Report

About dataset:

We will do our analysis based on online-retail dataset. It contains all the transactions occurring for a UK-based non-store online retail between 01/12/2009 and 09/12/2011. The company mainly sells unique all-occasion gift-ware.

Problem statement:

In this project I have made my own problem statements and have tried to solve those using different techniques.

i. Based on customer behaviour we try to achieve customer segmentation. This is an unsupervised learning technique therefore KMean is used to achieve customer segmentation.
ii. Making recommendation system based on country. This is achieved by using the apriori algorithm.

This report contain five major parts. These are:

i. Cleaning and pre-processing
ii. Basic Analysis
iii. Cohort Analysis
iv. RFM analysis (Customer segmentation)
v. Recommended system using apriori

1. Cleaning and Pre-processing:

We will first start the cleaning and pre-processing of the dataset. To achieve this following steps are taken:

i. Deleted all null values as most of the null values were Customer ID as our most analysis will be based with respect to Customer ID it's better to delete the null values instead of assuming and imputing it.
ii. Deleted rows where the items have been cancelled.
iii. Deleted duplicates rows.
iv. Created a new column 'Revenue' by using formula quantity * price.
v. Converted 'InvoiceDate' to datetime format and then made a new column where only month and year ('month_year') are extracted from the InvoiceDate.
vi. After doing initial cleaning and pre-processing we have 779495, 9 as our dataset shape.

2. Basic analysis:

i. After grouping 'Revenue' by 'month_year' following observations were made:

a. There is a sharp increase in sales in the month of 'November'. This could be the reason due to people buying Christmas gifts in the month of November.
b. A sharp decrease in December could be the reason as people already bought the gifts in November.

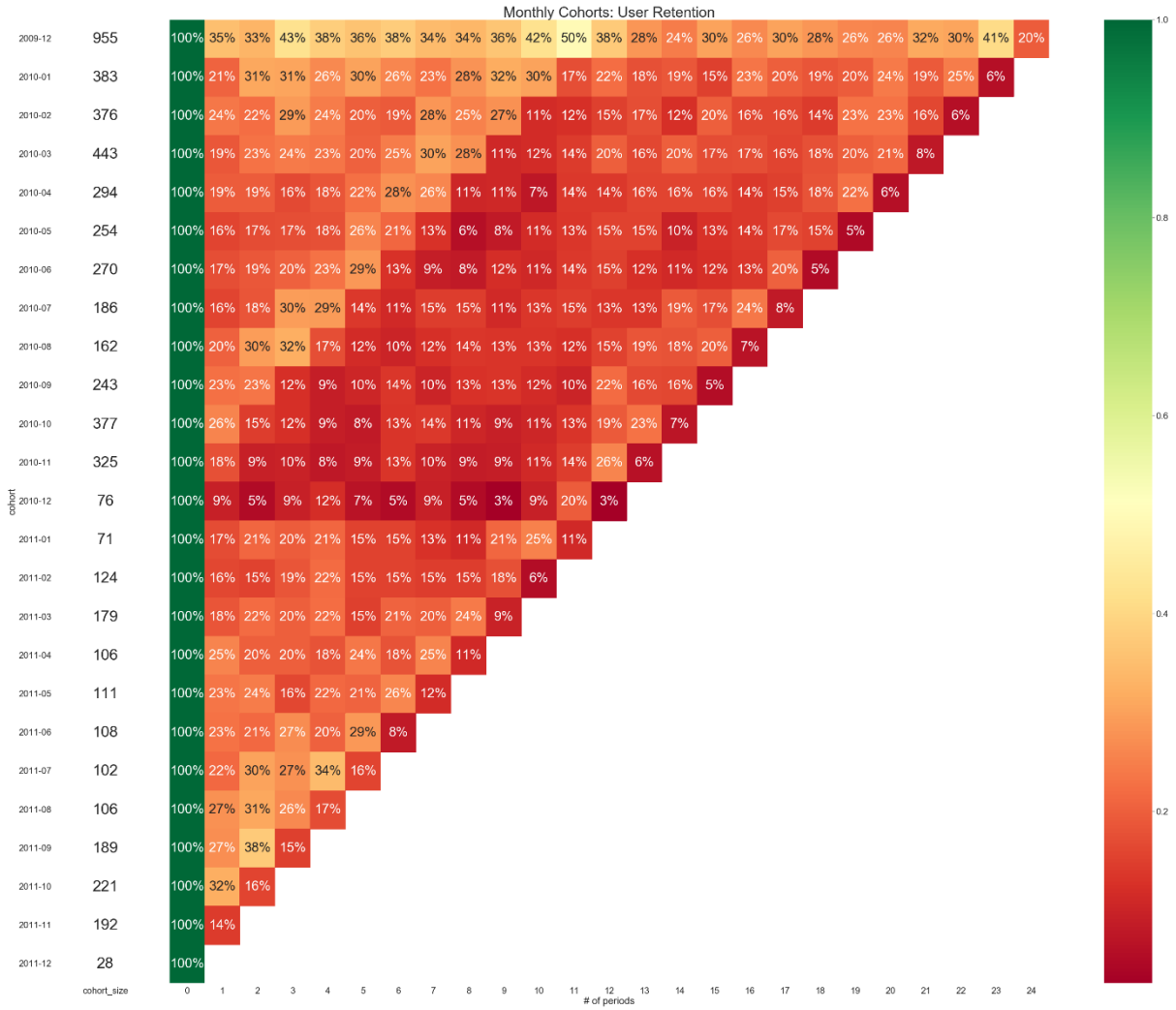
ii. There is a total of **5881** unique customers in our dataset.
iii. **72.35%** of customers have ordered more than once.
iv. The top three countries which purchased the most are the United Kingdom, Germany, and EIRE.
v. The top three products which gives the maximum revenue are 'REGENCY CAKESTAND 3 TIER', 'WHITE HANGING HEART T-LIGHT HOLDER', 'PAPER CRAFT , LITTLE BIRDIE'
vi. The top three product which are sold the most are 'WORLD WAR 2 GLIDERS ASSTD DESIGNS', 'WHITE HANGING HEART T-LIGHT HOLDER', 'PAPER CRAFT, LITTLE BIRDIE'.
vii. More than 80% purchased are done from the United Kingdom still top three customers which spent the most are from the United Kingdom, Netherlands and EIRE.

3. Cohort Analysis:

It is a part of behavioural analytics that examines data from the large set into related groups before analysis.

i. First, create a new column named 'cohort' which takes the minimum InvoiceDate of each customer. Since we are not provided with the first date when a customer visited the website I'm assuming the first day of the transaction as the first day visit to the website.
ii. 'order_month' is the column where only month and year from InvoiceDate is extracted.
iii. A new data frame is created with respect to 'cohort' and 'order_month'. It has two more columns namely 'n_customer' which is the number of unique customers for that cohort and order_month. Another column is 'period_number' which is the difference of 'order_month' and 'cohort'.
iv. After pivoting it we achieve a cohort table for retention since we have 25 months from 2009-12 to 2011-12 therefore we get a period ranging from 0 to 24.
v. To get the retention matrix we divide the whole column with its first column respectively.
vi. It can be observed we have received many Nan values basically these are future values which is not possible to predict.

Analysis from retention matrix:

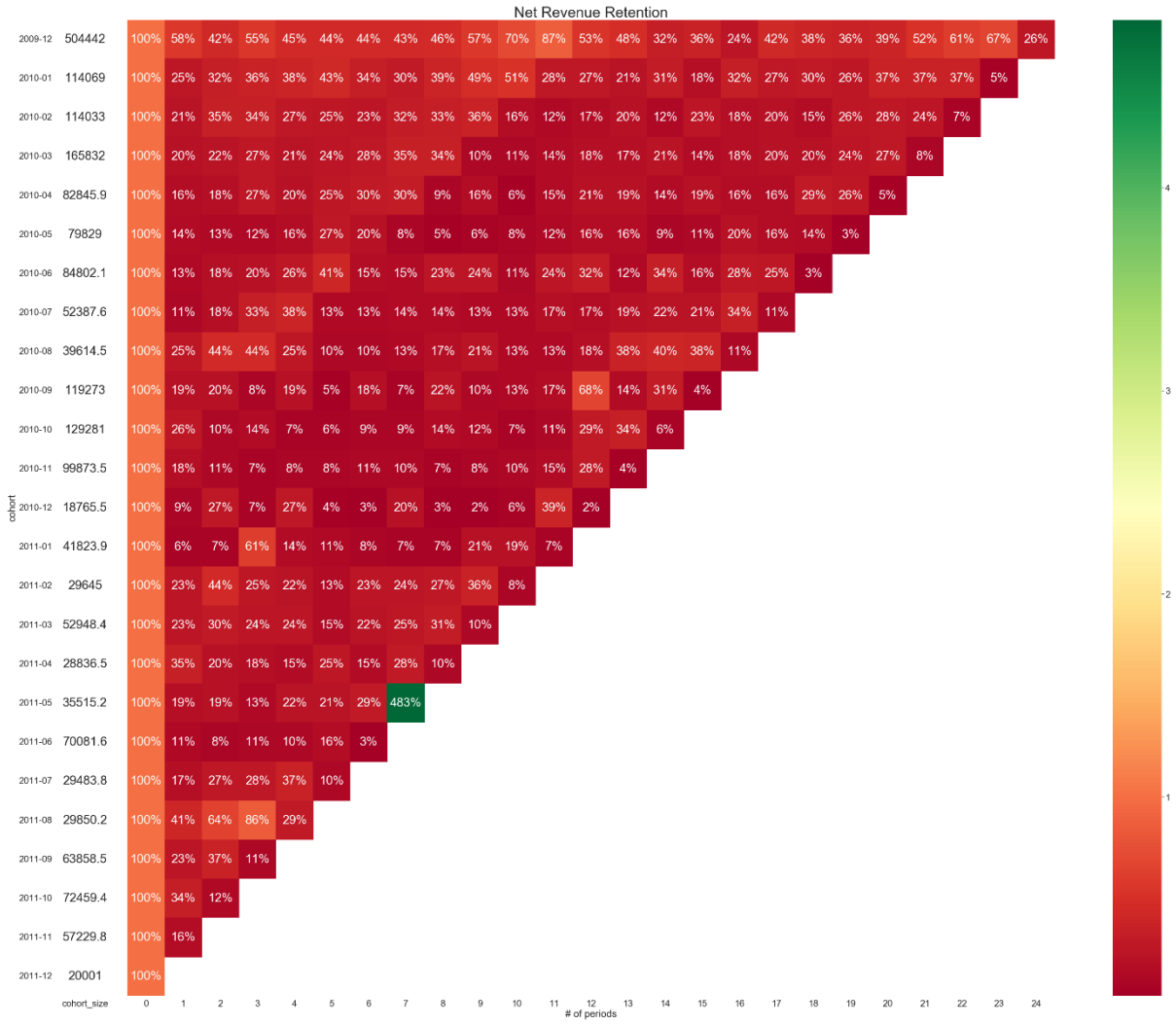


i. Customers who joined in 2009-12 out of 955 customers only 35% of it showed up in its followed month and 33% of 955 showed up in 2010-02. In a similar way cohort analysis of this retention matrix is done.

ii. Customer joined in 2010-12 has pretty bad customer retention for its followed months. That is 100%, 9%, 5%, 9%, 12%, 7% and so on.

Revenue Retention:

i. We Group wrt cohort, order_month and aggregating with revenue sum and follows the same steps as mentioned above.



Analysis:

i. One of the major observations is normally revenue retention of December is one of the lowest but customers who did the first transaction on 2011-05 for has 483% revenue retention for December. Maybe some offers were displayed for December 2010.

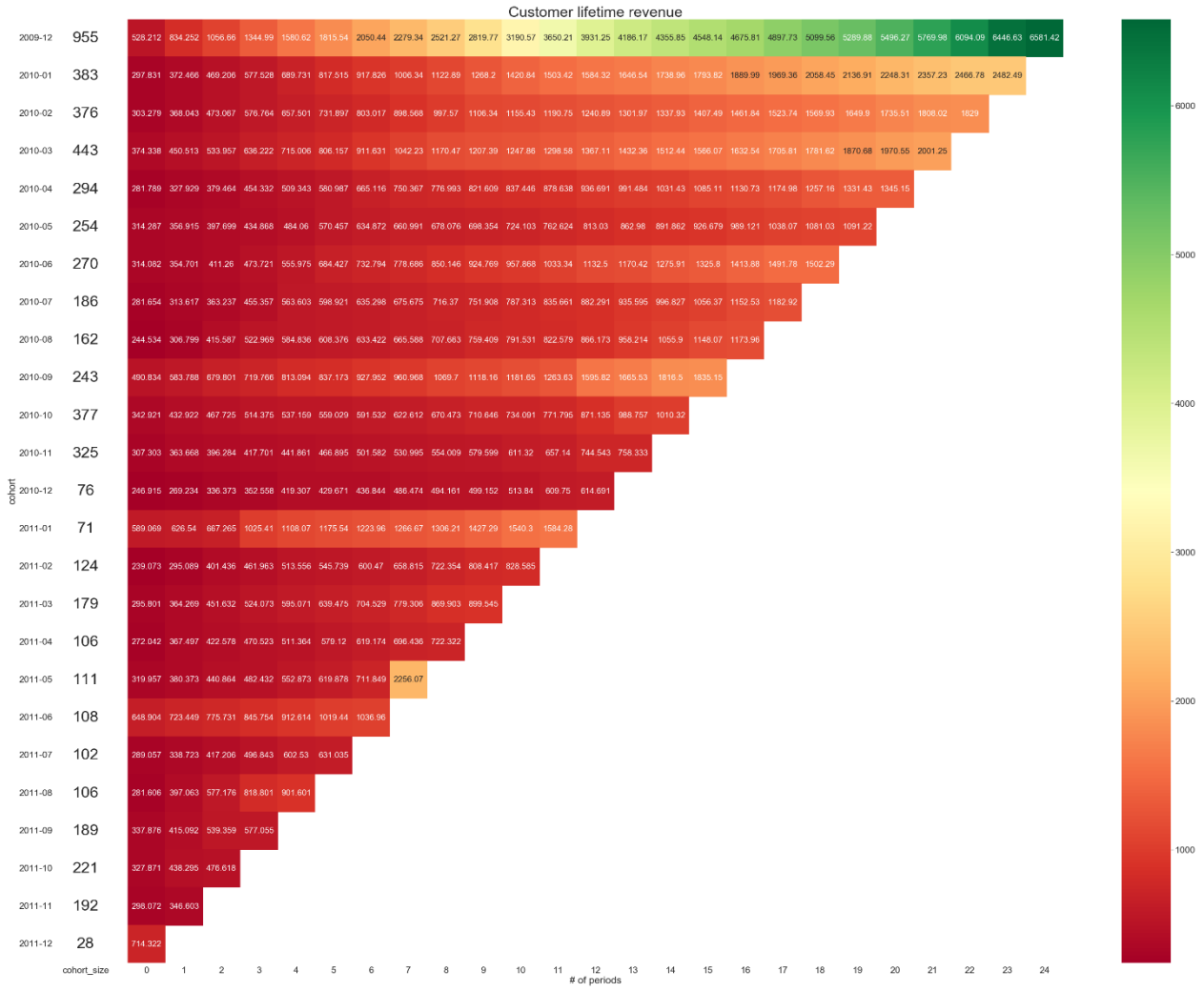
ii. From customer retention and revenue retention matrix, it is observed that for 2011-08 though the customer retention were 27%,31%,26% and 17% for its following month but have observed pretty good revenue retention of 41%,64%,86% and 29% respectively for the followed months. Meaning though there were few customers as compared to their joining date those few customers did spend a pretty good amount.

iii. For customers joined in 2010-09 have highest revenue retention of 68% for its September 2010. This is the highest revenue retention observed for September 2010.

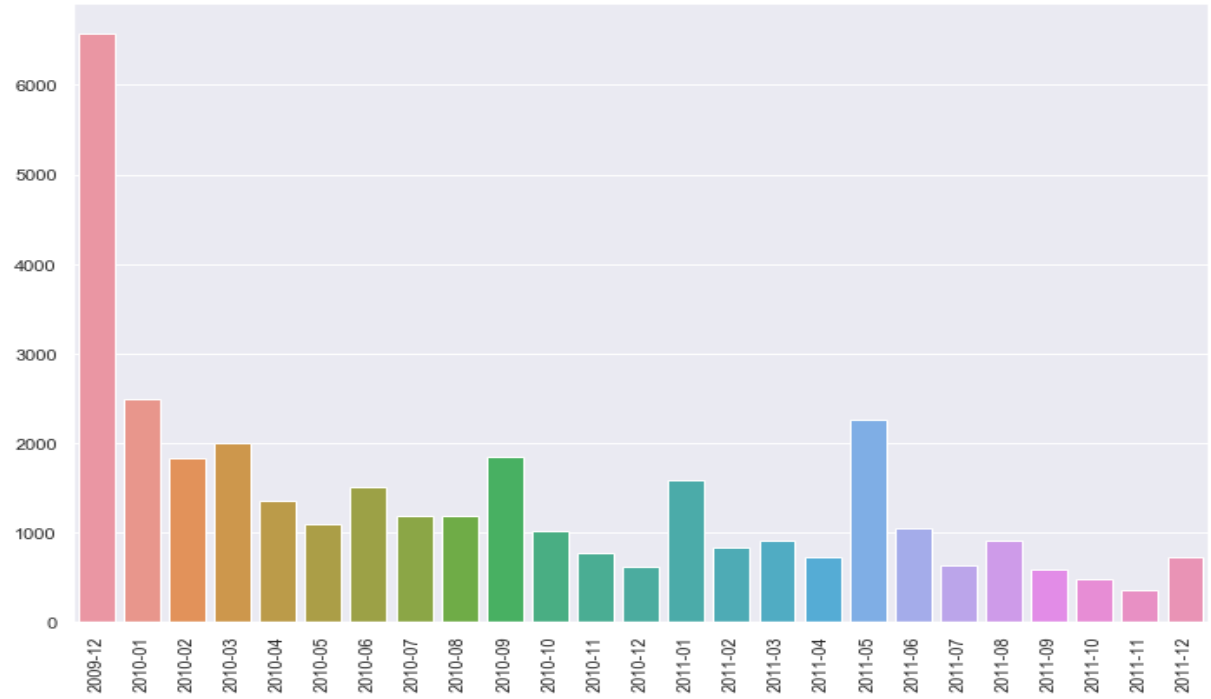
Customer lifetime revenue:

i. Find the cumulative revenue wrt period.

ii. To get the customer lifetime revenue matrix we have to divide the cumulative revenue matrix with total number of customer in each cohort



For each customer we acquired on 2009-12 we made on average 6581.42 at the end. Similarly, we can achieve for other months.



The average revenue per customer is highest for customers joined in 2009-12 followed by 2010-01 and 2011-05.

4. Recency frequency and monetary: (Customer Segmentation)

We will be using one of the most common yet important customer analysis methods known as RFM analysis i.e Recency, Frequency and Monetary.

Recency: Time elapsed since a customer's last activity or transaction.

Frequency: How often has a customer transacted or interacted with the brand during a particular period?

Monetary: How much a customer has spent with the brand during particular period?

To perform RFM analysis following steps were taken:

i. Made three different columns namely 'recency', 'frequency' and 'monetary' with respect to customer id in a new dataset
ii. For recency: Subtraction between the last transaction date and today's date (09-12-2011) is performed. While doing this we can get the number of days he/she has been inactive giving us its recent engage status.
iii. For frequency: It is the total number of unique Invoice per customer, with this we can get how many times a particular customer has purchased the product.
iv. For monetary: It is the sum of total revenue with respect to each customer with this we get the total amount of money spent by each customer.
v. We get a total of 5881 unique customers from the dataset we call this dataset rfm.
vi. Rows containing monetary as 0 is deleted (monetary 0 were observed for a few customers as few of the test product were free of cost)

vii. Columns ‘frequency’ and ‘monetary’ are highly skewed as some months some customers tend to spend a lot like in the month of ‘November’.

viii. Now for simplicity, we label these columns with scores. I’ll be using 5 buckets to divide the overall score for each column.

ix. For recency_score: Customers with lower recency are more valuable than those with high recency for instance someone who shopped 5 days ago is more likely to become a customer than someone whose last seen was 100 days ago. We are using qcut() to achieve this

x. For monetary_score: Customer with high monetary gets the 5 score and customer with low monetary gets 1 score as we again divide the monetary into 5 buckets.

xi. Made a new column which is the concatenation of RFM score. Here I chose to concatenate instead of addition as addition doesn’t give a clear picture out of RFM which one had the most effect. For instance RFM values of 1+4+5 = 10 and 5+4+1=10 so we don’t get the idea is it because the recency score is low and monetary score is high or vice versa.

RFM analysis:

After grouping these columns following observation were made:

recency_score	count	min_recency	max_recency	std_recency	avg_recency
5	1215	0	19	5.912250	8.799177
4	1140	21	58	11.272954	36.140351
3	1180	59	189	39.299417	107.621186
2	1175	190	410	70.411754	311.291915
1	1168	411	738	99.029827	544.557363

It shows that recency_score of 5 has recency range of 0-19.

For monetary score following are the observation made:

monetary_score	count	min_monetary	max_monetary	std_monetary	avg_monetary
1	1176	2.95	285.56	67.338328	162.007092
2	1175	285.60	608.86	91.790177	422.661371
3	1176	609.30	1220.90	176.902027	883.295207
4	1175	1220.91	2909.86	476.950501	1895.428624
5	1176	2910.23	580987.04	30846.980336	11413.072297

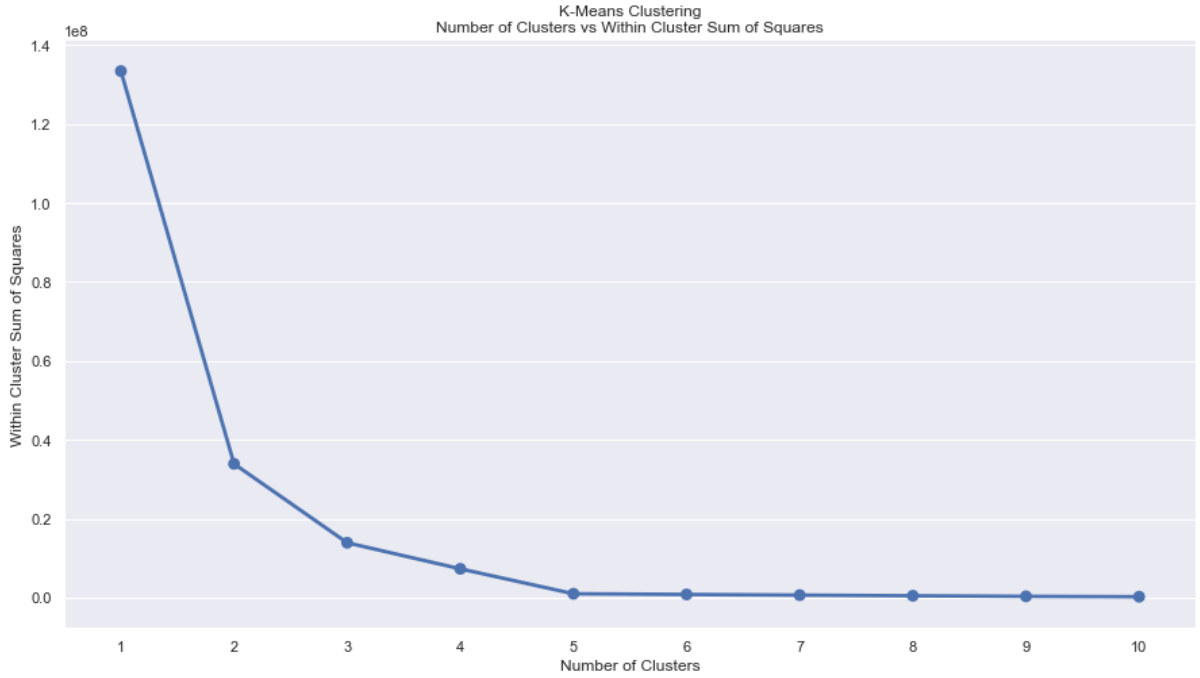
It can be observed that the standard deviation for score 5 is way too high as we had highly skewed data we get. This highlights the potential drawback of using quantile-based discretization on skewed data

For frequency score following are the observation made:

frequency_score	count	min_frequency	max_frequency	std_frequency	avg_frequency
1	1176	1	1	0.000000	1.000000
2	1175	1	2	0.485698	1.619574
3	1176	2	4	0.656592	3.068027
4	1175	4	8	1.228229	5.768511
5	1176	8	398	24.430446	19.988946

Now I have used kmean clustering to find the customer segmentation.

For this I have used elbow method to find the number of clusters.



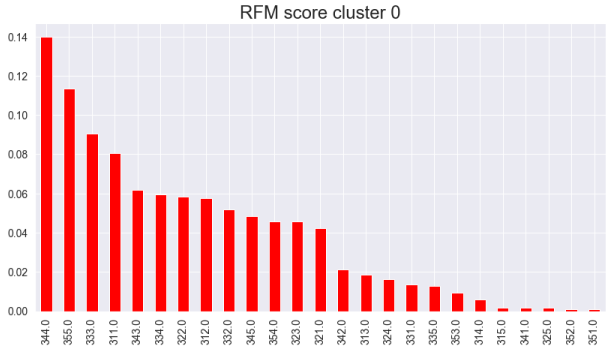
From the above figure, we can think of getting 4 clusters. But to evaluate the number of clusters I have used silhouette score which will give us a clear idea on choosing the number of clusters.

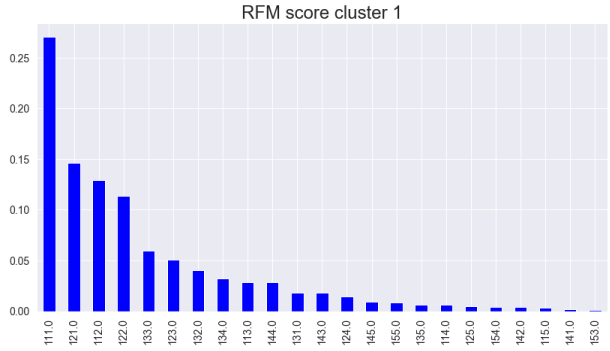
```
For n_clusters = 2 The average silhouette_score is : 0.6475079339078442
For n_clusters = 3 The average silhouette_score is : 0.667817156015989
For n_clusters = 4 The average silhouette_score is : 0.7037999109612837
For n_clusters = 5 The average silhouette_score is : 0.8470104076808928
For n_clusters = 6 The average silhouette_score is : 0.8061947258229936
```

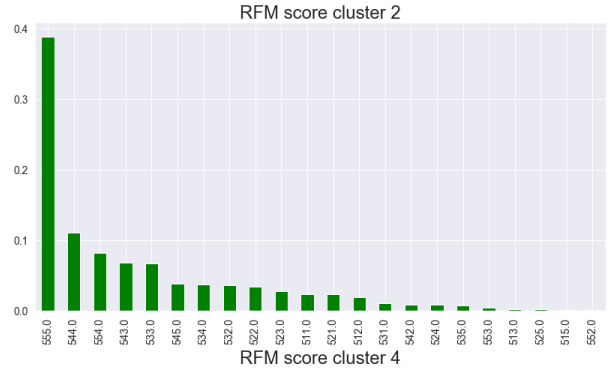
In this we clearly observe choosing cluster 5 is recommended as it has the highest silhouette score.

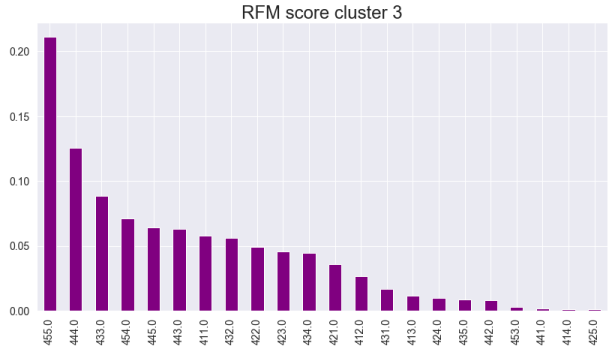
Therefore I choose the number of clusters as 5

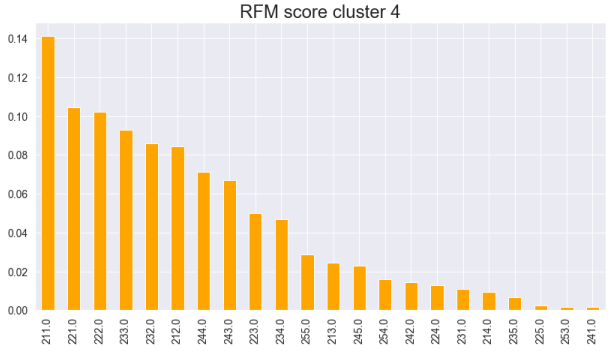
After applying kmean I have received the clusters as 1 < 4 < 0 < 3 < 2 here cluster 2 being the most valuable customer and cluster 1 being the least valuable.











From the above graphs, it's clear that clustering is highly correlated with recency scores. It can be observed that customers who have high recency score have high frequency and monetary score too. Indicating active customers tends to spend more.

Let's try to understand using two clusters cluster 2 and cluster 1 which according to the above analysis is the most important and least important clusters respectively.

Now if you see the top five RFM score of cluster 2 which are:

555,544,554,543,533,545 in this the first digit represent recency score, the second digit represent frequency score and the third digit represent the monetary score. As it can be observed all the three scores are pretty high indicating "Most valuable customers". In this cluster, it is interesting to note that more than 35% of data have a high RFM score of 555.

Similarly, the top five RFM scores of cluster 1 are:

111,121,112,122,133. It can be observed all the three scores are pretty less as compared to cluster 2 indicating “Churned customers”. In this cluster, more than 25% of data have the lowest RFM score of 111.

According to graphs I have classified these clusters as:

Cluster 0: “Promising customers”

Cluster 1: “Churned customers”

Cluster 2: “Most valuable customers”

Cluster 3: “Potential valuable customers”

Cluster 4: “About to churn customers”

I have applied these segmentations in the dataset. Further, this segmentation column can be merged with the original dataset using “Customer ID” as the primary key and we can develop a machine learning algorithm where we will use the merged dataset to train to predict these labels.

5. Recommendation system:

Tried making a recommendation system based on country. It suggests most items bought together. I have implemented this using the apriori algorithm.

In apriori algorithm we basically encounter support, confidence and lift.

Support: Each item’s frequency of occurrence.

Confidence: It refers to the likelihood that an item B is also bought if item A is bought. It can be calculated by finding the number of transactions where A and B are bought together, divided by a total number of transactions where A is bought.

Lift: It refers to the increase in the ratio of sale of B when A is sold. Lift (A –> B) can be calculated by dividing Confidence (A -> B) divided by Support (B).

An antecedent is an item found within the data. A consequent is an item found in combination with the antecedent.

Steps involved implementing this:

i. For this, I have used only the top 5 countries where the sales are maximum.
ii. Created 5 different baskets by grouping with respect to countries mainly United Kingdom, Germany, France, EIRE, France and Netherlands
iii. Basically, each basket represents what items were bought with respect to each transaction.
iv. Anything below or equal to 0 is made as 0 and anything above or equal to 1 is made as 1 (one hot encoding).
v. Then later association rule is applied by using the inbuilt association_rules function and have been sorted with respect to confidence and lift. I have sorted it with respect to confidence and lift as confidence tell us how likely they are bought together i.e how likely item A and

item B are bought together and lift tell us likelihood of buying item A and B together x times more than the likelihood of just buying item B.

vi. min_threshold for lift is kept as 1. Lift of greater than 1 means products A and B are more likely to be bought together. Lift of less than 1 refers to the case where two products are unlikely to be bought together.

vii. Created a function recommender that return the consequents of the requested antecedents.

viii. It is to be noted that for min_support of UK I have taken 0.02 whereas for other countries I have taken 0.05. Looks like UK doesn't have many frequent itemset as compared to other countries.

ix. I have implemented a recommendation system in jupyter notebook where for each country I have tried to show a max of two recommended item for the top 5 product which is sold the most in that particular country.

x. Whereas for UK I'm using its first five antecedents.

xi. For instance items 'POSTAGE','LUNCH BOX WITH CUTLERY RETROSPOT', 'LUNCH BAG APPLE DESIGN' are bought together in France.