Jordan Tompkins

WGU  Data Mining II Task 1

WGU D212 Task 1

A1, Proposal of Question: The question that will be looked at in this analysis is if there are any meaningful clusters within the churn dataset when looking at the ‘Age’ and ‘Income’ variables? To do this we will be using k-means clustering.

A2, Defined goal: The goal of the analysis is to determine the subgroups, or clusters, of the dataset and see if the clusters created are optimal for analysis.

B1, Explanation of Clustering Technique: K-means clustering is considered one of the most simple and popular unsupervised machine learning algorithms. It begins with a group of randomly selected centroids and these are the starting points of the clusters. It then performs the process to optimize the position of the centroids until it either A). it achieves the defined number of iterations, or B). the centroids are stabilized and there are no change in the values anymore. To do k-means clustering, you need to define the number of centroids you want (aka ‘k’ number of centroids) and the algorithm will allocate all data points to the closest cluster. Once either one of the two stopping conditions mentioned above are achieved, the algorithm will then be completed. An article published in “Towards Data Science” back in 2018 explains the very simple process a little more in-depth and provides examples as well.

B2, Summary of Technique Assumption: There are a few assumptions regarding this technique that are important to take into consideration. One of those assumptions is that “Kmeans assumes spherical shapes of clusters (with radius equal to the distance between the centroid and the furthest data point)…” (Dabbura, 2018) It’s important to take note that the clusters are assumed to be spherical in shape. If they aren’t, issues might arise when trying to use this technique. KMeans uses distances between centroids and the furthest points, any shape that is not spherical is going to cause issues to arise. These shapes might not allow for good clusters to be determined, thus leading to potentially inappropriate clusters and information regarding the data.

B3, Packages or Libraries List: There were multiple different packages and libraries that were used in the analysis. One of the main ones was pandas which allows for us to handle the dataset and read in the data. The next was numpy which allows for any mathematical calculations and operations to be performed. KMeans from sklearn.cluster was also used, and this allowed for us to perform the actual K-means clustering algorithm. Seaborn and Matplotlib were imported to allow for the visualization of any graphs and charts necessary. StandardScaler had to be imported in order to scale the data. Finally, silhouette\_score from sklearn.metrics was imported because you aren’t able to use the traditional methods for getting the accuracy of a clustering technique. Silhouette\_score allows for us to do just that.

C1, Data preprocessing: One of the preprocessing goals of the data preparation was to make sure that the data was standardized. The reason being, “…clustering algorithms including kmeans uses distance-based measurements to determine the similarity between data points, its recommended to standardize the data to have a mean of zero and a standard deviation of once since almost always the features in any dataset would have different units of measurements” (Dabbura, 2018). Due to the variables that were being used having different measurements, the data needed to be standardized in order for the clustering technique to be used.

C2, Dataset Variables: The variables used in the kmeans clustering analysis performed were ‘Age’ and ‘Income’, both of which are continuous variables. ‘Age’ represents the age of the customer based on the sign-up information while ‘Income’ represents the annual income for said customer.

C3, Steps for Analysis: There were a few steps that were taken in order to perform the k-means clustering technique. The first was a new dataframe was created with the code below:

A white rectangular object with a black border

Description automatically generated

What this code allowed was to set up a dataframe with the specific variables that were being used in the clustering technique, rather than keeping all the data that was deemed unnecessary. The next step was to standardize this new dataframe:

A white background with black text

Description automatically generated

The data had to be standardized due to ‘Age’ and ‘Income’ being two different measurements. One was the customers age in years, while the ‘Income’ variable was measured in dollar amounts and was much larger.

C4, Cleaned Dataset: See copy of cleaned dataset uploaded separately.

D1, Output and Intermediate Calculations: KMeans clustering is a rather simple clustering technique that helps with the clustering analysis. After cleaning and standardizing the data, the optimal number of ‘k’ had to be determined. To do this, a visualization for the optimal number of ‘K’ value had to be created. It was determined that the optimal number of clusters was 3, as shown by the visualization below. Three was determined by looking for where there was a sort of elbow pit, on the graph.

A graph with a line

Description automatically generated

After determining the optimal number of clusters to be 3, that value was then put into the KMeans() equation as the n\_clusters. The data was then fitted and predicted to determine the prediction values that were then labeled as clusters. See D2 for all the code used for the clustering technique.

A screenshot of a computer

Description automatically generated

D2, Code Execution: See code below and in jupyter notebook attached that was used for KMeans clustering as well as the output of the code:

distortions = []

num\_cluster = range(1,11)

for k in num\_cluster:

kmeans = KMeans(n\_clusters=k, init="k-means++")

kmeans.fit(df\_scaled\_2)

distortions.append(kmeans.inertia\_)

plt.grid()

plt.plot(num\_cluster, distortions)

plt.xlabel("K Value")

plt.ylabel("Distortions")

plt.show()

kmeans = KMeans(n\_clusters = 3)

y\_predicted = kmeans.fit\_predict(df\_scaled\_2)

df\_new['Clusters'] = y\_predicted

score = silhouette\_score(df\_scaled\_2, kmeans.labels\_)

print(score)

df\_new.groupby('Clusters').mean()

E1, Accuracy of Clustering Technique: While KMeans clustering does not have the typically accuracy score, a way to determine the accuracy is to use the Silhouette score. What the silhouette score does, is it determines the degree of separation between the clusters. This value can range from -1 to 1 with -1 being the sample is assigned to the wrong cluster, 0 being that it is close to the neighboring clusters and 1 being that it is far away from the neighboring cluster. (Dabbura, 2018) The score determined here was 0.424. While this isn’t higher and closer to 1 like you might hope, it isn’t an awful score. A good cluster one would hope would have a score above 0.5, and this score threads that line.

E2, Results and implications: The table below shows the results of the clustering analysis, with three clusters:

A screenshot of a computer

Description automatically generated

The company can see that 3 clusters were indeed created, all with different average ages and average incomes. The first cluster had an aver age of 72 years old and an income around $29.9K. Cluster 2 had an average age of 53 years old and an income around $87.3K. The final cluster has an average age of 35 years old and income around $29.7K. The company can use these clusters to “group” the data into these separate subgroups and get a look into the structure of the data.

E3, Limitation: KMeans clustering is a good way to get a sense of the structure of the data. One limitation of this technique, however, is that the technique works well when the clusters have a spherical-like shape. If the clusters have a complicated shape, kmeans will not work well in clustering the data. KMeans relies on distance between points and centroids to work well in grouping the data together (Dabbura, 2018). If the shape of the clusters are complicated, or even as simple as a straight line, the algorithm will have a difficult time grouping the data together.

E4, Course of Action: There are many things that the company can do with the information gathered from the clustering analysis. If the company wanted to, they can use Kmeans clustering with just the two variables that have been used in the analysis. Using these variables, along with the 3 clusters determined with the Elbow plot, the company knows how their data is structured and can use that information to look into their business and determine if they want to change their business structure to better suite the determined structure. The company can see that one of the clusters has an average age of 53 years old and an income over $87K. The company can take this information into account when determining what types of customers they should be looking for, or can use the information provided in analysis and can even begin to predict certain features based on these clusters.

F, Panopto Recording: <https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=342950a0-92d1-4ab8-9a4c-b045016d4227>

G, Sources for Third-Party Code: There was no outside sources for data or code

H, Sources:

Dabbura, I. (2022, September 27). *K-means clustering: Algorithm, applications, evaluation methods, and drawbacks*. Medium. https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a#:~:text=Since%20clustering%20algorithms%20including%20kmeans,units%20of%20measurements%20such%20as

Towards Data Science. (2018, September 12). *Understanding K-means clustering in machine learning*. Medium. https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1