E-Commerce Customer Clustering

ECE 225A Final Project

Panayu Keelawat (PID: A59003809)
pkeelawa@ucsd.edu

ABSTRACT

E-commerce has been disrupting brick-and-mortar stores for years. Since the beginning of 2020, it has even been accelerated by the global pandemic. It seems that this trend is not going to slow down any time soon, so gaining insights from e-commerce customers is crucial. In this work, e-commerce customer clustering was performed on a public dataset using *k*-means and RFM analysis. The results showed that $k = 8$ was the best parameter for clustering obtaining a silhouette score of 0.3956. These 8 clusters have customer counts of 759, 618, 675, 780, 316, 444, 432, and 314. From clustering results, some patterns could be noticed. For example, two groups could be spotted as loyal customers, who purchased more frequently than other groups. Big spenders, who spend more than £700 per receipt, could also be seen as well. However, this work can be improved by scrutinizing the time of year, types of items purchased, and quantity per item.

1. INTRODUCTION

E-commerce has revolutionized our way of living. We can now mostly order products online and wait for them to be delivered directly to our homes. This phenomenon has disrupted the world's retail ecosystem and made several shopping malls file for bankruptcy [1]. Some might call this ongoing event the "retail apocalypse." In addition, due to the acceleration caused by the global pandemic in 2020, e-commerce has become one of the fastest-growing business sectors during this time. The rise of online shopping happens because of the promotion of the social-distancing campaign. Also, in various cities, there are lockdowns, which boost up the online transaction even more.

However, the rise of e-commerce has long been noticed by analysts for decades. Chen et al. have collected UK-based online shopping transactions since 2010 [2]. Although typically e-commerce datasets are proprietary, his group made actual transactions during 2010 and 2011 available publicly through The UCI Machine Learning Repository [3]. Afterward, the dataset was uploaded to Kaggle [4].

The dataset has received great attention from the Kaggle community. Some members use this dataset for exploration, classification, or clustering. For instance, Daniel analyzed segments of customers from this dataset [5], Fink predicted sales based on the learned knowledge [6], etc.

It is evident that we can gain a lot of useful information from this dataset. If we know more about possible clusters of customers, it will enable us to serve each customer with their specific needs. Although some data scientists have already performed customer clustering, there are still rooms for discovery. In the field of market research, one of the great tools for customer segmentation is the RFM model [7]. The term RFM stands for Recency, Frequency, and Monetary. Another method that has received substantial attention to is the *k*-means clustering. These two methods can be applied together to acquire the best model.

It is undeniable that everything is going to be more reliant on technology. The e-commerce market grows rapidly every year, and this trend is not going to slow down any time soon. Knowing more about types of e-commerce customers is for any company in the market. Gaining insights from customers will help improve the service, and companies will be able to serve a better experience to customers.

2. BACKGROUND

2.1 *k*-means clustering

k-means clustering is an algorithm for partitioning data into segments. It is one of the most popular unsupervised machine learning algorithms. Basically, it aims to group similar data points together under the constraint of fixed *k* clusters. The center of each cluster is referred to as a centroid. The objective is to minimize the within-cluster sum of squares. Thus, it can be written formally as:

$$\text{argmin}_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} ||\mathbf{x} - \boldsymbol{\mu}_i||^2 \quad (1)$$

where k is the number of clusters, S is a set of S_i which is a set of observations in the i^{th} cluster, $\boldsymbol{x}$ and $\boldsymbol{\mu_i}$ are a data point and the mean of points (centroid) of S_i.

There are several algorithms that can be applied to solve this problem. Nevertheless, this project was implemented using scikit-learn, which uses either Lloyd's or Elkan's algorithm [8].

2.2 Silhouette score

Because *k*-means is an unsupervised algorithm, there are no labels to evaluate its performance. Silhouette score is commonly used to determine suitable *k* for clustering. It can be computed as:

$$\frac{b - a}{\max(a, b)} \quad (2)$$

where a is the mean intra-cluster distance, and b is the mean nearest-cluster distance. Therefore, this score measures how far the data points are from the nearest cluster compared to their own cluster on average. A higher silhouette score is more desirable than a lower one.

2.3 RFM analysis

RFM is a method commonly used by market researchers to analyze customer value. This method considers 3 dimensions of customers, Recency – the number of days since the last purchase, Frequency – the number of purchases in a certain period, Monetary – the total amount of their spending. The ideal customer should have 3 high values. To be specific, they should have recently purchased, have high frequency, and spend a lot. In order to obtain quantitative values representing these virtues, analysts typically rank customers and determine ranges to assign a class to each customer's value. In this work, quartiles were used to represent classes of RFM.

3. DATASET OVERVIEW

The dataset size is 45.6 MB, and it consists of 541909 rows and 8 columns. Rows reflect types of items purchased by a customer. The columns are InvoiceNo, StockCode, Description, Quantity, InvoiceData, UnitPrice, CustomerID, and Country. Due to the fact that this dataset was collected from a UK-based company, the majority of customers come from the UK. In this work, related columns are InvoiceNo, Quantity, UnitPrice, and CustomerID, as they are considered as RFM values. Table 1 below describes their details.

Table 1. Description of related columns

Column name	Description
InvoiceNo	Unique invoice number for each receipt
Quantity	Number of items purchased
UnitPrice	Price per 1 unit of item
CustomerID	Unique customer ID

4. METHODOLOGY

4.1 Data cleaning

Before feeding data into the *k*-means model, cleaning and verifying the correctness of the data are essential. Rows that are null are removed. Some rows contain negative Quantity or UnitPrice values. That is because some customers have returned purchased goods to the store. Since this work focuses on clusters of customers based on RFM, which analyzes customer purchases, rows that have a negative value in one of these columns are neglected.

Recency, Frequency, and Monetary must be computed. As this work used *k*-means clustering algorithm, the total money spent could be replaced by average spending per invoice, because it does not affect the optimization. The most recent transaction in the dataset was used as an offset for others. Considering one customer may have multiple transactions, rows with the same CustomerID were merged by averaging other values. It was found there were 4,338 unique CustomerIDs.

Numbers should be normalized before *k*-means clustering; otherwise, the algorithm would be less effective. For example, the lowest purchase frequency is one, while the highest is 209. Money spent per invoice ranges from £3.45 to £84,236.25. Therefore, normalization was performed by segmenting values from each column into quartiles.

4.2 *k*-means and visualization

The function KMeans from scikit-learn [8] was employed to perform *k*-means clustering. The experiment tested 8 candidates, *i.e.,* 2 to 9, for a suitable *k*. After clustering, silhouette scores were calculated by calling a built-in function in scikit-learn, so the performance of the models can be evaluated and compared. The silhouette plot was visualized based on barh from Matplotlib [9] to display the effectiveness of the model. Also, the silhouette score was plotted as a dotted line as well. The depiction of the experimental methodology can be seen in Fig. 1.

5. RESULTS AND DISCUSSION

Silhouette scores from k of 2 to 9 are shown in Fig. 2. It can be obviously seen that $k = 8$ obtained the highest score of 0.3956. The minimum score was 0.3324 based on $k = 4$. Although the silhouette score decreased when $k = 4$, the score overall increased as k increased, and it peaked at $k = 8$. Especially in the range of $k = 4$ to 7, we can see a significant rise in silhouette score. The slope of the graph is relatively steep compared to other parts. After $k = 7$, the slope starts to be shallower and lies more along the x-axis.

The best performing k, which is $k = 8$, was selected to be further visualized as a silhouette plot. After using an 8-means model to calculate scores for all data points, they were grouped by clusters and plotted separately with different group colors along the y-axis. The plot can be seen in Fig. 3. It can be noticed that despite large k, the thickness of clusters was just about uniform. Cluster 4 seemed to have the greatest number of customers, while cluster 8 had the least. The vertical green line represents the overall silhouette score, which equals 0.3956 as mentioned before. There were no clusters with below-average silhouette scores, so the score was an appropriate portrayal for all clusters.

Clusters can be further analyzed by investigating their statistics, which are shown in Table 2. As you may see that there are some evident differences across clusters, while the number of customers is evenly distributed. Cluster 4 was the largest group having 780 members, but cluster 8 was the smallest group with 314 customers. Cluster 2 had the lowest recency obtaining 330.71 days of latest purchase, while cluster 5 had the highest recency of 90.12 latest days of transaction. Regarding frequency, cluster 4 had the highest frequency of 9.87, and cluster 3 had the lowest frequency of 1.17 on average. It was found that customers in cluster 5 spend the most per receipt at £781.77 on average, while an average customer in cluster 3 spends the least at £164.10.

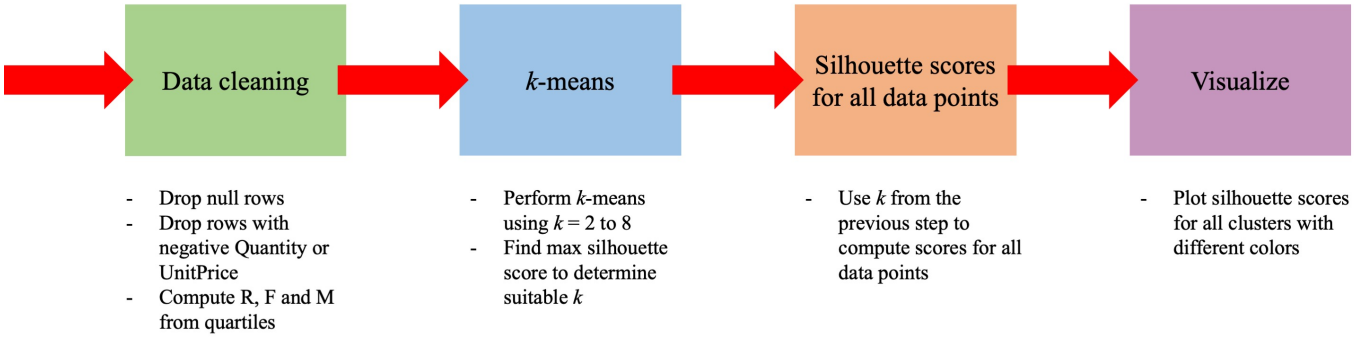

Figure 1. Experimental methodology.

Figure 2. Silhouette scores from diverse *k*.

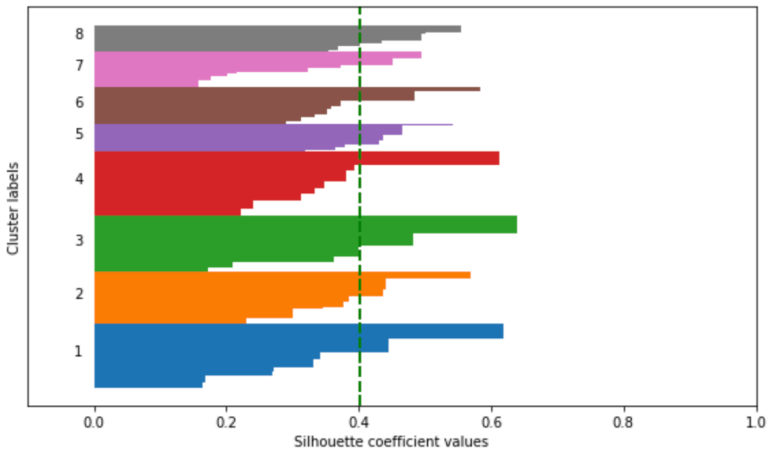

Figure 3. Silhouette plot for all clusters.

Table 2. Cluster statistics

Cluster number	Avg. latest purchase (days)	Avg. frequency	Avg. monetary per invoice (pounds)	Counts (customers)
1	108.34	1.31	701.64	759
2	330.71	8.00	197.03	618
3	90.12	1.17	164.10	675
4	329.28	9.87	603.45	780
5	308.36	1.49	781.77	316
6	314.75	1.44	165.95	444
7	167.28	3.60	181.06	432
8	159.95	4.59	584.83	314

Considering customers in cluster 1, even though they did not purchase frequently, the average payment per receipt was relatively high. They were also fairly new to the system, because they mostly recently purchased. Customers from cluster 2 were customers who often purchased with lower money spent per invoice. However, their latest transactions, for some reason, were comparably high. Probably there were promotions in some period of time that made them purchased frequently in that period. This is possible, because e-commerce companies generally try to lure customers into their platforms with promotions to gain momentum. Customers in cluster 3 were the newest customers. Their frequency and monetary per invoice were still low. In the perspective of recency and frequency, cluster 4 was similar to cluster 2. Nevertheless, they spent much more money than customers in cluster 2 with £603.45 per slip. These customers might have high purchasing power but were careful about their spending. If the price is reasonable, customers of this type will not hesitate to purchase in a considerable amount. Cluster 5 represented big spenders who did not buy things on the platform often and have not recently purchased anything, but they spent the most on one invoice. Customers in cluster 6 have purchased only once on average a long time ago. Cluster 7 portrayed customers who bought relatively frequently with a small monetary per invoice. Customers from cluster 8 would be the most desirable ones, because they have recently used the platform to purchase something, they had high frequency, and they spent much on one receipt.

Although customer cluster 8 is the most preferable cluster, the number of customers in this category is comparably low. Actually, it has the lowest counts among other groups. This cluster should be further analyzed about their purchase behavior. If we know more about their preferences, we can improve the strategy to better serve their needs, and possibly gain more traction from other potential customers. We also should try to bring back big spenders who used to purchase goods on the platform. Most of them were in cluster 4, which is the largest category in terms of customer counts. It will be great if we can find a way to attract more big spenders to the platform. The analysis can be performed by finding patterns in their purchase dates, types of items purchased, quantity per item, and other relevant factors. Serving needs basing on their personalities, or personalized marketing, is a technique that should be implemented to make use of existing databases in the age of big data analytics.

6. CONCLUSION

In this work, e-commerce customer clustering has been performed using *k*-means clustering and RFM analysis. The dataset was obtained from Kaggle, which was collected by Chen et al. Data cleaning was conducted to verify and format the data into computable instances. Rows that were null or had negative Quantity or UnitPrice were removed. Values from RFM analysis, *i.e.*, Recency, Frequency, and Monetary, were calculated before feeding into *k*-means models with various *k* values ranging from 2 to 9. The results showed that the clustering obtained the highest silhouette score of 0.3956 with $k = 8$. The 8-means model was further used to determine silhouette coefficient values of every data point in the dataset. The silhouette plot suggested that the clusters were evenly distributed, and the overall silhouette score was a reasonable representation for all 8 clusters. According to cluster statistics, different behaviors of customers could be detected and analyzed. For instance, big spenders and return customers could be spotted in the results. Cluster 8 represented

customers with the most desirable RFM values. However, the total counts in cluster 8 were the least among other categories. We could further analyze the dataset in order to know more about customers, so that we could improve the quality of service and serve the best experience to customers. Personalized marketing is also suggested as it is possible in the era of big data analytics.

7. REFERENCES

[1] M. Bain. (2017, 14 Dec. 2020). *America's vast swaths of retail space have become a burden in the age of e-commerce* [Online]. Available: https://qz.com/1032723/theres-much-more-empty-retail-space-in-the-us-than-in-other-countries-on-a-per-capita-level

[2] D. Chen, S. L. Sain, and K. Guo, "Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining," *Journal of Database Marketing & Customer Strategy Management,* vol. 19, no. 3, pp. 197-208, Aug. 2012.

[3] D. Chen. (2015, 14 Dec. 2020). *Online Retail Data Set* [Online]. Available: https://archive.ics.uci.edu/ml/datasets/online+retail

[4] Carrie. (2017, 14 Dec. 2020). *E-commerce data* [Online]. Available: https://www.kaggle.com/carrie1/ecommerce-data

[5] F. Daniel. (2018, 19 Dec. 2020). *Customer segmentation* [Online]. Available: https://www.kaggle.com/fabiendaniel/customer-segmentation

[6] L. Fink. (2020, 19 Dec. 2020). *E-commerce sales forecast* [Online]. Available: https://www.kaggle.com/allunia/e-commerce-sales-forecast

[7] A. M. Hughes, *Strategic database marketing*. Chicago, IL, USA: Probus Publishing, 1994.

[8] scikit-learn. (2020, 19 Dec. 2020). *K-means clustering* [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

[9] Matplotlib. (2020, 19 Dec. 2020). *barh* [Online]. Available: https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.barh.html