# Capstone Project 1 – Online Retail Store

# Table of Contents

# Problem Statement

An online retail store want analyze the customer purchasing behavior. Want to segment their customers on their purchasing behavior. Expecting some insights also about customer purchasing history.

# Project Objective

As we have to give insights for Customers purchasing history and segment them and have to analyze their purchasing pattern we will do these tasks for overall store as well as for specific country's  customers using strong evidence which comes from the data itself.

# Data Description

## Dataset Information:

### The online_retail.csv contains 387961 rows and 8 columns.

| Feature Name | Description |
|---|---|
| Invoice | Invoice number |
| StockCode | Product ID |
| Description | Product Description |
| Quantity | Quantity of the product |
| InvoiceDate | Date of the invoice |
| Price | Price of the product per unit |
| CustomerID | Customer ID |
| Country | Region of Purchase |

Some more information of the data.

For Given data there are 387961 records and 8 features and these 8 features are recorded for each transaction as follow:

1. Invoice: contains invoice number and it is unique for each new transaction. Note that there are some entries are start with ‘C’ which are cancelled transactions.

    a. Stock Code: contains unique identity for each product,

    b. Description contains details about the products and for sum transactions like sample, discount etc contains details about such transactions.

    c. Quantity shows number of single products for given transaction. E.g., there 2 number of parle biscuit and 3 number of chocolates

    d. Invoice date shows the date of transaction when it happens.

    e. Price contains price of product per unit.

    f. CustomerID : it is unique id for each customer.

    g. Country : it shows the country of the Customer.

# Data Preprocessing Steps and Inspiration

## The preprocessing of the data included the following steps:

.

1. Step 1: Loading the data
2. Step 2: checking shapes and data types of features
3. Step 3: Data cleaning, checking for duplicates and missing values.
4. Step 4: Performing EDA:

    a. Adding some columns for analysis
    b. Checking for outliers.
    c. Examine the plots of patterns of customers using created and added features.
    d. Checking the distribution of all the necessary columns.
    e. Creating some data frames to derive the insights from it.
    f. Analyzing for the country which has most number of customers.

5. Step 5: Creating Segmentations:
    RFM Analysis using following :
        a. Using K-Means clustering
        b. RMF Quartiles
        c. Segmentation plot of RMF grid
        d. Using Weighted RMF Score

6. Step 6: Analyzing the segmentation pattern and doing customers segmentation.

# Choosing the Algorithm for the Project

I Choose RMF analysis for this project

1) RFM Analysis:

RFM analysis is a marketing technique used to analyze customer behavior based on three key metrics: Recency, Frequency, and Monetary Value. It is commonly used in retail and e-commerce industries to segment customers and tailor marketing strategies accordingly.

**Description:**

- **Recency (R)**: This metric measures how recently a customer made a purchase. Customers who have made purchases more recently are typically considered more valuable, as they are more likely to make additional purchases in the future. Recency is often measured in terms of the number of days since the customer's last purchase.
- **Frequency (F)**: Frequency refers to how often a customer makes purchases. Customers who make frequent purchases are often more engaged and loyal to the brand. Frequency is typically measured as the total number of purchases made by the customer over a specific period, such as the past year.
- **Monetary Value (M)**: This metric represents the total monetary value of all purchases made by the customer. Customers who have spent more money are considered higher-value customers. Monetary value can be calculated as the sum of all purchase amounts made by the customer.

2) Algorithms for the RFM analysis :

- RFM segmentation involves clustering customers based on their Recency, Frequency, and Monetary Value scores. There are several techniques and algorithms that can be used for segmentation, including:
    1. **K-means Clustering**: K-means is a popular unsupervised clustering algorithm that partitions data into k clusters based on similarity. In RFM segmentation, you can use K-means to group customers into distinct segments based on their RFM scores. The algorithm aims to minimize the variance within clusters and maximize the variance between clusters.
    2. **Hierarchical Clustering**: Hierarchical clustering is another unsupervised clustering technique that organizes data into a hierarchical tree-like structure. In RFM segmentation,

hierarchical clustering can be used to create a dendrogram that visually represents the relationships between different customer segments. You can then cut the dendrogram at a certain level to identify distinct segments

3. **RFM Score-based Segmentation**: Instead of using clustering algorithms, you can define segmentation rules based on RFM scores. For example, you can create rules to classify customers as "High Value" if their RFM scores are above a certain threshold, "Medium Value" if their scores fall within a middle range, and "Low Value" if their scores are below a threshold.
4. **RFM Quartiles**: Another approach is to segment customers into quartiles based on their RFM scores. For each RFM metric (Recency, Frequency, Monetary Value), divide customers into four equal groups (quartiles) based on their scores. This approach categorizes customers into segments such as "Best Customers," "High Potential," "Low-Value," etc., based on their position within the quartiles.
5. **RFM Grid Analysis**: RFM grid analysis involves creating a two-dimensional grid where each axis represents one RFM metric (Recency, Frequency, Monetary Value). Customers are then categorized into segments based on their location within the grid. This approach provides a visual representation of customer segments and their RFM scores.
6. **RFM Weighted Score**: You can assign weights to each RFM metric based on their relative importance to your business objectives. For example, you may decide that Recency is more important than Frequency and Monetary Value. You can then calculate a weighted RFM score for each customer and use these scores to segment customers into different groups.

# Assumptions

The following assumptions were made in order to create the segmentation for RFM analysis for online retail store customers project.

1. **Independence of transactions:** RFM assumes that each transaction is independent of others. In other words, the behavior of a customer in one transaction does not influence their behavior in subsequent transactions. While this assumption simplifies the analysis, it may not always hold true, especially in cases where there are dependencies between transactions or purchases.
2. **Stationarity of behavior:** RFM assumes that customer behavior remains relatively stable over the analyzed period. It assumes that past behavior is indicative of future behavior. However, customer behavior may change over time due to various factors such as changes in preferences, life events, or external influences. Therefore, RFM analysis should be periodically reviewed and updated to reflect changes in customer behavior.
3. **Equal time intervals:** RFM typically divides the analysis period into equal time intervals (e.g., months or quarters). This assumption implies that each time period carries equal weight in the analysis. However, in practice, different time intervals may have varying significance. For example, recent transactions may be more indicative of current preferences than older transactions.
4. **Homogeneity within segments:** RFM segmentation assumes that customers within the same segment exhibit similar behavior. While segments may share common RFM characteristics, there may still be variability within segments. It's essential to recognize this variability and refine segments based on additional criteria or segmentation techniques if needed
5. **Monetary value as a proxy for profitability:** RFM analysis often uses monetary value (i.e., total spending) as a proxy for customer profitability. While high monetary value customers are generally more profitable, there may be exceptions, such as customers who make frequent but low-value purchases or customers who generate high revenue but also incur high servicing costs. Therefore, RFM-based segmentation should be supplemented with profitability analysis to ensure accurate customer prioritization.
6. **Data quality and completeness:** RFM analysis relies on accurate and complete transactional data, including recency, frequency, and monetary value metrics. Inaccurate or incomplete data can lead to biased segmentation results and erroneous insights. Therefore, it's crucial to validate and clean the data before conducting RFM analysis.

# Model Evaluation and Technique

The following techniques and steps were involved in the evaluation of the model

1. Load necessary Libraries
2. Load the dataset
3. Perform Exploratory Data Analysis (EDA) on the dataset
    a. Find the shape or size of the data
    b. Check for invalid and null entries
    c. Explore data description
    d. Dropping all the unnecessary data which is note use full for our analysis.
    e. Bar and Line plots to understand the behavior of customers over the 2 years.
    f. Checking the behavior of customers country wise.
4. Creating K-means and other RFM analysis segmentation methods to group similar type of customers.
5. Understanding the segmentations. And providing insights

Some of them are:

Preparation of data for RMF analysis:

some of values in invoice data are start with "c" these are cancelled transactions.

```
cancelled=data[data['InvoiceNo'].astype(str).str.contains('C')]
cancelled
```

[9] ✓ 0.5s

| | InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country |
|---|---|---|---|---|---|---|---|---|
| 141 | C536379 | D | Discount | -1 | 12/1/2010 9:41 | 27.50 | 14527.0 | United Kingdom |
| 154 | C536383 | 35004C | SET OF 3 COLOURED FLYING DUCKS | -1 | 12/1/2010 9:49 | 4.65 | 15311.0 | United Kingdom |
| 235 | C536391 | 22556 | PLASTERS IN TIN CIRCUS PARADE | -12 | 12/1/2010 10:24 | 1.65 | 17548.0 | United Kingdom |
| 236 | C536391 | 21984 | PACK OF 12 PINK PAISLEY TISSUES | -24 | 12/1/2010 10:24 | 0.29 | 17548.0 | United Kingdom |
| 237 | C536391 | 21983 | PACK OF 12 BLUE PAISLEY TISSUES | -24 | 12/1/2010 10:24 | 0.29 | 17548.0 | United Kingdom |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 540449 | C581490 | 23144 | ZINC T-LIGHT HOLDER STARS SMALL | -11 | 12/9/2011 9:57 | 0.83 | 14397.0 | United Kingdom |
| 541541 | C581499 | M | Manual | -1 | 12/9/2011 10:28 | 224.69 | 15498.0 | United Kingdom |
| 541715 | C581568 | 21258 | VICTORIAN SEWING BOX LARGE | -5 | 12/9/2011 11:57 | 10.95 | 15311.0 | United Kingdom |
| 541716 | C581569 | 84978 | HANGING HEART JAR T-LIGHT HOLDER | -1 | 12/9/2011 11:58 | 1.25 | 17315.0 | United Kingdom |
| 541717 | C581569 | 20979 | 36 PENCILS TUBE RED RETROSPOT | -5 | 12/9/2011 11:58 | 1.25 | 17315.0 | United Kingdom |

9288 rows × 8 columns

As you can see there are 9288 records are from cancelled transactions. Also you can see some stockcodes are containing alphabets.Moreover, Quantity columns are showing negative values. and Descriptin column has in appropriate format. we need to clean all this thing first

Above we are dropping cancelled transactions from the dataset.

```
data[data['Quantity']<0]
```

[10] ✓ 0.0s

| | InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country |
|---|---|---|---|---|---|---|---|---|
| 141 | C536379 | D | Discount | -1 | 12/1/2010 9:41 | 27.50 | 14527.0 | United Kingdom |
| 154 | C536383 | 35004C | SET OF 3 COLOURED FLYING DUCKS | -1 | 12/1/2010 9:49 | 4.65 | 15311.0 | United Kingdom |
| 235 | C536391 | 22556 | PLASTERS IN TIN CIRCUS PARADE | -12 | 12/1/2010 10:24 | 1.65 | 17548.0 | United Kingdom |
| 236 | C536391 | 21984 | PACK OF 12 PINK PAISLEY TISSUES | -24 | 12/1/2010 10:24 | 0.29 | 17548.0 | United Kingdom |
| 237 | C536391 | 21983 | PACK OF 12 BLUE PAISLEY TISSUES | -24 | 12/1/2010 10:24 | 0.29 | 17548.0 | United Kingdom |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 540449 | C581490 | 23144 | ZINC T-LIGHT HOLDER STARS SMALL | -11 | 12/9/2011 9:57 | 0.83 | 14397.0 | United Kingdom |
| 541541 | C581499 | M | Manual | -1 | 12/9/2011 10:28 | 224.69 | 15498.0 | United Kingdom |
| 541715 | C581568 | 21258 | VICTORIAN SEWING BOX LARGE | -5 | 12/9/2011 11:57 | 10.95 | 15311.0 | United Kingdom |
| 541716 | C581569 | 84978 | HANGING HEART JAR T-LIGHT HOLDER | -1 | 12/9/2011 11:58 | 1.25 | 17315.0 | United Kingdom |
| 541717 | C581569 | 20979 | 36 PENCILS TUBE RED RETROSPOT | -5 | 12/9/2011 11:58 | 1.25 | 17315.0 | United Kingdom |

10624 rows × 8 columns

There are 10624 records with negative quantities. reason could be return of the products,some discounts etc. lets try to check for unit price also.

```
data[data['UnitPrice']<0]
```

[11] ✓ 0.0s

| | InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country |
|---|---|---|---|---|---|---|---|---|
| 299983 | A563186 | B | Adjust bad debt | 1 | 8/12/2011 14:51 | -11062.06 | NaN | United Kingdom |
| 299984 | A563187 | B | Adjust bad debt | 1 | 8/12/2011 14:52 | -11062.06 | NaN | United Kingdom |

```
data=data[data['UnitPrice']>=0]
```

[12] ✓ 0.1s

So, there negative unit price. but by seeing in description it is clear that it is for utilisation. so we can drop them.

```
#checking for in canlled transactions any record is there with the positive quantity so we can drop all cancelled transactions
cancelled[cancelled['Quantity']>0]
```

[13] ✓ 0.0s

| InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country |
|---|---|---|---|---|---|---|---|

No record is there, that shows that all recordsare negative quantity records in cencelled data.so we can ndrop all these records from the main data

```
data=data[data['Quantity']>0]
```

[14] ✓ 0.1s

```
data.shape
```

[15] ✓ 0.0s

```
(531283, 8)
```

There are negative values in Quantity and in unit price. May they are showing for return, sample, or for self-use products also to adjust the bad-debt so we are dropping this all records from the data.

```
#cheking for any records having zero quantity and zero unit price because it means nothing to us.
data[(data['Quantity']==0) | data['UnitPrice']==0]
```

✓ 0.0s

| | InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country |
|---|---|---|---|---|---|---|---|---|
| 622 | 536414 | 22139 | NaN | 56 | 12/1/2010 11:52 | 0.0 | NaN | United Kingdom |
| 1970 | 536545 | 21134 | NaN | 1 | 12/1/2010 14:32 | 0.0 | NaN | United Kingdom |
| 1971 | 536546 | 22145 | NaN | 1 | 12/1/2010 14:33 | 0.0 | NaN | United Kingdom |
| 1972 | 536547 | 37509 | NaN | 1 | 12/1/2010 14:33 | 0.0 | NaN | United Kingdom |
| 1987 | 536549 | 85226A | NaN | 1 | 12/1/2010 14:34 | 0.0 | NaN | United Kingdom |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 535334 | 581211 | 22142 | check | 14 | 12/7/2011 18:36 | 0.0 | NaN | United Kingdom |
| 536981 | 581234 | 72817 | NaN | 27 | 12/8/2011 10:33 | 0.0 | NaN | United Kingdom |
| 538504 | 581406 | 46000M | POLYESTER FILLER PAD 45x45cm | 240 | 12/8/2011 13:58 | 0.0 | NaN | United Kingdom |
| 538505 | 581406 | 46000S | POLYESTER FILLER PAD 40x40cm | 300 | 12/8/2011 13:58 | 0.0 | NaN | United Kingdom |
| 538554 | 581408 | 85175 | NaN | 20 | 12/8/2011 14:06 | 0.0 | NaN | United Kingdom |

1179 rows × 8 columns

We can see that there are 1179 records with 0-unit price also some of them having null value in description and customer. also, we will drop all these null customer ids.
why?
because, without customer identification, it's challenging to analyse and segment customers effectively based on their purchasing behaviour or other attributes. The dataset used for segmentation analysis is complete and consistent. Complete data enables more robust and reliable segmentation models, leading to more actionable insights for marketing strategies, product recommendations, and customer engagement initiatives.

```
data['Description'] = data['Description'].str.lower()  # Convert to lowercase
data['Description'] = data['Description'].str.replace('[^\w\s]', '')  # Remove punctuation
print('No of Unique descriptions are',data['Description'].nunique())
```

[24] ✓ 0.5s

```
No of Unique descriptions are 3865
```

```
# Function to remove non-numeric characters from a string and convert it to int
def clean_and_convert_to_int(value):
    # Remove non-numeric characters (keep only digits)
    cleaned_value = ''.join(filter(str.isdigit, value))
    # Convert to integer
    return int(cleaned_value) if cleaned_value else None  # Convert to None if cleaned value is empty

# Apply the function to 'InvoiceNo' and 'StockCode' columns
data['StockCode'] = data['StockCode'].apply(clean_and_convert_to_int)

# Print the first few rows to verify the changes
data.head()
```

Correcting the format for Stock code and Description.

```
data['Month'] = data['InvoiceDate'].dt.month
data['Year'] = data['InvoiceDate'].dt.year
data['Month/year']=pd.to_datetime(data[['Year','Month']].assign(day=1))
data['Week_day']=data['InvoiceDate'].dt.day_name()
data['Quarter']=data['InvoiceDate'].dt.quarter
data['Hour']=data['InvoiceDate'].dt.hour
```

[33] ✓ 0.2s Python

```
data['Total_Amount']=data['Quantity']*data['UnitPrice']
```

[34] ✓ 0.0s Python

```
data
```

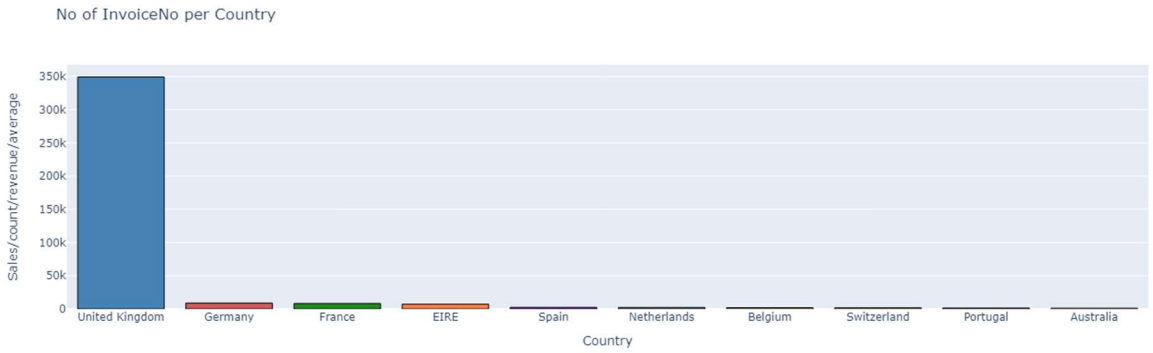
[35] ✓ 0.0s Python

| | InvoiceNo | StockCode | Description | Quantity | InvoiceDate | UnitPrice | CustomerID | Country | Month | Year | Month/year | Week_day | Quarter | Hour | Total_Amount |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 536365 | 85123.0 | white hanging heart tlight holder | 6 | 2010-12-01 08:26:00 | 2.55 | 17850.0 | United Kingdom | 12 | 2010 | 2010-12-01 | Wednesday | 4 | 8 | 15.30 |
| 1 | 536365 | 71053.0 | white metal lantern | 6 | 2010-12-01 08:26:00 | 3.39 | 17850.0 | United Kingdom | 12 | 2010 | 2010-12-01 | Wednesday | 4 | 8 | 20.34 |
| 2 | 536365 | 84406.0 | cream cupid hearts coat hanger | 8 | 2010-12-01 08:26:00 | 2.75 | 17850.0 | United Kingdom | 12 | 2010 | 2010-12-01 | Wednesday | 4 | 8 | 22.00 |
| 3 | 536365 | 84029.0 | knitted union flag hot water bottle | 6 | 2010-12-01 08:26:00 | 3.39 | 17850.0 | United Kingdom | 12 | 2010 | 2010-12-01 | Wednesday | 4 | 8 | 20.34 |
| 4 | 536365 | 84029.0 | red woolly hottie white heart | 6 | 2010-12-01 08:26:00 | 3.39 | 17850.0 | United Kingdom | 12 | 2010 | 2010-12-01 | Wednesday | 4 | 8 | 20.34 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 392653 | 581587 | 22613.0 | pack of 20 spaceboy napkins | 12 | 2011-12-09 12:50:00 | 0.85 | 12680.0 | France | 12 | 2011 | 2011-12-01 | Friday | 4 | 12 | 10.20 |
| 392654 | 581587 | 22899.0 | childrens apron dolly girl | 6 | 2011-12-09 12:50:00 | 2.10 | 12680.0 | France | 12 | 2011 | 2011-12-01 | Friday | 4 | 12 | 12.60 |

Spaces: 4 Cell 113 of 121

Search ENG IN 07:03 17-04-2024

Adding some columns like year, month, day, weekday, hour, quarter to analyse the behaviour of customer.




Most number of customers for this online retail store are from the United Kingdom. Hence, to create use full segmentation we do RFM analysis only for the UK based customers.

## Evolution of the models:

RFM Analysis :

- The first step to create a data frame using

  present date for out RFM analysis metrics.

  (recency, frequency, monetary).
- Using different methods to segment the customers:
  1) K-Means clustering:
     - Choosing best K value using elbow method and silhouette_score.
  2) RFM Quantile
  3) Plot RFM Grid
  4) weighted RFM score

- Understanding of the segment

Let’s see this evolution

```
#creating a data frame and sotring columns for our metrices
rfm = data_uk.groupby('CustomerID').agg({'InvoiceDate': lambda x: (presence - x.max()).days, 'InvoiceNo': lambda x: len(x), 'Total_Amount': lambda x: x.sum()})
rfm['InvoiceDate'] = rfm['InvoiceDate'].astype(int)
rfm.rename(columns={'InvoiceDate': 'recency',
                    'InvoiceNo': 'frequency',
                    'Total_Amount': 'monetary_value'}, inplace=True) #changing column names
```

[67] ✓ 1.6s

```
rfm.head()
```

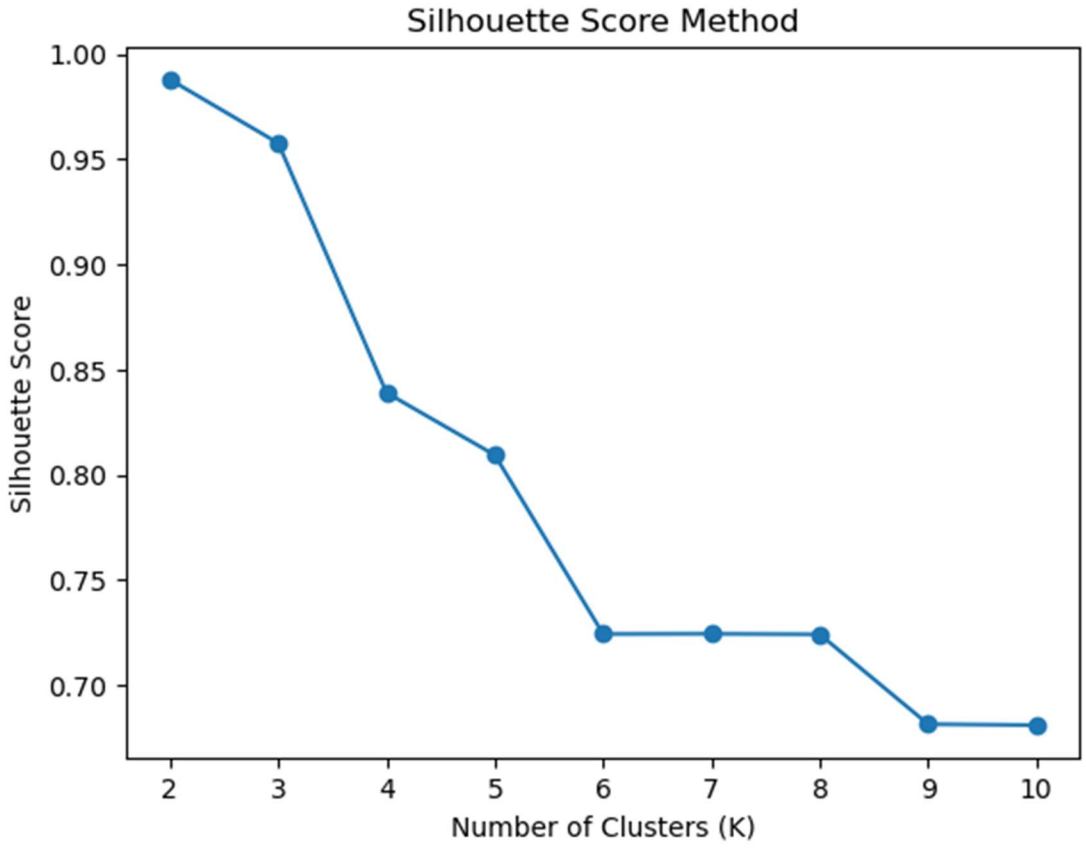
[68] ✓ 0.0s

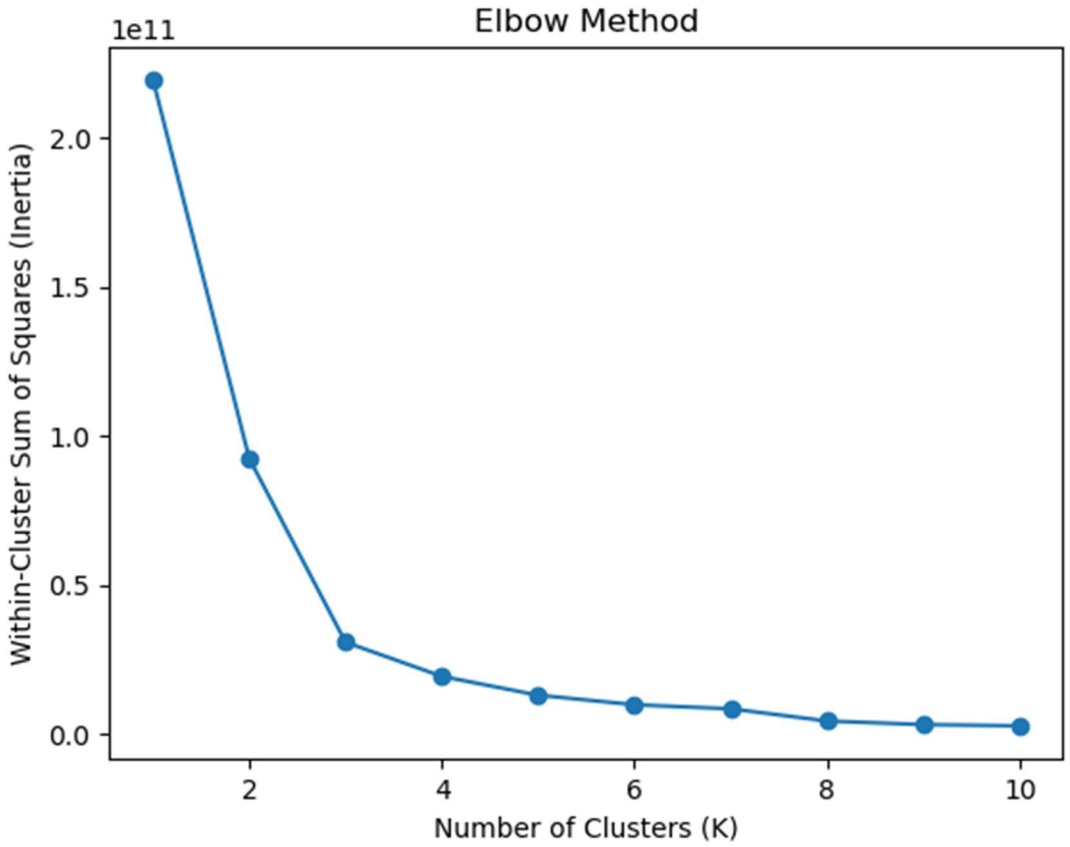
| CustomerID | recency | frequency | monetary_value |
|---|---|---|---|
| 12346.0 | 327 | 1 | 77183.60 |
| 12747.0 | 4 | 103 | 4196.01 |
| 12748.0 | 2 | 4412 | 33053.19 |
| 12749.0 | 5 | 199 | 4090.88 |
| 12820.0 | 5 | 59 | 942.34 |

After the first step.

## Step 2 : Applying methods for segmentation.

Firstly, K-Means clustering






From the both observation we can observe the best fit value of k is 3.As in elbow method at after k=3 we can see sudden drop. And in silhouette score for k=3 there is sudden drop.
Hence it is clear that there are 3 segments of customers.

Now we will apply rest three methods for segmentation.And after that our dataframe looks like below :

78] ✓ 0.0s

| CustomerID | recency | frequency | monetary_value | clusters | RecencyQuartile | FrequencyQuartile | MonetaryValueQuartile | RFMSegment | WeightedRFMScore |
|---|---|---|---|---|---|---|---|---|---|
| 12346.0 | 327 | 1 | 77183.60 | 2 | 3 | 0 | 3 | 303 | 15600.520 |
| 12747.0 | 4 | 103 | 4196.01 | 0 | 0 | 3 | 3 | 033 | 872.102 |
| 12748.0 | 2 | 4412 | 33053.19 | 2 | 0 | 3 | 3 | 033 | 7935.238 |
| 12749.0 | 5 | 199 | 4090.88 | 0 | 0 | 3 | 3 | 033 | 880.376 |
| 12820.0 | 5 | 59 | 942.34 | 0 | 0 | 2 | 2 | 022 | 208.668 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 18280.0 | 279 | 10 | 180.60 | 0 | 3 | 0 | 0 | 300 | 178.620 |
| 18281.0 | 182 | 7 | 80.82 | 0 | 3 | 0 | 0 | 300 | 109.264 |
| 18282.0 | 9 | 12 | 178.05 | 0 | 0 | 0 | 0 | 000 | 43.710 |
| 18283.0 | 5 | 721 | 2045.53 | 0 | 0 | 3 | 3 | 033 | 627.906 |
| 18287.0 | 44 | 70 | 1837.28 | 0 | 1 | 2 | 3 | 123 | 410.456 |

3920 rows × 9 columns

```
rfm.RFMSegment.unique()
```

79] ✓ 0.0s

```
array(['303', '033', '022', '300', '222', '203', '211', '011', '113',
       '111', '311', '233', '132', '333', '212', '121', '122', '220',
       '301', '223', '232', '322', '201', '200', '133', '321', '032',
       '101', '002', '110', '023', '302', '123', '013', '010', '210',
       '131', '100', '323', '001', '310', '000', '313', '031', '312',
       '012', '213', '102', '202', '320', '221', '021', '112', '120',
       '020', '332', '103', '231', '331', '030', '003'], dtype=object)
```

**The evaluation report suggests the following:**

See, from the K-Means clusters we got 3 segmentations but can't getting clear idea about the customer behavior properly. Similarly
By grid and weights.
But from the RFM quartile we can clearly do our desire analysis.

### 1.Recency and Recency quartile:

Observe that if the recency of customer is more than recency quartile is 3. Means that customer not purchased anything in recent time. E.g. customer id 12346 purchased 327 days ago so it has 3 recency quartile.

### 2. Frequency and Frequency Quartile:

In the case of frequency, the higher frequency high frequency quartile. If customer is more frequent it has 3 frequency quartiles.

### 3. Monetary and Monetary Quartile:

Similar to Frequency. If customer spends more it has 3 monetary quartile.

Hence, we can see the RFM pattern ‘303’,’001’,’300’,’033’… from this we can understand customers behavior.

‘033’ having this type of behavior such customers are champion customers to the retail store.

‘030’ are the most loyal ones.

‘003’ are the most money spenders.

‘010’,’013’,’001’,’002’ are the new customers.

‘111’,’121’,’211’,’210’,’101’,’200’ type of customers are from UK are on the verge of sleeping out.

‘300’,’100’ type of customers already out.

Champion Customers from UK are 407 and 10.38%
Loyal Customers from UK are 969 and 24.72%
Money Spender Customers from UK are 980 and 25.0%
New customers from UK are 73 and 12.22%
Customers from UK who are on the verge of sleep out 552 and 14.08%
Lost Customers from UK are 456 and 11.63%

# Inferences from the Project

From the EDA we derive some things which will important for customer behavior.

To do some analysis we created some columns which is already mention earlier. And created some data frames and plot them into the bar plot and line plot.

Now we will create some data frames to do more analysis.

```
count_per_hour=data.groupby('Hour')['InvoiceNo'].count().reset_index()
count_per_week_day=data.groupby('Week_day')['InvoiceNo'].count().reindex(['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Sunday']).reset_index()
count_per_month=data.groupby('Month')['InvoiceNo'].count().reset_index()
count_per_quarter=data.groupby('Quarter')['InvoiceNo'].count().reset_index()
count_per_year=data.groupby('Year')['InvoiceNo'].count().reset_index()
monthly_sales=data.groupby('Month')['Total_Amount'].sum().reset_index()
monthly_avg_sales=data.groupby('Month')['Total_Amount'].mean().reset_index()
Quarterly_revenue=data.groupby('Quarter')['Total_Amount'].sum().reset_index()
Querterly_avg_revenue=data.groupby('Quarter')['Total_Amount'].mean().reset_index()
Yearly_revenue=data.groupby('Year')['Total_Amount'].sum().reset_index()
Yearly_avg_revenue=data.groupby('Year')['Total_Amount'].mean().reset_index()
```

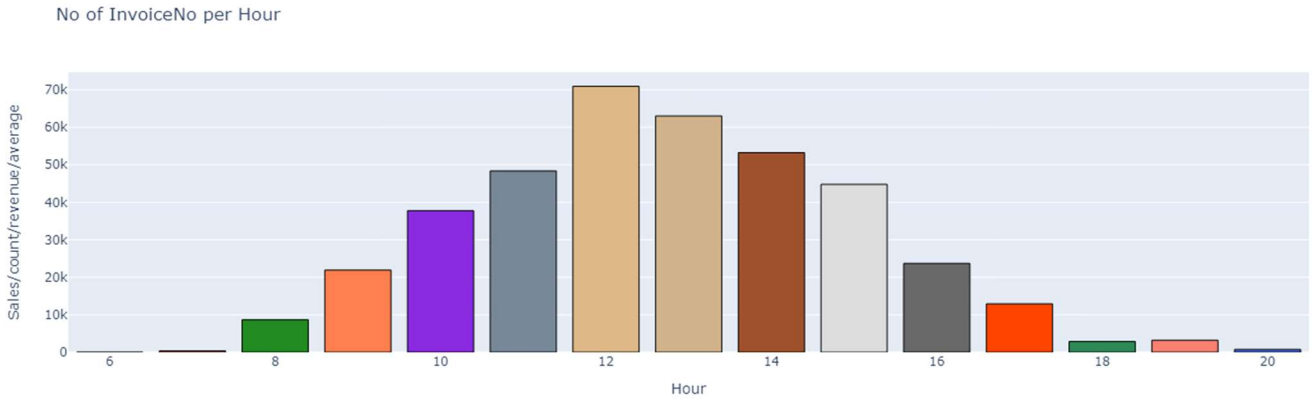
[36] ✓ 0.1s Python

```
data['Date']=data['InvoiceDate'].dt.date
Daily_sales=data.groupby('Date')['Total_Amount'].sum().reset_index()
```

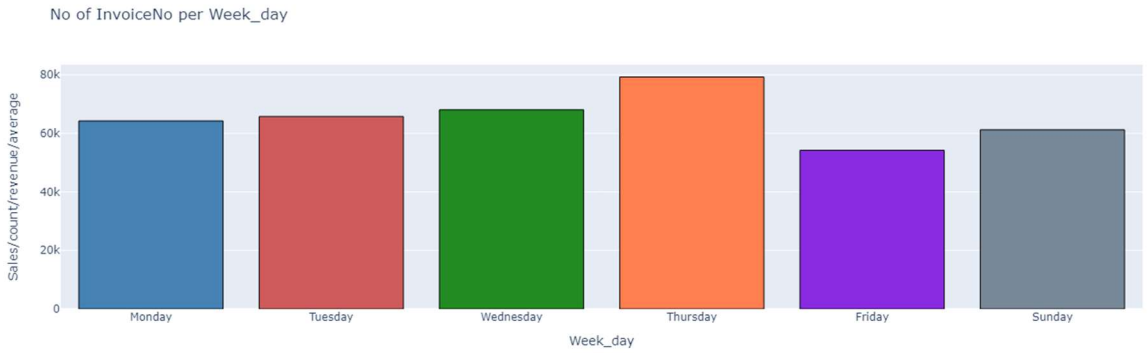
[37] ✓ 0.1s Python

```
#plot of no of orders are placing hourly
def plot_line_graphs(dataframe,xax,yax) :
    trace=go.Line(x=dataframe[xax],y=dataframe[yax],mode='lines',line=dict(color='blue', width=2), marker=dict(color='red', size=10))
    layout=go.Layout(title='No of {} per {}'.format(yax,xax),xaxis_title=xax,yaxis_title='Sales/count/revenue/average')
    fig=go.Figure([trace],layout=layout)
    return fig.show()
def plot_Bar_graphs(dataframe,xax,yax) :
    plt.figure(figsize=(15,10))
    color=['steelblue','indianred','forestgreen','coral','blueviolet','lightslategray','burlywood','tan','sienna','gainsboro','dimgray','orangered','seagreen','salmon','royalblu
    trace=go.Bar(x=dataframe[xax],y=dataframe[yax],marker=dict(color=color, line=dict(color='black', width=1)))
    layout=go.Layout(title='No of {} per {}'.format(yax,xax),xaxis_title=xax,yaxis_title='Sales/count/revenue/average')
    fig=go.Figure([trace],layout=layout)
    return fig.show()
```

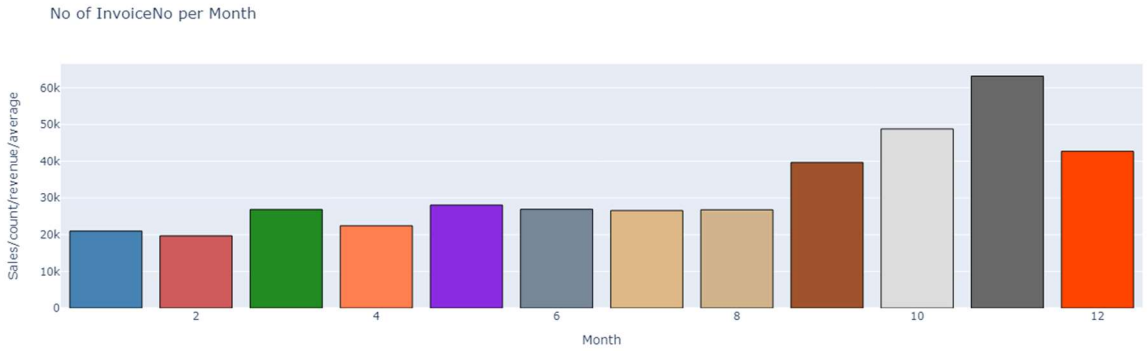
[38] ✓ 0.0s Python



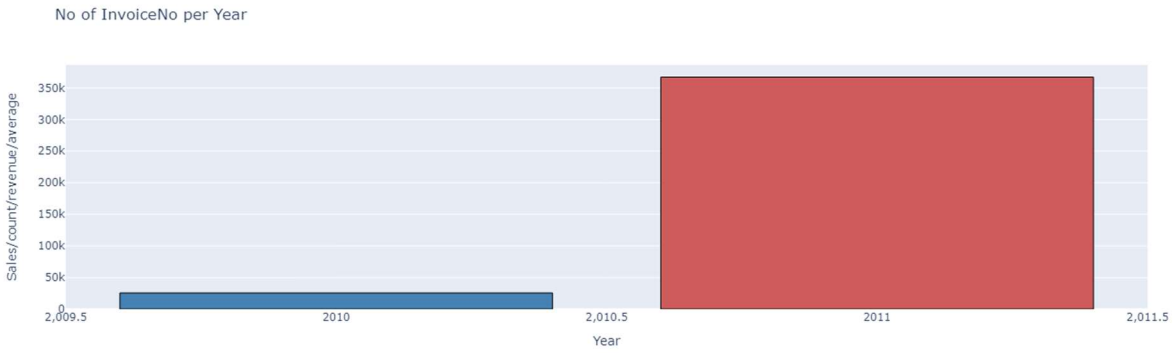

From the above graph we can see that most of customers are parchasing between 9 a.m to 4 p.m.




Major orders are placed in the mid week days.




Most number of orders are placed in the year end.




Majority of orders are placed in the year of 2011.


As we can see customers spending more in the end of the year.obviously major orders are also placed in the end of the year.

There are 4346 unique customers different cities.
In which UK hase 3920 customers. Second is Germany with the 94 customers. So obviously UK has a more orders and company is earning the most from it.
Top 10 customers who spends more are :

| CustomerID | Total_Amount |
|---|---|
| 14646.0 | 280206.02 |
| 18102.0 | 259657.30 |
| 17450.0 | 194390.79 |
| 16446.0 | 168472.50 |
| 14911.0 | 143711.17 |
| 12415.0 | 124914.53 |
| 14156.0 | 117210.08 |
| 17511.0 | 91062.38 |
| 16029.0 | 80850.84 |
| 12346.0 | 77183.60 |

All top customers are from UK, Netherlands, EIRE, Australia.

# Conclusion

Some of the important findings from the report include the followings:

1. Most of customers are buying between 9 to 4. May be because me retail store is of office related stuff. Try to give some sale offer in the night time for 1 or 2 hours.it will help to increase more orders in the night time also.
2. There are 0 sales on the Saturday, Make Saturday as special day of the retail store and give some special offers on that day.
3. In the end of the year more orders and sales observed. Try to organize “New year sale” in the January and February month.
4. Majority of customers are from the UK. Hence give them chance to access some special products first. Also advertise your store more in other countries through Social media, TV etc.
5. Give your top 10 customers special deals and promote them into advertisement which will poke other customers to spend more on your store.
6. From the customer segmentation, following steps are recommended:
    a. Based on RFM analysis for segmentation we can see that there are large percentage of customers in UK are loyal and money spenders. So according to this there should be some special campaigns for those customers.
    b. There are some percentages of customers are on the verge of sleep or already lost them. So do some extra effort to keep them. through some campaigns or providing discounts or special offers.

7. Regarding data:
    a. Large number of customer ids are note present.
    b. Stock code are not in valid form contains different letters/alphabets
    c. In Description also missing values are there and not in proper form.
    d. Each stock code should be assigned to only one product. This rule not kept in this data set

# References

For RFM analysis :
https://www.analyticsvidhya.com/blog/2021/07/customer-segmentation-using-rfm-analysis/

For K-Means Clusters: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.