Springboard Project 11.5 (Clustering)

Using Wine data

1. *Create a data frame where each row has the requested columns.*

   After importing all necessary packages and datasets, I merged the offers and transactions data sets. Next, I filled all NA values with zero since values in the data frame are binary markers of presence/absence in column.

2. *Look at the number of clusters (k) using the elbow method.*

   First, I generated an elbow plot showing the sum of squares vs. the number of clusters. There was no clear “elbow,” but 5 seemed to be the best choice to limit overall number of clusters. It was also a place where reduction in sum of squares began to taper off.

3. *Look at the number of clusters (k) using the silhouette method.*

   Through generating a silhouette plot and the visualized clusters for k-values 4 through 8, I found k=5 had the best silhouette score. It also aligned with my analysis from the elbow method.

4. *Use PCA to plot and evaluate the clusters.*

   First, I created a data frame with customer name, cluster ID, and the two principal components (labeled x and y). I then plotted the clusters according to this data frame. An argument can be made for choosing 3 or 4 clusters, but the other potential k-values (including 5) look less appealing on this plot.

5. *Using a new PCA object, plot the explained variance and look for the elbow point. This value is one possible value for the optimal number of dimensions. What is it?*

   I constructed the elbow plot from the new PCA values provided. The results showed a clearer elbow plot than the first one I created, and the optimal dimension value was 3.

6. *Using a new PCA object, plot the explained variance and look for the elbow point. This value is one possible value for the optimal number of dimensions. What is it?*

   I constructed the elbow plot from the new PCA values provided. The results showed a clearer elbow plot than the first one I created, and the optimal dimension value was 3.

7. *Try clustering using the following algorithms: affinity propagation, spectral clustering, agglomerative clustering, DBSCAN. How do their results compare? Which performs the best? Explain why you think it performs the best.*

   After running each algorithm and plotting results, it looked to me like affinity propagation performs best. Its metrics aren't the best, but its clusters are visually the best. The clusters don't overlap and appear to have separately nicely.