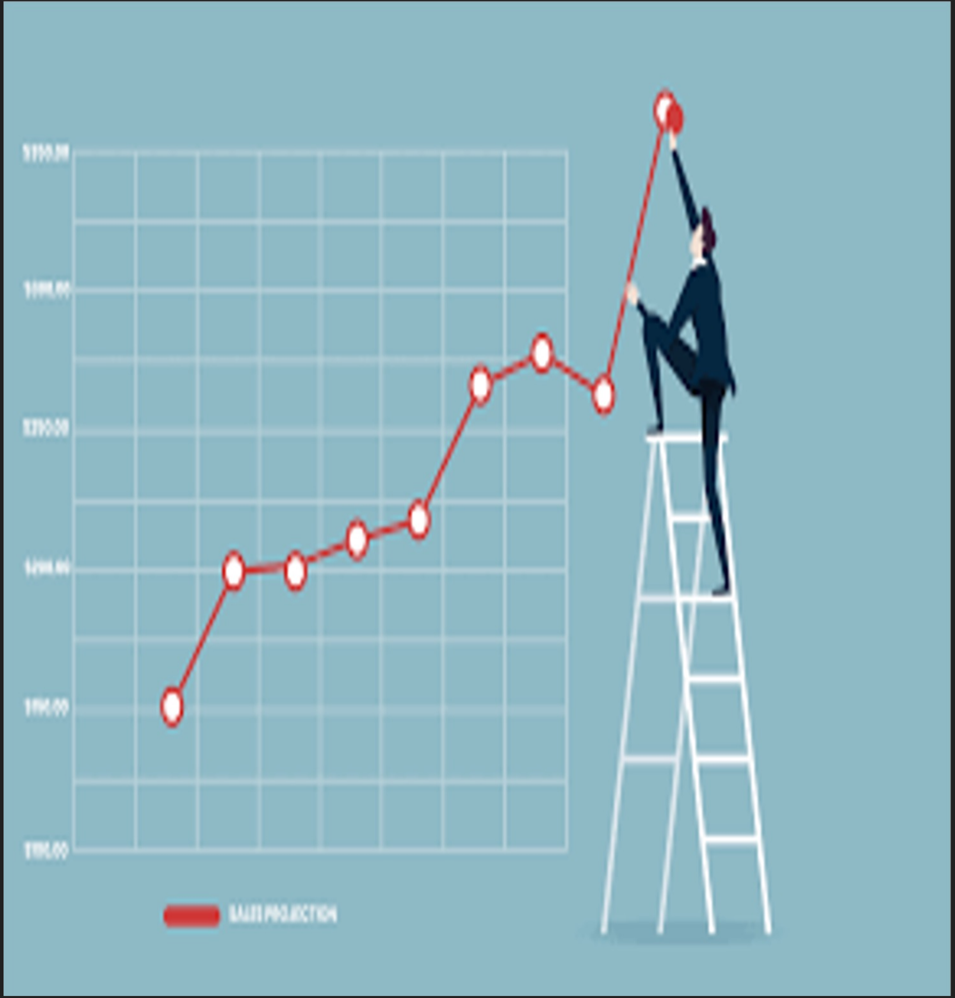


Customer Segmentation

by
Rime Saad

Table of contents:

- Problem Statement
- Reading the Data in Jupyter Notebook
- How the Data Looks Like
- Sales by Country
- Explore Price Column
- Explore Quantity Column
- Explore Description, Stockcode and Customer ID Columns

Customer Cohorts

Problem Statement

How would the business achieve deeper insight into customer profiles and behaviour and higher accuracy sales forecasting

What will be covered in this presentation

- Data cleaning

- Recency, Frequency and Monetary Value and RFM score

- Segmenting customers using kmeans and hierarchical clustering

- Sales Forecasting

Reading the data:

- Create an excel object
  data_excel =
  pd.ExcelFile('../raw_data/online_retail/online_retail_II.xlsx')

- Read the excel sheets
  Data_excel.sheet_names⇒ Year 2009-2010, Year 2010-2011

- Read the data into a dataframe
  data = pd.read_excel(‘online_retail_II.xlsx',sheet_name=[0,1])
  - This is a dictionary with 2 keys
  - Each key has a dataframe
  - Each dataframe has 8 columns and about 500,000 rows

- df= pd.concat(data.values(),ignore_index=True)

The data

Shape:

- 1,067,371 rows and 8 columns
- 5,835 customers and 4,615 products

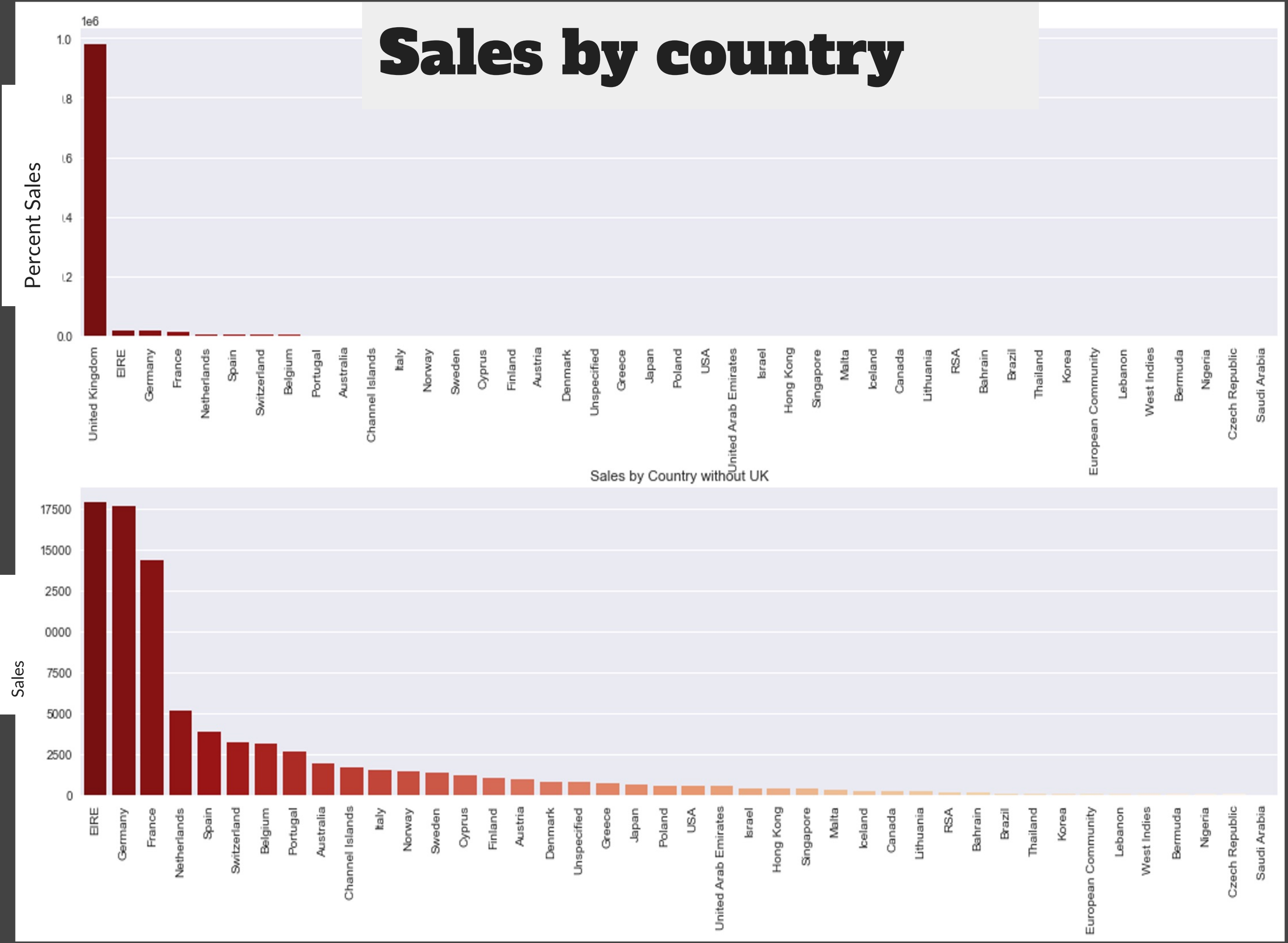
Columns Names:

- InvoiceNumber: number of purchase per customer
- Stockcode: number per product
- Description: description of product, there are 4,382 nans
- Quantity: amount ordered
- Price: price of single product ordered
- InvoiceDate: date of purchase
- Customer ID: 243,007 nans
- Country: 43 countries

There was one new feature created:

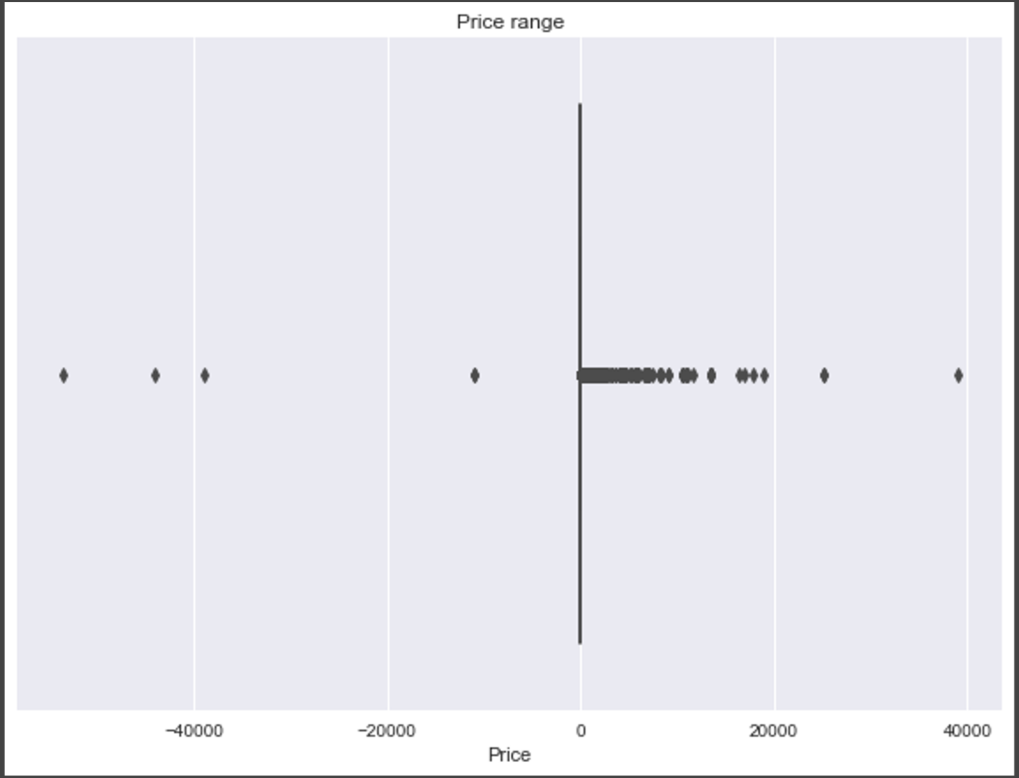
- Sales = quantity*price

Sales by country



Sales by country shows that 90% of the sales are for UK customers, but there are customers in 52 other countries

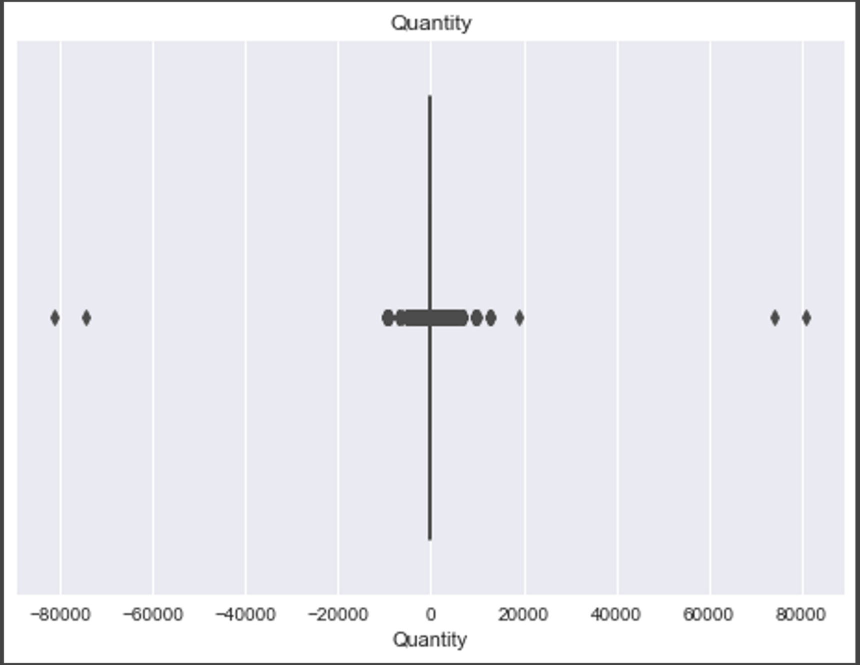
Price



There are some negative and zero prices. Some of the reasons are:

- Some of these items had the description of “adjusted bad debt’
- And some had descriptions that indicated that the item was damaged, so they were returned items
- There were also entries with price equal 0. These entries were deleted they are not considered to be sales

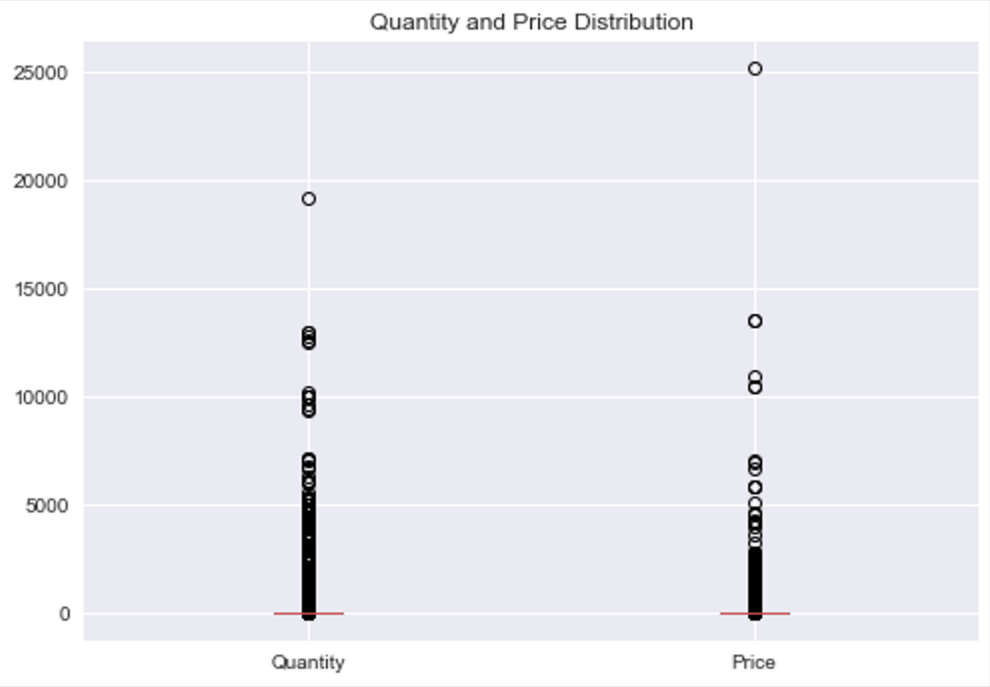
Quantity



There are 19,000 negative quantities that, upon further investigation, turned out to be canceled orders. So there were duplicates in the dataframe where orders were made and then cancelled.
Solution:.

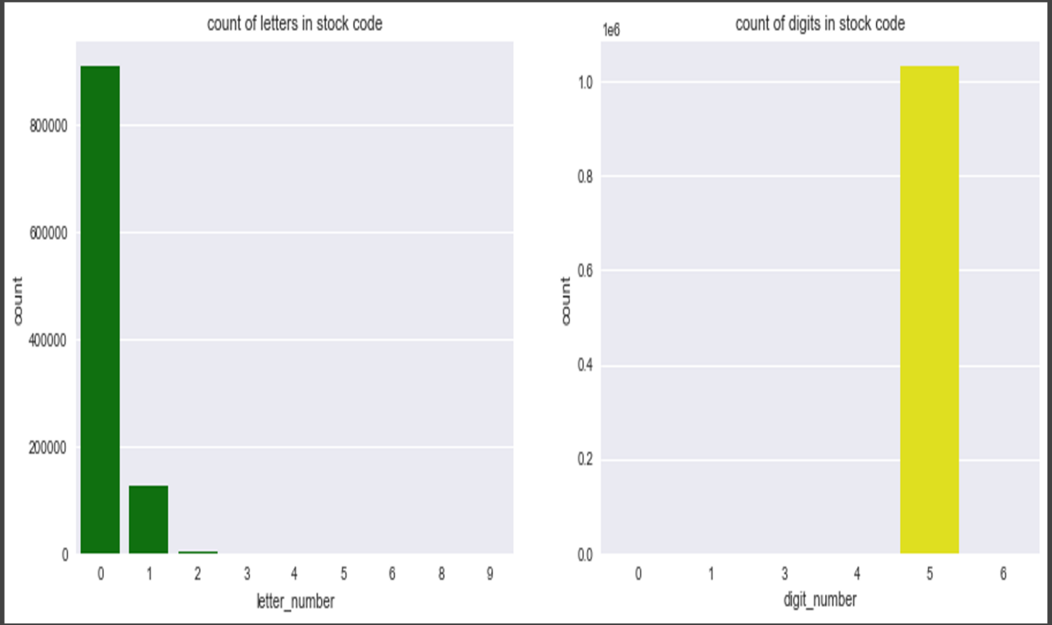
- Two dataframes are created, one for positive quantity and one for negative.
- A function is implemented that takes in the negative quantity and positive quantity dataframes and the columns to check on, iterate through dataframe rows and match on stockcode, quantity, customer id and country, if there a match then these duplicates are removed from the dataframe.

Quantity and Price after cleaning



All positive values and no zeros.

Investigation of the description and stockcode features



There are 4,900 unique stockcodes. To investigate if the code is for an actual product, I looked at the length of the code and made these observations:

- Most stock codes are 5 digits long, some with only numbers and some containing letters. Codes of this length are valid product codes.
- Some examples of codes that were only letters were “test and gift vouchers” and “Test”. Test is deleted but gift voucher is considered to be a sale so it is retained

Description

Most descriptions are all in uppercase but some are a mixture of upper and lower and the latter can contain strange descriptions like:

- gift vouchers
- adjusted
- Check?
- dotcom
- manual

Gift vouchers are retained but all the other ones had to be removed

Customer ID

- 30% of Customers ID is nan
- I divided the dataframe into two, one with customers with nan and one with all other customers
- There are 5,834 customer

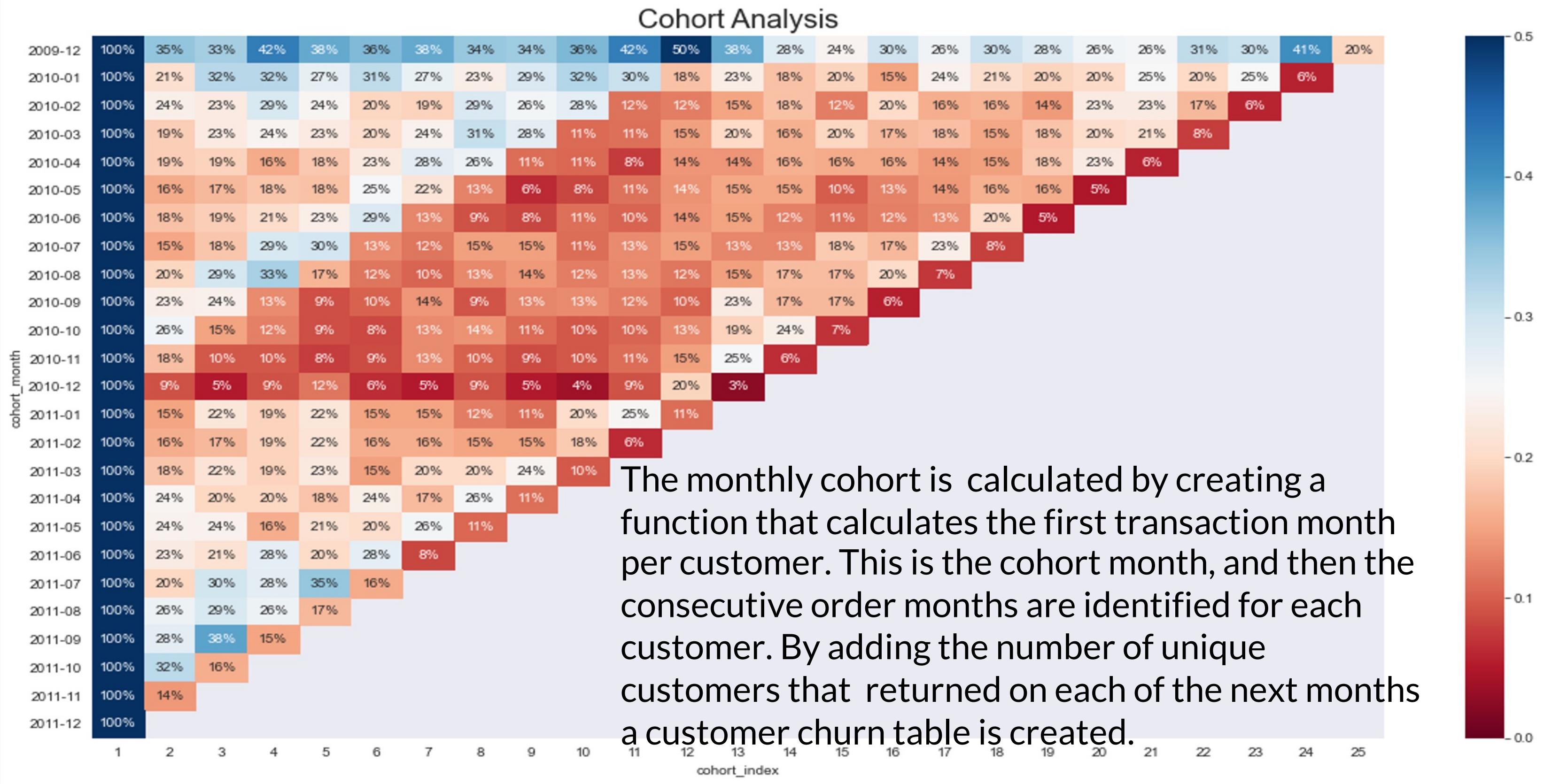
Link to notebook
https://github.com/rime11/customer_segmentation_sales_forecasting/blob/master/Notebooks/data_wrangling.ipynb

Customer Cohorts

Benefits of creating customer cohorts

- Give a better understanding on how customer behaviors affect the business.
- Once customer behavior is better understood the business can reduce customer churn.
- Increase customer lifetime value.
- Increase customer engagement.

Customer churn using customer cohorts chart



cohort_month	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
2009-12	100%	35%	33%	42%	38%	36%	38%	34%	34%	36%	42%	50%	38%	28%	24%	30%	26%	30%	28%	26%	26%	31%	30%	41%	20%
2010-01	100%	21%	32%	32%	27%	31%	27%	23%	29%	32%	30%	18%	23%	18%	20%	15%	24%	21%	20%	20%	25%	20%	25%	6%	
2010-02	100%	24%	23%	29%	24%	20%	19%	29%	26%	28%	12%	12%	15%	18%	12%	20%	16%	16%	14%	23%	23%	17%	6%		
2010-03	100%	19%	23%	24%	23%	20%	24%	31%	28%	11%	11%	15%	20%	16%	20%	17%	18%	15%	18%	20%	21%	8%			
2010-04	100%	19%	19%	16%	18%	23%	28%	26%	11%	11%	8%	14%	14%	16%	16%	16%	14%	15%	18%	23%	6%				
2010-05	100%	16%	17%	18%	18%	25%	22%	13%	6%	8%	11%	14%	15%	15%	10%	13%	14%	16%	16%	5%					
2010-06	100%	18%	19%	21%	23%	29%	13%	9%	8%	11%	10%	14%	15%	12%	11%	12%	13%	20%	5%						
2010-07	100%	15%	18%	29%	30%	13%	12%	15%	15%	11%	13%	15%	13%	13%	18%	17%	23%	8%							
2010-08	100%	20%	29%	33%	17%	12%	10%	13%	14%	12%	13%	12%	15%	17%	17%	20%	7%								
2010-09	100%	23%	24%	13%	9%	10%	14%	9%	13%	13%	12%	10%	23%	17%	17%	6%									
2010-10	100%	26%	15%	12%	9%	8%	13%	14%	11%	10%	10%	13%	19%	24%	7%										
2010-11	100%	18%	10%	10%	8%	9%	13%	10%	9%	10%	11%	15%	25%	6%											
2010-12	100%	9%	5%	9%	12%	6%	5%	9%	5%	4%	9%	20%	3%												
2011-01	100%	15%	22%	19%	22%	15%	15%	12%	11%	20%	25%	11%													
2011-02	100%	16%	17%	19%	22%	16%	16%	15%	15%	18%	6%														
2011-03	100%	18%	22%	19%	23%	15%	20%	20%	24%	10%															
2011-04	100%	24%	20%	20%	18%	24%	17%	26%	11%																
2011-05	100%	24%	24%	16%	21%	20%	26%	11%																	
2011-06	100%	23%	21%	28%	20%	28%	8%																		
2011-07	100%	20%	30%	28%	35%	16%																			
2011-08	100%	26%	29%	26%	17%																				
2011-09	100%	28%	38%	15%																					
2011-10	100%	32%	16%																						
2011-11	100%	14%																							
2011-12	100%																								

The monthly cohort is calculated by creating a function that calculates the first transaction month per customer. This is the cohort month, and then the consecutive order months are identified for each customer. By adding the number of unique customers that returned on each of the next months a customer churn table is created.

Link to notebook
https://github.com/rime11/customer_segmentation_sales_forecasting/blob/master/Notebooks/customer_cohort_analysis.ipynb

RFM Data Frame

RFM is a marketing technique used to rank and group customers based on the recency, frequency and monetary value of their transactions, the purpose of which is to identify the top customers. This is done to be able to perform targeted marketing campaigns. Businesses are able to give their customers a quantitative value that is measurable and easy to understand.

RFM Definition

Recency:

The number of days since the last transaction

code:

```
data_rfm = df.groupby('Customer ID').agg( 'InvoiceDate': lambda x:
(snapshot_date - max(x)).days, 'Invoice': 'count',
                                        'Revenue': 'sum'})
```

Frequency:

The number of transactions in the past period

Code:

```
recency_q = pd.qcut(data_rfm['Recency'],q=4,labels=labels) ⇒
labels = range(4,0,-1)
```

Monetary Value:

How much the customer spent in total in that period

Code

```
frequency_q = pd.qcut(data_rfm['Frequency'],q=4,labels =
labels_f) ⇒ labels_f = range(1,5)
```

How the dataframe looks like

Customer ID Recency Frequency Monetary Value

RFM_segment 529 24 RFM_score

12346 169.36

2 121 222 4

12347 4921.53

75 444 46 12

12348 1658.40

19 323 172 8

12349 3678.69

310 444 10 12

12350 294.40

211 4

This new dataframe will be used to create customer clusters.

Clustering

Clustering is used to find similar data points and then group them together in the same cluster. This process helps create targeted marketing and improves business strategy.

Kmeans clustering:

An unsupervised learning algorithm used to cluster customers according to their purchase history and past activities and is the most popular clustering algorithm. It is a centroid based algorithm, meaning it creates random centers for data and tries to minimize the distances between the points and their centroid. This process is repeated for different number of centroids until the optimal number of clusters is reached.

How it is done?

First, a range of cluster numbers is chosen, then different metrics are used to measure quality of clusters. The purpose is to minimize inter cluster distances and maximize the distance between the clusters.

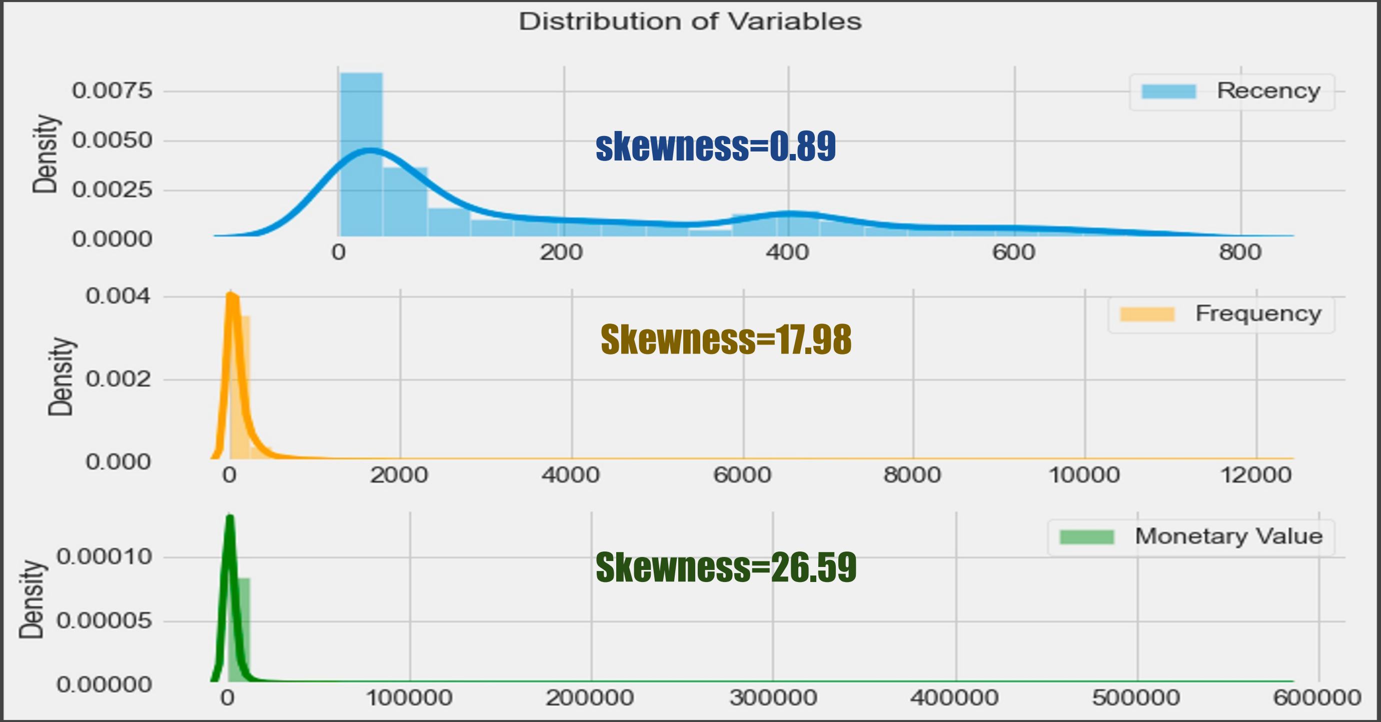
What is kmeans clustering?

Finding the right number of clusters is very important to the analysis, too many clusters and each point will start representing a cluster, too few and the clusters do not represent the data. This is also a very subjective problem and requires some domain knowledge.

kmeans requires some data preprocessing.

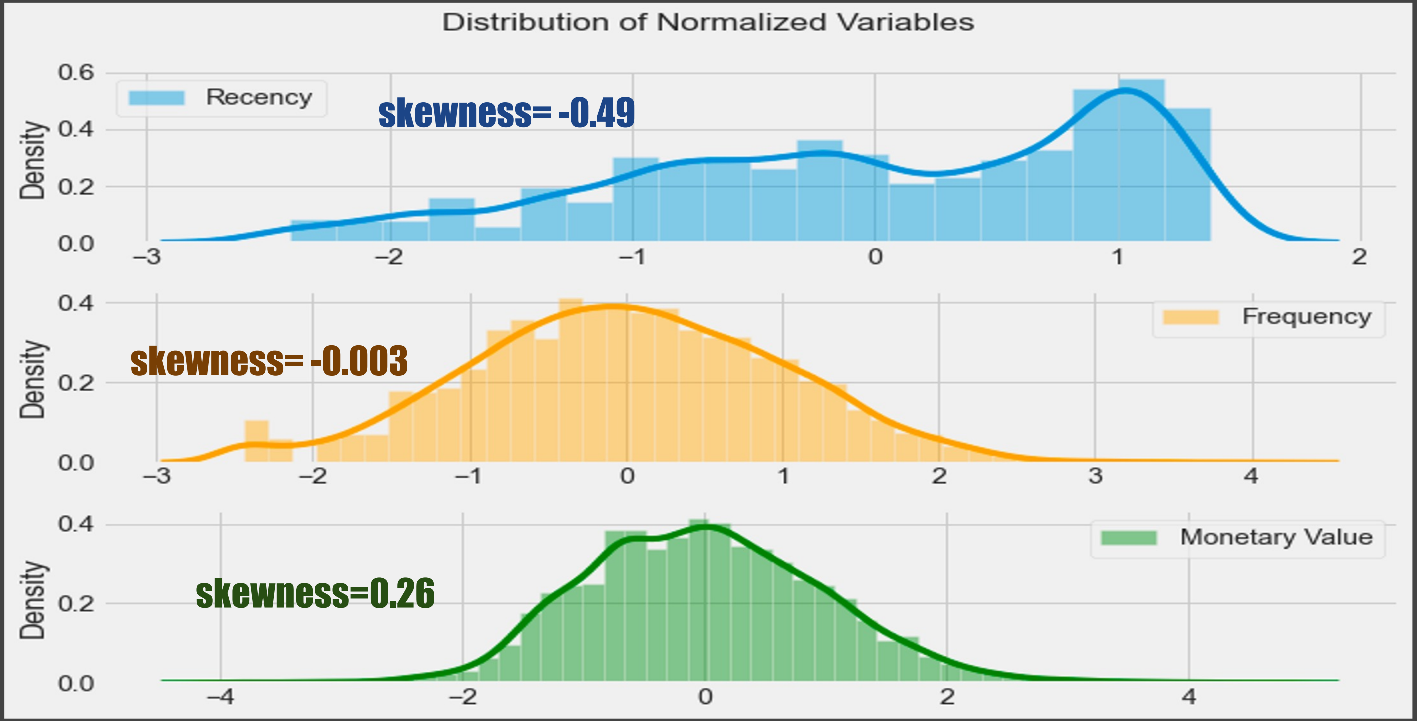
1. it requires that all the features have a symmetric distribution, meaning no skewness
2. variables have the same average value, meaning their means are the same, so they need to be standardized. This ensures that each variable gets equal weight in the kmeans calculation.
3. variables have the same variance.

Explore Distribution of Features for Kmeans



All distributions are skewed to the right. Skewness can be measured using skew() method. It needs to be between -0.5 and 0.5 for data to be symmetrical. One way to fix this is to apply logarithmic transformation.

Distributions after Log Transformation and Standardization of Data

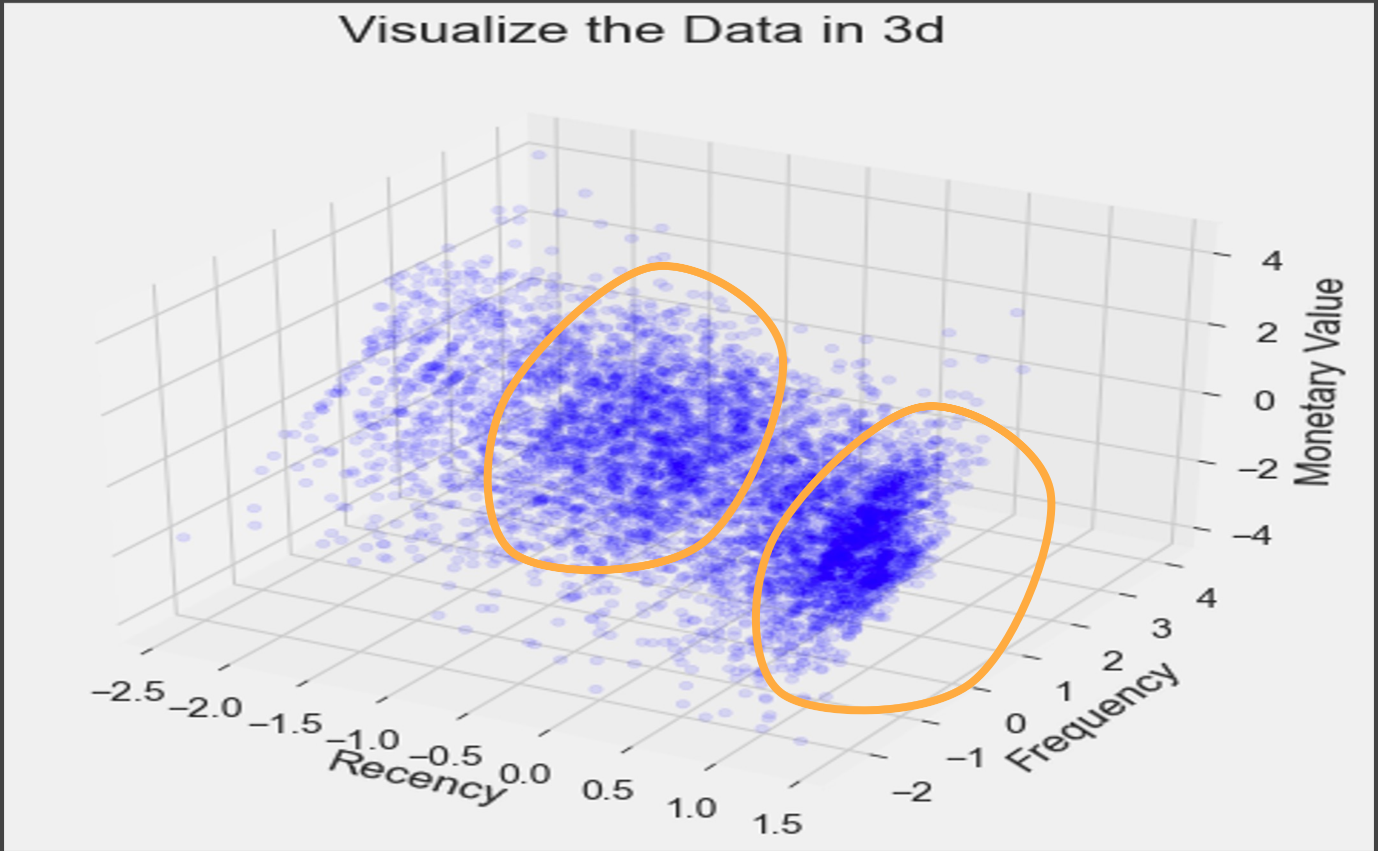


Kmeans and the Hopkins test.

Before running kmeans, which is an unsupervised algorithm, it needs to be determine if the data set contains clusters or no. One of the tests used is the Hopkins test. The steps are:

1. generate data that is randomly distributed across the data space
2. sample actual data
3. measure distance of nearest neighbor between the generated data and the original data, as well as between the sampled data and original data.
4. Hopkins statistic is the distance between the generated data divided by the sum of that distance and the distance of the original sampled data.
   If H = 0.5 then there is no cluster tendency
   if H = 1 or H = 0 then there is cluster tendency

The hopkin statistic is 0.9 which is close to 1 meaning there is cluster tendency in the data.



There are some dense areas in the front and middle

There are three metrics to measure how good the clusters are:

The elbow method

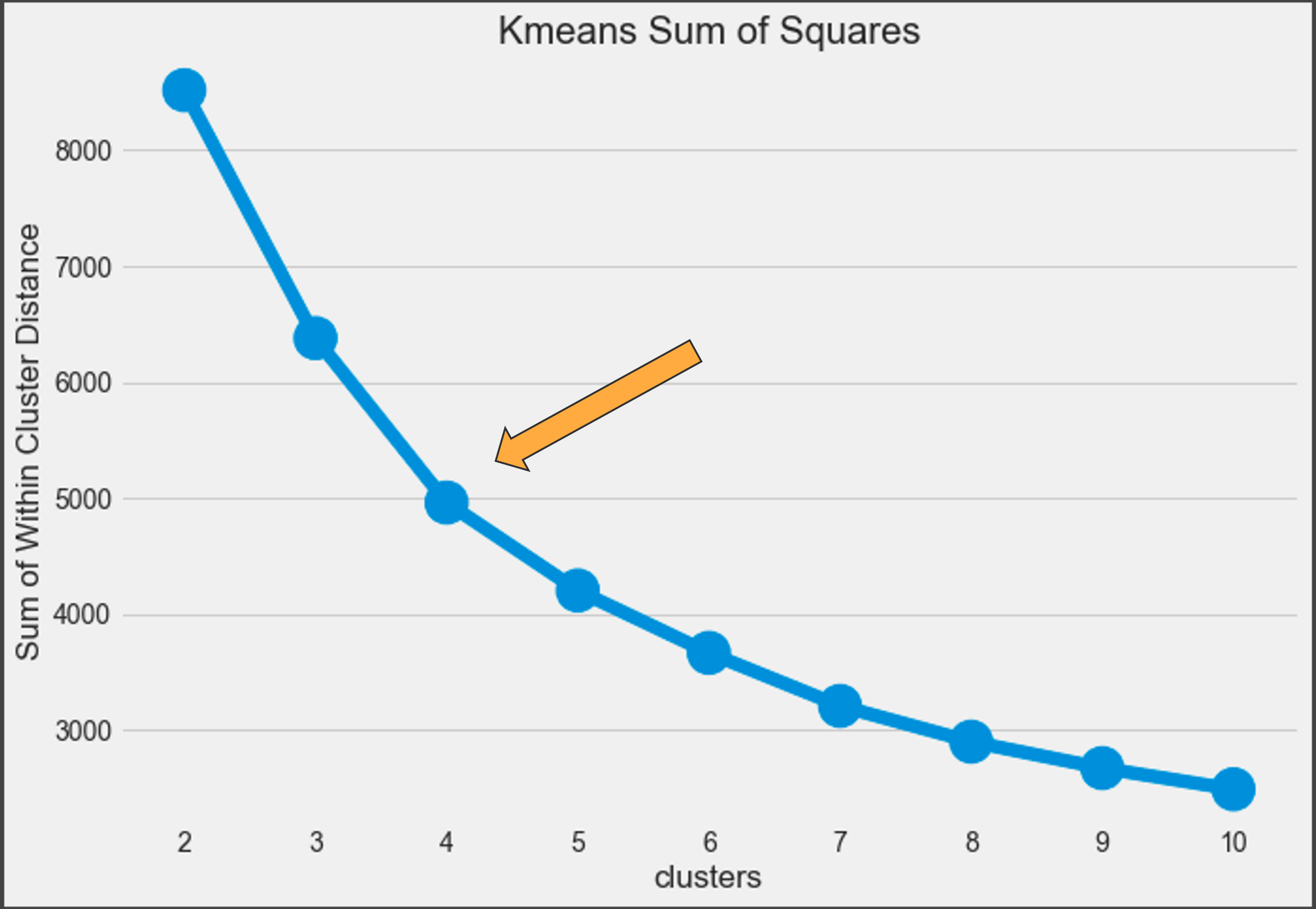
Measures within-cluster variance, or the compactness of the cluster. The lower this value, the higher the compactness of clusters formed. This distance is plotted against the number of clusters. This is called the elbow method because what we look for is a slowing down in the decrease of the distance, which looks like an elbow

The Silhouette Index

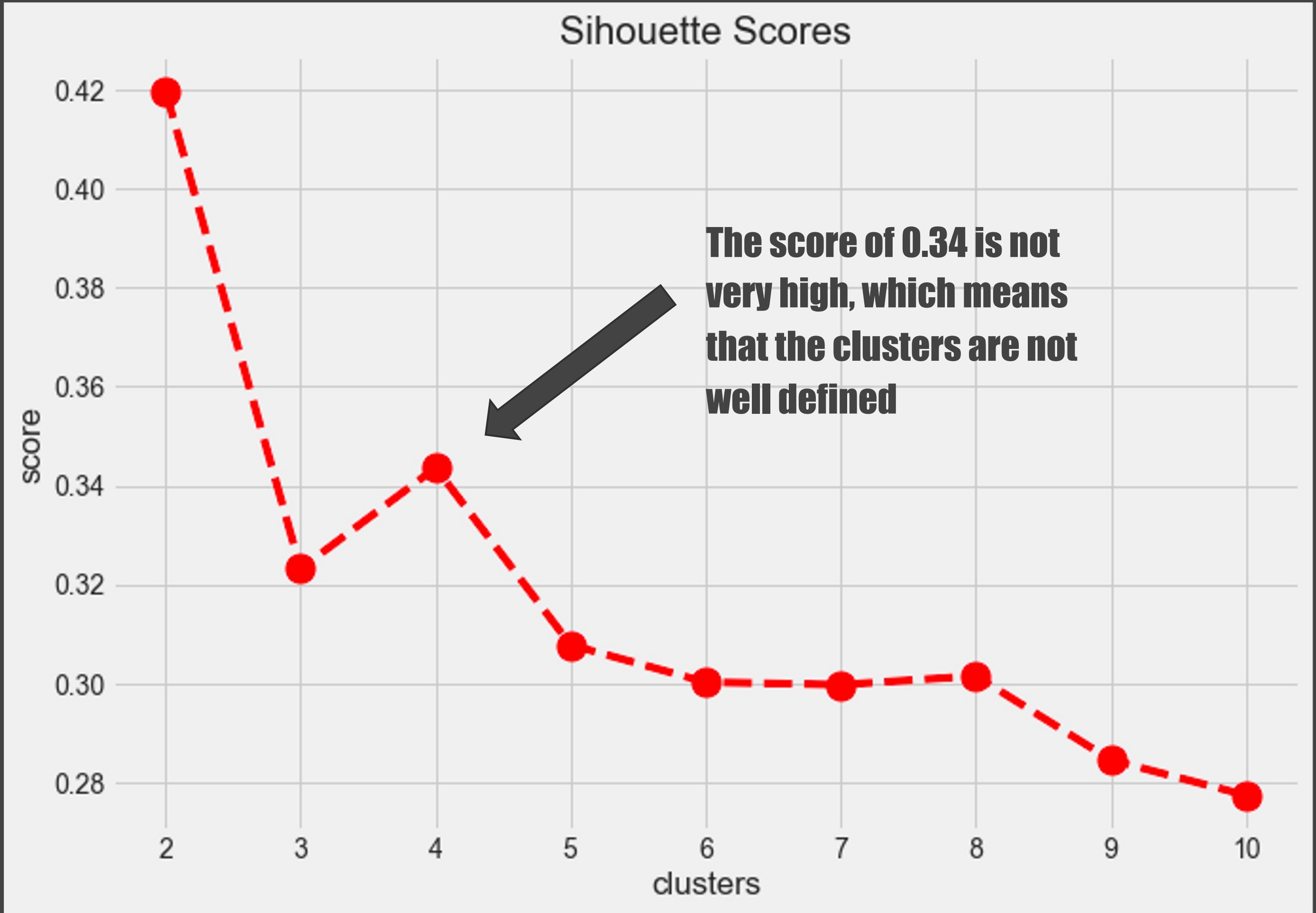
Measures the distance between each data point, the centroid of the cluster it was assigned to and the closest centroid belonging to another cluster. Values close to 0 indicate overlapping clusters, and values closer to 1 indicate a better separated dense clusters. The formula is b-a/max(a,b): b-a is the distance between the centers

David Bouldin score

Measures intra-cluster similarity and inter-cluster differences. Lower value indicates a better model with a better separation between clusters

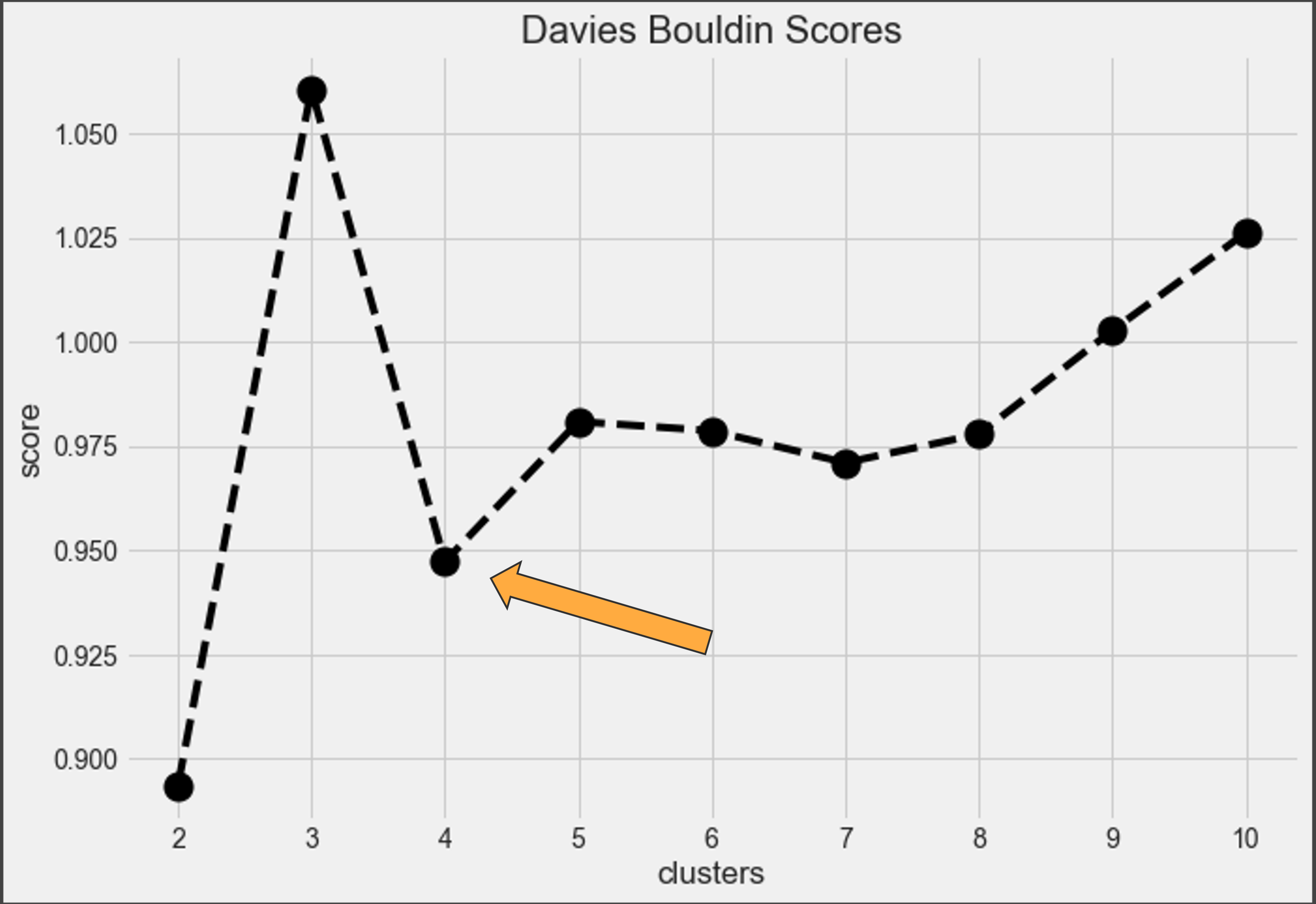


The decrease in distance slows down at 4, but it is not conclusive



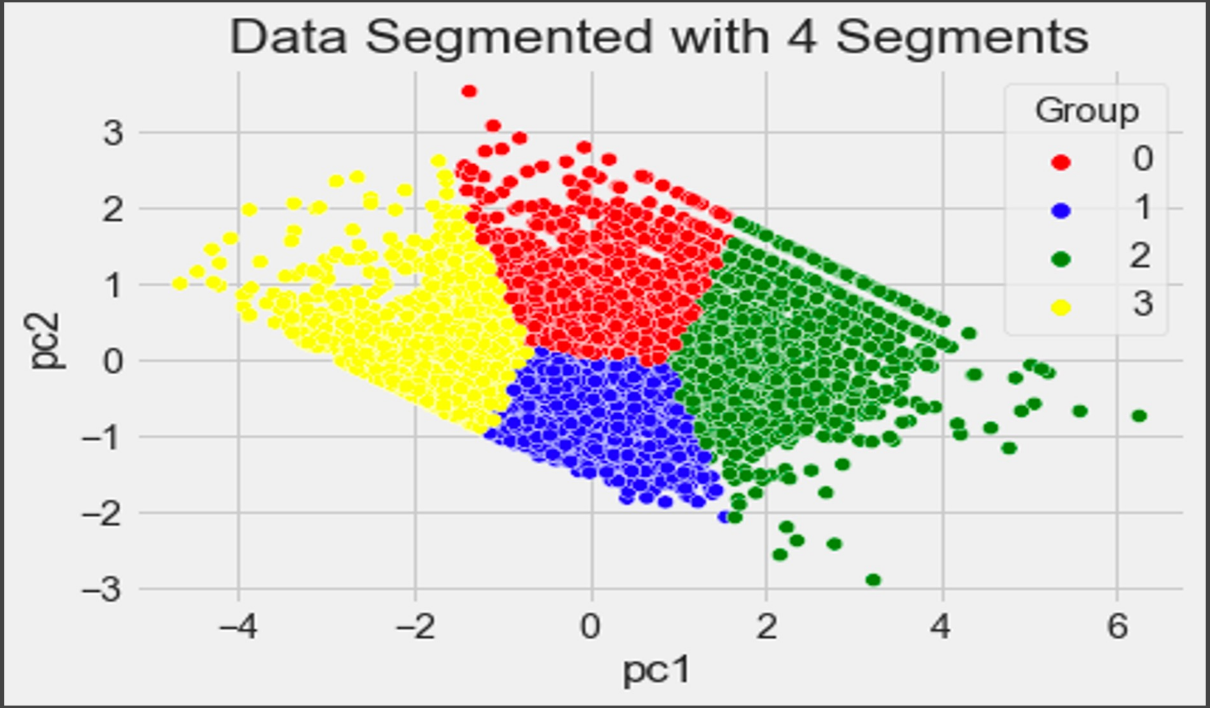
The score of 0.34 is not very high, which means that the clusters are not well defined

The Silhouette score measures the degree of separation between clusters. So I am looking for the highest value. From the graph I see that the local maxima is at 4.



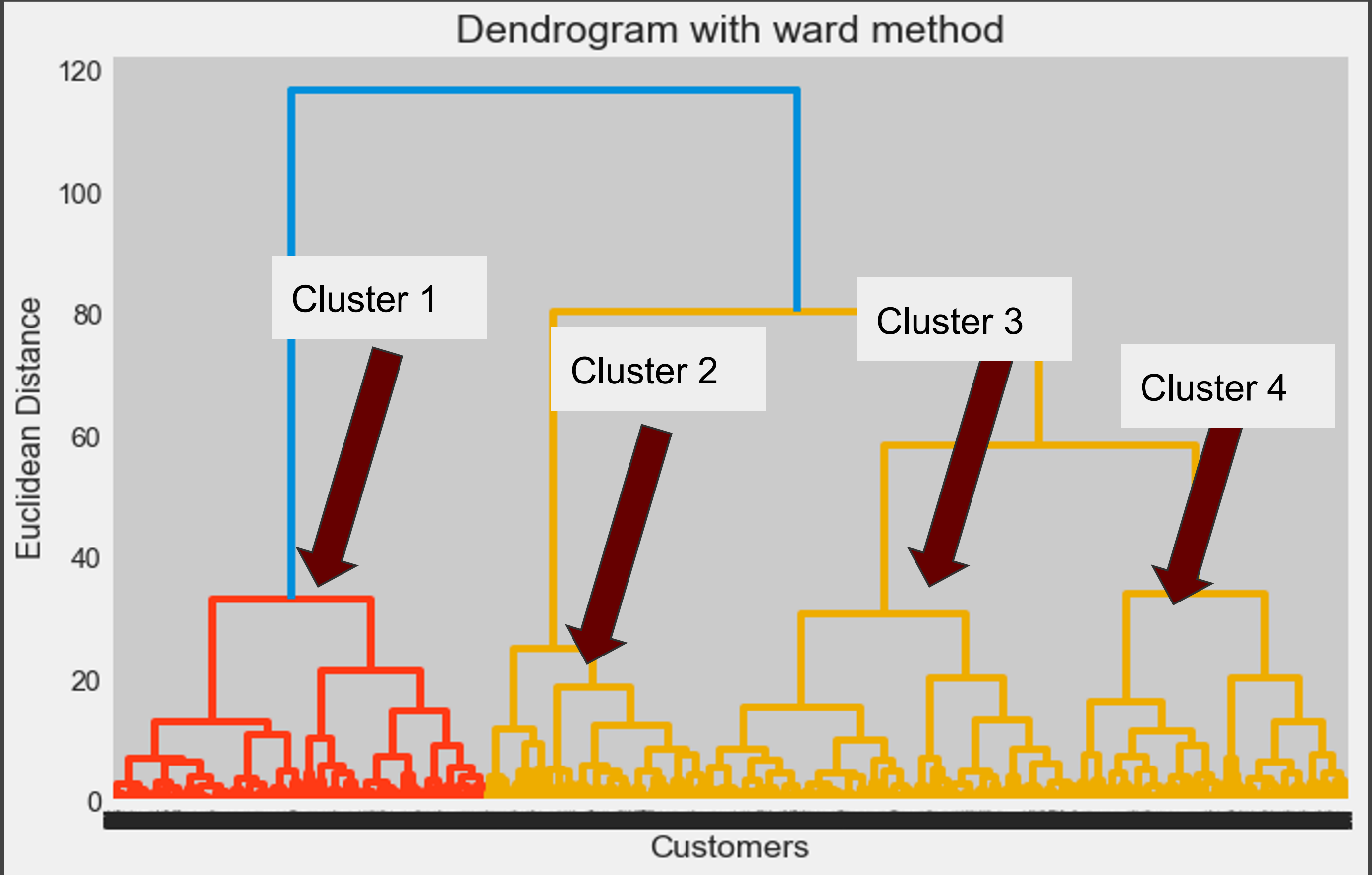
For Davies Bouldin we are looking for the lowest score before an increase, which is at 4 clusters on the graph.

How the data looks like after segmenting into 4 clusters using PCA



The clusters are not well separated, which explains the low Silhouette score

Using Hierarchical Clustering



Therefore,the data can be clustered in 4 groups but it can be clusters in more groups if the business requires it.

This is how the clusters look like

		Recency	Frequency	MonetaryValue		
Group	mean		mean		mean	count
0	1251	32		56	781	
1	1751	409		20	264	
2	1009	19		445	7290	
3	1720	216		112	1787	

Highest spending and very active customers

Second highest spending but not very active. These customers can become more active with some targeted marketing

Link to notebook