Corey Munk

COMP 4432

Assignment 5

**Problem 1a. Kmeans**

Questions:

1. What method(s) did you use to identify an appropriate value for K? Why did you select this method? (5 pts)

I used an elbow plot and a silhouette plot. These methods allowed me to analyze the data to discern an analytical result for the number of clusters.

1. What value did you select for K? Does your EDA support this choice? (2 pts)

I selected 3 for K since my EDA suggested there were 3 clusters present in the x1\_vals dataset.

1. How many iterations were required before your model converged? (2 pts)

Four iterations were required for my model to converge.

1. What were the values for each of your cluster centroids? (2 pts)

[ [1.24234291 0.25210859], [-1.09550672 -1.24100805], [-0.14683618 0.98889946] ]

1. What kmeans measure serves as a proxy for cluster coherence? What value did your model return? Discuss your interpretation of this value. (5 pts)

Sum of Squared Errors serves as a proxy for cluster coherence, which is also referred to as inertia. My model returned an inertia of 163, which could indicate that the model has good cohesion however it would be best to use this number to compare to other clustering models.

**Problem 1b. Silhouette Plot**

Questions:

1. What information does a silhouette analysis provide? (5 pts)

The silhouette analysis tells us how cohesive and separated each cluster is w/ respect to other clusters

for a given number of clusters.

1. Describe how to interpret the individual silhouette plots in terms of cluster cohesion and separation. What are the key features of the graphs that aid your evaluation? (5 pts)

Cohesiveness can be determined by looking at the bar height and sign within each cluster (taller positive bars means more cohesion in that cluster), while separation can be determined by space between each cluster bar (the more space the more separated each cluster is). The plot with the highest average silhouette score and taller positive bars could indicate that its cluster number is ideal.

1. From the silhouette plots you created, what value of K affords the best cluster assignments? (3 pts)

The silhouette plot for 3 clusters appears to be most ideal, which corroborates what the scatter and

elbow plot suggested from part 1a.

**Problem 2. External Validation**

Questions:

1. What method did you select to assess cluster agreement and why? (5 pts)

I used Adjusted Rand Score as it is a common external validations metric for clustering when labels are known.

1. What do your results of this assessment suggest? (5 pts)

The Adjusted Rand Score of .98 suggests very good agreement between the actual and predicted labels as 1.0 is the best value.

**Problem 3a – Concepts: Interpreting SSE**

Total SSE is the sum of the SSE for each separate attribute in the Kmeans algorithm.

1. What does it mean if the SSE for one variable is low for all clusters?

This suggests that the data likely has a lower variance for that variable in each cluster.

1. Low for just one cluster?

This suggests that the data likely has a low variance for that particular variable only for that cluster.

1. High for all clusters?

This suggests that the data likely has a higher variance for that variable in each cluster.

1. High for just one cluster?

This suggests that the data likely has a higher variance for that particular variable only for that cluster.

1. How could you use the per variable SSE information to improve your clustering? (5 pts)

Variables with low SSE for all clusters are likely more important in describing the data. While variables

with high SSE for one or more clusters may not be a good feature in driving the clustering for the data.

**Problem 3b. Local and Global Objective Functions**

A group of circles and squares

Description automatically generated

**Problem 3c. Density clustering**

1. Label all points as ‘core points’, ‘boundary points’, and ‘noise’. (5 pts)

|  |  |  |
| --- | --- | --- |
| **Point** | **Point(s) w/in EPS** | **Label** |
| A | 1 | noise |
| B | 3 | core |
| C | 4 | core |
| D | 2 | boundary |
| E | 3 | core |
| F | 4 | core |
| G | 2 | boundary |
| H | 1 | noise |
| I | 3 | core |
| J | 3 | core |
| L | 3 | core |
| M | 3 | core |

1. What is the clustering result (i.e., how will the data cluster)? (5 pts)A graph with circles and dots

   Description automatically generated

**Problem 3d. Entropy vs. SSE**

Assume you are given a data set of objects, each of which is assigned to one of two classes, and suppose that C1 and C2 are two clusterings produced from this data set. If entropy judges C1 to be a more accurate clustering than C2, is it necessary that SSE will also judge C1 to be a more accurate clustering than C2? (5 pts)

SSE may not also judge C1 to be more accurate. This is because entropy and SSE measure different characteristics of clusters. SSE is more focused on distances between data and their centroids while entropy ………….

**Problem 4. Selecting an Appropriate Clustering Algorithm**

Questions:

1. What method(s) did you use to identify an appropriate value for K? Why did you select this method(s)?

I used an elbow plot and a silhouette plot. These methods allowed me to analyze the data to discern an analytical result for the number of clusters.

1. What value did you select for K? Does your EDA support this choice? (5 pts)

I selected 3 for K since my EDA suggested there were 3 clusters present in the x4\_vals dataset.

1. Why is the algorithm that you selected for your final clustering model the best choice among those you evaluated? (5 pts)

I selected …….. because….

**Problem 5. Comparing Algorithms**

Compare the relative strengths and weaknesses of the clustering methods listed in Problem 4. Include your observations in a table form. (10 pts)

|  |  |
| --- | --- |
| **Model** | **Observation** |
| Kmeans | Kmeans and Gaussian Mixture yielded very similar clustering results with only a select number of data points having a different clustering assigned. |
| Gaussian Mixture |
| DBSCAN | DBSCAN seemed to do well in showing data points which could be outliers as well as mostly clustering the data into three clusters which my EDA suggested was the correct number of clusters. However, there were two small clusters that ended up showing up for a total of five clusters not including the outlier category. |
| Agglomerative | Agglomerative was more similar to Kmeans and Gaussian Mixture but the middle cluster traded some data points into the far right cluster group. |

**Problem 6. Dimensionality Reduction (10 pts)**

Please import the following dataset – network.csv

Dimensional reduction is a standard preprocessing step before clustering for high dimensional datasets. This owes to the fact that distance measures do not work well in high dimensional spaces. It’s worth noting, however, that the ability to cluster data owes more to the feature signal-to-noise ratio than to the number of dimensions.

For this problem we are going to read a dataset that contains performance information from LTE cell sites operated by a rural provider. The dataset is comma delimited and contains 33 columns. The ‘day’ column is a string in the 2020-12-16 format (year-month-day). You will need to convert this to pandas datetime. The BTS\_ID column is a unique identifier for the cell site and radio. The first number is the cell site and the number after the dash indicates the radio. Do not open and save the dataset in excel. Excel will convert all the BTS\_ID’s that is thinks look vaguely like a date into a date (10-1 will be converted to October 1st). The remaining columns are different performance measurements for how well the mobile phones are doing while on that site. Some measures are best when they are very small values and some are best when they are large. It would require a deep dive into the LTE cell phone standard to explain all the measurements. Some of the key values are FDD\_Cell\_DL\_MAC\_Data\_Volume\_MB which shows how much data was downloaded in MegaBytes. RRC\_Drop\_Pct is the percentage of data sessions that were dropped due to bad coverage.

It is very time consuming for the engineers in charge of this network to look at all 31 measurements for each cell site to see if there are any problems. We need to simplify the problem by doing dimensionality reduction and finding sites that are not “normal”. The goal of this analysis is to find cell sites that are performing differently than the normal sites by seeing how close their performance is to all the other sites. We are going to reduce the 31 dimensions down to two dimensions and then find outliers from the main cluster.

Part 1: filter the dataframe down to just the last day, 2021-06-13. Drop the day column and save the BTS\_ID column to another variable. Standard scale the data and run PCA with two principal components. Join the BTS\_ID data back into the PCA dataframe so it can be used with the plot. Plot the data using plotly scatter plot with the hover data showing the BTS\_ID. Labeling all points in scatter plot is going to be impossible to read, so hover data allows you to read the points you want and not overlap with the others.

The plot should look like this when you hover over BTS\_ID 143-3:A graph showing a number of data

Description automatically generated

Part 2: Run DBSCAN on the PCA data from above to find outliers, list BTS\_ID's that are outliers (the points DBSCAN calls noise). Use eps=1 and min\_samples=2 to get a reasonable number of outliers. Make the labels into a column in the PCA dataframe. Next, change the labels in the dataframe to be -1 if it is an outlier and 0 if it is not. Now do a plotly plot with the color being the labels column of the dataframe so we can clearly see the outliers.

The plot should look like this when you hover over BTS\_ID 143-3: A screen shot of a graph

Description automatically generated

Part 3: Perform PCA on all the dates in the dataset except "2021-06-13" just like you did above with a single date. Compute the mean Euclidian distance from the center of the plot (coordinate 0,0) for each BTS\_ID. Remember each BTS\_ID has measurements for each day so get the mean distance across all the days. Compute the distance from the center of the plot for each BTS\_ID using the "2021-06-13" data.

Print a list of all the outlier sites are farther from the "normal", meaning center of the plot, than their mean over the last 6 months. Print a list of all the sites that are getting better (meaning heading toward the other sites performance in the center of the cluster).

The sites that are outliers and getting worse should be reported to the engineers in charge of system performance. The outliers that are getting better show the engineers are making some progress.

Are the engineers making progress with the system performance?