CS5810 – Programming for Data Analysis

K-Means Clustering

Clustering is the process of partitioning a group of data points into a small number of clusters. It is an unsupervised learning algorithm, wherein we group together similar set of data. K-means clustering is a clustering method in which forms pre-defined set of clusters, say K clusters. The procedure follows a simple and easy way to classify a given dataset through a certain number of clusters (K clusters).

The main idea of K-means clustering is to define K centres, one for each cluster. The centres are placed at different locations to form unique clusters. Then associate each data point to the nearest cluster centre until no data point is pending. After that we calculate the K new centroids, a new binding of data points to cluster centres is found to form new clusters. This process continues till the cluster centres converge. The final clusters formed is the output of the K-means clustering algorithm.

Algorithmic Steps for K-means Clustering

Let $X = \{x_1, x_2, x_3, \ldots\ldots, x_n\}$ be the set of data points and $C = \{c_1, c_2, \ldots\ldots, c_K\}$ be the set of centers.

1) Randomly select *'K'* cluster centres.

2) Calculate the distance between each data point and cluster centres.

3) Assign the data point to the cluster centre whose distance from the cluster centre is minimum of all the cluster centres.

4) Recalculate the new cluster centre using:

$$C_i = (1/n_i) \sum_{i=1}^{n_i} x_i$$

where, *'n'* represents the number of data points in i^{th} cluster.

5) Recalculate the distance between each data point and new obtained cluster centres.

6) If no data point was reassigned then stop, otherwise repeat the step 3 to 6.

Gene Expression Data

The gene expression data consists of

- Each row represents a gene (gi)
- Each column represents a time point (or condition (tj)
- Each entry (ei,j) is the expression of gene I at time (or condition) j.

After clustering the data using 2 different distance metrics, such as Euclidean and Pearson Correlation, we obtain the following graphical outputs as shown below. In my program, I have given the choice to the user to choose the distance metric. If the choice is 1, the distance metric chosen is Euclidean and if the choice is 2, the distance metric chosen is Pearson Correlation. The sum squared distance is also plotted with the number of iterations. The sum squared distance is inversely

proportional to the number of iterations because at each iteration, the data points which are closest to the cluster centre is calculated. Since this is calculated for different cluster centres at each iteration, only the data points closer together forms a cluster, so the distance values decreases with the increase in the number of iterations.

Distance Metric: Euclidean distance

Let the number of clusters to be formed, that is, K = 3 and the chosen distance metric is Euclidean. The clusters are visualised as follows:

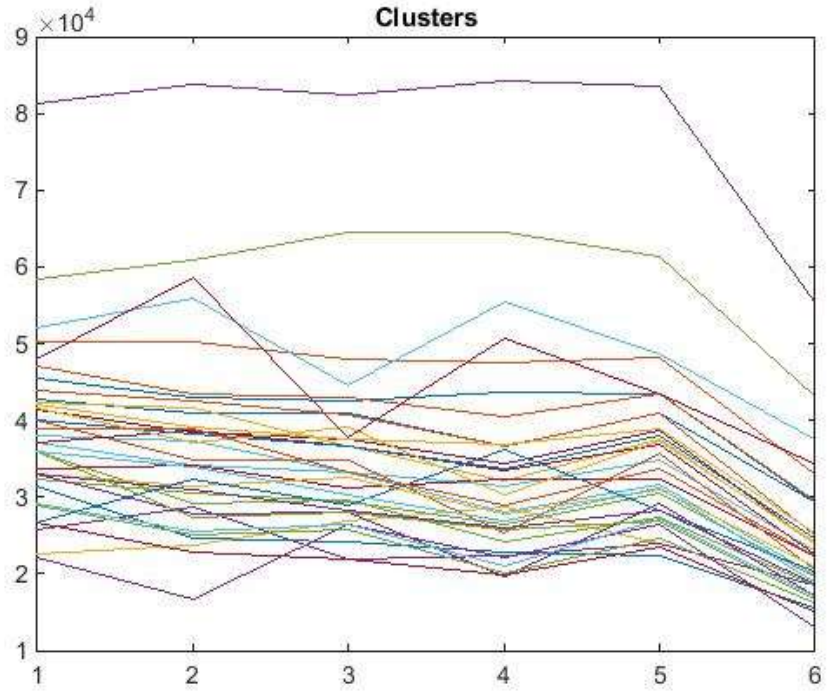

Fig 1: Number of genes in cluster 1

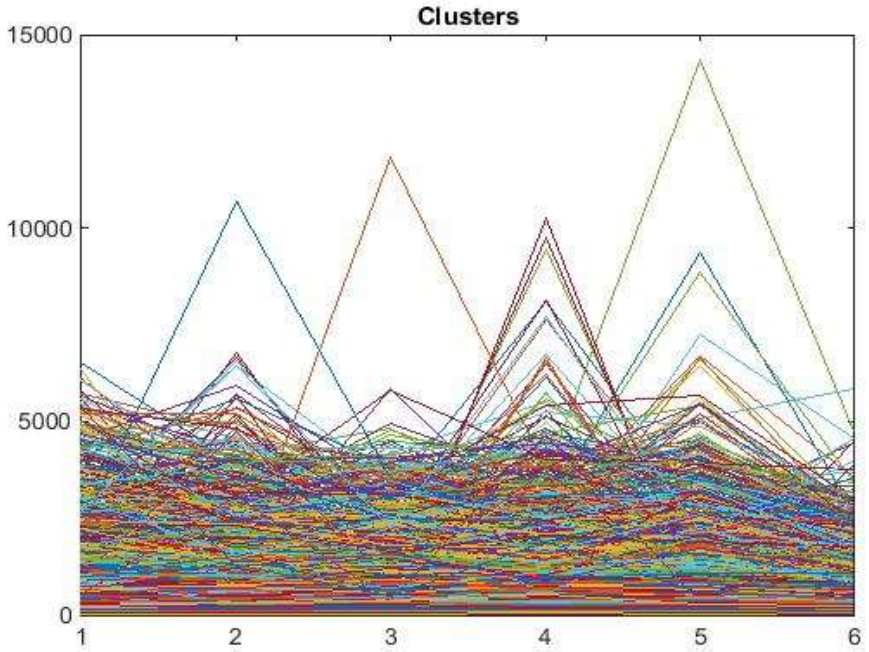

Fig 2: Number of genes in cluster 2

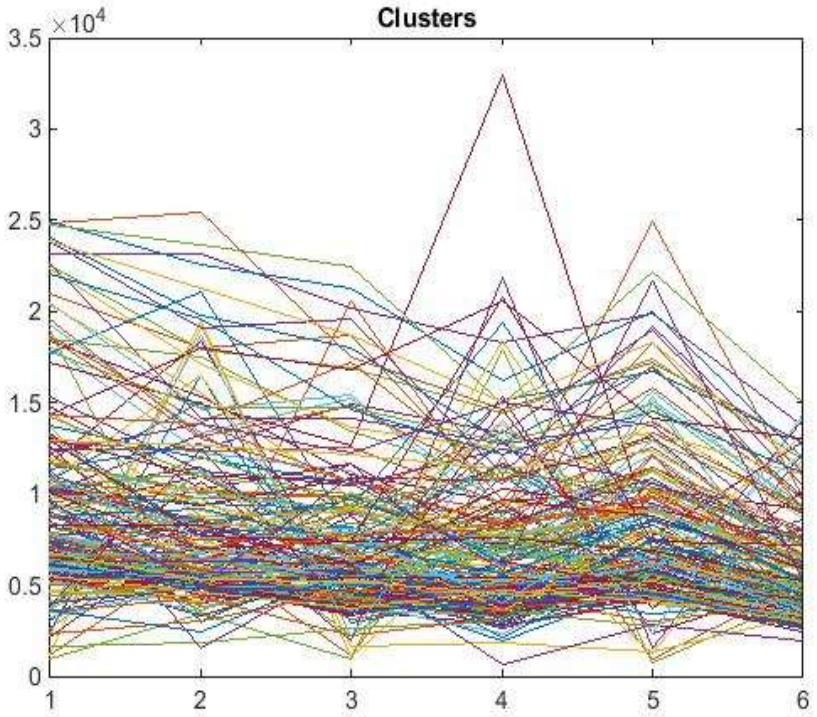

Fig 3: Number of genes in cluster 3

The sum squared distance is calculated and is plotted against the number of iterations and the graphical output is as follows:

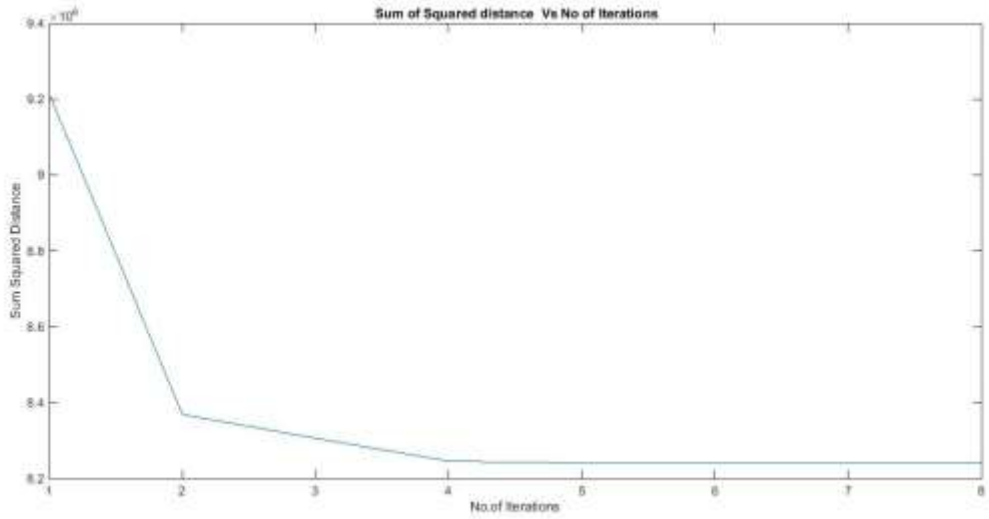

Fig 4: Sum Squared Distance Vs No. of Iterations

This graph clearly proves that the sum squared distance value decreases with the increase in the number of iterations.

Distance Metric: Pearson Correlation distance

Let the number of clusters to be formed, that is, K = 3 and the chosen distance metric is Pearson Correlation. The clusters are visualised as follows:

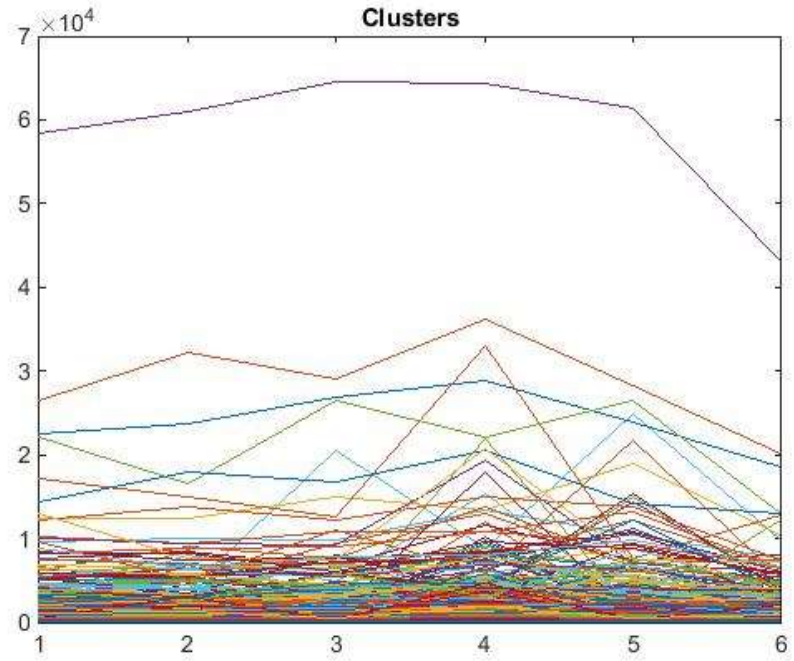

Fig 5: Number of genes in cluster 1

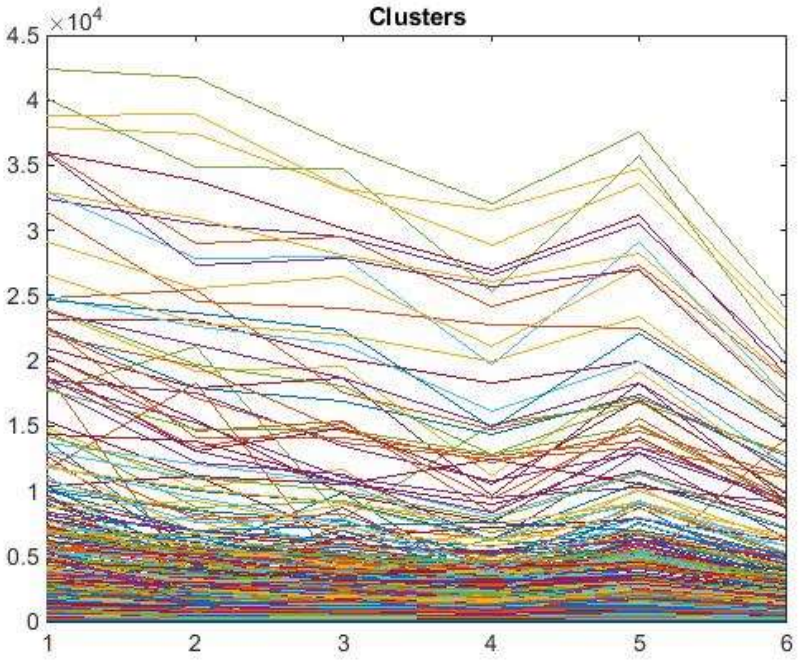

Fig 6: Number of genes in cluster 2

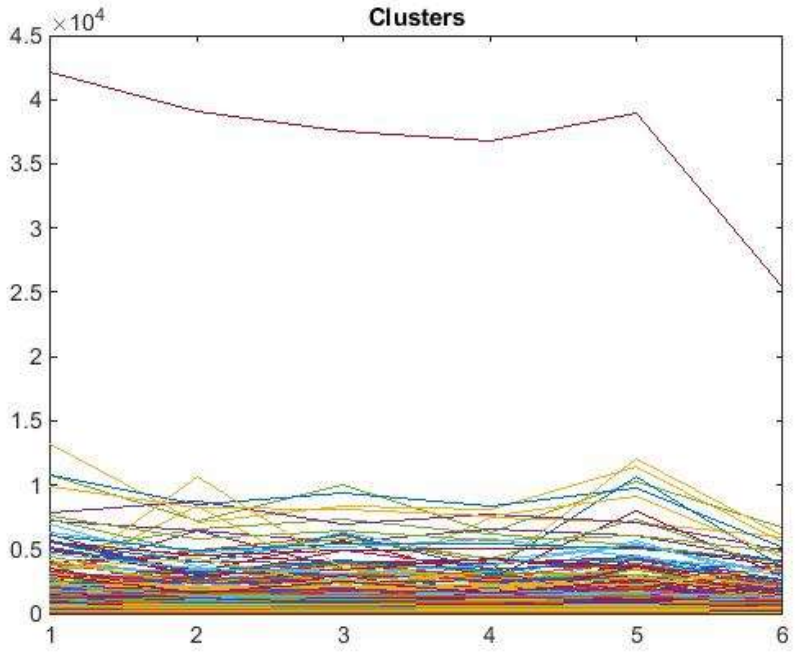

Fig 7: Number of genes in cluster 3

The sum squared distance is calculated and is plotted against the number of iterations and the graphical output is as follows:

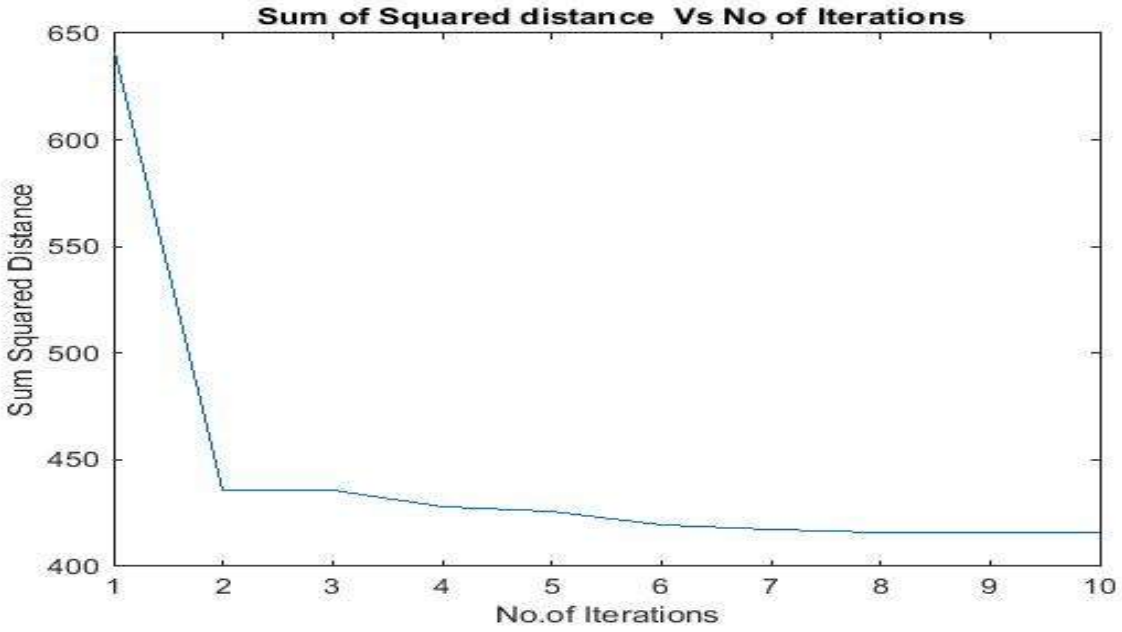

Fig 8: Sum Squared Distance Vs No. of Iterations

This graph clearly proves that the sum squared distance value decreases with the increase in the number of iterations.

Human Hereditary Disease Data

The human hereditary disease data consists of

•each node corresponds to a distinct disorder
•size of each node is proportional to the number of genes participating in the corresponding disorder
•the link thickness is proportional to the number of genes shared by the disorders it connects.

After clustering the data using 2 different distance metrics, such as Euclidean and Pearson Correlation, we obtain the following graphical outputs as shown below. In my program, I have given the choice to the user to choose the distance metric. If the choice is 1, the distance metric chosen is Euclidean and if the choice is 2, the distance metric chosen is Pearson Correlation. The sum squared distance is also plotted with the number of iterations. The sum squared distance is inversely proportional to the number of iterations because at each iteration, the data points which are closest to the cluster centre is calculated. Since this is calculated for different cluster centres at each iteration, only the data points closer together forms a cluster, so the distance values decreases with the increase in the number of iterations.

The disease data points belonging to a particular disease type is assigned a colour, the graphical output of the same is as follows:

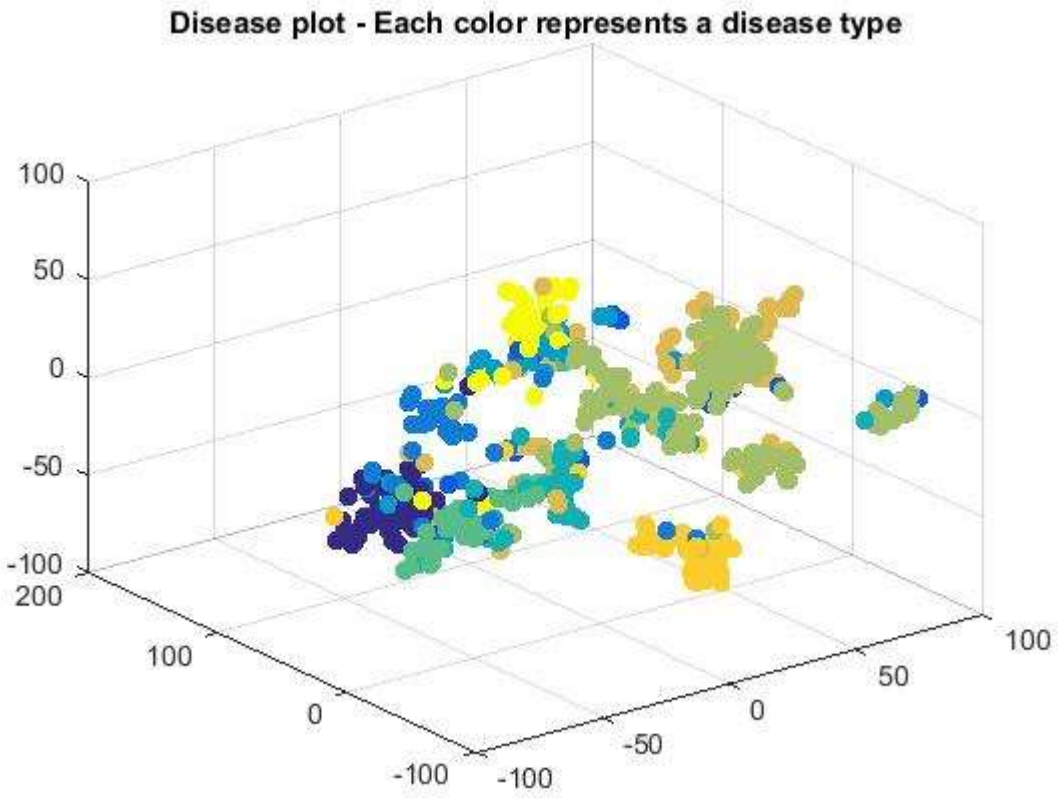

Fig 9: Scatter plot of the disease data points – colours assigned based on the disease type

Distance Metric: Euclidean distance

Let the number of clusters to be formed, that is, K = 3 and the chosen distance metric is Euclidean.

The clusters are visualised as follows:

The cluster is formed and is graphically visualised by assigning a different colour to each cluster.

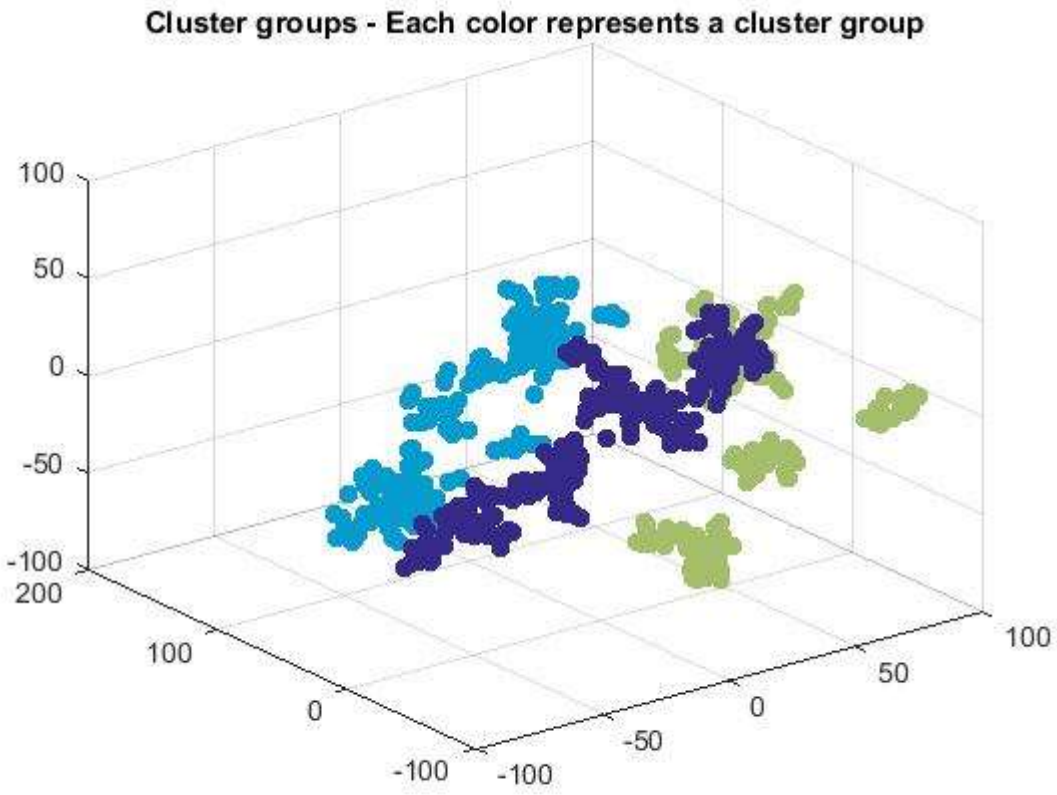

Fig 10: Scatter plot of the clusters – colours assigned to each clusters

The sum squared distance is calculated and is plotted against the number of iterations and the graphical output is as follows:

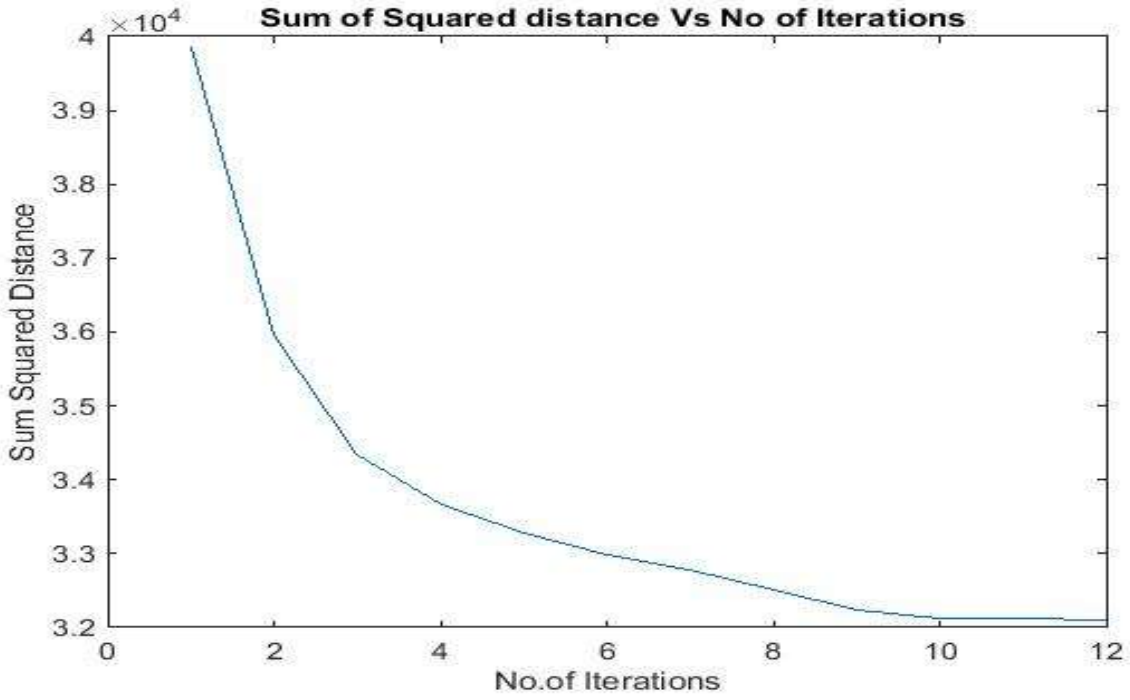

Fig 11: Sum Squared Distance Vs No. of Iterations

This graph clearly proves that the sum squared distance value decreases with the increase in the number of iterations.

Distance Metric: Pearson Correlation distance

Let the number of clusters to be formed, that is, K = 3 and the chosen distance metric is Pearson Correlation.

The clusters are visualised as follows:

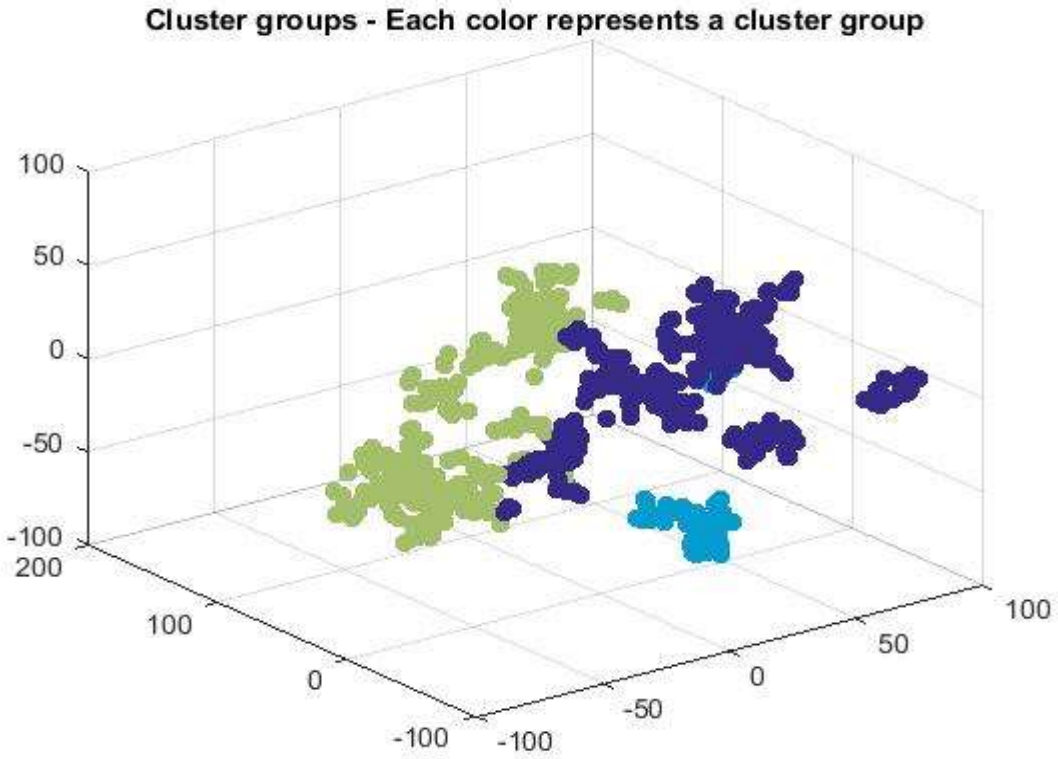

Fig 12: Scatter plot of the clusters – colours assigned to each clusters

The sum squared distance is calculated and is plotted against the number of iterations and the graphical output is as follows:

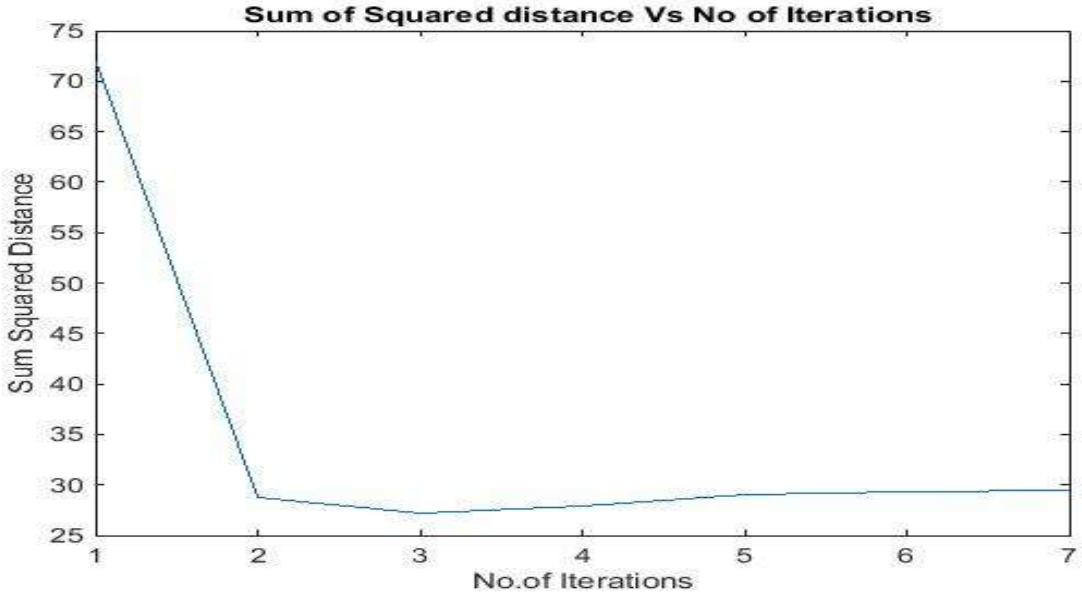

Fig 13: Sum Squared Distance Vs No. of Iterations

This graph clearly proves that the sum squared distance value decreases with the increase in the number of iterations.

Consideration of different distance metric in K- means Clustering Algorithm

Euclidean Distance metric is preferred more than Pearson Correlation because K-Means is implicitly based on pairwise Euclidean distances between points, because the sum of squared deviations from centroid is equal to the sum of pairwise squared Euclidean distances divided by the number of points. The term "centroid" is itself from Euclidean geometry, it is multivariate mean in Euclidean space. Non-Euclidean distances will generally not span Euclidean space. Hence, the cluster groups formed are better in Euclidean distance metric than Pearson Correlation distance metric.

Procedures followed for the datasets in K- means Clustering Algorithm

In my program for gene expression dataset, certain pre-processing of the data was needed, and it was executed for two different distance metrics, that is, Euclidean distance and Pearson Correlation distance. Then, cluster groups are formed and the graphical output of each cluster group is obtained. The sum squared distance is plotted against the number of iteration to obtain their dependence on each other.

Similarly, for human hereditary dataset, the data is loaded and executed for two different distance metrics, that is, Euclidean distance and Pearson Correlation distance. The diseases are also plotted by giving a colour scheme for different disease types. Then, cluster groups are formed and the graphical output of a scatter plot is obtained with each cluster having a different colour scheme. The sum squared distance is plotted against the number of iteration to obtain their dependence on each other.

Conclusion

Thus K-means clustering was implemented on 2 different datasets, that is, on gene expression data and on human hereditary data. In both the datasets, both Euclidean distance metric and Pearson Correlation distance metric has been implemented in K-means clustering algorithm and the results have been discussed. The sum squared distance have also been calculated and plotted against the number of iterations.