Journal of Engineering and Applied Sciences 12 (11): 2827-2830, 2017
ISSN: 1816-949X

Calculation and Visualization Phylogeny of Clusters Using Java API (Application Programming Interface)

[1]Farhana Desai and [2]R.K. Kamat
[1]Symbiosis Institute of Computer Studies and Research, Pune, Maharashtra, India
[2]Shivaji University, Kolhapur, Maharashtra, India

Abstract: To measure similarity between clusters in genetic sequences graphical access is the simplest way to do it. Hence, to show the hierarchical relationships between clusters pie chart and bar graph are used in this study which are developed using java Application Programming Interface (API). The input to the graph is the set of all closest clusters calculated from the distance matrix. The combination of a distance matrix using clustering algorithm with the visualization technique used provides a powerful description for data extraction and representation.

Key words: Java, residue, cluster, pie chart, bar graph

INTRODUCTION

One of the fields of phylogeny is cluster analysis that generates genetic information which depends on the evolution of an organism. Cluster analysis is a technique used for classifying information into known clusters without having any previous knowledge about which residues belong to which cluster. Clustering is considered to be a method of unsupervised learning, to study statistical data analysis used in many fields including data mining, image analysis, machine learning, pattern recognition and bioinformatics. As there are three types of clustering methods namely agglomerative, divisive and hierarchical. Here in this study, hierarchical clustering method is used to combine the known clusters.

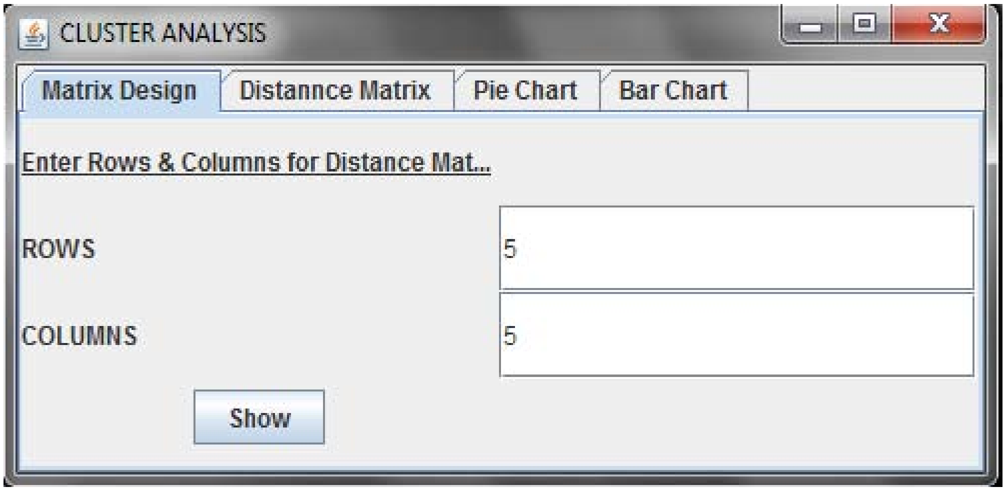
Hierarchical clustering is one of the most oftenly used mathematical technique which attempts to group residues into small clusters and further group clusters into higher-level systems. The result of a hierarchical tree is a dendogram which is a tree structure used to represent the hierarchy of clusters. A single cluster containing all observations is the root of the tree while the individual observations are the leaves. The steps to first join the two closest residues with the ancestor and further forming a cluster by adding again the closest residue with their ancestor. At each step, we select the two closest residues and join them to form a clade. Then replace the two just joined residues with their ancestor. This reduces the size of the data matrix by one. The distances from the new ancestor to the remaining sequences will be calculated.

Many of the tools have also used the bar graph to represent the clusters but it specifies only the graph with number of clusters and the cluster size. Treelink is an application that allows an automated integration of datasets to phylogenetic trees resulting in displaying the distribution of selected data attributes in the form of branches and nodes visually (Allende *et al*., 2015). The visualization of clustering is represented in the form of bra graph where the vertical axis represents the distance and the horizontal axis represents the number of each sequence (Chen *et al*., 2006). Scoredist calculates the closest distance between the clusters and represent in the form of a dendogram (Archer and Robertson, 2007). CTree is an implementation of hierarchical clustering algorithm in a star format (Sonnhammer and Hollich, 2005). CLUMP (clustering through MST in parallel) is one of the Minimum Spanning Tree (MST) based clustering techniques. Since, all of the previous work displays the visualization of clusters which do not easily describe the cluster names and their distances. Whereas the current study displays the cluster name along with its distance alongside the pie chart and bar chart which becomes easy to understand.

MATERIALS AND METHODS

The clusters from the data matrix are identified which acts as the input to develop the graphs namely pie chart and bar graph using collections, swings, two-dimensional array and j freechart of JAVA. The data matrix is taken as the input, for which the user has to enter the dimensions of the matrix shown in Fig. 1 which then automatically generates a java swing table of that dimensions (Fig. 2). The values are entered in the table and the table gets

Corresponding Author: Farhana Desai, Symbiosis Institute of Computer Studies and Research, Pune, Maharashtra, India

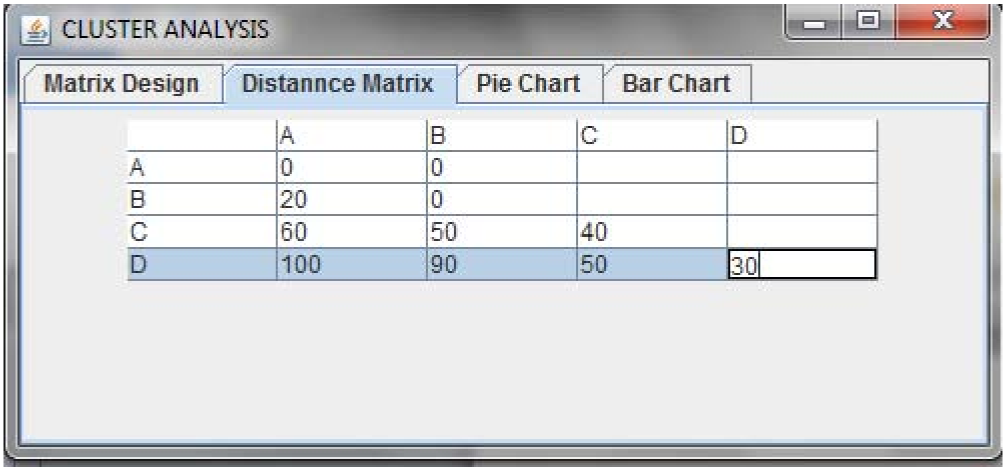


Enter Rows & Columns for Distance Mat...

ROWS 5

COLUMNS 5

Fig. 1: Dimensions to be entered to create a distance matrix



	A	B	C	D
A	0	0		
B	20	0		
C	60	50	40	
D	100	90	50	30

Fig. 2: The distance matrix to be filled in the table

stored in a two-dimensional integer array. To find the closest clusters from the data matrix a hierarchical clustering algorithm is used called as unweighted pair group method. Clusters from the distance matrix are calculated using three member functions namely to find the least element in the distance matrix to find the new column name of the minimum element and to find the distance of the closest clusters. The algorithm to find the closest cluster and its equivalent distance is shown:

The input of the distance matrix is stored in a two-dimensional integer array.
To find the minimum element all the positive integers from a two dimensional array are stored in an array list
Using the min method of the collection interface the minimum from the array list is found
The row name and the column name are retrieved from the index value of the minimum element. The new name of the column becomes the combination of the row name and the column name
The closest distance to the minimum element is calculated by taking the entire nearest element to it and adding those combinations
Similarly, the above steps will be repeated for all the closest clusters

To create a pie chart an instance of the class is created which requires the input as dataset which is stored in a hashmap as key is the cluster and value is the distance between the closest clusters. The pie chart is then assigned a label as cluster name: distance value. To display the pie chart a chart window is created where the pie chart is plotted.

To create a bar graph an instance of the class is created which requires the input as dataset which is stored in a hashmap as key is the cluster and value is the distance between the closest clusters. The x-axis denotes cluster and the y-axis denotes the distance which are parameters to the class of bar graph. The width and the color of bars are set. The classes and methods used to develop a pie chart and bar graph are found in the j freechart api of JAVA.

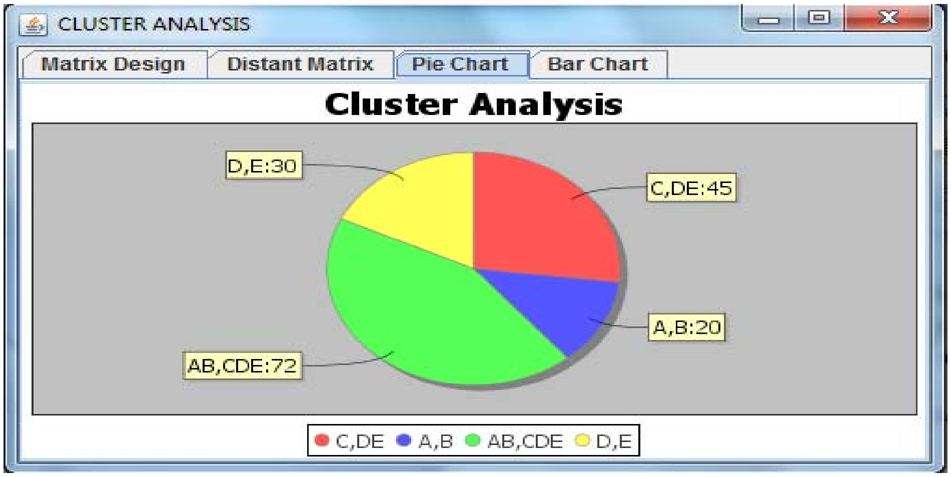


Fig. 3: Hierarchical clustering using pie chart

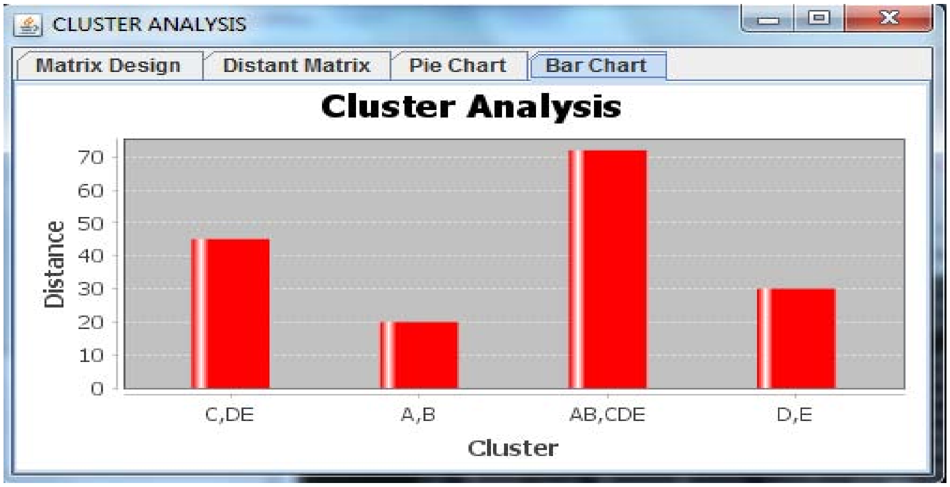


Fig. 4: Hierarchical clustering using bar graph

RESULTS AND DISCUSSION

The closest clusters are displayed in the form of a pie chart (Fig. 3) specifying the label of the chart as the cluster name: distance value. Each value of the pie chart shows the names of the closest clusters and the distance associated with them. Secondly, in the form of a bar graph (Fig. 4) here x-axis denotes the closest cluster names and he y-axis denotes the distance between the closest clusters. The closeness of the clusters helps to find the evolutionary change in phylogeny. Morphological character and phylogenetic analysis based on the SSU rRNA gene sequence and ITS sequence indicated that *Endoreticulatus* sp. Shengzhou is closely related to Endoreticulatus genus. Lastly the graphs are saved in an image file on the current drive.

CONCLUSION

Interpretation of the data becomes easy as the clusters derived from the data matrix are represented in the form of graphs. Many representations of the clusters are available like the pixel plot, star format, dots format but these formats do not easily describe the closest cluster names and their distances. The graph is also saved as an image file which can be used for future comparison. Since, the algorithm is build using only the java application programming interface it can be used on any operating system.

ACKNOWLEDGEMENT

I thank Dr. Haridas Acharya for his guidance and support.

REFERENCES

Allende, C., E. Sohn and C. Little, 2015. Treelink: Data integration, clustering and visualization of phylogenetic trees. BMC. Bioinf., 16: 404-414.

Archer, J. and D.L. Robertson, 2007. CTree: Comparison of clusters between phylogenetic trees made easy. Bioinf., 23: 2952-2953.

Chen, Y., K.D. Reilly, A.P. Sprague and Z. Guan, 2006. SEQOPTICS: A protein sequence clustering system. BMC. Bioinf., 7: S1-S10.

Sonnhammer, E.L.L. and V. Hollich, 2005. Scoredist: A simple and robust protein sequence distance estimator. BMC Bioinform., Vol. 6. 10.1186/1471-2105-6-108.