Birla Institute of Technology and Science-Pilani, Hyderabad Campus

Second Semester 2017-18

Data Mining (CS F415)

K-Means AND
Hierarchical Clustering – Agglomerative and Divisive

by

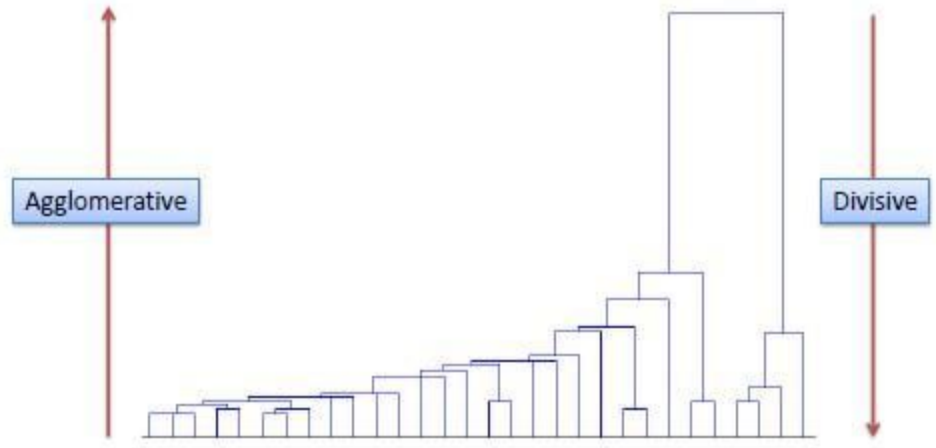
AMAN METHA	2016A7PS0066H
GARVIT JAIN	2016A7PS0080H
AYUSHI SRIVASTAV	2016A1PS0587H

Under the guidance of
Dr. Aruna Malapati

CLUSTERING

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same cluster are more similar to each other than to those in other clusters.

Hierarchical clustering involves creating clusters that have a predetermined ordering from top to bottom. For example, all files and folders on the hard disk are organized in a hierarchy. There are two types of hierarchical clustering: ***Divisive*** and ***Agglomerative***.



DIVISIVE METHOD

In *divisive* or *top-down clustering* method we assign all of the points to a single cluster and then partition the cluster to two least similar clusters. Finally, we proceed recursively on each cluster until there is one cluster for each point.

AGGLOMERATIVE METHOD

In *agglomerative* or *bottom-up clustering* method, we start with individual points as a cluster. Then, compute the distance between each of the clusters and join the two most similar clusters until we are only left with single cluster consisting of all the points

Data Set Used

Amino Acid Sequence

Data pre-processing

File was read and the DNA sequences were stored in a dictionary, where the key is the gene sequence name and the value contains the entire gene string.

A mapping was created from the unique gene sequences in the dataset to integers so that each sequence corresponded to a unique integer.

We used python multithreading function ThreadPool to speed-up the process.

Distance Matrix

Distance Matrix is an **N**x**N** matrix where a point (i, j) denotes the alignment distance between the $\mathbf{i}^{th}$ and the $\mathbf{j}^{th}$ DNA sequence strings.

Due to slower computation power of Python over C++, computation of edit distance between DNA sequence strings takes a much longer time (around 22000 seconds). As a result, this computation can not be repeated. Hence the distance matrix is stored as a pickle file to be reused later.

Linkage Matrix

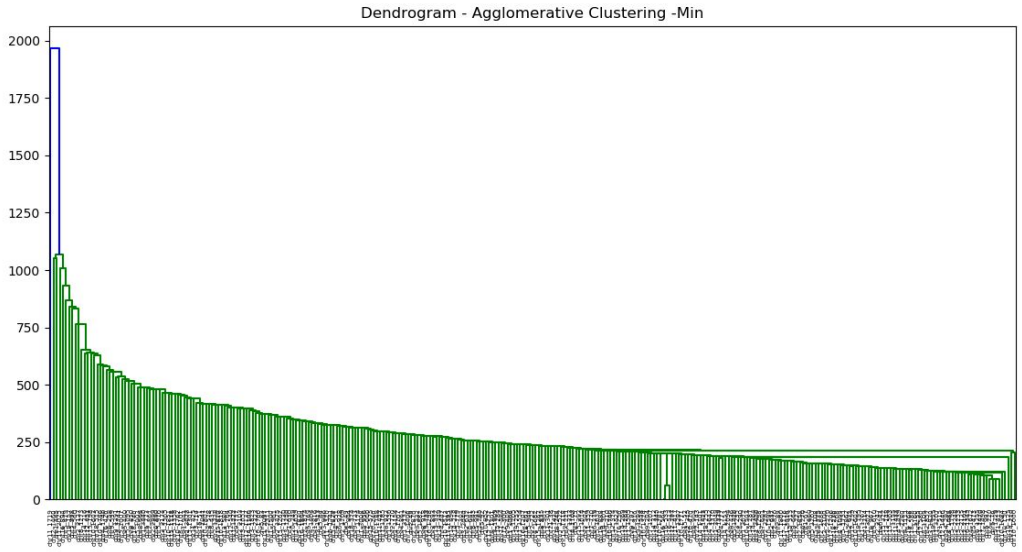
Scipy uses a special matrix called linkage matrix to draw dendrograms. The shape of the matrix is **2N-2** x **4**, where the $\mathbf{i}^{th}$ row represents the merging of two clusters to form the $\mathbf{(n+i)}^{th}$ cluster. The first and second columns of the matrix contain the clusters being merged, the third column contains the distance between the two clusters being merged, and the fourth column contains the number of elements in the merged cluster.

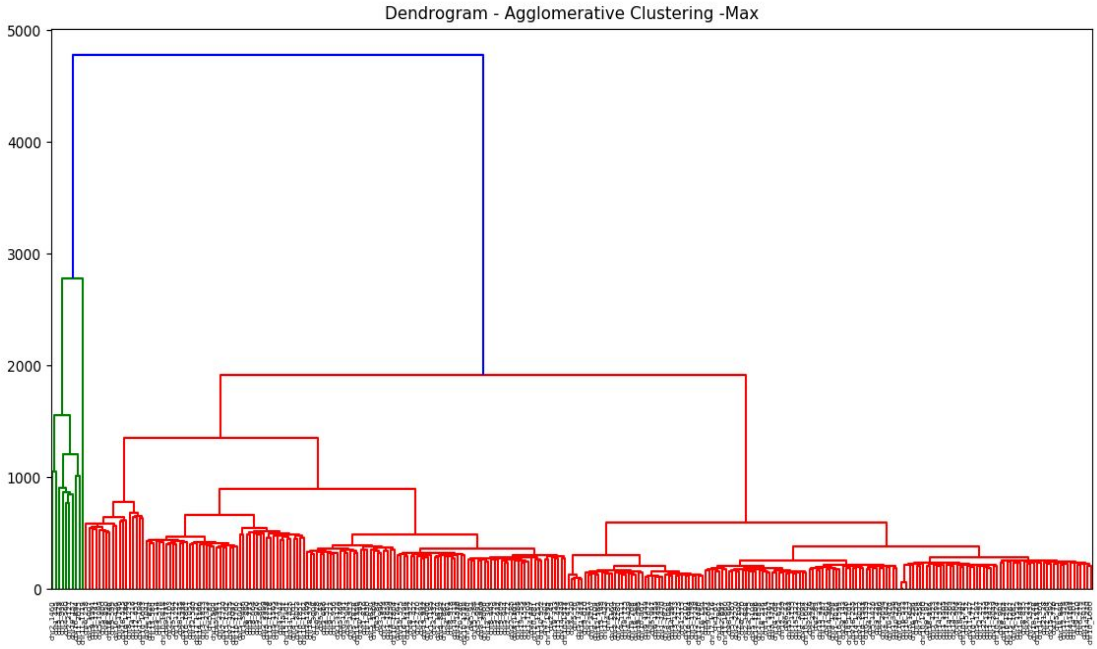
I. Agglomerative Clustering

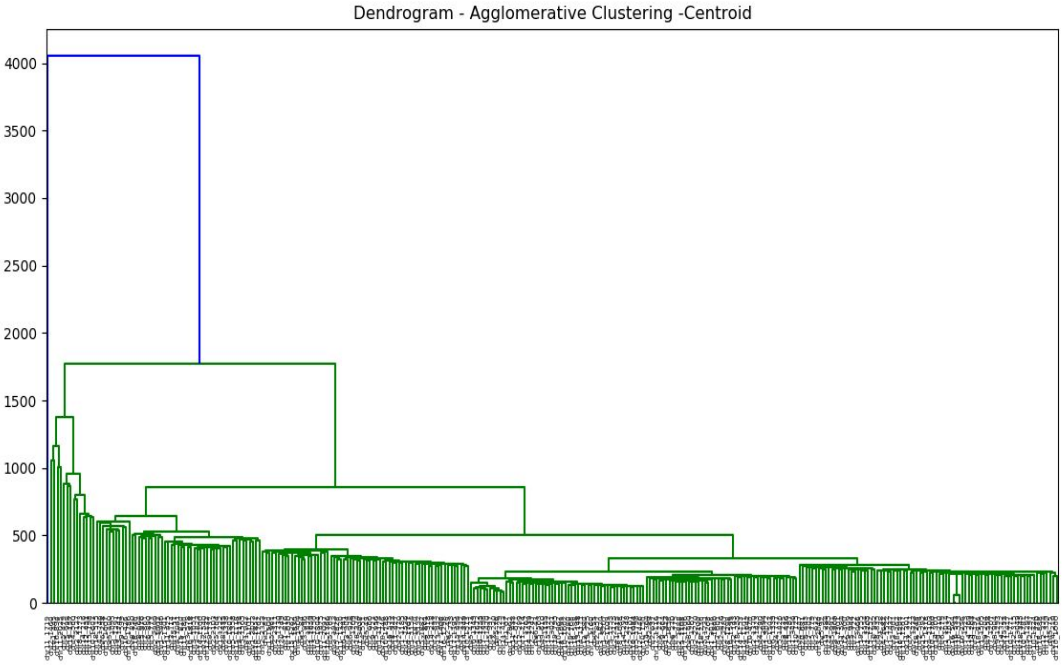
Formulas Used

Rows in the distance matrix are merged when we form a cluster and points in the newly formed cluster are modified so that they are not taken up for consideration again while forming new cluster. The newly formed cluster will have distances to all points outside the cluster using one of the heuristic (min, max, mean).

Heuristic	Formula
min	Min(d(a,b))
max	Max(d(a,b))
centroid	sum of all d(a,b) ---------- |A|*|B|




Dendrogram - Agglomerative Clustering -Max
5000
4000
3000
2000
1000
0


Dendrogram - Agglomerative Clustering -Centroid
4000
3500
3000
2500
2000
1500
1000
500
0

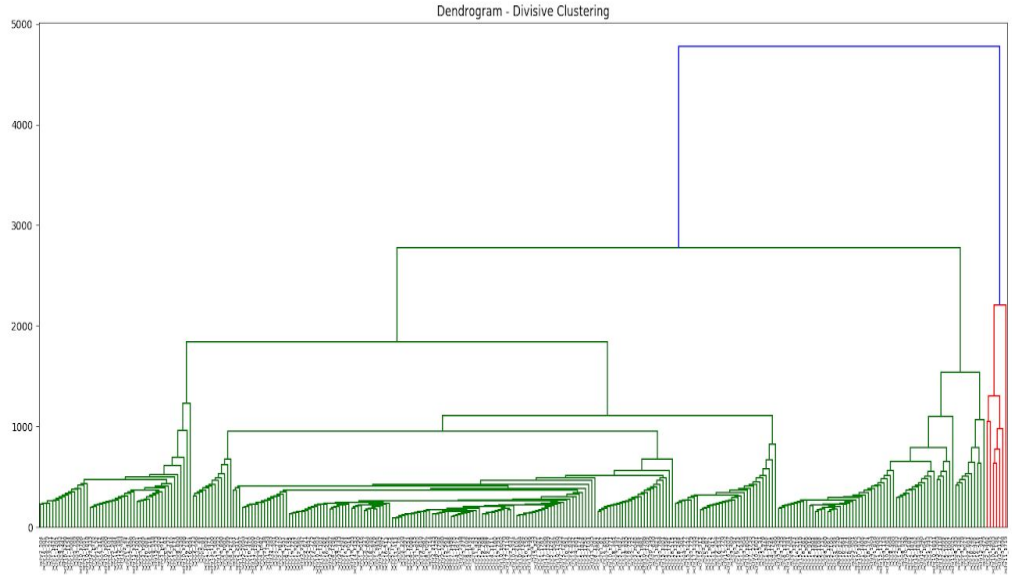
Agglomerative	Time Taken
Max	13.3
Min	11.5
Centroid	12.8

II. Divisive Clustering

We calculate the cluster with largest diameter and the select point from that cluster whose average distance from the rest of the points of that cluster, i.e. the point which is very different from most of the other points of that cluster. The point chosen is separated into a new cluster and remaining points in the cluster are rearranged. This is repeated until we have number of clusters equal to the total number of points.

Formula Used

Diameter of a cluster	Max(d(a,b))



Divisive	Time
	1.30

III. K-Means:

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

** To calculate centroids we took euclidean distances of each point in the cluster from the rest of the points in the cluster. Then making the point with min euclidean distance as centroid.

Simple: - Easy to understand and to implement.
Efficient: Time complexity: O(tkn), where n is the number of data points, k is the number of clusters, and t is the number of iterations.
Since both k and t are small. k-Means is considered a linear algorithm

k	Time taken	Number of iter.
3	.77	19
20	.10	9
50	.06	7

Comparison of Agglomerative and Divisive Clustering

Bottom-up(Agglomerative) clustering is much faster as compared to top down clustering.

- Agglomerative clustering completes execution in polynomial time
- Divisive clustering requires exponential time

Agglomerative Clustering cannot undo what has been done previously. If two clusters have been combined, they cannot be separated again.

The dendrograms generated by top-down and bottom up clustering are not same, but they are similar.

Comparison of K-means and Hierarchical

- **Time taken**
  K-means theoretically takes O(n) time while Hierarchical takes greater than polynomial time(O(n^2) for agglomerative and exponential for divisive).

- **Data type criteria**
  Distance matrix(as used in DNA clustering) is better suited for Hierarchical as compared to K-means.