

Lesson 8: Geographic Segmentation

Spatially Constrained Clustering Techniques

Dr. Kam Tin Seong
Assoc. Professor of Information Systems

**School of Information Systems,
Singapore Management University**

2020-5-16 (updated: 2021-09-18)

Content

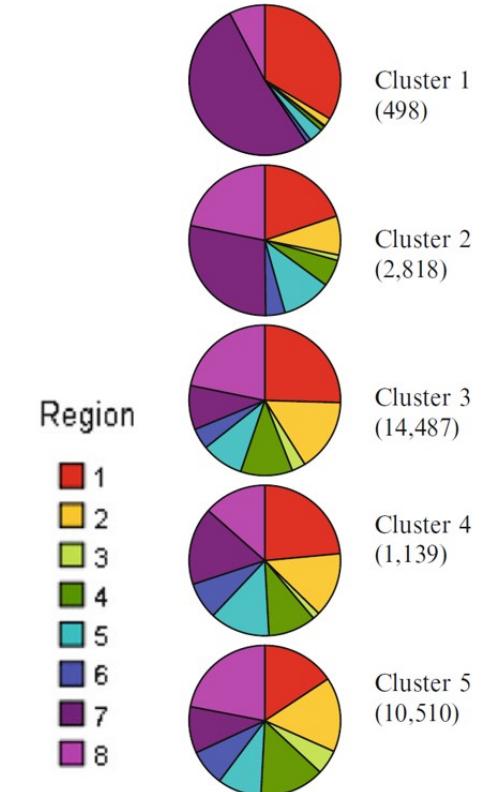
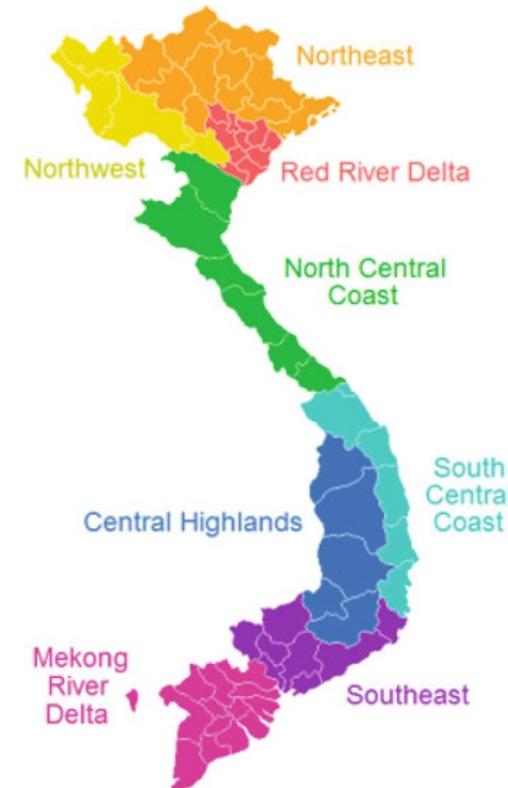
- What is Geographic Segmentation?
- Introducing Cluster Analysis
- Hierarchical Class Analysis
- Cluster Analysis Process
- Spatially Constrained Clustering Techniques

What is Geographical Segmentation?

- Geographic segmentation divides a target market by location so marketers can better serve customers in a particular area.
- This type of market segmentation is based on the geographic units themselves (countries, states, cities, etc.), but also on various geographic factors, such as climate, cultural preferences, populations, and more.

For business applications of geographic segmentation, refer to

<https://manychat.com/blog/geographic-segmentation/>

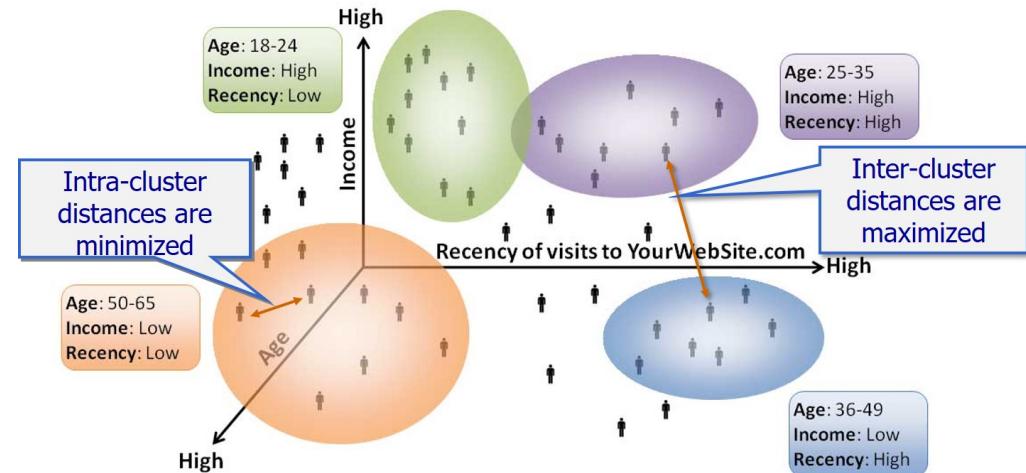


Advantages of Geographic Segmentation

- It's an effective approach for companies with large national or international markets because different consumers in different regions have different needs, wants, and cultural characteristics that can be specifically targeted.
- It can also be an effective approach for small businesses with limited budgets. They can focus on their defined area and not expend needless marketing dollars on approaches ill-suited for their target geographic segment.
- It works well in different areas of population density. Consumers in an urban environment often have different needs and wants than people in suburban and rural environments. There are even cultural differences between these three areas.

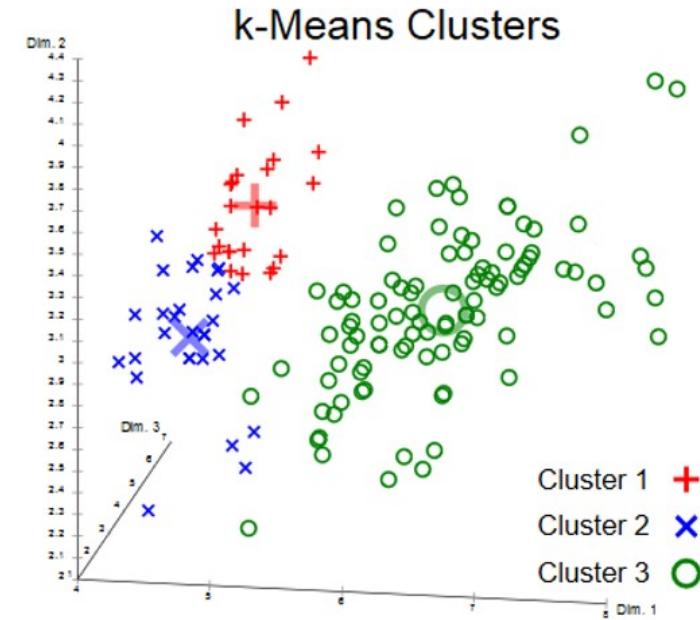
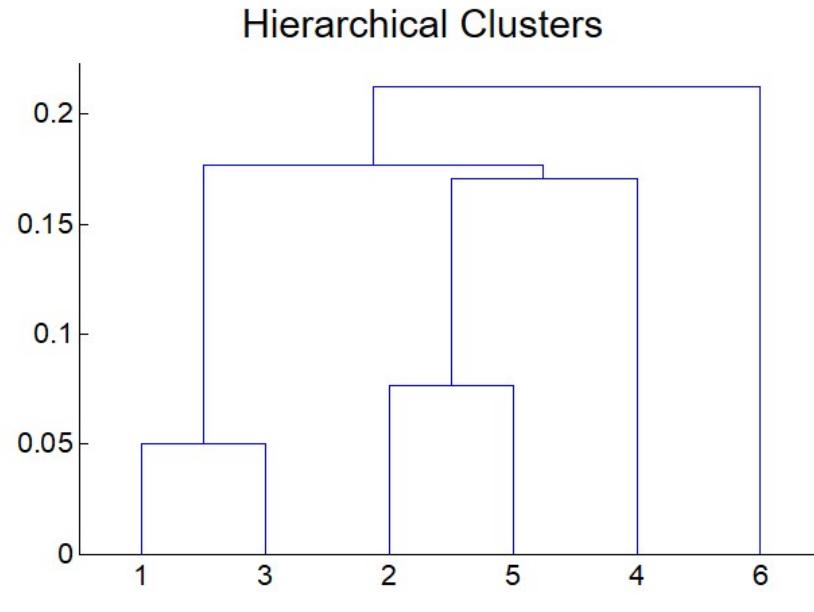
Cluster Analysis

- **Cluster analysis** or **Clustering** is the task of grouping a set of objects in such a way that objects in the same group (called cluster) are more similar (in some sense or another) to each other than to those in another group (clusters).
- In modern machine learning age, it is belong to the family of exploratory data mining.
- It has been used in many fields including Machine Learning, Pattern Recognition, Image Analysis, Information Retrieval, Bioinformatics, Data Compression, and Computer Graphics.



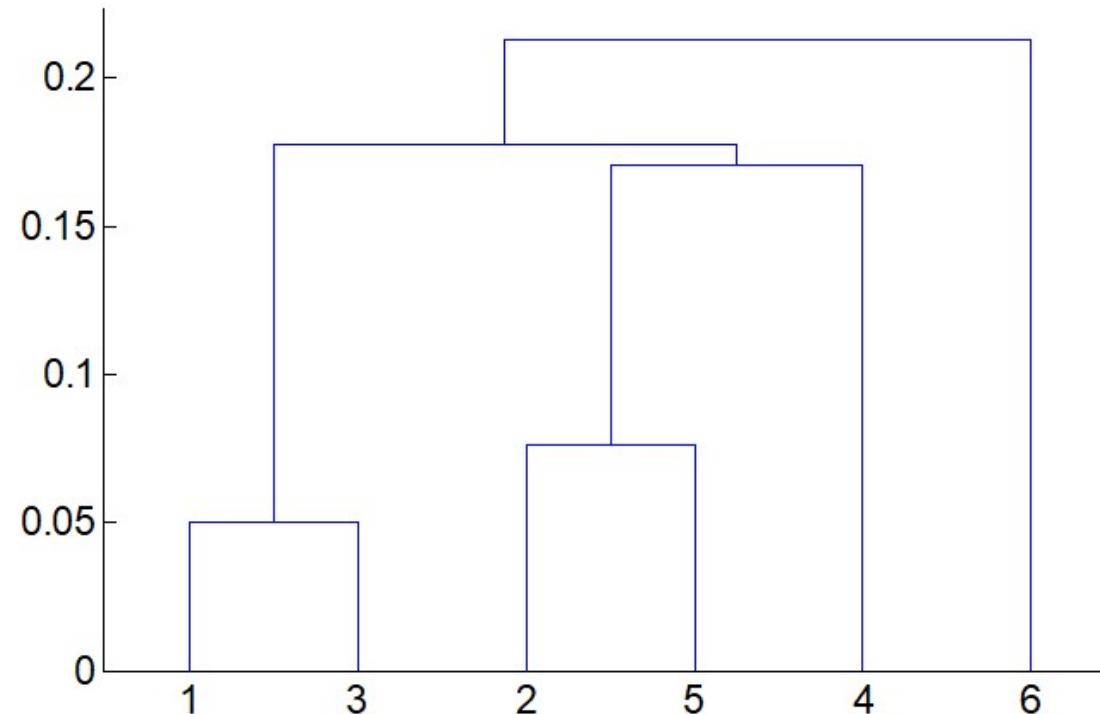
Typology of Cluster Analysis Techniques

- Hierarchical clustering
 - A set of nested clusters organized as a hierarchical tree.
- Partitioning clustering (also known as k-means)
 - A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset.



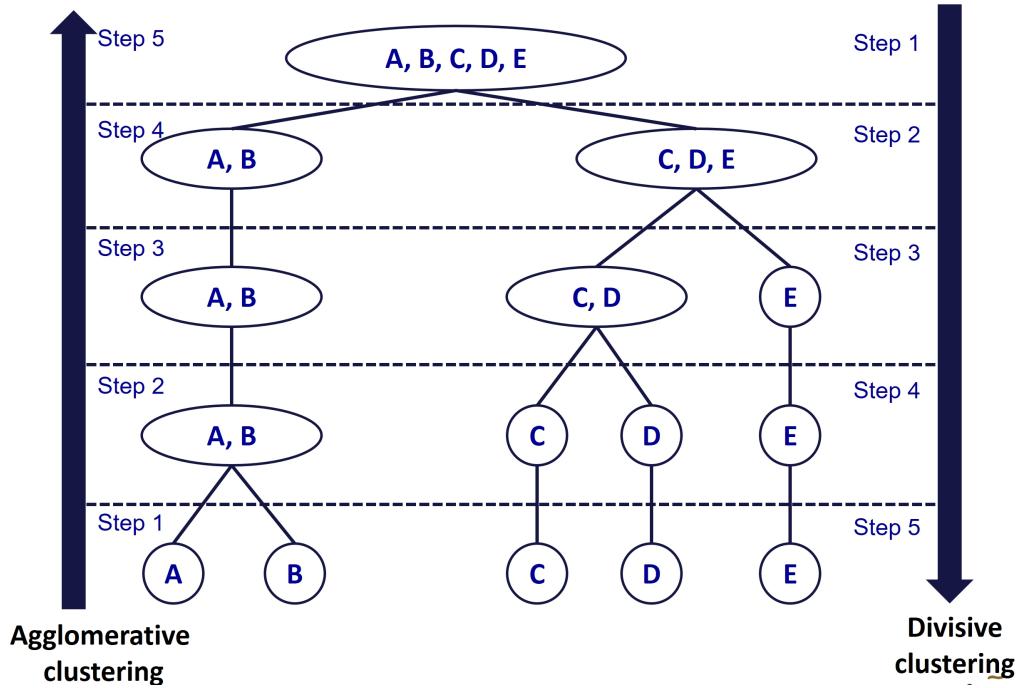
Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree.



Hierarchical Methods

- Agglomerative clustering: It's also known as **AGNES (Agglomerative Nesting)**. It works in a bottom-up manner. That is, each object is initially considered as a single-element cluster (leaf). At each step of the algorithm, the two clusters that are the most similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are member of just one single big cluster (root). The result is a tree which can be plotted as a dendrogram.
- Divisive hierarchical clustering: It's also known as **DIANA (Divise Analysis)** and it works in a top-down manner. The algorithm is an inverse order of AGNES. It begins with the root, in which all objects are included in a single cluster. At each step of iteration, the most heterogeneous cluster is divided into two. The process is iterated until all objects are in their own cluster.

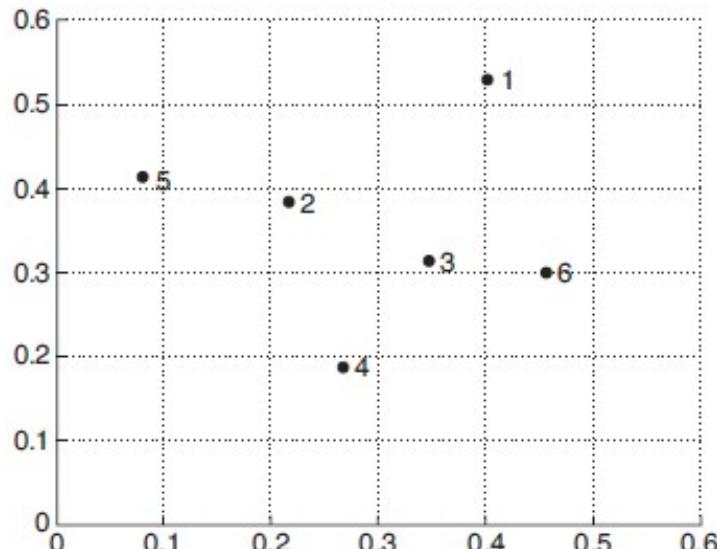


Basic Agglomerative Hierarchical Clustering Algorithm

- Basic algorithm is straightforward
1. *Compute the proximity matrix*
 2. *Let each data point be a cluster*
 3. ***Repeat***
 4. *Merge the two closest clusters*
 5. *Update the proximity matrix*
 6. ***Until*** only a single cluster remains

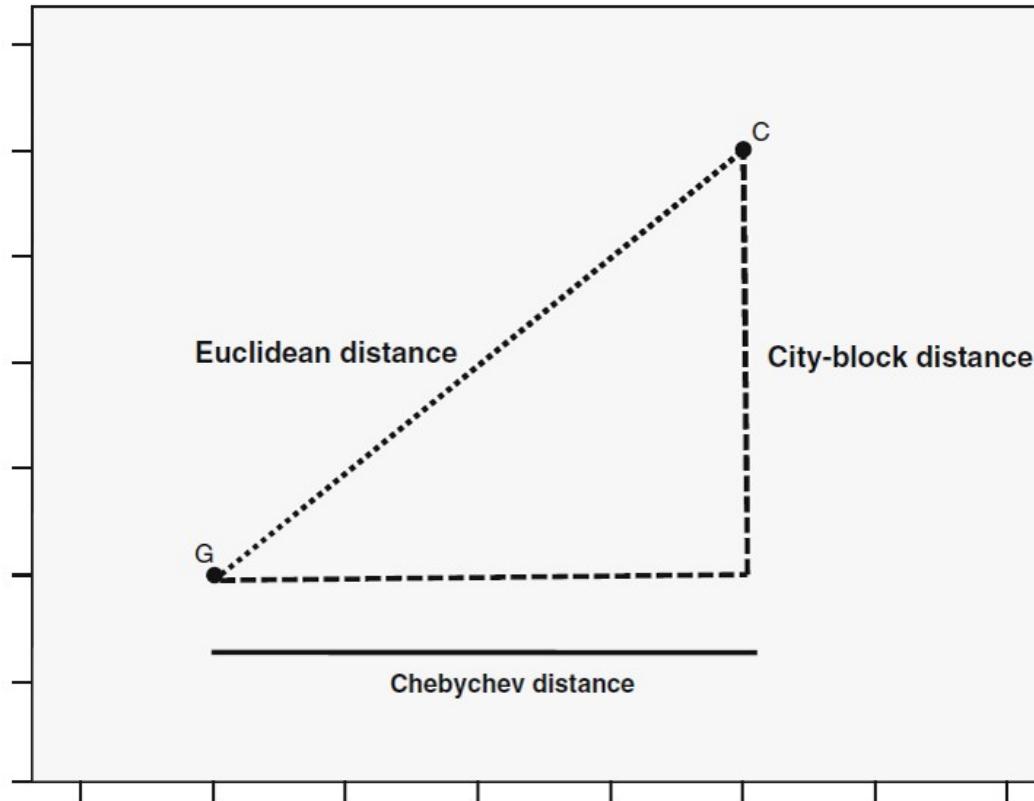
What is Proximity Matrix?

- Measures of Similarity or Dissimilarity.



Point	x Coordinate	y Coordinate
p1	0.40	0.53
p2	0.22	0.38
p3	0.35	0.32
p4	0.26	0.19
p5	0.08	0.41
p6	0.45	0.30

Three commonly used methods to calculate proximity matrix



Proximity matrix: Euclidean distance

- Euclidean distance formula:

$$D_{euclidean}(p_1, p_2) = \sqrt{(x_{p1} - x_{p2})^2 + (y_{p1} - y_{p2})^2}$$

- Proximity matrix of Euclidean distance

Point	x Coordinate	y Coordinate
p1	0.40	0.53
p2	0.22	0.38
p3	0.35	0.32
p4	0.26	0.19
p5	0.08	0.41
p6	0.45	0.30

	p1	p2	p3	p4	p5	p6
p1		0.23	0.22	0.37	0.34	0.24
p2	0.23		0.14	0.19	0.14	0.24
p3	0.22	0.14		0.16	0.28	0.10
p4	0.37	0.19	0.16		0.28	0.22
p5	0.34	0.14	0.28	0.28		0.39
p6	0.24	0.24	0.10	0.22	0.39	

Proximity matrix: City-block distance

- City-clock formula:

$$D_{City-block}(p_1, p_2) = |x_{p1} - x_{p2}| + |y_{p1} - y_{p2}|$$

- Proximity matrix of city-block distance

Point	x Coordinate	y Coordinate
p1	0.40	0.53
p2	0.22	0.38
p3	0.35	0.32
p4	0.26	0.19
p5	0.08	0.41
p6	0.45	0.30

	p1	p2	p3	p4	p5	p6
p1		0.33	0.26	0.48	0.44	0.28
p2	0.33		0.19	0.23	0.17	0.31
p3	0.26	0.19		0.22	0.36	0.12
p4	0.48	0.23	0.22		0.40	0.30
p5	0.44	0.17	0.36	0.40		0.48
p6	0.28	0.31	0.12	0.30	0.48	

Proximity matrix: Chebychev distance

- Chebychev distance formula:

$$D_{Chebychev}(p_1, p_2) = \max(|x_{p1} - x_{p2}|, |y_{p1} - y_{p2}|)$$

- Proximity matrix of Chebychev distance

Point	x Coordinate	y Coordinate
p1	0.40	0.53
p2	0.22	0.38
p3	0.35	0.32
p4	0.26	0.19
p5	0.08	0.41
p6	0.45	0.30

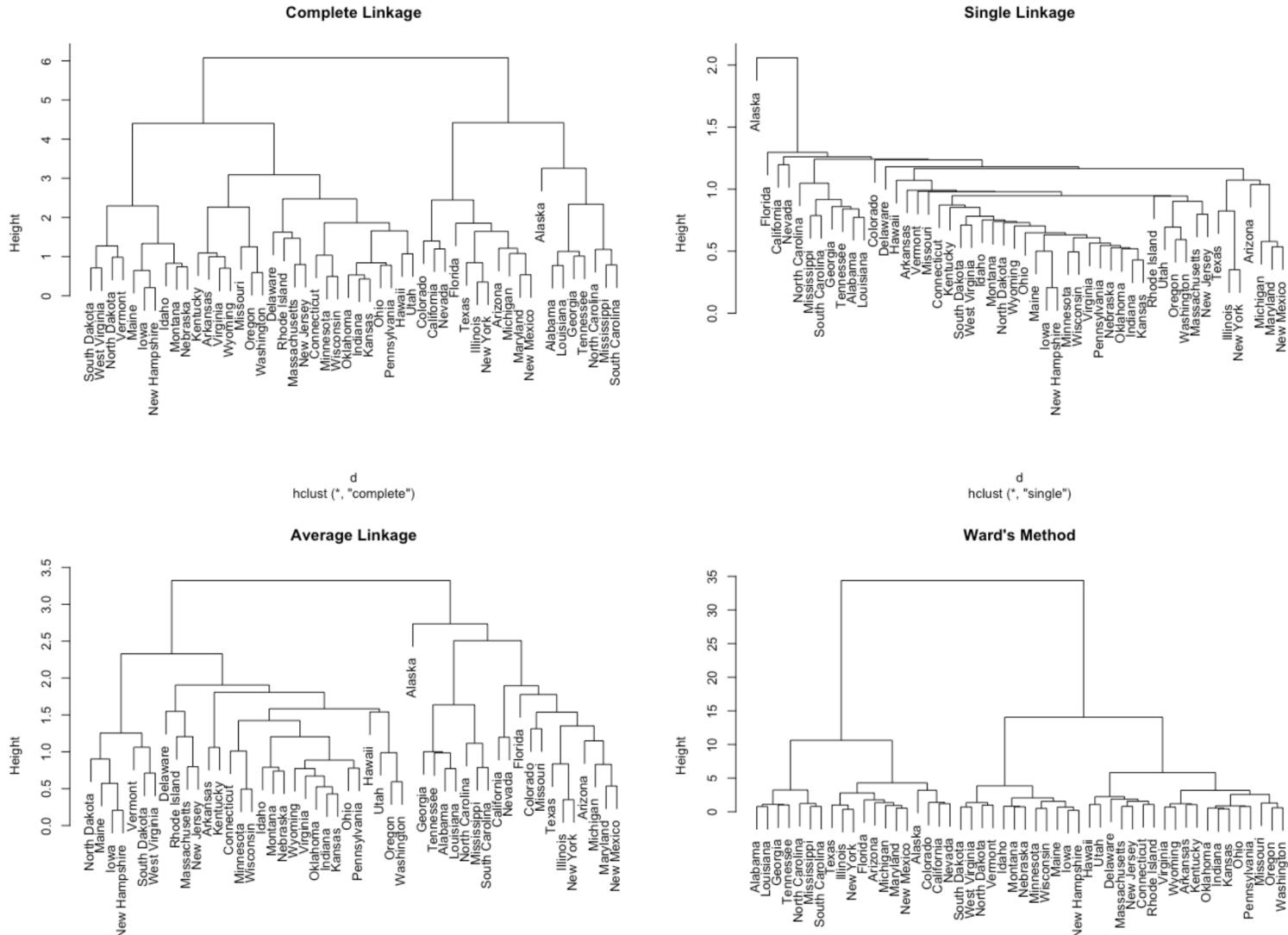
	p1	p2	p3	p4	p5	p6
p1		0.18	0.21	0.34	0.32	0.23
p2	0.18		0.13	0.19	0.14	0.23
p3	0.21	0.13		0.13	0.27	0.10
p4	0.34	0.19	0.13		0.22	0.19
p5	0.32	0.14	0.27	0.22		0.37
p6	0.23	0.23	0.10	0.19	0.37	

Agglomerative Hierarchical Clustering Algorithms

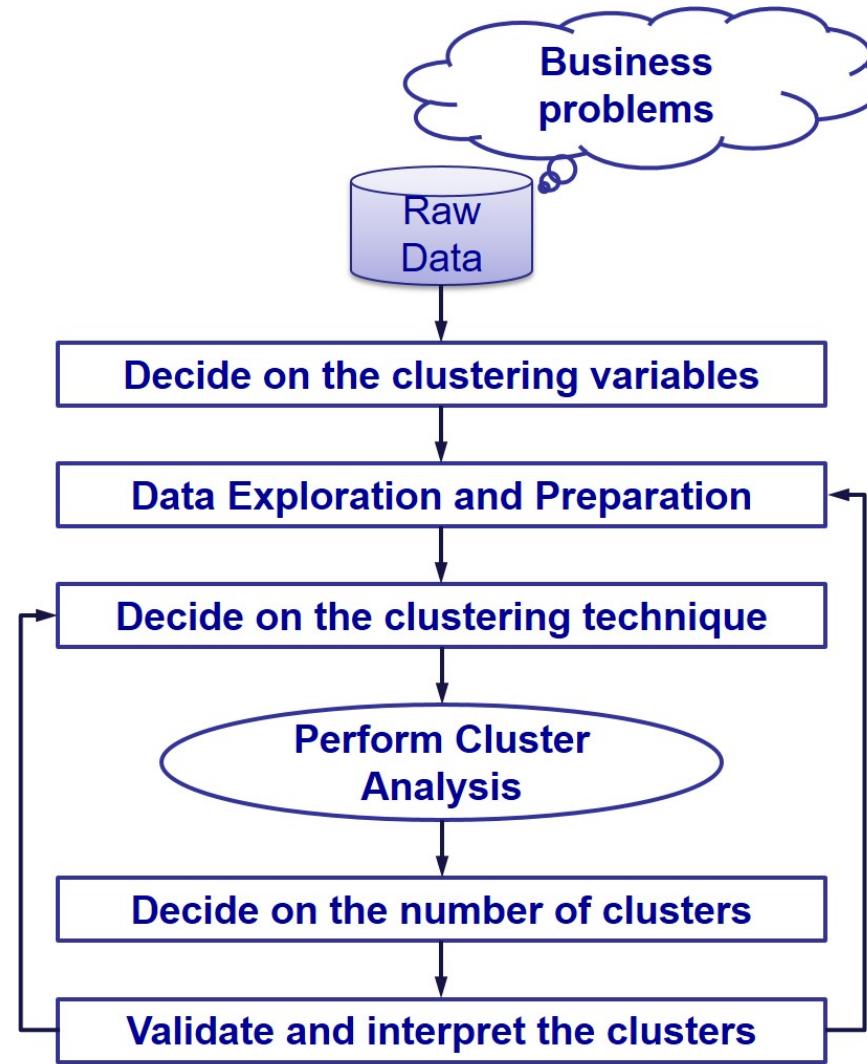
The most common types methods are:

- **Maximum** or **complete linkage** clustering: It computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2, and considers the largest value (i.e., maximum value) of these dissimilarities as the distance between the two clusters. It tends to produce more compact clusters.
- **Minimum** or **single linkage** clustering: It computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2, and considers the smallest of these dissimilarities as a linkage criterion. It tends to produce long, “loose” clusters.
- **Mean** or **average linkage** clustering: It computes all pairwise dissimilarities between the elements in cluster 1 and the elements in cluster 2, and considers the average of these dissimilarities as the distance between the two clusters.
- **Centroid** linkage clustering: It computes the dissimilarity between the centroid for cluster 1 (a mean vector of length p variables) and the centroid for cluster 2.
- **Ward's minimum variance** method: It minimizes the total within-cluster variance. At each step the pair of clusters with minimum between-cluster distance are merged.

Agglomerative Hierarchical Clustering Algorithms - Dendograms



Cluster Analysis Process



Data Preparation

To perform a cluster analysis in R, generally, the data should be prepared as follows:

- Rows are observations (individuals) and columns are variables.
- Input variables must be inline with the segmentation task.
- Ideally, the input variables must be in continuous data type.
- Any missing value in the data must be removed or estimated.

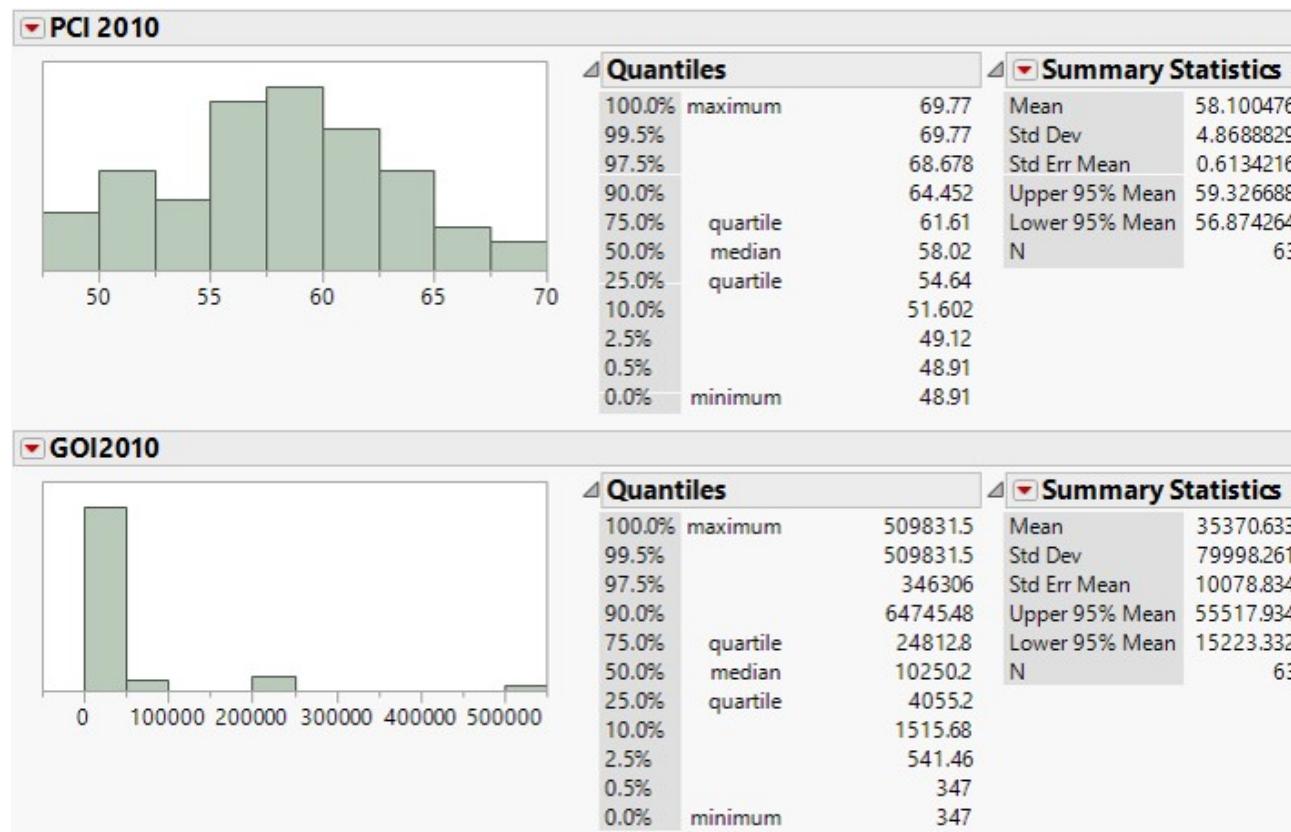
Avoid “Garbage-In, Garbage-Out”



Univariate EDA

Checking the distribution of the cluster variables,

- if their data ranges differences are very large.



Variable standardisation techniques

- Z-score

$$Z(x_{ij}) = \frac{x_{ij} - \bar{x}_j}{\sigma_j}$$

- Min-Max

$$MM(x_{ij}) = \frac{x_{ij} - x_{min}}{x_{max} - x_{min}}$$

- Decimal scaling

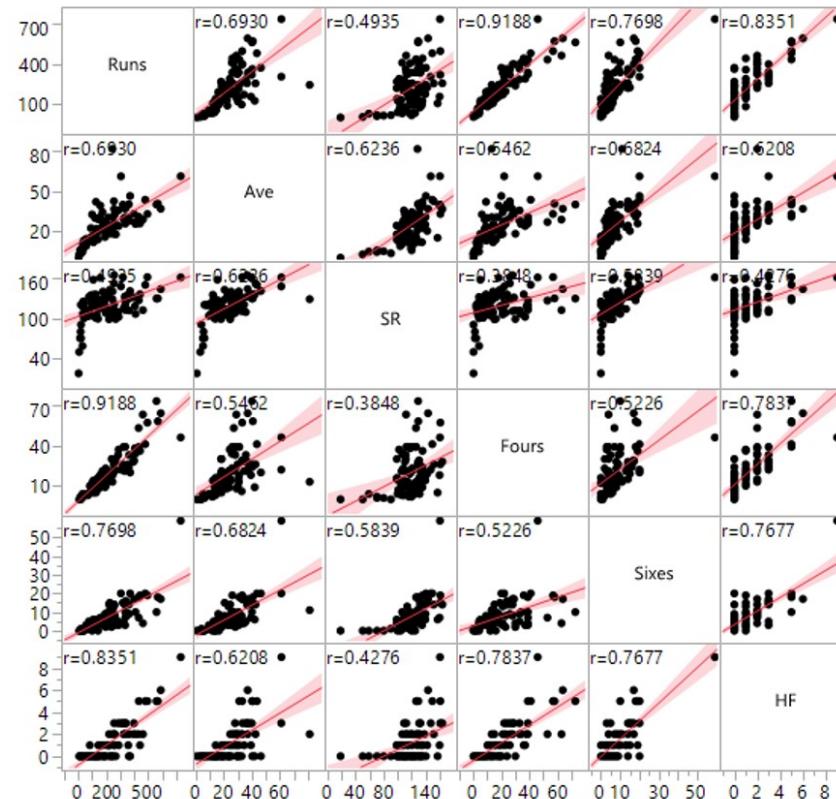
$$DS(x_{ij}) = \frac{x_{ij}}{10^c}$$

*where c is the smallest integer
such that max[|DS(x_{ij})|] < 1*

Bivariate EDA

Checking if the input variables are highly correlated (i.e. correlation coefficient ≥ 0.85).

Avoid “The Curse of Multi-collinearity”

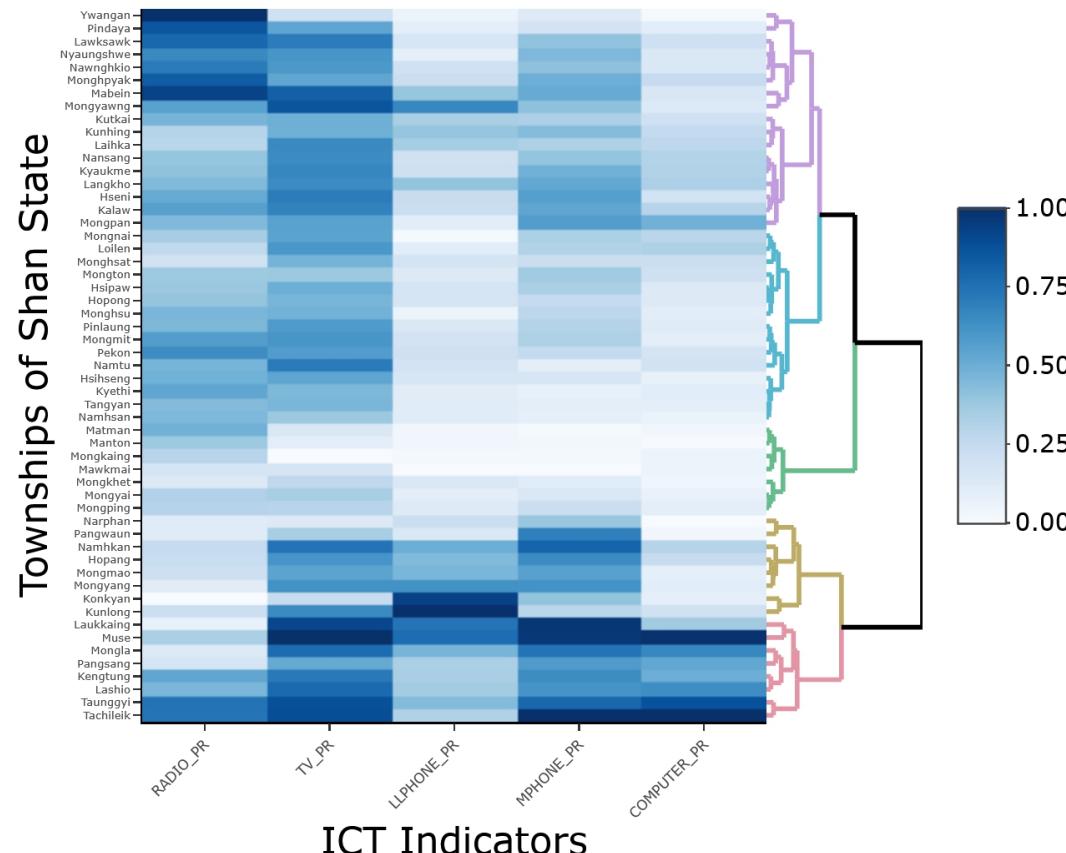


Visual interpretation of hierarchical clusters: Dendrogram

Visual interpretation of hierarchical clusters:

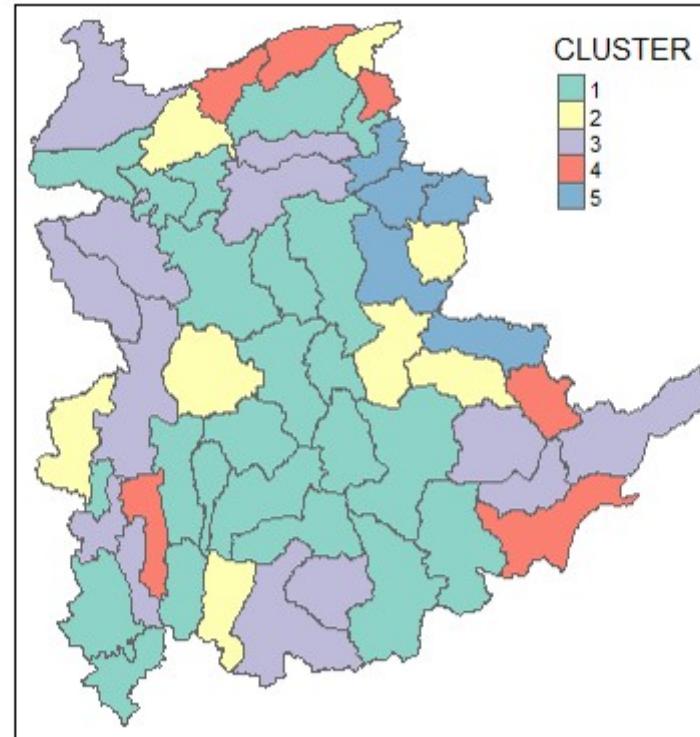
Dendrogram with heatmap

Geographic Segmentation of Shan State by ICT indicators



Limitation of non-spatial clustering algorithm

- Spatially fragmented regions.



Spatially Constrained Clustering Methods

- Grouping contiguous objects that are similar into new aggregate areal units
 - tension between attribute similarity.
- Grouping of similar observations
 - locational similarity: group spatially contiguous observations only.

Introducing SKATER method

- Spatial Kluster analysis by Tree Edge Removal Assuncao et al (2006) algorithm.
- Construct minimum spanning tree from adjacency graph.
- Prune the tree (cut edges) to achieve maximum internal homogeneity.

Reference: Assunção, R. M ; Neves, M. C ; Câmara, G ; Da Costa Freitas, C (2006) "Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees", *International Journal of Geographical Information Science*, Vol.20 (7), p.797-811 (<https://web-a-ebscohost-com.libproxy.smu.edu.sg/ehost/detail/detail?vid=0&sid=60b437ef-2130-43e1-808f-e1933ac625a8%40sdc-v-sessmgr03&bdata=JnNpdGU9ZWhvc3QtbGI2ZS2zY29wZT1zaXRI#db=asn&AN=21895448>)

Contiguity as a Graph

- Network connectivity based on adjacency between nodes (locations).
- Edge value reflects dissimilarity between nodes.

$$d(i, j) = d(x_i, x_j) = \sum_{l=1}^n (\mathbf{x}_{il} - \mathbf{x}_{jl})^2$$

- Objective is to minimize within-group dissimilarity (maximize between-group).



Minimum Spanning Tree Algorithm (Assuncao et al 2006)

- Connectivity graph $G = (V, L)$, V vertices (nodes), L edges path
 - a sequence of nodes connected by edges v_1 to v_k : $(v_1, v_2), \dots, (v_{k-1}, v_k)$.
- Spanning tree
 - tree with n nodes of G unique path connecting any two nodes $n-1$ edges.
- Minimum spanning tree
 - spanning tree that minimizes a cost function minimize sum of dissimilarities over all nodes.

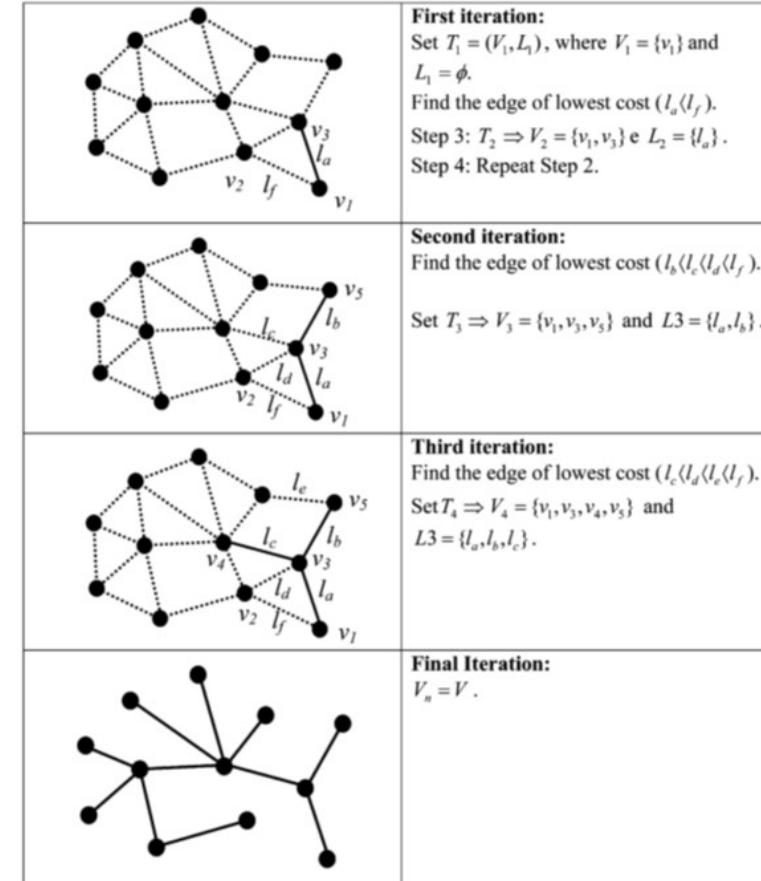


Figure 2. Construction of the minimum spanning tree.

SKATER - A heuristic for fast tree partitioning

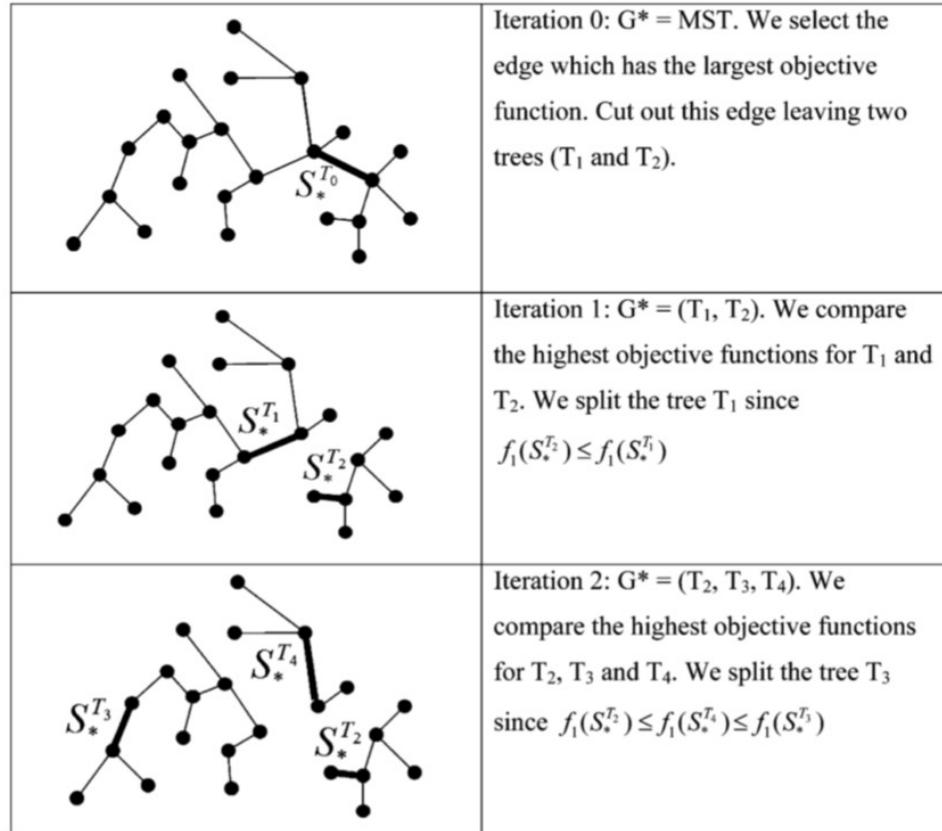
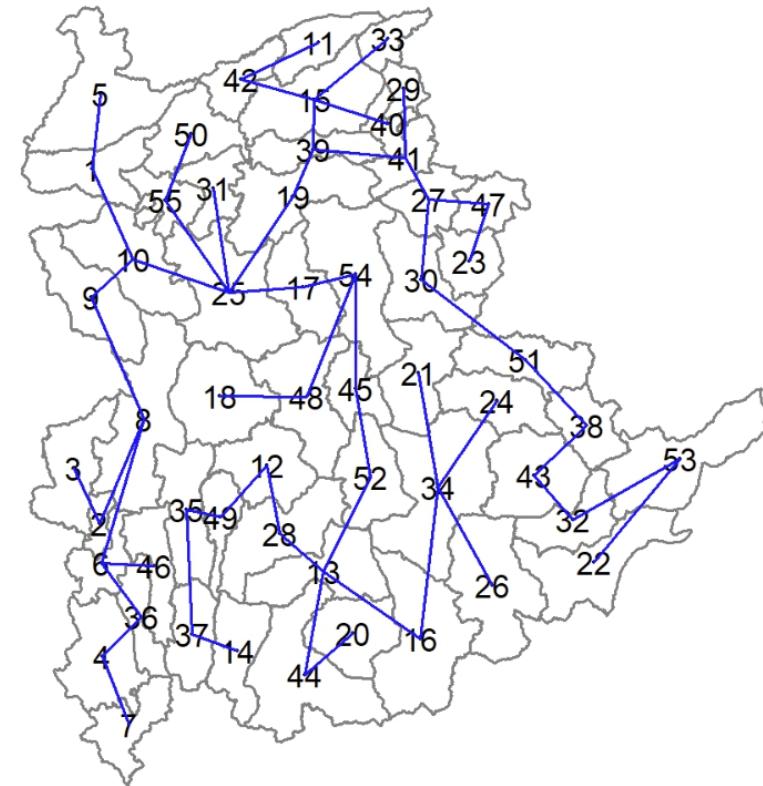


Figure 3. Partitioning of the MST.



Spatially Constrained Clustering using SKATER

