Renee Chien
Mentored by Daniel Suen
SPA Directed Reading Program
17 December 2021

Cluster Analysis Directed Reading Program Writeup

Over the course of this quarter, I learned about clustering as an unsupervised way to find subgroups within data sets. Specifically, the DRP explored the two methods of clustering data that were covered in the textbook *An Introduction to Statistical Learning*— **k-means clustering** and **hierarchical clustering**.

These two methods differ in that, while k-means is a clustering method that partitions data points into a given number of clusters, hierarchical clustering is a method that allows for there to be any number of clusters (specifically, from 1 cluster of all points to as many data points there are), that can later be decided by interpreting a tree of clusters called a ***dendrogram***.

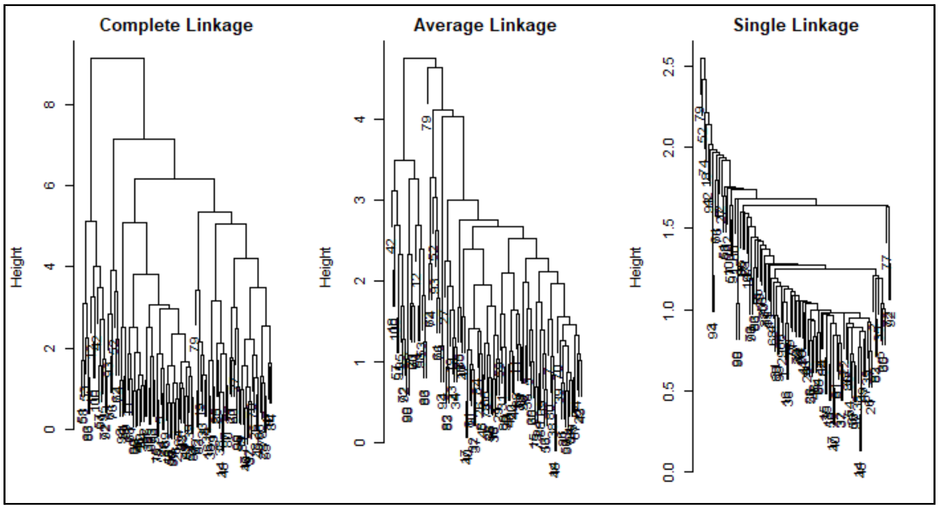
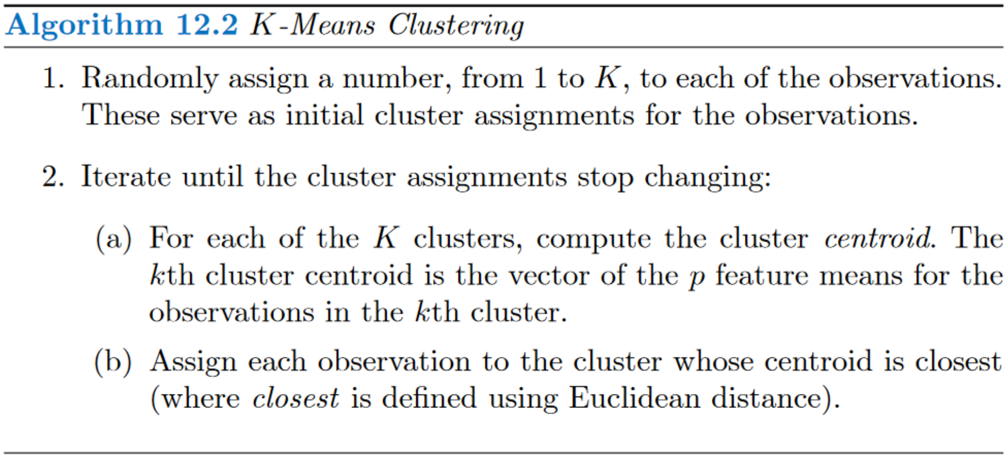


Fig 1. Dendrograms for my data set. Dendrograms can be constructed using different modes of linkages, which determine how singular data points and/or clusters of them are joined.

Furthermore, while in k-means clustering, Euclidean distance is used as the measure of "difference" between points (also called ***dissimilarity***)— and thus is only applicable to observations with continuous features— hierarchical clustering may use different types of dissimilarity measures, including ones that define difference between categorical variables.

The following is the algorithm for how k-means clustering is conducted:



Algorithm 12.2 *K-Means Clustering*

1. Randomly assign a number, from 1 to K, to each of the observations. These serve as initial cluster assignments for the observations.
2. Iterate until the cluster assignments stop changing:
   (a) For each of the K clusters, compute the cluster *centroid*. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster.
   (b) Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

K-means Clustering Algorithm. (Source: An Introduction to Statistical Learning, 2nd ed. page 519.)

Below is a visual representation of steps of the algorithm, using a data set with two features (and thus two "dimensions".)

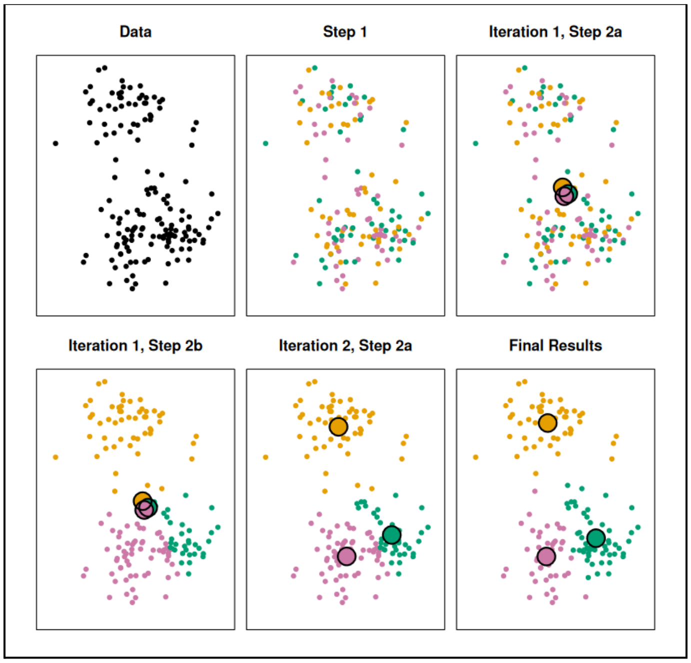
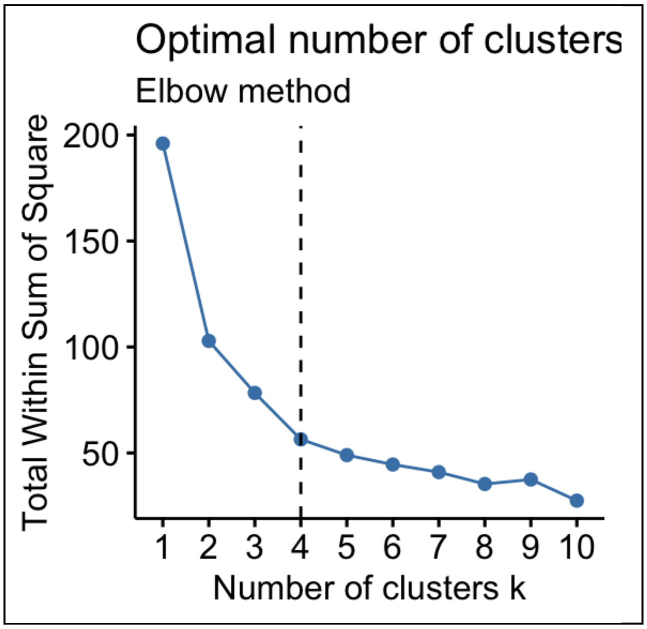


Fig 3. K-means Visualization. (Source: An Introduction to Statistical Learning, 2nd ed. page 520.)

Depending on how the data points are initialized (through the randomization process), the outcomes of k-means can vary, and each individual iteration might not yield the most "optimal" or tight clustering results. R thus runs a number of different initializations to completion and finds one that yields the best clustering results, which are the ones that have minimized dissimilarity between points within clusters.

In implementing k-means clustering on real data where we don't know the "true" clusters, it's not always clear what number of clusters— **k**— to use. I learned about the "elbow method" to determine a point where extra clusters produce diminishing returns when it comes to minimizing the "within sum of squares".



Elbow method. *Image Source*

The textbook gave some code for simulations that I was able to run, and later implement to my data set. I used my 2021 Spotify Wrapped playlist— the top 100 songs I listened to between January and November this past year on Spotify's platform— as my data set. Daniel helped me grab the values of track features (see slide 13 of my slide deck for descriptions of track features based on Spotify's API) using a python program, on seven of which I was then able to run k-means and hierarchical clustering. While the **WSS v. Number of Clusters** plot didn't give a clear "elbow" point, I decided on 4 clusters for both methods based on the interpretability of 4 clusters as opposed to 5, or 6, or 7 clusters etc.

The two methods of clustering the data set into 4 clusters yielded similar results— they shared the detection of four clusters as follows: 1. An acoustic/instrumental cluster, 2. A "speechy"/more spoken-word/less melodic cluster, 3. A sadder pop/dance cluster, and 4. A Happier pop/dance cluster. The number of songs in each cluster across the two methods were similar. I would say that the fact that the two methods somewhat corroborated each other, that these are pretty solid clustering results.

Two thoughts have been with me since my presentation earlier this week. Michael Pearce pointed out, helpfully, that k-means clustering doesn’t work well with many features. I think if I were to run this clustering again, I would eliminate some track features to reduce the dimensionality of the data. Another thing Michael pointed out was that I had two bigger clusters— the sad and happy pop/dance clusters. I joked that I should take those clusters and then re-run clustering on them to parse them up, but now I’m actually not sure, as I remembered that clustering methods are very sensitive to perturbations to data, including taking subsets out.

More research is needed. :)

My acknowledgements go to Daniel Suen, my amazingly patient, incredibly smart, and morally supportive mentor who has become a great friend of mine despite our MBTI incompatibilities, and David Marcano, who has gone to great lengths administratively and in his explanations to ensure the DRP group’s success. Townsend Cocke, my fellow mentee, has been great company as well. The DRP organizers, notably Anna Neufeld, who is patient and accommodating regarding my technology and logistical needs. Michael Pierce for putting me on the spot with very good questions at the final presentation. Jamie, who helped me rehearse my presentation, and my family.