Project 3: Customer Segmentation

Lara Clasen
Spring Term 2020
LLClasen.github.io

Abstract

This data science project aims to use machine learning to analyze mall customer spending by way of segmenting those customers in various ways. Customer segmentation is a useful mechanism for businesses to have at their fingertips. Companies can utilize such methods to target their consumers based on spending behaviors, gender, age, and many more characteristics. The goal is to find your "best" customer, depending upon what the situation may be. Using this targeted information to create adapted marketing efforts, they increase their chances of appealing to that grouping of customers. One of the best ways to attempt this type of method is by using what is called a K-means algorithm. This algorithm is ideal for clustering unlabeled data, or data lacking a predefined groups or categories, which is what we will be working with. K-means works to separate data into not only relevant groups, but the appropriate *number* of groups as well. This allows the groupings to be made organically. Not only is this a powerful algorithm for analyzing customer behavior, but it has business implications in terms of inventory management, bot detection, and more. One can monitor data once it has been sorted into a group; For example, if a specific data point changes its label over time, this could represent a meaningful signal. The data being used for this project consists of customer information for a business, and it comes from a research page where data has been collected for project purposes.

Customer Segmentation

Data Pre-Processing and Exploration

The data set that I have used consists of customer information for a shopping mall. The features included are age, gender, annual income, and spending score. Each customer is assigned a Customer ID for identification purposes. The spending score is a value provided for each customer which helps us to define their spending behavior. A customer who spends more has a higher score than one who spends less. There is a total of 200 observations in our data set. The file we are working with is named *Mall_Customers.csv* which contains all of our data.

The best way to begin to familiarize ourselves with the data is by running a couple of overview functions on the features. We will run the `head` and `summary` function on the entire set, and then continue by running these functions for some of the specific features (such as Age and Income Level) as well as look at the standard deviations for each. Running the str() function on the data set also gives us a general look at how it is laid out and the type of features we are going to be dealing with.

One great visualization for this data set will be to look at the gender distribution since we have this information for our customers. We can do this by creating a simple bar plot and we see that the number of female customers is higher than male customers. We can visualize this feature in a slightly different way by using a pie chart to show the ratio between the two genders (Figure 1). From our pie chart we know that the percentage of females is 56% while the percentage of males in the customer dataset is 44%.

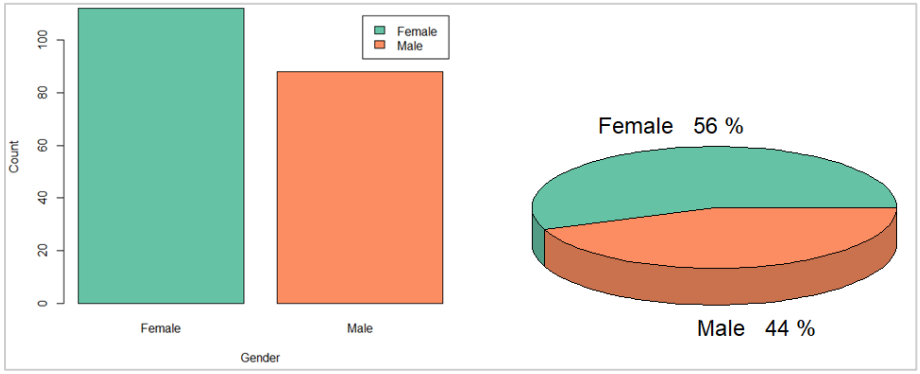

Figure 1: Distribution of the Gender feature

Customer Segmentation

The next feature we want to take a closer look at is customer age. We can look at the age distribution of our customer base by first viewing a summary of this feature, followed by another simple visualization, a histogram, of this data. A boxplot is also a useful way to represent this feature because it is a way of summarizing a set of data measured on an interval scale. It works by showing the shape of the distribution, its central value, and its variability. From these two visualizations we find that the maximum customer age is 70 while the minimum age is 18 years old.

Our third feature is that of the customers' annual income which can be viewed using a basic histogram. From these visualizations we can take away a few things; We can conclude that the minimum annual income is 15,000 and the maximum income is 137,000. Customers who earn an average of 70,000 are the most frequent in our histogram depiction of the data. The average salary of all customers is 60,560, which we found by running our summary() function on this variable. We can see this reflected on both charts. Our density plot displays a Normal Distribution of our data. Normal distribution is a symmetric distribution where the majority of our observations center around a peak, and the remaining values taper off in either direction.

Finally, we can address our fourth variable which is the spending score of our customers. Again, this is a score given to each customer based on their spending history, with a higher score suggesting that the customer spends more than one with a lower score. This feature is already categorized for us in a way, but if it had not already been, we would likely have wanted to "bin" this feature. By binning, we would separate spending amounts into segments and then assign a score based on which level of segment the customer fell into. To look at our spending score feature, we will first create a histogram of the data as well as another box plot. We can see that the minimum Spending Score is 1 and the maximum score is 99. From our summary we see that the average score is 50.20. We see all of these values reflected in the visualizations. Looking at our histogram we can tell that the customers with scores between 40 and 50 are the most frequent among all customers.

Customer Segmentation

K-means Algorithm

Now that we have come to understand our variables and how they are situated within our data set, we can move on to developing our K-means algorithm. K-means is a popular algorithm used for clustering analysis because of its simplicity. This algorithm works iteratively to partition data points into a predefined number of groups, or clusters, such that no point belongs to more than one group and points within each cluster are the most similar to one another. It is commonly used for many applications such as image segmentation, document clustering and, like in our case, customer segmentation.

The first thing we want to do when creating a K-means algorithm is to indicate how many clusters, k, we want in our final output. The algorithm then selects that defined number of objects at random from our data set, which gives us our initial cluster centers, or centroids. These centroids serve as the mean values for each of our clusters. Now, every other data point in the data set is assigned to a cluster based on its nearest centroid. This step is called “cluster assignment”, and this closeness is defined by the Euclidian Distance that is present between the object and each centroid. This measurement is defined as the “measure of the true straight line distance between two points” (Trevino, 2016).

Once the cluster assignment is complete, the algorithm calculates new means for each group based on its components. Using these new mean values, the cluster assignment process begins again to determine if there are points which need to be relocated based on closeness to the newfound mean of another cluster. As soon as the clusters determined by one iteration are identical to the determinations of the previous one, the process comes to an end.

Customer Segmentation

When working with this type of algorithm, it is up to us to first determine the number of clusters to be used. There are multiple methods for assisting us in making these determinations, three of which we will examine next. These three methods are the elbow method, the silhouette method, and the gap statistic method. Using the elbow method, we run the clustering on our dataset for a range of values, in our case from 1 to 10, and then for each value of k, an average score for all clusters is computed. We want to calculate the intra-cluster sum of squares, and then plot this based on the number of clusters. In other words, we plot the value of the cost function produced by different values of k. On our plot, the location of a bend is our indication of the optimal number of clusters. From the resulting plot we can anticipate that the appropriate number of clusters is somewhere around 5 or 6, as this is where the bend in the plot seems to appear.

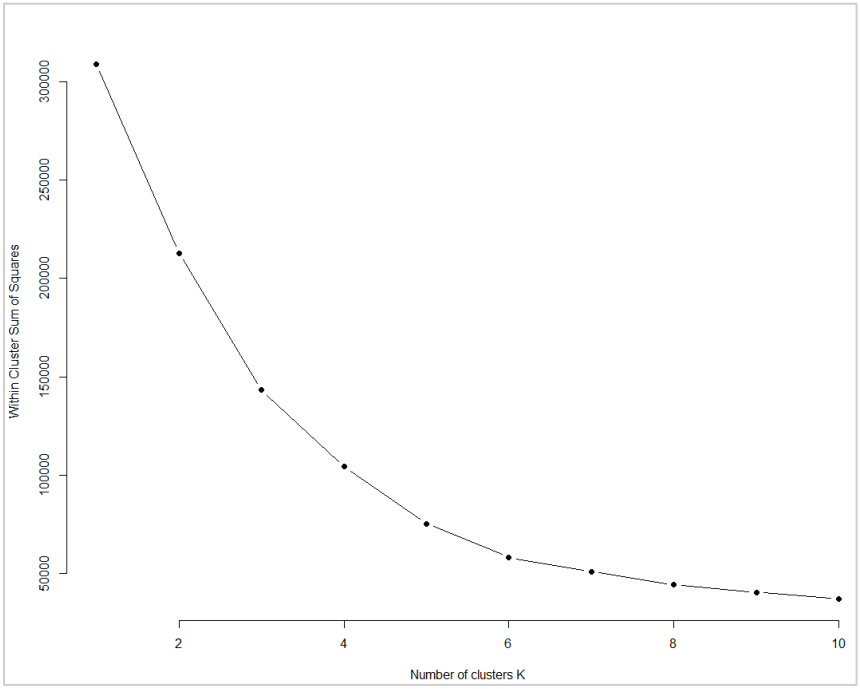

Figure 2: Elbow method plot

The average silhouette method is another option to help us make the determination of how many clusters to use. A silhouette plot displays for us the measurement of closeness for each point in one cluster to points in the neighboring cluster. This measurement has a range from -1 to +1, with

coefficients close to 1 signaling that the point is far from neighboring clusters and that of 0 telling us that the point is on or very close to the boundary of decision. A negative coefficient indicates that our point may have been assigned to the incorrect cluster. Our optimal cluster will have the highest average when we compute the average silhouette width. We will visualize silhouette plots for k values ranging from 2 to 10, and then use a new function to visualize the optimal number of clusters (Figure 3). This tells us that the optimal number of clusters to be used is six.

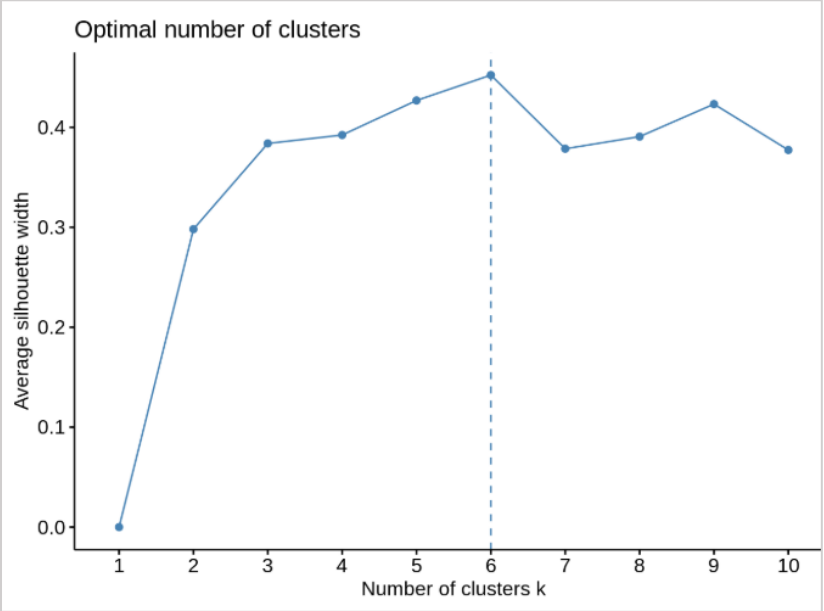

Figure 3: Silhouette method

The third method that we have at our disposal is called the gap statistic method. This method has the flexibility to be applied to other clustering methods, such as hierarchical clustering, as well. This method takes different k values and compares the total *within intra-cluster variation* with their expected values under null reference distribution of the data. The optimal cluster number will be the value at which the gap statistic is maximized. In other words, the clustering structure is far in distance from the random uniform distribution of points. Once again, we are shown that 6 appears to be the optimal number of clusters for our data.

Customer Segmentation

Finally, using our suggested k values from the methods used above, we can implement our K-means algorithm for this data set. We run the `kmeans()` function which performs the algorithm on our data matrix. The resulting table provides quite a bit of information. We are given the sizes of our 6 clusters, along with the means for each and the resulting clustering vector information which indicates the cluster to which each point is allocated. The function also returns a list of components that can be accessed. These components include *cluster, centers, totss, withinss, tot.withinss, betweens,* and *size*. For example, if we run the call `k6$betweens`, we are given the sum of between-cluster squares; By running `k6$tot.withinss`, we see the total intra-cluster sum of squares.

After running our `kmeans()` function we are ready to visualize the results. We will visualize the segmentation based on our features Annual Income and Spending Score. We can clearly see our six clusters and we are able to understand where each group falls in terms of classification based on these two features. For example, cluster 3 represents those customers who have a low annual income as well as a low spending score. These customers likely would not be the target customer for a business looking at marketing strategy. On the other hand, cluster 5 with a high annual income and a low spending score might be just the target for a company who wants to attempt to increase the spending habits of these customers. When we visualize in another manner, using income level and age, we see clear groupings as well, although not quite as clear-cut as with spending score. We can see that cluster 1 shares a similar spending score but has a large age range, while clusters 3 and 5 maintain both a similar age and a similar spending score.

Summary

K-means clustering has offered us the opportunity to analyze this data set with a business mind. By clustering our consumers into segments, we are able to take an focused approach to marketing and other business strategies to increase our reach and effectiveness. K-means has proven to be a valuable method for use with this particular data set, which leads one to consider many other use cases where

this would be a valuable analysis, such as at a grocery or clothing store. By combining multiple methods in assisting our initial cluster quantity decisions, we were able to make the most informed selection in order to obtain a well-developed result. This is an important success for companies all over the world and it is interesting to see how it progresses from step one to a complete analysis with a usable, real-world outcome.

Appendix

Main R packages used:

- plotrix: Offers specialized, easily customizable plots
- ggplot2: Data visualization package for R; Widely used for creating custom plots
- purrr: Functional programming toolkit for R
- cluster: Methods for cluster analysis
- gridExtra: Provides functions to work with grid graphics
- NbClust: For determining the relevant number of clusters in a data set

References

References

1. Choudhary, V. (2018, August 11). Mall Customer Segmentation Data. Retrieved from https://www.kaggle.com/vjchoudhary7/customer-segmentation-tutorial-in-python
    a. This is our main data source.
2. DataCamp. (n.d.). Cluster Analysis. Retrieved from https://www.statmethods.net/advstats/cluster.html

   a. Various cluster methods using the R language specifically.

3. Data Novia. (n.d.). K-Means Clustering in R: Algorithm and Practical Examples. Retrieved from https://www.datanovia.com/en/lessons/k-means-clustering-in-r-algorith-and-practical-examples/

   a. A much more in-depth look at K-means clustering use and practical examples.

4. Fonseca, Luiz. (2019, Aug 15). Estimating the number of clusters in a data set via the gap statistic. Retrieved from https://statweb.stanford.edu/~gwalther/gap

   a. This is a longer article discussing the use of the Gap Statistic Method for compare intracluster variation with expected values.

5. Halabisky, B. (n.d.). Euclidean Distance In 'n'-Dimensional Space. Retrieved from https://hlab.stanford.edu/brian/euclidean_distance_in.html

   a. Explains the basic principles of Euclidean Distance measurement.

6. Hanlon, Annmarie. (2020, April 10). The segmentation, targeting and positioning model. Retrieved from https://www.smartinsights.com/digital-marketing-strategy/customer-segmentation-targeting/segmentation-targeting-and-positioning/

   a. A useful review of how to translate consumer segmentation into a marketing strategy.

7. Nguyen, Tien Anh. (2018, July 3). Customer Segmentation: A Step By Step Guide For B2B. Retrieved from https://openviewpartners.com/blog/customer-segmentation/#.Xp9P1IhKgmI

   a. This is a great general overview of customer segmentation as a business concept, without getting into the programming aspect.

Customer Segmentation

8. Shopify. (n.d.). Customer Segmentation. Retrieved from https://www.shopify.com/encyclopedia/customer-segmentation

    a. Another resource for understanding customer engagement from a business perspective.

9. Team, D. F. (2019, May 3). Data Science K-means Clustering - In-depth Tutorial with Example. Retrieved from https://data-flair.training/blogs/k-means-clustering-tutorial/

    a. This article from Data Flair discusses K-means clustering as an unsupervised learning algorithm.

10. Tibshirani, Robert, Walther, Guenther, and Hastie, Trevor. (2000, Nov). Clustering Analysis in R using K-means. Retrieved from https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967

    a. Another resource for K-means usage and a real-world example.

11. Trevino, Andrea (2016, Dec 6). Introduction to K-means Clustering. Retrieved from https://blogs.oracle.com/datascience/introduction-to-k-means-clustering

    a. Useful description of the K-means algorithm and how it applies to machine learning.

12. Yadav, Jyoti. (2019, Aug 16). Selecting optimal number of clusters in KMeans Algorithm(Silhouette Score). Retrieved from https://medium.com/@jyotiyadav99111/selecting-optimal-number-of-clusters-in-kmeans-algorithm-silhouette-score-c0d9ebb11308

    a. Using the silhouette method to visualize with k-means clustering.