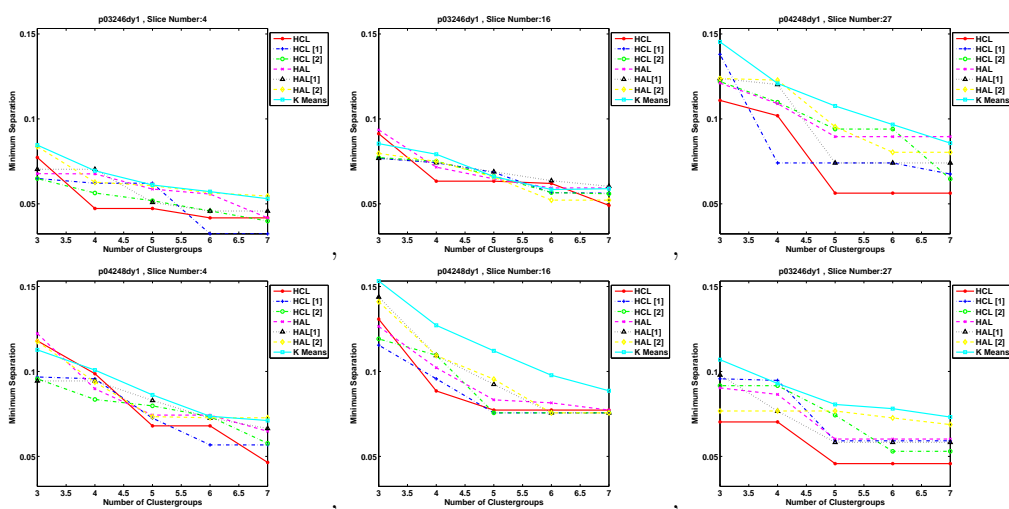


Figure 21. Minimum Separation Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

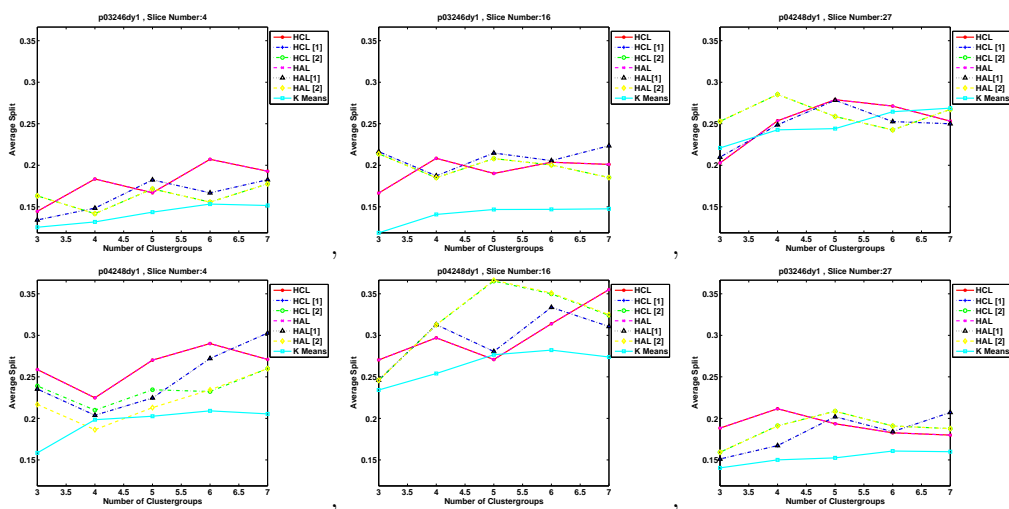


Figure 22. Average Split Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

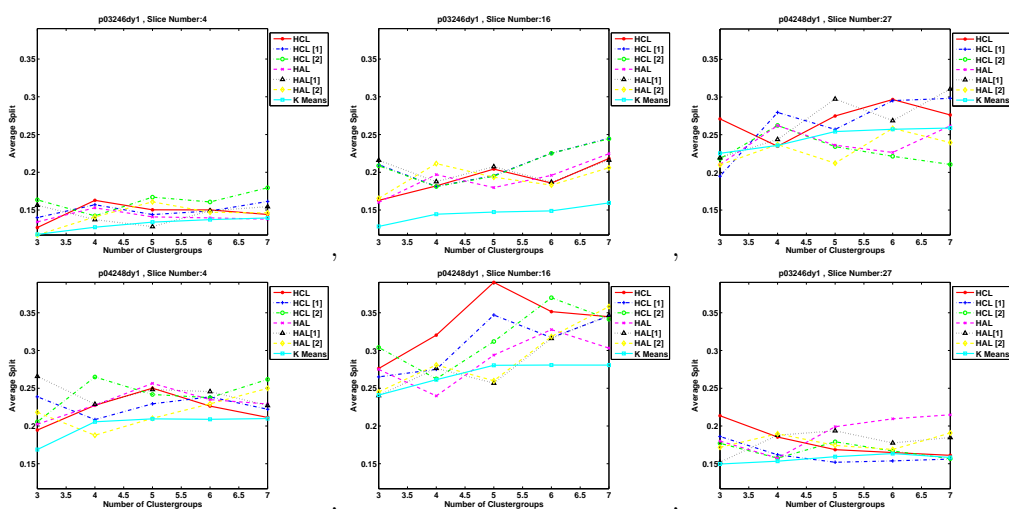


Figure 23. Average Split Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

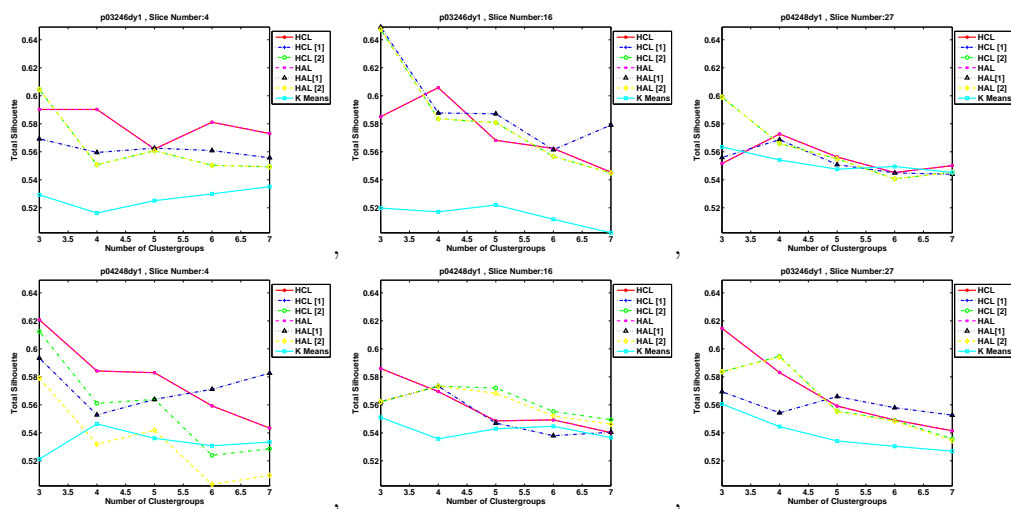


Figure 24. Silhouette Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

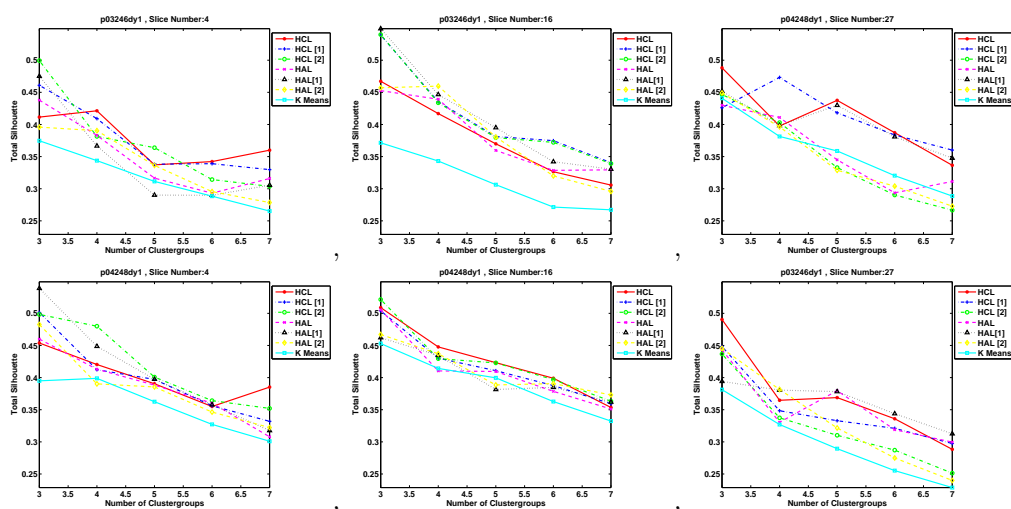


Figure 25. Silhouette Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

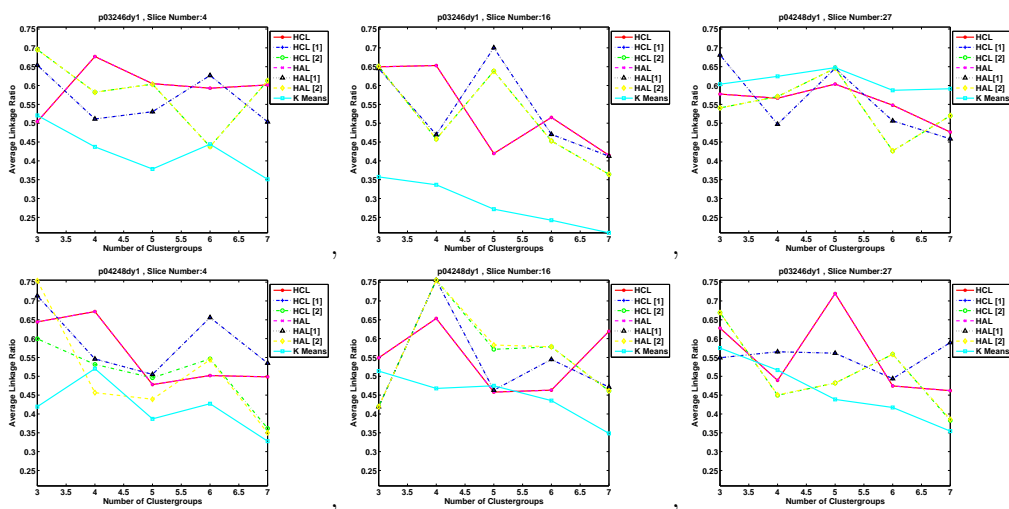


Figure 26. Average Linkage Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

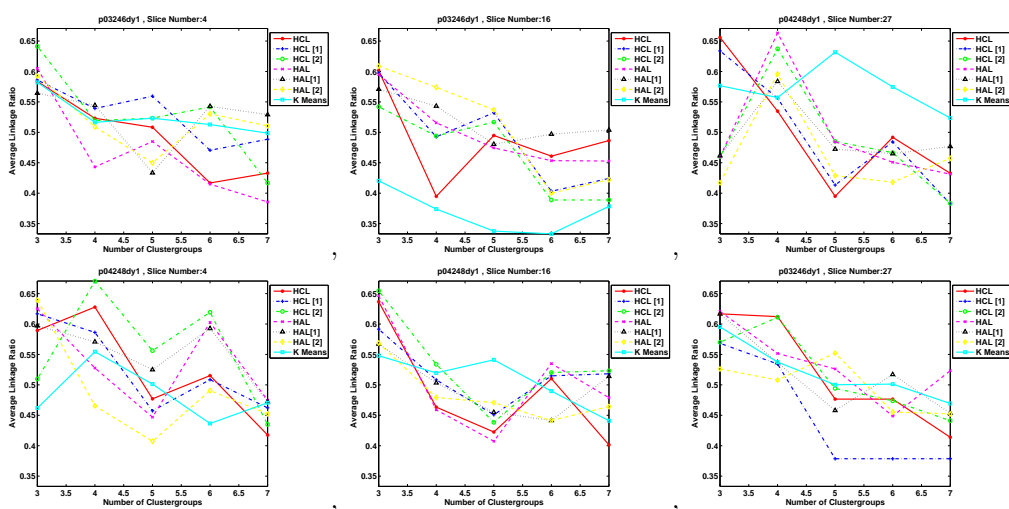


Figure 27. Average Linkage Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

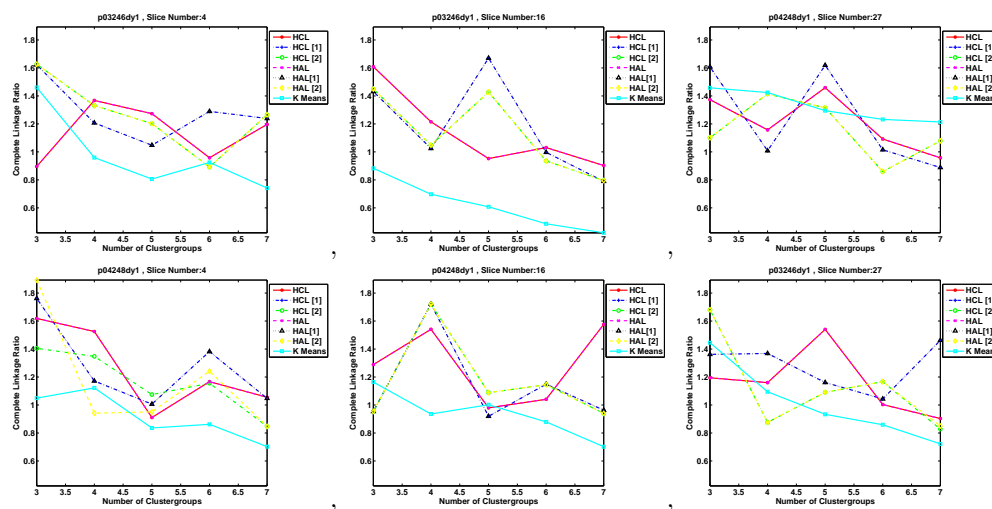


Figure 28. Complete Linkage Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

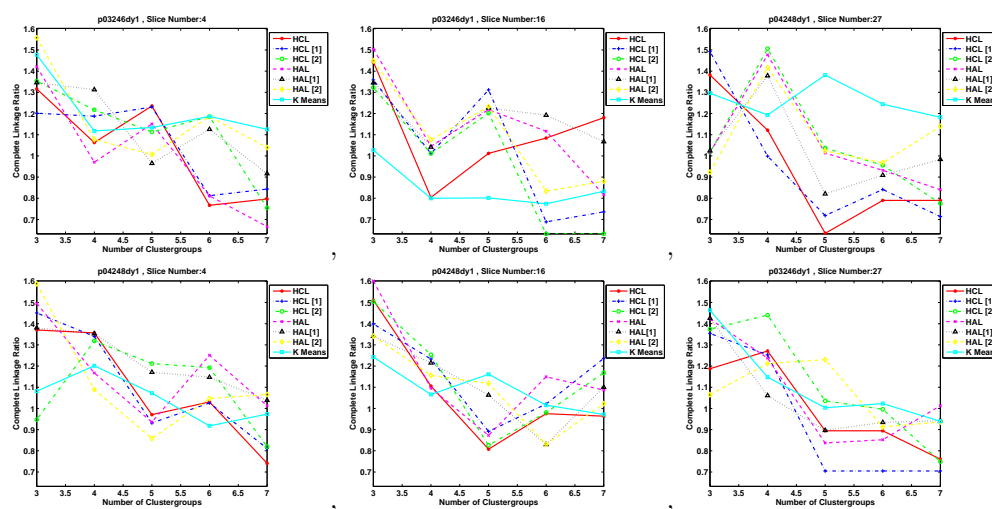


Figure 29. Complete Linkage Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248(TACs)

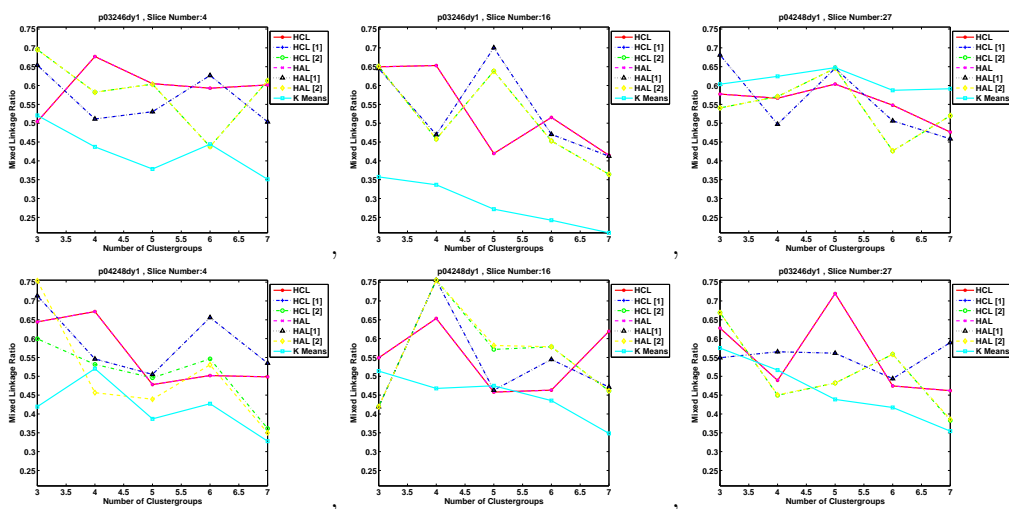


Figure 30. Combined Average Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (Integrals)

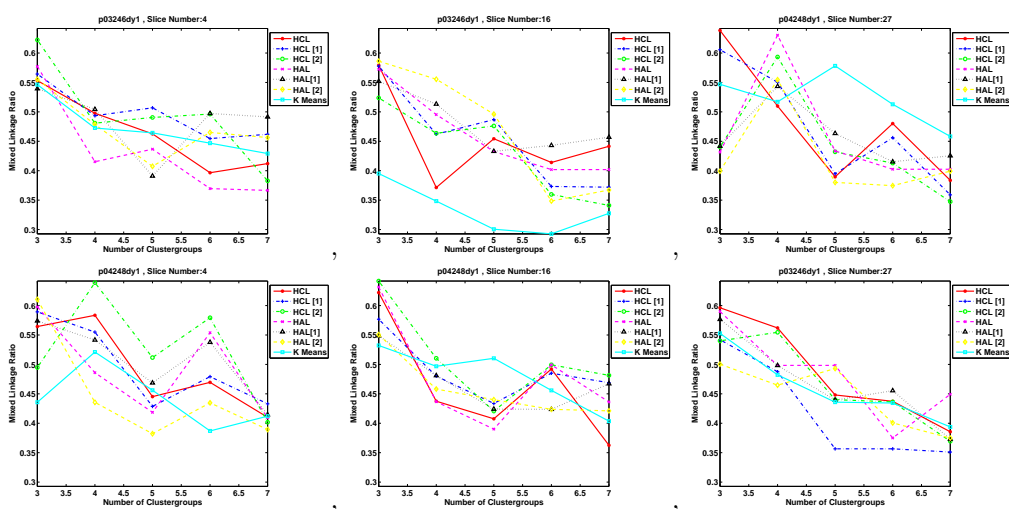


Figure 31. Combined Average Ratio Comparison Plots for slices 4, 16 and 27 of subjects #3246 and #4248 (TACs)

CHAPTER 6

Results-Observations and Conclusions

1. Data: Integrals

Measure	Observations
Avg. Distance from Mean (4.1)	All methods produce clusters with very similar values of this measure indicating that the average distance to mean for all clusters obtained from different methods are almost the same in most cases. This measure was observed to reduce as the number of clusters was increased.
Average Spread (4.4)	Again, little difference were observed between the average spread values among the different methods. This indicates that all the methods produce clusters that are <i>homogeneous</i> to a similar degree. This measure too was observed to steadily decrease as the number of clusters was increased from 3 to 7.
Max. Distance to the Mean (4.2)	This measure gives us an indication of the outliers and marginal data that were classified to a given cluster. No significant differences were observed in this measure amongst different methods indicating the clusters represented well classified data. This measure either remains constant or decreases as the number of clusters were increased.

Measure	Observations
Maximum Diameter (4.3)	In most of the cases, K-Means produced clusters with relatively greater diameter compared to other methods indicating that the clusters obtained from the K-Means method are slightly less compact as compared to other methods. No significant differences were observed between the hierarchical methods. As expected, this measure too was observed to either remain the same or decrease with an increase in the number of clusters
Separation (4.6)	This measure indicates the average distance between clusters obtained from different methods. The separation values obtained from K-Means algorithm were lower than those obtained from hierarchical methods. This is because the clusters obtained through the K-means methods are less compact compared to those obtained from Hierarchical methods.
Minimum Separation (4.7)	These values were observed to either remain the same or decrease as the number of clusters was increased. K-means was observed to return clusters with higher minimum separation values compared to the hierarchical methods. The low values of this measure indicates that the clusters generated by most methods are not very well separated.
Average Split (4.8)	In most instances, the clusters obtained from K-means methods had relatively lower average split values compared to those obtained from the Hierarchical methods. The difference between the average split values was observed to be not significantly large.
Total Energy (4.5)	As expected, K-means was observed to produce clusters with the lowest energy values in all instances. As the number of cluster groups were increased, the total energy values consistently decreased in the case of all the different clustering methods.

Measure	Observations
Avg. Silhouette Width (4.9)	<p>This measure, for all the methods on integral data, was observed to consistently range between 0.5 and 0.65. These high silhouette values indicate that the clusters contain well classified data. Maximum values obtained using this measure were analyzed to obtain clues regarding optimal number of clusters. Different methods were observed to give different results. 3 and 4 were most commonly observed to contain the peaks in most cases. The optimal number of clusters varied depending upon the slice selected. In the case of slice 4, 3 clusters was found to be the optimal value in the case of most methods. 4 clusters was found to be the consensus value in the case of slice 16 and in case of slice 27, the results varied between 4 and 5.</p>
Average Linkage Ratio (4.11)	<p>Significant differences in the average linkage ratio values were observed between different clustering methods. The maximum values across all the clusters were analyzed to obtain optimal number of clusters. Varying results were obtained for each individual slice and each subject. The overall consensus was observed to be 4 clusters for slice 4, and 5 clusters for slices 16 and 27.</p>

Measure	Observations
Complete Linkage Ratio (4.12)	<p>Using this measure, the optimal number of clusters was observed to range between 3 and 5. This measure does not use all the data points and hence suffers from the drawback of being influenced by noisy points. Usually, the level of confidence associated with this measure is lower compared to the other intra-inter measures [39].</p>
Combined Avg. Linkage Ratio (4.13)	<p>While using the integral data, this measure provides almost the same quantitative result as the average linkage ratio. This is an interesting observation and from the definition of the two measures, (4.11) and (4.13), it can be inferred that when we use integral data to obtain clusters, the clusters usually have the following property:</p> $\left(\frac{1}{n_j} + \frac{1}{n_k}\right) \sum_{x \in X_j, y \in X_k} d(x, y) \simeq \left\{ \sum_{x \in X_j} d(x, \mu_k) + \sum_{y \in X_k} d(y, \mu_j) \right\}$

2. Data: Time Activity Curves

Measure	Observations
Avg. Distance from Mean (4.1)	All methods produce clusters with very similar values of this measure indicating that the average distance to mean for all clusters obtained from different methods are almost the same in most cases. This measure was observed to reduce as the number of clusters was increased.
Average Spread (4.4)	In the case of this measure too, very few dissimilarities was observed between the average spread values among the different methods. This indicates that all the methods produce clusters that are <i>homogeneous</i> to a similar degree. This measure too was observed to steadily decrease as the number of clusters was increased from 3 to 7.
Max. Distance to the Mean (4.2)	This measure is useful in assessing the outliers and marginal data that were classified to a given cluster. This measure either remains constant or decreases as the number of clusters was increased.
Maximum Diameter (4.3)	Unlike what was observed in the case of integral data, K-Means did not produced clusters with relatively greater diameter compared to the hierarchical methods. Instead, in most cases it was observed to produce clusters with a lower maximum diameter compared to the hierarchical methods. This clearly shows the difference between using TAC data and integral data during clustering. No significant differences were observed between the hierarchical methods. As expected, this measure was observed to either remain the same or decrease with an increase in the number of clusters.

Measure	Observations
Separation (4.6)	This measure indicates the average distance between clusters obtained from different methods. The separation values obtained from K-Means algorithm were lower than those of the hierarchical methods. This indicates that the clusters obtained through the K-means method are less compact compared to those obtained from Hierarchical methods.
Minimum Separation (4.7)	These values were observed to either remain the same or decrease as the number of clusters was increased. K-means was observed to return clusters with higher minimum separation values compared to the hierarchical methods. The low values of this measure indicates that the clusters generated by most methods are not very well separated.
Average Split (4.8)	In most instances, the clusters obtained from K-means methods had relatively lower average split values compared to those obtained from the Hierarchical methods. The difference between the average split values was observed to be not significantly large. No noticeable trend was observed between the different hierarchical clustering results.
Total Energy (4.5)	As expected, K-means was observed to produce clusters with the lowest energy values in all instances. As the number of cluster groups were increased, the total energy values consistently decreased in the case of all the different clustering methods.

Measure	Observations
Avg. Silhouette Width (4.9)	<p>The average silhouette width values for all the methods on integral data was observed to consistently range between 0.25 and 0.55. Maximum values obtained using this measure were analyzed to gain a deeper insight regarding optimal number of clusters. Different methods were observed to give different results. 3 was the value most commonly observed to contain the peak for the average silhouette width measure in most cases. The optimal number of clusters varied depending upon the slice selected. In the case of slice 4, 3 clusters was found to be the optimal value in the case of most methods. In case of slice16, 4 clusters was found to be the consensus value and in the case of slice 27, the results varied between 4 and 5.</p>
Average Linkage Ratio (4.11)	<p>Significant differences in the average linkage ratio values were observed between different clustering methods. The maximum values across all the clusters was analyzed to obtain optimal number of clusters. Varying results were obtained for each individual slice and each subject. The overall consensus was observed to be 3 and 4 clusters for slice 4 depending upon the subject, and 3, 4 clusters for slices 16 and 27 respectively. The results of this measure, like most intra-inter cluster measures vary considerably from those obtained from integral data. This indicates that these measures are more sensitive to the data.</p>

Measure	Observations
Complete Linkage Ratio (4.12)	Using this measure, the optimal number of clusters was observed to be 3 for slices 4 and 16 and 4 for slice 27.
Combined Avg. Linkage Ratio (4.13)	Unlike the case for integral data, the TACs produce significantly different values for the average linkage ratio and the combined average linkage ratio. The overall profile of the comparison plots were again very similar between the two measures. The magnitudes were observed to be significantly different.

3. Significant Differences between TACs and Integrals

- For integral data, the corresponding average linkage and centroid linkage algorithms provide similar clusters in most cases, thereby providing identical values for each measure (notice 4 curves in most integral comparison plots as compared to TAC data comparison plots). Equivalently, this means that the most efficient HAL or HCL algorithm can be adopted without impacting the results of the clustering of integral data.
- Most of the intra-cluster measures and inter-cluster measures for the TAC data are relatively higher in magnitude compared to those for the integral data. The difference in the intra-cluster measures is greater as compared to the difference in the inter-cluster measures. This means that the TAC data produces clusters with significantly greater width and also those that are marginally better separated as compared to the clusters obtained from the integrals.
- The average silhouette width in the case of Integral data results are significantly greater as compared to TAC data results. This follows from the differences highlighted in the previous point. Since the intra-cluster measures are significantly higher in the case of TAC data and since the inter-cluster separation difference is not significantly high, this would lead to a lower difference between the average dissimilarity between the groups to the average dissimilarity within the groups in case of TAC data.

4. Optimal Number of Clusters

The results obtained for the optimal number of clusters for the different slices of two subjects are shown in tables 29 and 30 contained within Appendix III of this report. The optimal number of clusters are obtained by considering the maximum values for the average silhouette width, average linkage ratio, complete linkage ratio and combined average linkage ratio measures for each algorithm for a particular slice. These values can be directly obtained by looking at peaks in the method comparison plots shown in Chapter 5. In cases where the peak is difficult to observe, they can

be obtained from the quantitative values obtained by computing each measure that are listed in Appendix II. The optimal values were observed to be different for TAC and integral data. The following important issues related to determining the consensus value for the optimal number of clusters for each slice should be noted. Since the number of fast clustering algorithms used in this study outnumber the number of standard algorithms, it would be unfair to set the consensus purely based on the results obtained from the fast algorithms alone. Hence, the consensus is determined using a rule of thumb that requires atleast one of the fast algorithms and the standard algorithms to produce the same optimal number of clusters. In certain cases, especially while using Integral data, it was observed that all three methods (HAL/HCL,HAL1/HCL1,HAL2/HCL2) produce different results for optimal number of clusters. Again, in such instances, the consensus is indicated by a – meaning that it cannot be determined.

4.1. Conclusions. The important observations and conclusions made from the results about the optimal number of clusters shown in tables 29 and 30 are summarized below.

- The maximum values are more well pronounced in the case of integral data compared to TAC data. In case of TAC data, in most instances,these measures were observed to steadily decrease when the number of clusters was increased from 3 clusters to 7 clusters.
- The number of instances which agree with consensus for Integral and TAC data are summarized in Table 31. From these results a uniform trend is observed indicating that the integral data results for determining the optimal number of clusters are less consistent than those obtained from the multivariate TAC data.
- The hierarchical methods with average linkage (HAL,HAL1,HAL2) are the most consistent in the sense that they have the higher occurrence of the consensus value for the number of optimal clusters as compared to the other methods.

5. Conclusions

The current study successfully provides the basis for a validation strategy for comparing different clustering methods that can be used in the context of dynamic human PET data. No other studies of clustering for dynamic PET data, to the best of my knowledge, appear to have considered the use of clustering measures to identify the optimal number of clusters in the data. This may be a potential benefit of the results of the work since more insight into this information could lead to more efficient segmentation of the images thereby leading to better quantification of voxel level parameters. Another potential benefit and motivation behind the study was to compare the results obtained from the faster variants of hierarchical clustering to their more traditional counterparts. With the help of standard intra-cluster and inter-cluster measures that have been described in prior studies, we were able to quantify the cluster results with unique scalar values for each methods as a whole. The results indicate that the cluster results obtained from the less expensive hierarchical algorithms are comparable to those obtained from the traditional hierarchical algorithms that are more expensive. K-means was observed to produce meaningful results in most instances. It is significantly faster than the hierarchical algorithms but the average dissimilarity within the clusters was observed to be relatively high and the separation between clusters was also found to be relatively low compared to the clusters obtained from hierarchical methods. Overall, we determined the pre-processed PET data to be quite amenable to the traditional clustering strategies and from a visual perspective, the clusters were quite meaningful. This is mainly attributed to the appropriate weighting scheme that was employed to determine the dissimilarity between different pixels/voxels.

Four different intra-inter cluster measures were used to assess the overall goodness and optimality of the clusters. All of them, by definition, would indicate an optimal number of clusters based upon the maximum value of the measure over different number of clusters. The consensus value ranges between 3 and 5 for different slices and different subjects. The results greatly varied within different slices of the same subject and between different subjects. The possibility that these measures may

return sub-optimal clusters to be the most optimal cannot be fully ruled out. We are trying to get help of domain experts in determining an optimal number of clusters for a given slice based upon visual interpretation. This may again not be an effective strategy but it could be a good reference to start with. With the help of this knowledge, we can determine the number of instances when a particular measure conforms with or differs from the calls made by the domain experts.

6. Future Work

Patlak analysis [41] can be used to analyze the quality of cluster results by using them for quantifying the kinetic parameters in different healthy and affected subjects. These results can further be compared to those obtained by using traditional clustering methods and obtained by using other strategies that do not use clustering. Another immediate task is to compare the cluster results on the basis of the entire brain volume rather than individual slices. Due to the memory constraints involved in using the traditional clustering methods, only the faster variants of hierarchical algorithms and K-means algorithm would be possible to be compared in this mode. On completion of these tasks, it would be possible to research the use and validation of other efficient and popular clustering strategies. For example, Self organizing feature maps (SOM) [27], fuzzy clustering techniques, principle component analysis (PCA), random graphs and latent class models could be interesting strategies than could be tested in this domain.

Multivariate statistical techniques can be used to study the statistical significance of each cluster obtained from different clustering methods and also to compare different measure obtained from different clustering methods.

Other cluster validation measures such as the Davies-Bouldin Index, the modified Hubert's Γ statistic [39] can be used to improve the consensus on the optimal number of clusters.

Bibliography

- [1] Phelps, M.E., Positron Emission Tomography. In: Mazziotta, J. and Gilman, S., Eds.,1992, *Clinical Brain Imaging: Principles and Applications*, F.A. Davis Company, pp. 71-107
- [2] Iwata, Ren,2004, *Reference Book for PET Radiopharmaceuticals*, Cyclotron and Radioisotope Center, Tohoku University, Japan [<http://kakuyaku.cytic.tohoku.ac.jp/indexe.html>]
- [3] Hongbin Guo, Rosemary Renaut, Kewei Chen and Eric Reiman,2003, *Clustering Huge Data sets for Parametric PET imaging*,Biosystems, 71, 1-2, pp.81-92
- [4] Clarke, L.P., Velthuizen, R.P., Camacho, M.A., Heine, J.J., Vaidyanathan, M.,Hall, L.O., Thatcher, R.W., Silbiger, M.L.,1995,*MRI segmentation:methods and applications*,Magn.Reson.Imag. 13(3),pp.343-368
- [5] Kimura, Y., Senda, M., Alpert, N.,2002,*Fast formation of statistically reliable FDG parametric images based on clustering and principal components*, Phys. Med. Biol. 47(3),pp.455-468
- [6] Zhou, Y.,2000,*Model fitting with spatial constraint for parametric imaging in dynamic PET studies*,PhD. thesis, UCLA, Biomedical Physics, Advisor: Sung-Cheng Huang
- [7] Acton, P.D., Pilowsky, L.S., Kung, H.F., Ell, P.J., 1999,*Automatic segmentation of dynamic neuroreceptor single-photon emission tomography images using fuzzy clustering*, Eur. J. Nucl. Med. 26 (6), pp.582 - 590
- [8] Liptrot, M., Adams, K. H., Martiny, L., Pinborg, L. H., Lonsdale, M. N., Olsen, N. V., Holm, S., Svarer, C., Knudsen, G. M., 2004. *Cluster analysis in kinetic modelling of the brain: a noninvasive alternative to arterial sampling*. Neuroimage 21 (2), 483 - 493.
- [9] K.-P. Wong, D. Feng, S. R. Meikle, and M. J. Fulham, 2002, *Segmentation of dynamic PET images using cluster analysis*, IEEE Trans. Nuclear Science, 49, pp.200207
- [10] Sokal, R. R., and Michener, C. D. 1958. *A statistical method for evaluating systematic relationships*. University of Kansas Science Bulletin, 38, 1409 - 1438.
- [11] McQueen, J. 1967. *Some methods for classification and analysis of multivariate observations*. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 281 - 297.

- [12] Pavel Berkhin. *Survey of Clustering Data Mining Techniques*. Accrue Software. 2002.
- [13] Jain, A.K., Murty M.N., and Flynn P.J. (1999): *Data Clustering: A Review*, ACM Computing Surveys, Vol 31, No. 3, 264 - 323
- [14] A.K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [15] Kaufman L., and Rousseeuw P., 1990. *Finding groups in Data : An introduction to Cluster Analysis*. John Wiley and Sons, New York, NY.
- [16] Everitt B.S., Landau S., Leese M., 2001. *Cluster Analysis, 4th Edition*. Edward Arnold, London, UK.
- [17] Cheetham, A.H. and Hazel, J.E., 1969. *Binary (presence-absence) similarity coefficients*. J. Paleontol. 43: 113 - 136.
- [18] Boyce, A.J. 1969. *Mapping diversity. A comparative study of some numerical methods*. Numerical Taxonomy, Academic Press, New York.
- [19] Williams, W.T., Lambert, J.M. and Lance, G.N. 1966. *Multivariate methods in plant ecology. V. Similarity analysis and information analysis*. J. Ecol., 47, 83-101.
- [20] Diday, E. 1973. *The dynamic cluster method in non-hierarchical clustering*. J. Comput. Inf. Sci. 2, 61 - 88.
- [21] Symon, M. J. 1977. *Clustering criterion and multi-variate normal mixture*. Biometrics 77, 35 - 43.
- [22] Lance, G., and Williams W., 1967. *A general theory of classification sorting strategies*. Computer Journal, 9, 373 - 386.
- [23] Hubert, L. and Schultz, J., 1976. *Quadratic assignment as a general data analysis strategy*, BR J Math Statis Psychol, 29, 190-241
- [24] Dunn, D., *Well separated clusters and optimal fuzzy partitions*, J Cybernet 4, 95-104
- [25] Davies, D. L. and Bouldin D. W., 1979. *A Cluster Separation measure*, IEEE Trans Pattern Recognition Machine Intelligence, 1:2, 224-227
- [26] Jensen, R.E., 1969, *A dynamic programming algorithm for cluster analysis*. Operations Research 17, 1034 - 1057.
- [27] Kohonen, T. 1989. *Self-Organization and Associative Memory*. 3rd ed. Springer information sciences series. Springer-Verlag, New York, NY.

- [28] Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Co., Inc., Redwood City, CA.
- [29] Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P. 1983. *Optimization by simulated annealing*. Science 220, 4598 (May), 671 - 680.
- [30] Al-Sultan, K. S. 1995. *A tabu search approach to clustering problems*. Pattern Recogn. 28, 1443 - 1451.
- [31] Berry, M.J.A. and Linoff, G. 1996. *Data Mining Techniques For Marketing, Sales and Customer Support*. John Wiley & Sons, Inc., USA.
- [32] Hammersley, J.M., and Handscomb, D.C., 1965, *Monte Carlo Methods*. Methuen and Company Ltd., London
- [33] Wilks, S.S., 1963, *Mathematical Statistics*. John Wiley and Sons, Inc., New York
- [34] Rand, W.M., 1971, *Objective criteria for the evaluation of clustering methods*. J. American Statistical Association 66, 846 - 850.
- [35] Fowlkes, E.B., and Mallows, C.L., 1983, *A method for comparing two hierarchical clusterings*. J. American Statistical Association 78, 553 - 569.
- [36] Seber, G.A.F., 1984. *Multivariate Observations*, Wiley, New York.
- [37] Spath H. 1985, *Cluster Dissection and Analysis: Theory, FORTRAN Programs, Examples*, translated by J. Goldschmidt, Halsted Press, New York.
- [38] Halkidi, M., Batistakis, Y., Vazirgiannis M., 2001. *On Clustering Validation Techniques*. Journal of Intelligent Information Systems, 17:2/3, 107 - 145.
- [39] Bezdek, J. C. and Pal, N.R., *Some new indexes of cluster validity* IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics 28 (1998), no. 3, 301 - 315.
- [40] Ashburner, J., Haslam, J., Taylor, C., Cunningham, V., Jones, T., 1996, *A cluster analysis for the characterisation of dynamic PET data*, In: Myers, R., Cunningham, V., Bailey, D., Jones, T. (Eds.), *Quantification of Brain Function Using PET*. Academic Press, San Diego, CA, pp. 301 - 306
- [41] Patlak CS, Blasberg R, Fernstermacher JD., 1983, *Graphical evaluation of blood to brain transfer constants from multiple time-uptake data*. J. Cereb Blood Flow Metab 1983;3:1 - 7.

Appendix I : Statistical Results(ANOVA)

Cluster	HCL-HCL1	HCL-HCL2	HCL-HAL	HCL-HAL1	HCL-HAL2	HCL-K Means	HCL1-HCL2
3,1	0.00000	0.00000	1.00000	0.00000	0.00000	0.00001	0.93621
3,2	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.85987
3,3	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.77336
4,1	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.93621
4,2	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.20483
4,3	0.00052	0.00000	1.00000	0.00052	0.00000	0.00000	0.09039
4,4	1.00000	0.77336	1.00000	1.00000	0.77336	0.00000	0.77336
5,1	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00015
5,2	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00066
5,3	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.20483
5,4	0.00052	0.00000	1.00000	0.00052	0.00000	0.00000	0.09039
5,5	1.00000	0.77336	1.00000	1.00000	0.77336	0.00000	0.77336
Cluster	HCL1-HAL	HCL1-HAL1	HCL1-HAL2	HCL1-Kmeans	HCL2-HAL	HCL2-HAL1	HCL2-HAL2
3,1	0.00000	1.00000	0.93621	0.00000	0.00000	0.93621	1.00000
3,2	0.00000	1.00000	0.85987	0.00000	0.00000	0.85987	1.00000
3,3	0.00000	1.00000	0.77336	0.00000	0.00000	0.77336	1.00000
4,1	0.00000	1.00000	0.93621	0.00000	0.00000	0.93621	1.00000
4,2	0.00000	1.00000	0.20483	0.00000	0.00000	0.20483	1.00000
4,3	0.00052	1.00000	0.09039	0.00000	0.00000	0.09039	1.00000
4,4	1.00000	1.00000	0.77336	0.00000	0.77336	0.77336	1.00000
5,1	0.00000	1.00000	0.00015	0.55518	0.00000	0.00015	1.00000
5,2	0.00000	1.00000	0.00066	0.00000	0.00000	0.00066	1.00000
5,3	0.00000	1.00000	0.20483	0.00000	0.00000	0.20483	1.00000
5,4	0.00052	1.00000	0.09039	0.00000	0.00000	0.09039	1.00000
5,5	1.00000	1.00000	0.77336	0.00000	0.77336	0.77336	1.00000
Cluster	HCL2-Kmeans	HAL-HAL1	HAL-HAL2	HAL-Kmeans	HAL1-HAL2	HAL1-Kmeans	HAL2-Kmeans
3,1	0.00000	0.00000	0.00000	0.00001	0.93621	0.00000	0.00000
3,2	0.00000	0.00000	0.00000	0.00000	0.85987	0.00000	0.00000
3,3	0.00000	0.00000	0.00000	0.00000	0.77336	0.00000	0.00000
4,1	0.00000	0.00000	0.00000	0.00000	0.93621	0.00000	0.00000
4,2	0.00000	0.00000	0.00000	0.00000	0.20483	0.00000	0.00000
4,3	0.00000	0.00052	0.00000	0.00000	0.09039	0.00000	0.00000
4,4	0.00000	1.00000	0.77336	0.00000	0.77336	0.00000	0.00000
5,1	0.00134	0.00000	0.00000	0.00000	0.00015	0.55518	0.00134
5,2	0.00000	0.00000	0.00000	0.00000	0.00066	0.00000	0.00000
5,3	0.00000	0.00000	0.00000	0.00000	0.20483	0.00000	0.00000
5,4	0.00000	0.00052	0.00000	0.00000	0.09039	0.00000	0.00000
5,5	0.00000	1.00000	0.77336	0.00000	0.77336	0.00000	0.00000

Table 1. Subject:p03246dy1 ,Slice Number :16 Anova [INTEGRALS]

Cluster	HCL-HCL1	HCL-HCL2	HCL-HAL	HCL-HAL1	HCL-HAL2	HCL-K Means	HCL1-HCL2
6,1	1.00000	0.00015	1.00000	1.00000	0.00015	0.00000	0.00015
6,2	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00066
6,3	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.20483
6,4	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.06142
6,5	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.91771
6,6	1.00000	0.77336	1.00000	1.00000	0.77336	0.00000	0.77336
7,1	1.00000	0.00015	1.00000	1.00000	0.00015	0.00000	0.00015
7,2	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00066
7,3	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
7,4	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
7,5	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
7,6	0.00000	0.91771	1.00000	0.00000	0.91771	0.00000	0.00000
7,7	0.00001	0.77336	1.00000	0.00001	0.77336	0.00000	0.00000
Cluster	HCL1-HAL	HCL1-HAL1	HCL1-HAL2	HCL1-Kmeans	HCL2-HAL	HCL2-HAL1	HCL2-HAL2
6,1	1.00000	1.00000	0.00015	0.00000	0.00015	0.00015	1.00000
6,2	0.00000	1.00000	0.00066	0.00000	0.00000	0.00066	1.00000
6,3	0.00000	1.00000	0.20483	0.00000	0.00000	0.20483	1.00000
6,4	0.00000	1.00000	0.06142	0.00000	0.00000	0.06142	1.00000
6,5	0.00000	1.00000	0.91771	0.00000	0.00000	0.91771	1.00000
6,6	1.00000	1.00000	0.77336	0.00000	0.77336	0.77336	1.00000
7,1	1.00000	1.00000	0.00015	0.00000	0.00015	0.00015	1.00000
7,2	0.00000	1.00000	0.00066	0.00000	0.00000	0.00066	1.00000
7,3	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	1.00000
7,4	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	1.00000
7,5	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000	1.00000
7,6	0.00000	1.00000	0.00000	0.00000	0.91771	0.00000	1.00000
7,7	0.00001	1.00000	0.00000	0.00000	0.77336	0.00000	1.00000
Cluster	HCL2-Kmeans	HAL-HAL1	HAL-HAL2	HAL-Kmeans	HAL1-HAL2	HAL1-Kmeans	HAL2-Kmeans
6,1	0.00000	1.00000	0.00015	0.00000	0.00015	0.00000	0.00000
6,2	0.00000	0.00000	0.00000	0.00000	0.00066	0.00000	0.00000
6,3	0.00000	0.00000	0.00000	0.00000	0.20483	0.00000	0.00000
6,4	0.00000	0.00000	0.00000	0.00000	0.06142	0.00000	0.00000
6,5	0.00000	0.00000	0.00000	0.00000	0.91771	0.00000	0.00000
6,6	0.00000	1.00000	0.77336	0.00000	0.77336	0.00000	0.00000
7,1	0.00000	1.00000	0.00015	0.00000	0.00015	0.00000	0.00000
7,2	0.00000	0.00000	0.00000	0.00000	0.00066	0.00000	0.00000
7,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,6	0.00000	0.00000	0.91771	0.00000	0.00000	0.00000	0.00000
7,7	0.00000	0.00001	0.77336	0.00000	0.00000	0.00000	0.00000

Table 2. Subject:p03246dy1 ,Slice Number :16 Anova [INTEGRALS]

Cluster	HCL-HCL1	HCL-HCL2	HCL-HAL	HCL-HAL1	HCL-HAL2	HCL-K Means	HCL1-HCL2
3,1	0.63616	0.96592	0.00000	0.00000	0.00000	0.00000	0.66660
3,2	0.00000	0.00000	0.00000	0.97088	0.00000	0.00000	0.58133
3,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.88831
4,1	0.63616	0.96592	0.00000	0.00000	0.00000	0.00000	0.66660
4,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.19780
4,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.73275
4,4	0.55742	0.46849	0.12001	0.55742	0.46849	0.00000	0.88831
5,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00005	0.83684
5,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.17128
5,3	0.00000	0.00000	0.00157	0.00000	0.00000	0.00000	0.19780
5,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.73275
5,5	0.55742	0.46849	0.12001	0.55742	0.46849	0.00000	0.88831
Cluster	HCL1-HAL	HCL1-HAL1	HCL1-HAL2	HCL1-Kmeans	HCL2-HAL	HCL2-HAL1	HCL2-HAL2
3,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3,3	0.00000	1.00000	0.00000	0.00000	0.00000	0.88831	0.00000
4,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4,3	0.05286	1.00000	0.76275	0.00000	0.11274	0.73275	0.96844
4,4	0.03418	1.00000	0.88831	0.00000	0.02472	0.88831	1.00000
5,1	0.00000	0.00000	0.85777	0.00000	0.00000	0.00000	0.97847
5,2	0.00000	0.00000	0.00949	0.00000	0.00000	0.00000	0.19230
5,3	0.00047	0.00000	0.72377	0.00000	0.00000	0.00000	0.35031
5,4	0.05286	1.00000	0.76275	0.00000	0.11274	0.73275	0.96844
5,5	0.03418	1.00000	0.88831	0.00000	0.02472	0.88831	1.00000
Cluster	HCL2-Kmeans	HAL-HAL1	HAL-HAL2	HAL-Kmeans	HAL1-HAL2	HAL1-Kmeans	HAL2-Kmeans
3,1	0.00000	0.00000	0.00000	0.00046	0.00000	0.00000	0.00000
3,2	0.00000	0.00000	0.00006	0.00000	0.00000	0.00000	0.00000
3,3	0.00000	0.00000	0.98760	0.00000	0.00000	0.00000	0.00000
4,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
4,3	0.00000	0.05286	0.10393	0.00000	0.76275	0.00000	0.00000
4,4	0.00000	0.03418	0.02472	0.00000	0.88831	0.00000	0.00000
5,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00790	0.00000
5,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
5,3	0.00000	0.00000	0.00012	0.00000	0.00000	0.00000	0.00000
5,4	0.00000	0.05286	0.10393	0.00000	0.76275	0.00000	0.00000
5,5	0.00000	0.03418	0.02472	0.00000	0.88831	0.00000	0.00000

Table 3. Subject:p03246dy1 ,Slice Number :16 Anova [TACs]

Cluster	HCL-HCL1	HCL-HCL2	HCL-HAL	HCL-HAL1	HCL-HAL2	HCL-K Means	HCL1-HCL2
6,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.83684
6,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.17128
6,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.19780
6,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.73275
6,5	0.00000	0.00000	0.00000	0.00000	0.22681	0.00000	0.85425
6,6	0.00000	0.00000	0.12001	0.55742	0.46849	0.00000	1.00000
7,1	0.00000	0.00000	0.00000	0.11328	0.00000	0.00000	0.28918
7,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.51278
7,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.17128
7,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.19780
7,5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.73275
7,6	0.00000	0.00000	0.00000	0.00000	0.22681	0.00000	0.85425
7,7	0.00000	0.00000	0.00008	0.55742	0.46849	0.00000	1.00000
Cluster	HCL1-HAL	HCL1-HAL1	HCL1-HAL2	HCL1-Kmeans	HCL2-HAL	HCL2-HAL1	HCL2-HAL2
6,1	0.00000	0.00000	0.85777	0.00000	0.00000	0.00000	0.97847
6,2	0.00000	0.00000	0.00949	0.00000	0.00000	0.00000	0.19230
6,3	0.00000	0.00000	0.72377	0.00000	0.00000	0.00000	0.35031
6,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6,5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6,6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,1	0.00017	0.00000	0.46655	0.00000	0.00606	0.00000	0.08104
7,2	0.00000	0.00000	0.01519	0.00000	0.00000	0.00000	0.00199
7,3	0.00000	0.00000	0.00949	0.00008	0.00000	0.00000	0.19230
7,4	0.00047	0.00000	0.72377	0.00000	0.00000	0.00000	0.35031
7,5	0.05286	0.00000	0.00000	0.00000	0.11274	0.00000	0.00000
7,6	0.00012	0.00000	0.00000	0.00000	0.00006	0.00000	0.00000
7,7	0.03846	0.00000	0.00000	0.00000	0.03846	0.00000	0.00000
Cluster	HCL2-Kmeans	HAL-HAL1	HAL-HAL2	HAL-Kmeans	HAL1-HAL2	HAL1-Kmeans	HAL2-Kmeans
6,1	0.00000	0.00000	0.00000	0.00000	0.00000	0.80511	0.00000
6,2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6,3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6,4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
6,5	0.00000	0.05286	0.00000	0.00000	0.00000	0.00000	0.00000
6,6	0.00000	0.03418	0.02472	0.00000	0.88831	0.00000	0.00000
7,1	0.00000	0.00000	0.00002	0.00019	0.00000	0.00000	0.00000
7,2	0.00000	0.00000	0.00000	0.00024	0.00000	0.00000	0.00000
7,3	0.00525	0.00000	0.00000	0.00000	0.00000	0.00000	0.15427
7,4	0.00000	0.00000	0.00012	0.00000	0.00000	0.00000	0.00000
7,5	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,6	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
7,7	0.00000	0.00027	0.00037	0.00000	0.88831	0.00000	0.00000

Table 4. Subject:p03246dy1 ,Slice Number :16 Anova [TACs]

Appendix II : Quantitative Values of Validation Measures

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.2124	0.1255	0.1014	0.0869	0.0746
HCL1	0.1559	0.1184	0.0988	0.0911	0.0633
HCL2	0.1261	0.1184	0.1034	0.1034	0.0604
HAL	0.2124	0.1255	0.1014	0.0869	0.0746
HAL1	0.1559	0.1184	0.0988	0.0911	0.0633
HAL2	0.1261	0.1184	0.1034	0.1034	0.0604
K-Means	0.1702	0.1488	0.1305	0.1019	0.0942

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.2128	0.1577	0.1577	0.1181	0.1181
HCL1	0.2213	0.1909	0.1181	0.1181	0.1181
HCL2	0.2174	0.1908	0.1250	0.1250	0.1250
HAL	0.2128	0.1577	0.1577	0.1181	0.1181
HAL1	0.2213	0.1909	0.1181	0.1181	0.1181
HAL2	0.2174	0.1908	0.1250	0.1250	0.1250
K-Means	0.3168	0.2460	0.2241	0.2154	0.2126

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.1878	0.1585	0.1096	0.1096	0.1096
HCL1	0.1756	0.1447	0.1150	0.1150	0.0700
HCL2	0.1543	0.1543	0.1150	0.1150	0.1150
HAL	0.1878	0.1585	0.1096	0.1096	0.1096
HAL1	0.1756	0.1447	0.1150	0.1150	0.0700
HAL2	0.1543	0.1543	0.1150	0.1150	0.1150
K-Means	0.1941	0.1554	0.1388	0.1324	0.1216

Table 5. Max.Distance from Mean, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0401	0.0328	0.0266	0.0230	0.0199
HCL1	0.0413	0.0319	0.0264	0.0222	0.0194
HCL2	0.0415	0.0325	0.0268	0.0224	0.0197
HAL	0.0401	0.0328	0.0266	0.0230	0.0199
HAL1	0.0413	0.0319	0.0264	0.0222	0.0194
HAL2	0.0415	0.0325	0.0268	0.0224	0.0197
K-Means	0.0448	0.0348	0.0288	0.0241	0.0207

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0491	0.0396	0.0323	0.0270	0.0236
HCL1	0.0480	0.0398	0.0322	0.0275	0.0235
HCL2	0.0481	0.0398	0.0323	0.0276	0.0239
HAL	0.0491	0.0396	0.0323	0.0270	0.0236
HAL1	0.0480	0.0398	0.0322	0.0275	0.0235
HAL2	0.0481	0.0398	0.0323	0.0276	0.0239
K-Means	0.0440	0.0368	0.0300	0.0254	0.0222

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.0482	0.0370	0.0307	0.0255	0.0216
HCL1	0.0474	0.0363	0.0302	0.0253	0.0221
HCL2	0.0466	0.0366	0.0300	0.0253	0.0221
HAL	0.0482	0.0370	0.0307	0.0255	0.0216
HAL1	0.0474	0.0363	0.0302	0.0253	0.0221
HAL2	0.0466	0.0367	0.0301	0.0254	0.0222
K-Means	0.0460	0.0357	0.0292	0.0246	0.0214

Table 6. Avg.Distance from Mean, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.3094	0.2027	0.1577	0.1577	0.1262
HCL1	0.2235	0.1852	0.1767	0.1371	0.1106
HCL2	0.2465	0.1852	0.1540	0.1540	0.1086
HAL	0.3094	0.2027	0.1577	0.1577	0.1262
HAL1	0.2235	0.1852	0.1767	0.1371	0.1106
HAL2	0.2465	0.1852	0.1540	0.1540	0.1086
K-Means	0.2616	0.2204	0.2062	0.1616	0.1515

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.2894	0.2381	0.2381	0.1682	0.1682
HCL1	0.3188	0.2979	0.1682	0.1682	0.1682
HCL2	0.3158	0.2979	0.1822	0.1822	0.1822
HAL	0.2894	0.2381	0.2381	0.1682	0.1682
HAL1	0.3188	0.2979	0.1682	0.1682	0.1682
HAL2	0.3158	0.2979	0.1822	0.1822	0.1822
K-Means	0.4012	0.3272	0.3022	0.2927	0.2890

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.3105	0.2677	0.1635	0.1629	0.1629
HCL1	0.2888	0.2111	0.1818	0.1742	0.1192
HCL2	0.2544	0.2544	0.2042	0.1742	0.1742
HAL	0.3105	0.2677	0.1635	0.1629	0.1629
HAL1	0.2888	0.2111	0.1818	0.1742	0.1192
HAL2	0.2544	0.2544	0.2042	0.1742	0.1742
K-Means	0.2787	0.2265	0.2013	0.1925	0.1758

Table 7. Maximum Diameter, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0540	0.0440	0.0357	0.0308	0.0265
HCL1	0.0555	0.0431	0.0354	0.0297	0.0260
HCL2	0.0564	0.0436	0.0358	0.0299	0.0262
HAL	0.0540	0.0440	0.0357	0.0308	0.0265
HAL1	0.0555	0.0431	0.0354	0.0297	0.0260
HAL2	0.0564	0.0436	0.0358	0.0299	0.0262
K-Means	0.0603	0.0467	0.0385	0.0325	0.0278

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0665	0.0534	0.0434	0.0364	0.0317
HCL1	0.0654	0.0536	0.0434	0.0369	0.0314
HCL2	0.0657	0.0537	0.0435	0.0370	0.0320
HAL	0.0665	0.0534	0.0434	0.0364	0.0317
HAL1	0.0654	0.0536	0.0434	0.0369	0.0314
HAL2	0.0657	0.0537	0.0435	0.0370	0.0320
K-Means	0.0592	0.0496	0.0404	0.0344	0.0300

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.0651	0.0498	0.0409	0.0342	0.0291
HCL1	0.0642	0.0489	0.0405	0.0339	0.0296
HCL2	0.0635	0.0496	0.0404	0.0339	0.0295
HAL	0.0651	0.0498	0.0409	0.0342	0.0291
HAL1	0.0642	0.0489	0.0405	0.0339	0.0296
HAL2	0.0635	0.0497	0.0405	0.0340	0.0296
K-Means	0.0622	0.0484	0.0393	0.0332	0.0287

Table 8. Average Spread, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	149.2533	116.2821	78.7055	79.1347	58.3360
HCL1	170.3116	115.9347	101.4571	69.9426	66.8537
HCL2	234.5830	97.2516	83.4350	68.8573	64.9286
HAL	149.2533	116.2821	78.7055	79.1347	58.3360
HAL1	170.3116	115.9347	101.4571	69.9426	66.8537
HAL2	234.5830	97.2516	83.4350	68.8573	64.9286
K-Means	183.9854	151.0640	99.9635	65.8392	59.1852

Slice Number 16					
Alg	3	4	5	6	7
HCL	209.1631	190.7092	145.0967	116.0357	109.1489
HCL1	236.1823	202.7794	130.2194	125.5388	124.8807
HCL2	227.4741	232.6155	185.2719	171.2087	119.3876
HAL	209.1631	190.7092	145.0967	116.0357	109.1489
HAL1	236.1823	202.7794	130.2194	125.5388	124.8807
HAL2	227.4741	232.6155	185.2719	171.2087	119.3876
K-Means	188.3166	147.9956	143.0427	117.8402	117.1865

Slice Number 27					
Alg	3	4	5	6	7
HCL	306.9925	230.7886	110.7913	98.7417	82.5302
HCL1	205.3016	155.2984	143.5460	116.7734	108.9472
HCL2	179.7176	193.6394	164.8661	118.8105	91.8167
HAL	306.9925	230.7886	110.7913	98.7417	82.5302
HAL1	205.3016	155.2984	143.5460	116.7734	108.9472
HAL2	179.7176	183.3752	154.6020	108.5464	92.9251
K-Means	222.6227	165.5489	154.1451	118.5773	79.5150

Table 9. Total Energy, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0305	0.0305	0.0305	0.0189	0.0189
HCL1	0.0676	0.0384	0.0305	0.0305	0.0258
HCL2	0.0510	0.0487	0.0326	0.0317	0.0293
HAL	0.0305	0.0305	0.0305	0.0189	0.0189
HAL1	0.0676	0.0384	0.0305	0.0305	0.0258
HAL2	0.0510	0.0487	0.0326	0.0317	0.0293
K-Means	0.0635	0.0472	0.0388	0.0352	0.0264

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0772	0.0540	0.0487	0.0338	0.0273
HCL1	0.0569	0.0569	0.0508	0.0335	0.0334
HCL2	0.0591	0.0591	0.0566	0.0350	0.0289
HAL	0.0772	0.0540	0.0487	0.0338	0.0273
HAL1	0.0569	0.0569	0.0508	0.0335	0.0334
HAL2	0.0591	0.0591	0.0566	0.0350	0.0289
K-Means	0.0623	0.0548	0.0412	0.0339	0.0299

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.0365	0.0365	0.0365	0.0300	0.0300
HCL1	0.0667	0.0546	0.0321	0.0321	0.0294
HCL2	0.0702	0.0273	0.0273	0.0273	0.0273
HAL	0.0365	0.0365	0.0365	0.0300	0.0300
HAL1	0.0667	0.0546	0.0321	0.0321	0.0294
HAL2	0.0702	0.0218	0.0218	0.0218	0.0218
K-Means	0.0757	0.0579	0.0439	0.0397	0.0310

Table 10. Minimum Separation, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.2203	0.2454	0.2169	0.2517	0.2307
HCL1	0.2084	0.2081	0.2334	0.2090	0.2197
HCL2	0.2434	0.2005	0.2217	0.1971	0.2144
HAL	0.2203	0.2454	0.2169	0.2517	0.2307
HAL1	0.2084	0.2081	0.2334	0.2090	0.2197
HAL2	0.2434	0.2005	0.2217	0.1971	0.2144
K-Means	0.2034	0.1929	0.1957	0.1981	0.1905

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.2566	0.2875	0.2520	0.2552	0.2462
HCL1	0.3226	0.2653	0.2771	0.2586	0.2698
HCL2	0.3213	0.2637	0.2714	0.2543	0.2311
HAL	0.2566	0.2875	0.2520	0.2552	0.2462
HAL1	0.3226	0.2653	0.2771	0.2586	0.2698
HAL2	0.3213	0.2637	0.2714	0.2543	0.2311
K-Means	0.1934	0.2050	0.2001	0.1917	0.1863

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.2852	0.2830	0.2505	0.2305	0.2210
HCL1	0.2412	0.2353	0.2601	0.2321	0.2494
HCL2	0.2499	0.2646	0.2670	0.2391	0.2293
HAL	0.2852	0.2830	0.2505	0.2305	0.2210
HAL1	0.2412	0.2353	0.2601	0.2321	0.2494
HAL2	0.2499	0.2649	0.2673	0.2394	0.2296
K-Means	0.2265	0.2171	0.2066	0.2068	0.1996

Table 11. Separation, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.1445	0.1834	0.1669	0.2072	0.1928
HCL1	0.1341	0.1484	0.1824	0.1668	0.1826
HCL2	0.1632	0.1417	0.1716	0.1556	0.1776
HAL	0.1445	0.1834	0.1669	0.2072	0.1928
HAL1	0.1341	0.1484	0.1824	0.1668	0.1826
HAL2	0.1632	0.1417	0.1716	0.1556	0.1776
K-Means	0.1254	0.1318	0.1436	0.1533	0.1516

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.1665	0.2084	0.1902	0.2036	0.2011
HCL1	0.2164	0.1873	0.2148	0.2056	0.2235
HCL2	0.2138	0.1850	0.2081	0.2004	0.1852
HAL	0.1665	0.2084	0.1902	0.2036	0.2011
HAL1	0.2164	0.1873	0.2148	0.2056	0.2235
HAL2	0.2138	0.1850	0.2081	0.2004	0.1852
K-Means	0.1187	0.1408	0.1467	0.1469	0.1476

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.1883	0.2116	0.1935	0.1827	0.1799
HCL1	0.1511	0.1672	0.2020	0.1840	0.2071
HCL2	0.1592	0.1911	0.2087	0.1909	0.1876
HAL	0.1883	0.2116	0.1935	0.1827	0.1799
HAL1	0.1511	0.1672	0.2020	0.1840	0.2071
HAL2	0.1592	0.1908	0.2085	0.1908	0.1876
K-Means	0.1404	0.1501	0.1524	0.1607	0.1599

Table 12. Average Split, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.5902	0.5902	0.5620	0.5811	0.5730
HCL1	0.5693	0.5595	0.5626	0.5609	0.5557
HCL2	0.6045	0.5507	0.5607	0.5503	0.5493
HAL	0.5902	0.5902	0.5620	0.5811	0.5730
HAL1	0.5693	0.5595	0.5626	0.5609	0.5557
HAL2	0.6045	0.5507	0.5607	0.5503	0.5493
K-Means	0.5292	0.5163	0.5251	0.5299	0.5351

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.5850	0.6058	0.5682	0.5623	0.5455
HCL1	0.6491	0.5877	0.5871	0.5615	0.5791
HCL2	0.6475	0.5836	0.5808	0.5567	0.5449
HAL	0.5850	0.6058	0.5682	0.5623	0.5455
HAL1	0.6491	0.5877	0.5871	0.5615	0.5791
HAL2	0.6475	0.5836	0.5808	0.5567	0.5449
K-Means	0.5198	0.5171	0.5220	0.5118	0.5022

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.6147	0.5831	0.5592	0.5492	0.5415
HCL1	0.5694	0.5544	0.5660	0.5580	0.5527
HCL2	0.5837	0.5947	0.5556	0.5489	0.5357
HAL	0.6147	0.5831	0.5592	0.5492	0.5415
HAL1	0.5694	0.5544	0.5660	0.5580	0.5527
HAL2	0.5837	0.5942	0.5552	0.5484	0.5345
K-Means	0.5607	0.5445	0.5342	0.5305	0.5270

Table 13. Average Silhouette Width, Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.5042	0.6769	0.6050	0.5929	0.6016
HCL1	0.6540	0.5113	0.5305	0.6267	0.5036
HCL2	0.6954	0.5823	0.6036	0.4378	0.6127
HAL	0.5042	0.6769	0.6050	0.5929	0.6016
HAL1	0.6540	0.5113	0.5305	0.6267	0.5036
HAL2	0.6954	0.5823	0.6036	0.4378	0.6127
K-Means	0.5206	0.4372	0.3784	0.4442	0.3511

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.6497	0.6532	0.4195	0.5153	0.4149
HCL1	0.6452	0.4694	0.7003	0.4705	0.4124
HCL2	0.6506	0.4578	0.6382	0.4528	0.3643
HAL	0.6497	0.6532	0.4195	0.5153	0.4149
HAL1	0.6452	0.4694	0.7003	0.4705	0.4124
HAL2	0.6506	0.4578	0.6382	0.4528	0.3643
K-Means	0.3574	0.3365	0.2717	0.2424	0.2090

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.6280	0.4892	0.7198	0.4744	0.4618
HCL1	0.5493	0.5651	0.5613	0.4938	0.5895
HCL2	0.6690	0.4497	0.4821	0.5585	0.3827
HAL	0.6280	0.4892	0.7198	0.4744	0.4618
HAL1	0.5493	0.5651	0.5613	0.4938	0.5895
HAL2	0.6690	0.4509	0.4821	0.5585	0.3856
K-Means	0.5750	0.5165	0.4386	0.4172	0.3545

Table 14. Modified Dunn's Index (Average Linkage), Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.8959	1.3676	1.2732	0.9581	1.1965
HCL1	1.6226	1.2073	1.0476	1.2893	1.2389
HCL2	1.6264	1.3314	1.2021	0.8920	1.2654
HAL	0.8959	1.3676	1.2732	0.9581	1.1965
HAL1	1.6226	1.2073	1.0476	1.2893	1.2389
HAL2	1.6264	1.3314	1.2021	0.8920	1.2654
K-Means	1.4602	0.9594	0.8063	0.9247	0.7416

Slice Number 16					
Alg	3	4	5	6	7
HCL	1.6084	1.2151	0.9524	1.0309	0.9025
HCL1	1.4313	1.0245	1.6702	0.9969	0.7897
HCL2	1.4492	1.0484	1.4277	0.9355	0.7955
HAL	1.6084	1.2151	0.9524	1.0309	0.9025
HAL1	1.4313	1.0245	1.6702	0.9969	0.7897
HAL2	1.4492	1.0484	1.4277	0.9355	0.7955
K-Means	0.8827	0.6974	0.6076	0.4861	0.4220

Slice Number 27					
Alg	3	4	5	6	7
HCL	1.1944	1.1596	1.5405	1.0033	0.9034
HCL1	1.3618	1.3682	1.1608	1.0435	1.4620
HCL2	1.6803	0.8758	1.0910	1.1669	0.8280
HAL	1.1944	1.1596	1.5405	1.0033	0.9034
HAL1	1.3618	1.3682	1.1608	1.0435	1.4620
HAL2	1.6803	0.8758	1.0910	1.1669	0.8496
K-Means	1.4452	1.0945	0.9333	0.8589	0.7223

Table 15. Modified Dunn's Index (Complete Linkage), Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.5042	0.6769	0.6050	0.5929	0.6016
HCL1	0.6540	0.5113	0.5305	0.6267	0.5036
HCL2	0.6954	0.5823	0.6036	0.4378	0.6127
HAL	0.5042	0.6769	0.6050	0.5929	0.6016
HAL1	0.6540	0.5113	0.5305	0.6267	0.5036
HAL2	0.6954	0.5823	0.6036	0.4378	0.6127
K-Means	0.5206	0.4372	0.3784	0.4442	0.3511

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.6497	0.6532	0.4195	0.5153	0.4149
HCL1	0.6452	0.4694	0.7003	0.4705	0.4124
HCL2	0.6506	0.4578	0.6382	0.4528	0.3643
HAL	0.6497	0.6532	0.4195	0.5153	0.4149
HAL1	0.6452	0.4694	0.7003	0.4705	0.4124
HAL2	0.6506	0.4578	0.6382	0.4528	0.3643
K-Means	0.3574	0.3365	0.2717	0.2424	0.2090

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.6280	0.4892	0.7198	0.4744	0.4618
HCL1	0.5493	0.5651	0.5613	0.4938	0.5895
HCL2	0.6690	0.4497	0.4821	0.5585	0.3827
HAL	0.6280	0.4892	0.7198	0.4744	0.4618
HAL1	0.5493	0.5651	0.5613	0.4938	0.5895
HAL2	0.6690	0.4509	0.4821	0.5585	0.3856
K-Means	0.5750	0.5165	0.4386	0.4172	0.3545

Table 16. Modified Dunn's Index (Combined Average), Subject:3246 Data:Integrals

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.1709	0.1562	0.1240	0.1240	0.1237
HCL1	0.1736	0.1387	0.1145	0.1145	0.1070
HCL2	0.1588	0.1366	0.1317	0.1079	0.1079
HAL	0.1633	0.1531	0.1247	0.1247	0.1247
HAL1	0.1785	0.1315	0.1315	0.1046	0.0996
HAL2	0.1607	0.1352	0.1284	0.1103	0.1103
K-Means	0.1569	0.1314	0.1145	0.1050	0.1031

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.1906	0.1906	0.1530	0.1299	0.1240
HCL1	0.2282	0.1917	0.1419	0.1419	0.1327
HCL2	0.2300	0.1907	0.1455	0.1455	0.1455
HAL	0.2226	0.1720	0.1524	0.1346	0.1346
HAL1	0.2418	0.1762	0.1432	0.1233	0.1233
HAL2	0.2225	0.1606	0.1422	0.1422	0.1273
K-Means	0.2947	0.2472	0.2191	0.1932	0.1593

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.2216	0.1612	0.1612	0.1608	0.1608
HCL1	0.2124	0.1870	0.1870	0.1870	0.1870
HCL2	0.2119	0.1715	0.1569	0.1545	0.1531
HAL	0.1876	0.1586	0.1552	0.1668	0.1355
HAL1	0.2145	0.1633	0.1633	0.1497	0.1497
HAL2	0.2109	0.1739	0.1508	0.1508	0.1508
K-Means	0.2078	0.1924	0.1623	0.1611	0.1603

Table 17. Max.Distance from Mean, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0626	0.0539	0.0499	0.0464	0.0439
HCL1	0.0601	0.0539	0.0501	0.0465	0.0446
HCL2	0.0599	0.0534	0.0490	0.0462	0.0435
HAL	0.0623	0.0546	0.0506	0.0483	0.0455
HAL1	0.0618	0.0547	0.0506	0.0464	0.0442
HAL2	0.0624	0.0534	0.0496	0.0468	0.0456
K-Means	0.0617	0.0543	0.0498	0.0465	0.0443

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0711	0.0624	0.0566	0.0533	0.0516
HCL1	0.0691	0.0624	0.0568	0.0508	0.0485
HCL2	0.0690	0.0625	0.0567	0.0506	0.0484
HAL	0.0700	0.0623	0.0572	0.0534	0.0490
HAL1	0.0687	0.0623	0.0567	0.0530	0.0516
HAL2	0.0702	0.0623	0.0567	0.0532	0.0506
K-Means	0.0670	0.0601	0.0553	0.0524	0.0499

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.0785	0.0721	0.0644	0.0615	0.0595
HCL1	0.0795	0.0712	0.0658	0.0647	0.0637
HCL2	0.0798	0.0713	0.0665	0.0628	0.0622
HAL	0.0791	0.0717	0.0627	0.0613	0.0606
HAL1	0.0798	0.0721	0.0646	0.0612	0.0600
HAL2	0.0808	0.0720	0.0666	0.0622	0.0605
K-Means	0.0787	0.0703	0.0654	0.0625	0.0603

Table 18. Avg.Distance from Mean, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.2841	0.2673	0.2163	0.2163	0.2082
HCL1	0.2975	0.2506	0.2035	0.2035	0.1960
HCL2	0.3013	0.2475	0.2224	0.1876	0.1876
HAL	0.2859	0.2859	0.2410	0.2410	0.2410
HAL1	0.3275	0.2393	0.2393	0.1871	0.1871
HAL2	0.2618	0.2429	0.2413	0.1876	0.1801
K-Means	0.2527	0.2230	0.1926	0.1764	0.1715

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.3408	0.3408	0.2707	0.2497	0.2116
HCL1	0.3630	0.3275	0.2497	0.2497	0.2336
HCL2	0.4001	0.3275	0.2562	0.2562	0.2562
HAL	0.3354	0.3208	0.2634	0.2359	0.2359
HAL1	0.3831	0.3180	0.2593	0.2175	0.2038
HAL2	0.3312	0.3079	0.2497	0.2497	0.2365
K-Means	0.3904	0.3614	0.3169	0.2883	0.2458

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.4077	0.2812	0.2812	0.2812	0.2812
HCL1	0.4019	0.3133	0.3133	0.3133	0.3133
HCL2	0.3943	0.2731	0.2636	0.2636	0.2612
HAL	0.3544	0.2864	0.2864	0.2812	0.2366
HAL1	0.3328	0.3136	0.3136	0.2740	0.2665
HAL2	0.4055	0.3345	0.2612	0.2612	0.2546
K-Means	0.3141	0.2857	0.2526	0.2427	0.2427

Table 19. Maximum Diameter, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0868	0.0749	0.0696	0.0642	0.0599
HCL1	0.0836	0.0753	0.0700	0.0638	0.0610
HCL2	0.0831	0.0742	0.0683	0.0645	0.0606
HAL	0.0862	0.0762	0.0706	0.0671	0.0628
HAL1	0.0857	0.0760	0.0706	0.0651	0.0617
HAL2	0.0863	0.0741	0.0692	0.0656	0.0635
K-Means	0.0855	0.0754	0.0696	0.0652	0.0623

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0988	0.0857	0.0782	0.0739	0.0714
HCL1	0.0949	0.0856	0.0783	0.0697	0.0668
HCL2	0.0948	0.0858	0.0783	0.0696	0.0667
HAL	0.0971	0.0860	0.0789	0.0740	0.0680
HAL1	0.0944	0.0855	0.0781	0.0733	0.0708
HAL2	0.0974	0.0857	0.0783	0.0737	0.0703
K-Means	0.0932	0.0838	0.0771	0.0733	0.0699

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.1095	0.1003	0.0895	0.0856	0.0828
HCL1	0.1102	0.0989	0.0905	0.0885	0.0870
HCL2	0.1109	0.0995	0.0924	0.0872	0.0864
HAL	0.1102	0.0998	0.0870	0.0851	0.0840
HAL1	0.1105	0.0997	0.0891	0.0849	0.0835
HAL2	0.1118	0.1001	0.0927	0.0868	0.0846
K-Means	0.1096	0.0981	0.0914	0.0874	0.0845

Table 20. Average Spread, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	207.5209	204.9716	183.1791	183.2614	159.3749
HCL1	229.8254	206.4586	190.5589	189.6196	181.2347
HCL2	203.2316	180.7153	175.5200	166.5811	166.1840
HAL	195.9626	179.6550	153.8724	147.4171	147.4069
HAL1	282.2082	180.6763	158.7179	149.7156	142.0133
HAL2	198.5040	191.3905	187.5309	138.6171	137.3985
K-Means	194.2802	189.1931	155.4193	166.5267	149.0693

Slice Number 16					
Alg	3	4	5	6	7
HCL	358.0641	347.7981	295.8929	263.4576	247.7051
HCL1	405.6591	358.5525	312.5544	311.5502	283.6521
HCL2	408.3560	356.1473	311.1938	310.0173	309.9596
HAL	371.4993	362.5819	280.1489	283.1806	281.1771
HAL1	363.7765	349.0002	287.5511	273.6006	268.1273
HAL2	351.2118	336.9215	275.3664	273.6068	269.3822
K-Means	312.1097	304.2221	277.5425	252.3583	240.5760

Slice Number 27					
Alg	3	4	5	6	7
HCL	376.7170	323.7267	350.9184	371.1124	342.3015
HCL1	428.4053	314.4729	312.5411	304.3016	319.4406
HCL2	406.4668	323.6375	292.5190	297.3633	287.6863
HAL	352.1553	299.0104	299.1977	285.2270	283.4420
HAL1	328.6738	324.9384	332.4017	290.3264	273.3339
HAL2	383.6615	324.8287	294.9583	284.2788	272.6102
K-Means	356.7461	305.9105	271.2002	268.2276	282.8642

Table 21. Total Energy, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.0773	0.0473	0.0473	0.0418	0.0418
HCL1	0.0649	0.0621	0.0621	0.0324	0.0324
HCL2	0.0649	0.0564	0.0518	0.0458	0.0399
HAL	0.0678	0.0676	0.0587	0.0558	0.0418
HAL1	0.0704	0.0704	0.0509	0.0458	0.0458
HAL2	0.0837	0.0627	0.0607	0.0565	0.0546
K-Means	0.0846	0.0695	0.0611	0.0572	0.0530

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.0913	0.0633	0.0633	0.0620	0.0491
HCL1	0.0768	0.0742	0.0685	0.0566	0.0563
HCL2	0.0772	0.0751	0.0663	0.0566	0.0559
HAL	0.0935	0.0716	0.0647	0.0594	0.0594
HAL1	0.0768	0.0742	0.0689	0.0636	0.0603
HAL2	0.0797	0.0750	0.0658	0.0522	0.0522
K-Means	0.0855	0.0792	0.0661	0.0584	0.0589

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.0703	0.0703	0.0457	0.0457	0.0457
HCL1	0.0957	0.0947	0.0592	0.0592	0.0592
HCL2	0.0916	0.0916	0.0743	0.0530	0.0530
HAL	0.0903	0.0865	0.0602	0.0602	0.0602
HAL1	0.0979	0.0767	0.0582	0.0582	0.0582
HAL2	0.0767	0.0767	0.0767	0.0726	0.0687
K-Means	0.1069	0.0929	0.0806	0.0781	0.0732

Table 22. Minimum Separation, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.2207	0.2439	0.2193	0.2114	0.1993
HCL1	0.2340	0.2425	0.2175	0.2118	0.2185
HCL2	0.2582	0.2168	0.2341	0.2231	0.2370
HAL	0.2314	0.2350	0.2137	0.2054	0.1980
HAL1	0.2555	0.2152	0.1957	0.2085	0.2101
HAL2	0.2097	0.2180	0.2303	0.2083	0.2030
K-Means	0.1983	0.1976	0.1972	0.1941	0.1929

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.2772	0.2787	0.2888	0.2602	0.2858
HCL1	0.3292	0.2753	0.2762	0.2952	0.3082
HCL2	0.3305	0.2764	0.2756	0.2947	0.3072
HAL	0.2662	0.2906	0.2603	0.2684	0.2917
HAL1	0.3394	0.2848	0.2930	0.2628	0.2853
HAL2	0.2728	0.3060	0.2753	0.2564	0.2734
K-Means	0.2152	0.2225	0.2165	0.2136	0.2211

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.3308	0.2796	0.2487	0.2366	0.2294
HCL1	0.3030	0.2589	0.2338	0.2304	0.2281
HCL2	0.2975	0.2541	0.2675	0.2462	0.2305
HAL	0.2969	0.2520	0.2825	0.2924	0.2882
HAL1	0.2644	0.2894	0.2797	0.2557	0.2622
HAL2	0.2965	0.2942	0.2632	0.2480	0.2674
K-Means	0.2522	0.2392	0.2363	0.2365	0.2266

Table 23. Separation, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.1268	0.1628	0.1503	0.1502	0.1441
HCL1	0.1400	0.1570	0.1440	0.1487	0.1614
HCL2	0.1635	0.1423	0.1670	0.1607	0.1795
HAL	0.1343	0.1529	0.1409	0.1396	0.1379
HAL1	0.1564	0.1373	0.1280	0.1490	0.1544
HAL2	0.1169	0.1409	0.1608	0.1471	0.1461
K-Means	0.1177	0.1272	0.1340	0.1373	0.1394

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.1628	0.1820	0.2042	0.1857	0.2183
HCL1	0.2101	0.1813	0.1955	0.2253	0.2450
HCL2	0.2087	0.1811	0.1948	0.2251	0.2443
HAL	0.1601	0.1969	0.1797	0.1959	0.2246
HAL1	0.2160	0.1876	0.2075	0.1870	0.2162
HAL2	0.1655	0.2115	0.1935	0.1830	0.2060
K-Means	0.1283	0.1444	0.1473	0.1488	0.1594

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.2136	0.1854	0.1687	0.1648	0.1611
HCL1	0.1861	0.1619	0.1519	0.1538	0.1562
HCL2	0.1771	0.1574	0.1791	0.1669	0.1567
HAL	0.1800	0.1575	0.1990	0.2095	0.2148
HAL1	0.1520	0.1878	0.1937	0.1776	0.1847
HAL2	0.1718	0.1898	0.1743	0.1689	0.1907
K-Means	0.1497	0.1534	0.1594	0.1635	0.1583

Table 24. Average Split, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.4114	0.4213	0.3372	0.3423	0.3600
HCL1	0.4609	0.4091	0.3382	0.3390	0.3297
HCL2	0.4994	0.3814	0.3639	0.3143	0.3037
HAL	0.4379	0.3831	0.3165	0.2931	0.3160
HAL1	0.4751	0.3666	0.2901	0.2898	0.3056
HAL2	0.3959	0.3904	0.3363	0.2955	0.2785
K-Means	0.3746	0.3437	0.3114	0.2884	0.2652

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.4672	0.4169	0.3699	0.3264	0.3057
HCL1	0.5392	0.4353	0.3812	0.3748	0.3406
HCL2	0.5398	0.4337	0.3796	0.3722	0.3391
HAL	0.4526	0.4393	0.3596	0.3286	0.3294
HAL1	0.5491	0.4466	0.3950	0.3420	0.3304
HAL2	0.4571	0.4597	0.3805	0.3200	0.2959
K-Means	0.3714	0.3431	0.3063	0.2715	0.2672

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.4904	0.3647	0.3689	0.3359	0.2884
HCL1	0.4483	0.3484	0.3329	0.3211	0.2974
HCL2	0.4367	0.3374	0.3102	0.2872	0.2514
HAL	0.4432	0.3305	0.3796	0.3189	0.2996
HAL1	0.3941	0.3805	0.3782	0.3439	0.3126
HAL2	0.4454	0.3815	0.3217	0.2749	0.2400
K-Means	0.3813	0.3270	0.2894	0.2553	0.2289

Table 25. Average Silhouette Width, Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.5836	0.5231	0.5083	0.4167	0.4330
HCL1	0.5860	0.5392	0.5594	0.4705	0.4886
HCL2	0.6414	0.5193	0.5233	0.5418	0.4169
HAL	0.6051	0.4431	0.4850	0.4148	0.3855
HAL1	0.5644	0.5444	0.4334	0.5428	0.5291
HAL2	0.5926	0.5091	0.4499	0.5304	0.5106
K-Means	0.5822	0.5165	0.5232	0.5129	0.4986

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.6016	0.3948	0.4948	0.4609	0.4865
HCL1	0.5975	0.4927	0.5322	0.4032	0.4243
HCL2	0.5420	0.4941	0.5168	0.3888	0.3888
HAL	0.5950	0.5157	0.4744	0.4535	0.4528
HAL1	0.5708	0.5431	0.4805	0.4970	0.5038
HAL2	0.6089	0.5741	0.5370	0.3999	0.4222
K-Means	0.4203	0.3739	0.3376	0.3332	0.3784

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.6168	0.6122	0.4766	0.4766	0.4139
HCL1	0.5683	0.5343	0.3786	0.3786	0.3786
HCL2	0.5700	0.6111	0.4942	0.4738	0.4412
HAL	0.6208	0.5516	0.5263	0.4488	0.5233
HAL1	0.6167	0.5367	0.4580	0.5171	0.4533
HAL2	0.5262	0.5079	0.5523	0.4554	0.4522
K-Means	0.5952	0.5375	0.5002	0.5012	0.4695

Table 26. Modified Dunn's Index (Average Linkage), Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	1.3161	1.0629	1.2356	0.7668	0.7967
HCL1	1.2014	1.1873	1.2310	0.8122	0.8434
HCL2	1.3531	1.2175	1.1131	1.1854	0.7548
HAL	1.4213	0.9701	1.1509	0.8115	0.6662
HAL1	1.3456	1.3126	0.9650	1.1256	0.9164
HAL2	1.5575	1.0776	1.0067	1.1854	1.0397
K-Means	1.4773	1.1180	1.1339	1.1865	1.1253

Slice Number 16					
Alg	3	4	5	6	7
HCL	1.4436	0.8040	1.0120	1.0841	1.1802
HCL1	1.3588	1.0114	1.3114	0.6886	0.7360
HCL2	1.3223	1.0114	1.2016	0.6322	0.6322
HAL	1.5045	1.0456	1.2180	1.1166	0.8165
HAL1	1.3451	1.0417	1.2264	1.1921	1.0674
HAL2	1.4492	1.0759	1.2329	0.8333	0.8800
K-Means	1.0269	0.7998	0.8016	0.7741	0.8328

Slice Number 27					
Alg	3	4	5	6	7
HCL	1.1874	1.2712	0.8946	0.8946	0.7607
HCL1	1.3542	1.2529	0.7049	0.7049	0.7049
HCL2	1.3744	1.4406	1.0357	0.9960	0.7490
HAL	1.4224	1.2376	0.8370	0.8522	1.0129
HAL1	1.4263	1.0610	0.8966	0.9343	0.9422
HAL2	1.0649	1.2124	1.2302	0.9125	0.9360
K-Means	1.4629	1.1477	1.0034	1.0229	0.9393

Table 27. Modified Dunn's Index (Complete Linkage), Subject:3246 Data:TACs

Slice Number 4					
Alg	3	4	5	6	7
HCL	0.5532	0.4976	0.4626	0.3966	0.4121
HCL1	0.5642	0.4934	0.5068	0.4547	0.4619
HCL2	0.6222	0.4808	0.4905	0.4963	0.3828
HAL	0.5771	0.4153	0.4366	0.3696	0.3665
HAL1	0.5394	0.5043	0.3909	0.4972	0.4915
HAL2	0.5553	0.4785	0.4075	0.4650	0.4564
K-Means	0.5462	0.4726	0.4644	0.4469	0.4291

Slice Number 16					
Alg	3	4	5	6	7
HCL	0.5792	0.3714	0.4544	0.4141	0.4415
HCL1	0.5777	0.4621	0.4868	0.3733	0.3720
HCL2	0.5237	0.4636	0.4761	0.3597	0.3410
HAL	0.5723	0.4954	0.4327	0.4019	0.4019
HAL1	0.5519	0.5132	0.4330	0.4430	0.4570
HAL2	0.5862	0.5555	0.4962	0.3486	0.3672
K-Means	0.3952	0.3484	0.3006	0.2928	0.3276

Slice Number 27					
Alg	3	4	5	6	7
HCL	0.5962	0.5619	0.4481	0.4373	0.3855
HCL1	0.5419	0.4875	0.3566	0.3566	0.3511
HCL2	0.5400	0.5545	0.4401	0.4363	0.3695
HAL	0.5898	0.4983	0.4983	0.3751	0.4493
HAL1	0.5766	0.4983	0.4394	0.4554	0.3730
HAL2	0.5004	0.4647	0.4930	0.4007	0.3748
K-Means	0.5527	0.4822	0.4359	0.4344	0.3938

Table 28. Modified Dunn's Index (Combined Average), Subject:3246 Data:TACs

*Appendix III : Optimal Number of Clusters obtained from Intra-Inter
Cluster Measures*

Slice Number 4								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(3,4)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(7,3)	(3,3)
Dunn's Index (4.11)	(4,3)	(3,3)	(3,3)	(4,3)	(3,3)	(3,3)	(3,3)	(3,3)
Dunn's Index (4.12)	(4,3)	(3,5)	(3,3)	(4,3)	(3,3)	(3,3)	(3,3)	(3,3)
Dunn's Index (4.13)	(4,3)	(3,3)	(3,3)	(4,3)	(3,3)	(3,3)	(3,3)	(3,3)

Slice Number 16								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(4,3)	(3,3)	(3,3)	(4,3)	(3,3)	(3,4)	(5,3)	(3,3)
Dunn's Index (4.11)	(4,3)	(5,3)	(3,3)	(4,3)	(5,3)	(3,3)	(3,3)	(-,3)
Dunn's Index (4.12)	(3,3)	(5,3)	(3,3)	(3,3)	(5,3)	(3,3)	(3,3)	(3,3)
Dunn's Index (4.13)	(4,3)	(5,3)	(3,3)	(4,3)	(5,3)	(3,3)	(3,3)	(-,3)

Slice Number 27								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(3,3)	(3,3)	(4,3)	(3,3)	(3,3)	(4,3)	(3,3)	(3,3)
Dunn's Index (4.11)	(5,3)	(7,3)	(3,4)	(5,3)	(7,3)	(3,5)	(3,3)	(-,3)
Dunn's Index (4.12)	(5,4)	(4,3)	(3,4)	(5,3)	(4,3)	(3,5)	(3,3)	(-,3)
Dunn's Index (4.13)	(5,3)	(4,3)	(3,4)	(5,3)	(4,3)	(3,3)	(3,3)	(-,3)

Table 29. Optimal Cluster groups obtained for subject: 3246 using 4 different inter-intra measures. (a,b) refers to the optimal cluster groups obtained from Integrals and TAC data respectively. Consensus result is shown in the last column

Slice Number 4								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(3,3)	(4,4)	(3,3)
Dunn's Index (4.11)	(4,4)	(3,3)	(3,4)	(4,3)	(3,3)	(3,3)	(4,4)	(3,3)
Dunn's Index (4.12)	(3,3)	(3,3)	(3,4)	(3,3)	(3,3)	(3,3)	(4,4)	(3,3)
Dunn's Index (4.13)	(4,4)	(3,3)	(3,4)	(4,3)	(3,3)	(3,3)	(4,4)	(-,3)

Slice Number 16								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(3,3)	(4,3)	(4,3)	(3,3)	(4,3)	(4,3)	(3,3)	(4,3)
Dunn's Index (4.11)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(3,3)	(4,3)
Dunn's Index (4.12)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(3,3)	(4,3)
Dunn's Index (4.13)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	(3,3)	(4,3)

Slice Number 27								
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means	Consensus
Silhouette (4.9)	(4,3)	(4,4)	(3,3)	(4,3)	(4,3)	(3,3)	(3,3)	(4,3)
Dunn's Index (4.11)	(5,3)	(3,3)	(5,4)	(5,4)	(3,4)	(5,4)	(5,5)	(5,4)
Dunn's Index (4.12)	(5,3)	(5,3)	(5,4)	(5,4)	(5,4)	(5,4)	(3,5)	(5,4)
Dunn's Index (4.13)	(5,3)	(3,3)	(5,4)	(5,4)	(3,4)	(5,4)	(5,3)	(5,4)

Table 30. Optimal Cluster groups obtained for subject: 4248 using 4 different inter-intra measures. (a,b) refers to the optimal cluster groups obtained from Integrals and TAC data respectively. Consensus result is shown in the last column

Consensus Occurrences per algorithm for all measures							
	HCL	HCL1	HCL2	HAL	HAL1	HAL2	K-Means
Consensus Occurrences	(11,17)	(15,19)	(16,18)	(12,24)	(16,24)	(17,21)	(8,17)

Table 31. Number of consensus value occurrences over all measures. (A,B) refers to the consensus matches obtained from 24 individual values for Integrals and TAC data respectively.