Hierarchical Clustering based on Voronoi edges: The R package HCV

Hsu Hao Yun : zoez5230100@gmail.com

September 14, 2021

Abstract

The R package `HCV` implement the hierarchical agglomerative clustering algorithm which involving the spatial homogeneity by considering the Voronoi edges from R package `alphahull`. The constraint of Voronoi edges give the spatial proximity property at each step of iterations. Bseides, we also offer several methods to find the optimal number of clusters under the restarint of spatial homogeneity.

1 Introduction

Today, analyzing spatial data is a crucial step in data mining. Be distinguished from the conventional data structure, the spatial data emphasize that if the two individuals are more proximity on geometry domain, then the spatial feature of these two individuals are more strongly related than the distant pairs.

Clustering analysis is a practical means of dividing data into several subsets, with extracting the centroids or prototypes among the subsets, then we have a clear interpretation of the diverse trends of data. The two main branches of clustering method are partitional clustering and hierarchical clustering, the former measures on the within-group sum of square (WSS) and require user specify the number of clusters in advance, the letter builds the dissimilarity matrix and incorporates two clusters if the dissimilarity between these clusters is the minimum among the lower triangular section of dissimilarity matrix. Nevertheless, the applicability of partitional clustering methods to spatial data is not clear. This paper aims at proposing a novel number of clusters selection method based on hierarchical clustering.

Another significant issue in cluster analysis is determining a proper number of clusters. The widely used methods are the internal indices and external indices, however, this traditional methods only depend on the within sum of squared matrix of non-spatial data attributes, which leak off the condition on geometry domain, hence, we propose a novel internal index call SMI (Spatial Mixture Index) for determining the optimal number of clusters, see chapter 3.2.

2 Spatial clustering techniques

This section provides a brief literature review on the spatial partition diagram and hierarchical clustering algorithm.

2.1 Voronoi diagram and Delaunay Triangulation

The Voronoi diagram is a efficient algorithm for constructing a partition diagram on geometry domain, see 1, a best property of the diagram is that we add a new point A on the diagram, and we wish the point is the nearest to point B, we only have to locate the point A to the cell which centroid is point B. Interestingly, if the point A locate on the boundary of two cells, then nearest points from point A is the two centroids of this two cells.

The Delaunay Triangulation is a dual graph of Voronoi diagram, see 2. The Triangles in the graph is constructed by three nearest points, namely, the edges on the Delaunay triangle represent the minimum cost path of the two endpoints.

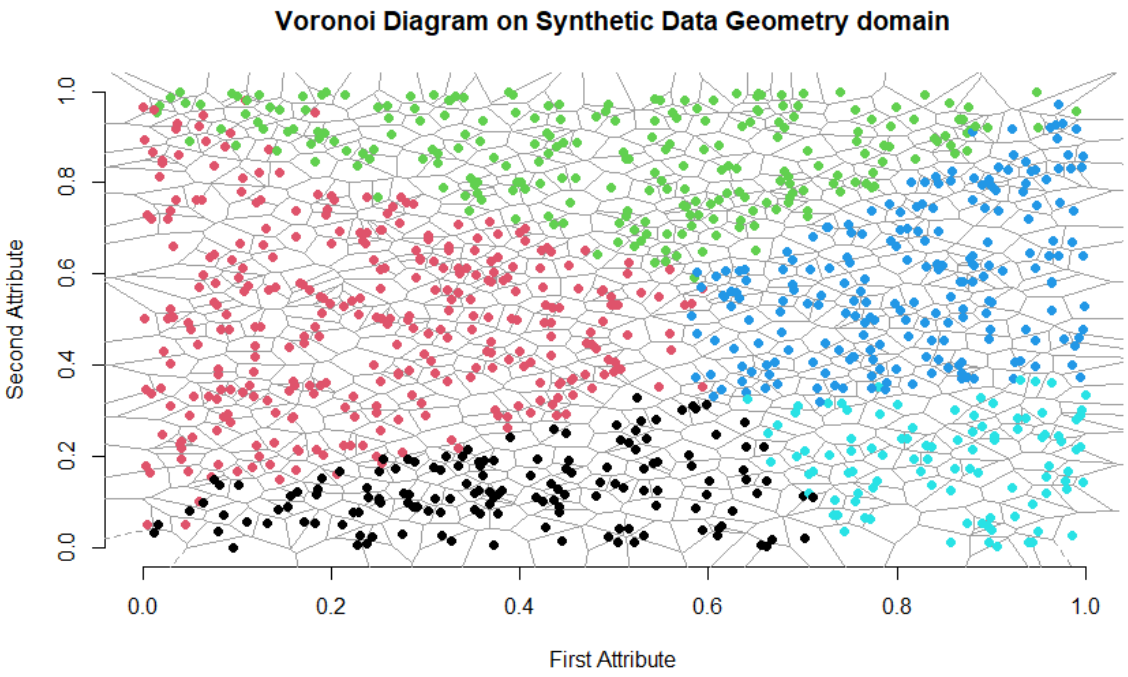


Figure 1: The Voronoi diagram on synthetic data

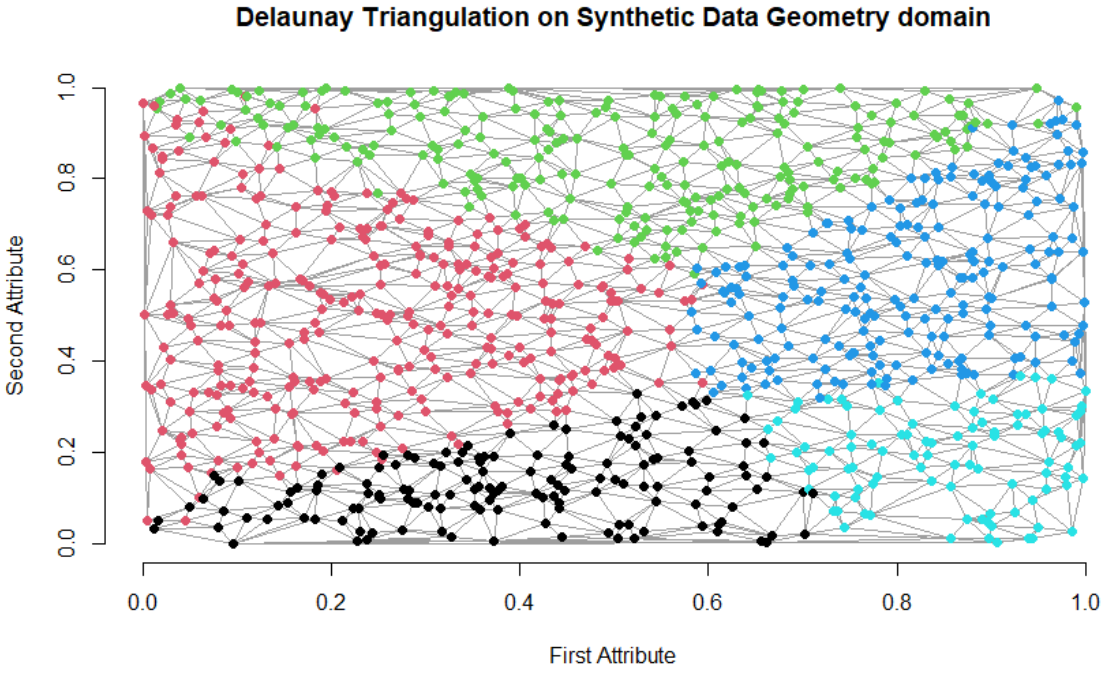


Figure 2: The Delaunay Triangulation on synthetic data

2.2 Hierarchical clustering

Hierarchical clustering algorithm is a stepwise approaching technique that iterates on the dissimilarity matrix and incorporates two clusters exactly once in each step, consequently, we have a dendrogram at

the end of the iteration. There are two commonly used hierarchical algorithms with opposite directions.

- AGNES : The algorithm works in a bottom-up manner. Initially, each individual is considered as a single cluster. Then it merges two clusters if they have the highest spatially homogeneous.
- DIANA : The algorithm works in a top-down manner. Initially, the whole data is considered as a cluster. Then it split up into two clusters if they have the highest spatially heterogeneous.

Let $x_i(s_i) = (x_{i1}, \ldots, x_{ip})'$ be a p-dimensional data located at s_i for $i = 1, \ldots, n$. Euclidean distance is often used to measure the distance or dissimilarity between a pairs of points. A general form of the measurement is Minkowski distance, see equation 1. Nevertheless, a cluster nay contain more than one single point, hence, several linkage methods for determining the distance has been proposed. The well known method single, complete, average, Ward's method are all follow the Lance-Williams formula with different linear coefficient, see equation 2, where C_k is the cluster aggregate by cluster i and cluster j.

$$d(x(s_i), x(s_j)) = (\sum_{k=1}^{p}(|x_k(s_i) - x_k(s_j)|)^r)^{\frac{1}{r}} \tag{1}$$

$$d(C_h, C_k) = \alpha_i d(C_i, C_h) + \alpha_j d(C_j, C_h) + \beta d(C_i, C_j) + \gamma |d(C_h - C_i) - d(C_h, C_j)| \tag{2}$$

2.3 Hierarchical clustering based on Voronoi edges

The idea of HCV algorithm we proposed is generally based on the common edges of the Voronoi diagram. If two points have a common edge in the Voronoi diagram, then they are considered as spatial proximity and geometry connectedness. In each iteration of quintessential hierarchical agglomerative clustering algorithm, we only focus on the groups which are connected, namely, the groups join the aggregate competition must have a common edge in Voronoi diagram, the general steps for HCV algorithm is as following.

1. Construct the Voronoi diagram on geometry domain
2. Build the dissimilarity matrix D
3. Build the adjacency matrix A according to equation 3
4. Aggregate i group and j group if $\arg \min_{i,j,A_{ij}=1} D_{ij}$
5. update dissimilarity matrix and adjacency matrix according to equation 2 and 4
6. Repeat step 3 to 5 until every points are in one single cluster

$$A_{ij} = \begin{cases} 0, & \text{if cluster } i \text{ and cluster } j \text{ have no common Voronoi edge} \\ 1, & \text{if cluster } i \text{ and cluster } j \text{ have an common Voronoi edge} \end{cases} \tag{3}$$

$$A_{hk}^{\{t\}} = \begin{cases} 0, & \text{if } A_{jk}^{\{t-1\}} = 0 \text{ and } A_{ik}^{\{t-1\}} = 0 \\ 1, & \text{if } A_{jk}^{\{t-1\}} = 1 \text{ or } A_{ik}^{\{t-1\}} = 1 \end{cases} \tag{4}$$

3 Implementation

There are two main functions in `HCV` package, `HierarchicalVoronoi` and `SMI`, the former is used to implement the HCV algorithm, the letter is used to determining the optimal number of clusters under the constraint of spatial homogeneity.

3.1 HierarchicalVoronoi

For implementing the HCV algorithm, we have to specify the constraint domain (geometry domain) which used to construct the Voronoi diagram and the optimization doamin (non geometry domain) which used to construct the dissimilarity matrix. In the function `HierarchicalVoronoi`, the data type of constraint domain and optimization domain are asked to be matrix form, the constraint domain is a $n \times 2$ matrix with n be the sample size and optimization domain is a $n \times p$ matrix with p be the number of features, if you are already have the dissimilarity matrix or adjacency matrix, you may set the parameter `diss = 'precomputed'` for input optimization domain as an $n \times n$ dissimilarity matrix and `adjacency = True` for input constraint domain as a $n \times n$ adjacency matrix. Finally, choose a linkage method for the distance between clusters, the default is `'ward.D'` method. The output of HCV algorithm is a list of objects with class `hclust`, therefore, the R base function `cutree` can be used to obtain the cluster labels of each point. The following example code is an easy way to implement the HCV algorithm.

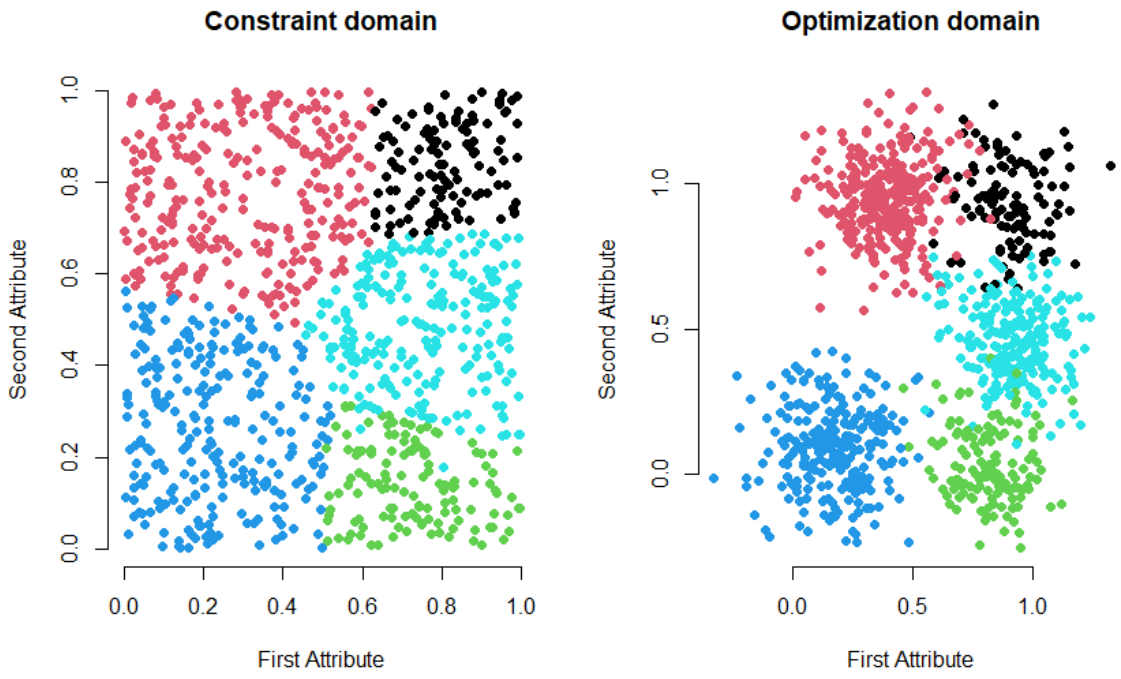
```
> # Generate 1000 points with constraint domain (geo) and optimization domain (feat)
> spatialData <- synthetic_data(5,100,0.02,1000,2)
> result <- HierarchicalVoronoi(spatialData$geo,
+                               spatialData$feat, linkage = 'ward.D')
> result
Cluster method   : ward.D
Distance         : euclidean
Number of objects: 1000

> names(result)
[1] "label_matrix" "merge"        "height"       "method"       "dist.method"
[6] "labels"       "order"

> class(result)
[1] "hclust"

> par(mfrow = c(1,2))
> plot(spatialData$geo, col = cutree(result, 5), pch = 19, main = 'Constraint domain',
+      frame = F, xlab = 'First Attribute', ylab = 'Second Attribute')
> plot(spatialData$feat, col = cutree(result, 5), pch = 19, main = 'Optimization domain',
+      frame = F, xlab = 'First Attribute', ylab = 'Second Attribute')
```

The label_matrix is the matrix which store the result of `cutree`, and result$label_matrix[i,] equivalent to `cutree(result, i)`. The other attributes is exactly the same to the attribute of class `hclust`.



3.2 SMI

SMI is an internal index which involving the property of spatial proximity by considering the edge length of Delaunay Triangulation. Let $x_i(s_i) = (x_{i1}, \ldots, x_{ip})'$ be a p-dimensional data located at s_i for $i = 1, \ldots, n$, see equation 5, where μ_k and δ_k represent the clsuter k centroids for optimization domain and constraint domain respectively, and e_k is the average Delaunay edges length in cluster k, then the definition of SMI index is define as following.

$$\text{SMI} = \frac{1}{K}\sum_{k=1}^{K} f(\alpha_k)\text{WSS}_f^k + (1 - f(\alpha_k))\text{WSS}_{geo}^k \tag{5}$$

$$\text{WSS}_{opt}^k = \sum_{x_i \in C_k} (x_i - \mu_k)'(x_i - \mu_k) \tag{6}$$

$$\text{WSS}_{con}^k = \sum_{x_i \in C_k} (s_i - \delta_k)'(s_i - \delta_k) \tag{7}$$

The definition of α_k and function f is as following, where e_k represent the average Delaunay edges length of cluster k :

$$\begin{aligned} \text{WSS}_f^k &= \sum_{x_i \in C_k} (x_i - \mu_k)'(x_i - \mu_k) \\ \text{WSS}_f &= \sum_{n=1}^{K} \text{WSS}_f^n \\ \alpha_k &= \frac{Ke_k - \sum_{j=1}^{K} e_j}{\sum_{j=1}^{K} e_j} \\ f(x) &= (1 + e^{-x})^{-1} \end{aligned} \tag{8}$$

A clearly explain of the formula can be found in my another researcher paper Determining the optimal number of clusters under the constraint of Spatial Homogeneous, the usage example code is shown below :

```
> SMI(spatialData$geo, spatialData$feat, result, 30)
$bestCluster
[1] 3

$index
 [1] 638.606660 231.535392 119.637256  74.297295  59.297407  48.714523  40.508281
 [8]  34.691950  29.736230  26.683105  24.074248  21.772144  19.968204  18.481191
[15]  16.810873  15.682021  14.289012  12.247916  11.508806  10.445373   9.909160
[22]   9.341543   8.862649   8.407029   7.911264   7.542482   7.220705   6.801251
[29]   6.478649

$ratio
 [1] 0.63743662 0.48328739 0.37897861 0.20189009 0.17847129 0.16845576 0.14358374
 [8] 0.14284928 0.10267358 0.09777189 0.09562514 0.08285540 0.07446905 0.09037935
[15] 0.06715011 0.08882842 0.14284375 0.06034577 0.09240166 0.05133500 0.05728208
[22] 0.05126495 0.05140901 0.05897029 0.04661480 0.04266201 0.05809034 0.04743286
```

The index of `SMI` result is the value of SMI result from cluster 2 to the maximum cluster you specify ($max_k = 30$), the attribute ratio is the different ratio of index, and the best cluster is determined by selecting the maximum ratio cluster.

References