CS395T: Large Scale Data-Mining

Lecture 8

Clustering (Contd.)

1 Agglomerative Clustering

A number of agglomerative clustering algorithms were presented in lecture 7 with different measures of distance between two clusters. Out of these, the Nearest neighbor and farthest neighbor clustering algorithms are greedy and satisfy local objective criteria. Instead, we can also use a global objective function used in K-means clustering:

$$\mathcal{J} = \min_{k=1\ldots K} \sum_{\mathbf{x} \in C_k} ||\mathbf{x} - m_k||^2. \tag{1}$$

An agglomerative clustering algorithm that at each step tries to minimize this objective function is described below.

2 Stepwise-Optimal hierarchical Clustering

- Let $\hat{c} = n$ and $C_i = \{\mathbf{x}_i\}$, $i = 1 \ldots n$.
- While $\hat{c} \leq K$

  (i) Find the pair of distance clusters, say C_i and C_j whose merger increases the objective function as little as possible.

  (ii) Merge C_i and C_j, delete C_j.

  (iii) $\hat{c} \leftarrow \hat{c} - 1$.

  EndWhile

Let $J_i = \sum_{\mathbf{x} \in C_i} ||\mathbf{x} - m_i||^2$ and $J_k = \sum_{\mathbf{x} \in C_k} ||\mathbf{x} - m_k||^2$. Merging C_i and C_k results in the new mean $\hat{m}_i$ to be:

$$\hat{m}_i = \frac{n_i m_i + n_k m_k}{n_i + n_k} = m_i + \frac{n_k(m_k - m_i)}{n_i + n_k} = m_k + \frac{n_i(m_i - m_k}{)} n_i + n_k. \tag{2}$$

Increased objective function (Equation (1)) value is:

$$\begin{aligned}
&\textstyle\sum_{\mathbf{x} \in C_i} ||\mathbf{x} - \hat{m}_i||^2 + \sum_{\mathbf{x} \in C_k} ||\mathbf{x} - \hat{m}_k||^2 \\
&= \textstyle\sum_{\mathbf{x} \in C_i} ||\mathbf{x} - m_i - \frac{n_k(m_k - m_i)}{n_i + n_k}||^2 + \sum_{\mathbf{x} \in C_k} ||\mathbf{x} - m_k - \frac{n_i(m_i - m_k)}{n_i + n_k}||^2 \\
&= J_i + \tfrac{n_k^2 n_i}{(n_i + n_k)^2} ||m_i - m_k||^2 + J_k + \tfrac{n_i^2 n_k}{(n_i + n_k)^2} ||m_i - m_k||^2 \\
&= (J_i + J_k) + \tfrac{n_i n_k}{n_i + n_k} ||m_i - m_k||^2
\end{aligned} \tag{3}$$

Hence among all pairs C_i and C_k, chose the pair such that:

$$d_{opt}(C_i, C_j) = \sqrt{\frac{n_i n_k}{n_i + n_k}} ||m_i - m_k|| \tag{4}$$

3 K-Means Clustering

Goal is to produce K clusters.

1. Start with some partitioning of the data into K clusters: $C_1^{(0)}, C_2^{(0)}, \ldots, C_K^{(0)}$. Set iteration count $t \leftarrow 0$.

2. Update Mean of each cluster based on current data partitioning:

$$m_i^{(t)} = \frac{1}{n_i^{(t)}} \sum_{\mathbf{x} \in C_i^{(t)}} \mathbf{x} \tag{5}$$

3. Update partition i.e. distribution of data points in different clusters based on the new cluster means. For each $\mathbf{x}$, find the mean vector $m_i^{(t)}$ nearest to it, say the closest mean vector is $m_j^{(t)}$. Put $\mathbf{x}$ in $C_j^{(t+1)}$. Thus:

$$C_j^{(t+1)} = \{\mathbf{x} : ||\mathbf{x} - m_j^{(t)}|| \leq ||\mathbf{x} - m_i^{(t)}||, i = 1, 2, \ldots, K\} \tag{6}$$

4. If not "converged", $t \leftarrow t + 1$ and go to step 2.

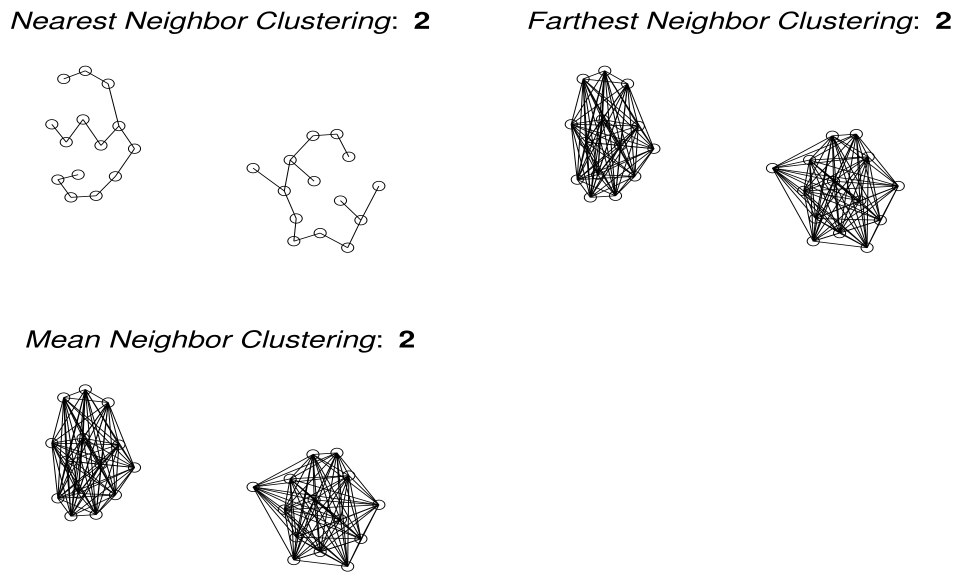

Figure 1: Dataset 1: Nearest neighbor tends to make chain like clusters and Farthest neighbor and mean form spherical clusters.

4 Demo on different Clustering techniques

In this section, some of the results of various agglomerative clustering algorithms and k-Means clustering algorithms on 2-D toy datasets are shown to highlight the properties of different clustering algorithms. The key is to first figure out the bias of your datasets and then apply that clustering technique whose bias matches the bias of that dataset. The idea is that there is no one clustering algorithm that is best for all datasets.

Figures 1 through 3 show the results of three agglomerative clustering algorithm on three datasets from Duda and Hart. For all these datasets, the clusters are obvious when viewed. Note how the nearest neighbor clustering technique tends to make elongated chain like clusters while the farthest neighbor and mean clustering tend to form spherical clusters.

Figures 4 through 6 show the results of K-means clustering algorithm on three datasets. Each figure shows four different runs of k-means with different initializations. Note how sensitive K-means is to these initializations. It generally converges to a local minima near reachable from the initial partitioning.

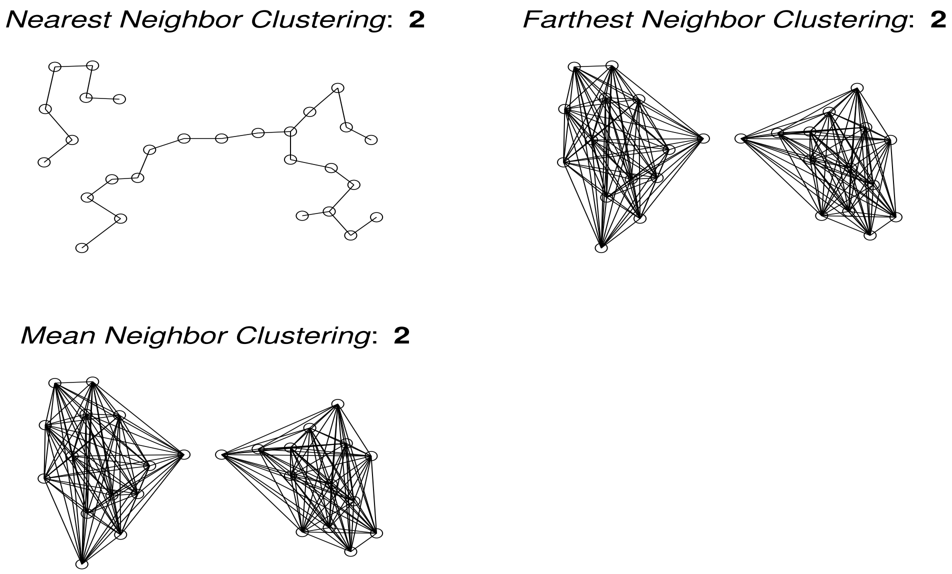

Figure 2: Dataset 2: Nearest neighbor tends to make chain like clusters and Farthest neighbor and mean form spherical clusters

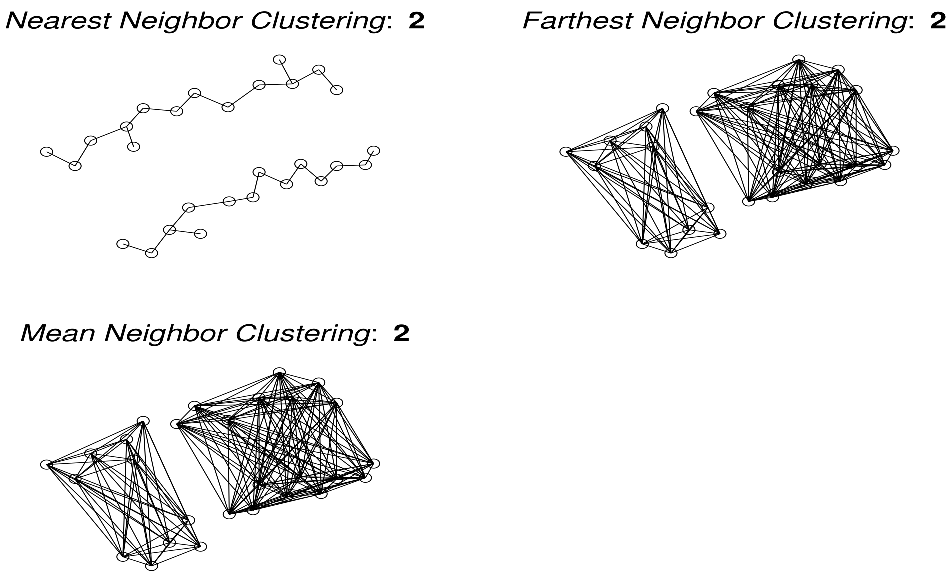

Figure 3: Dataset 3

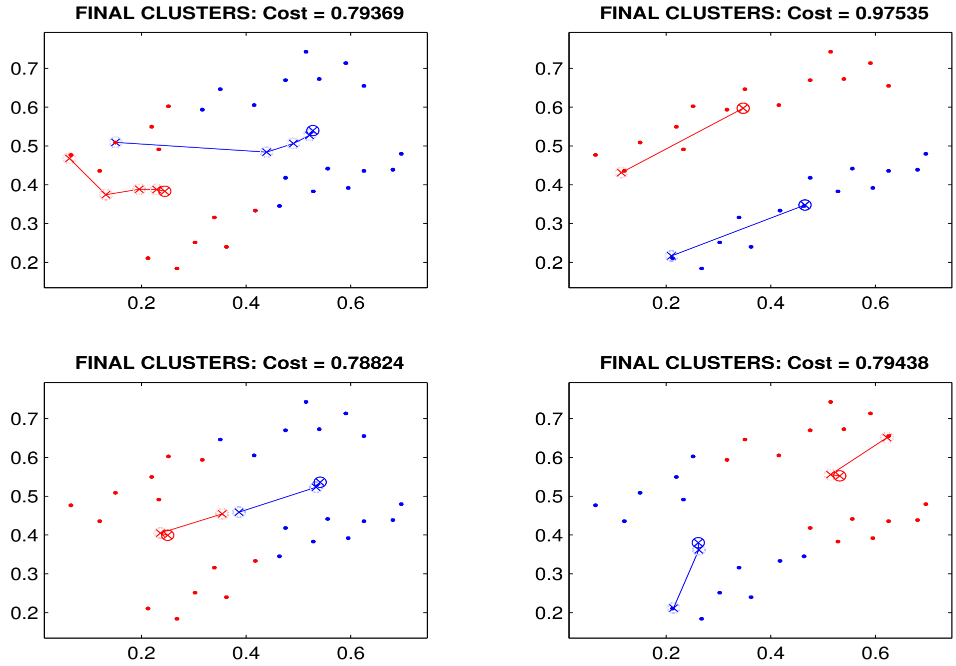

Figure 4: Dataset 1: Four runs. Two local minima possible depending on initial clusters.

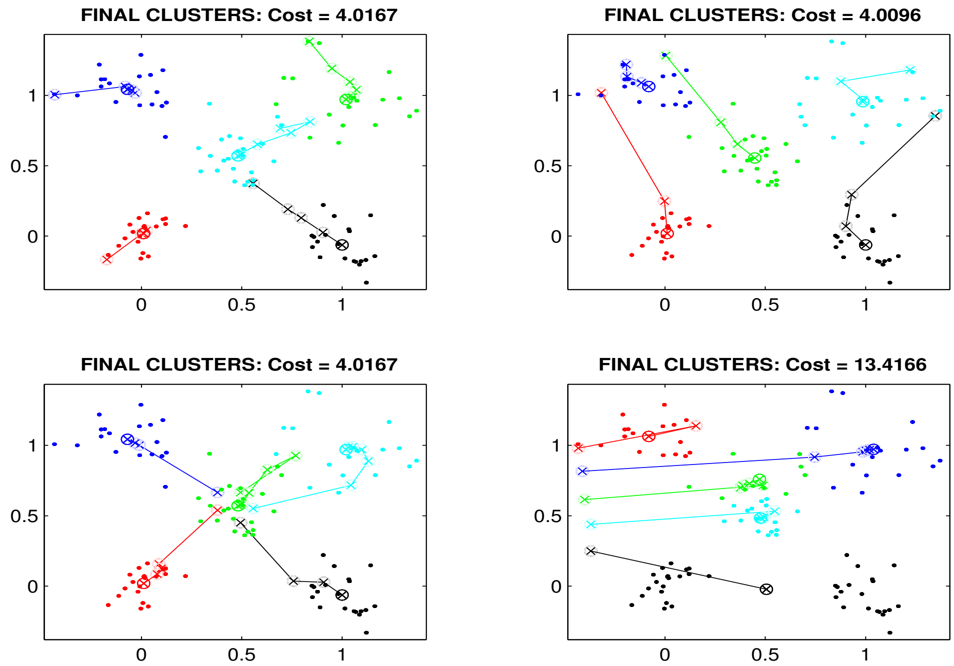

Figure 5: Dataset 2: 5 cluster data. Different results based on initialization.

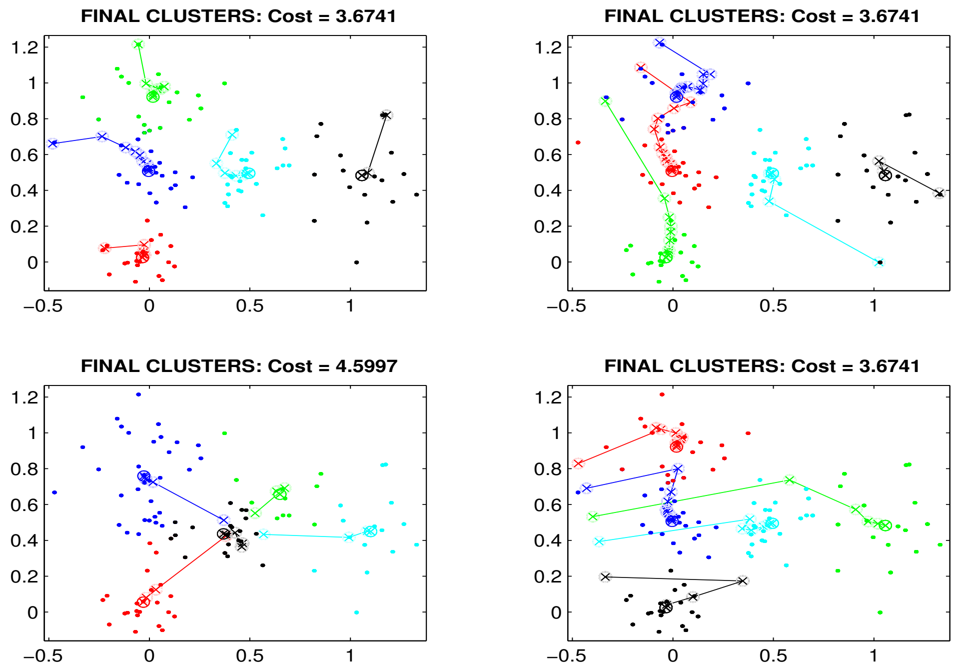

Figure 6: Dataset 3: 5 clusters data with less obvious cluster boundaries. Again different results based on initialization.