Cairo University

Faculty of Computers and Artificial Intelligence

Operations Research and Decision Support Dept

جامعة القاهرة

كلية الحاسبات والذكاء الاصطناعي

قسم بحوث العمليات ودعم اتخاذ القرار

PROJECT PROPOSAL

CUSTOMER SEGMENTATION USING MACHINE LEARNING

“Know your customers”

Name	ID
Asmaa Adel Omar	20180460
Mohamed Essam Galal	20180231
Mohamed Hany khariy	20180242
Rech Raymondo Malek	20180402

Supervised by:

Dr. Doaa Saleh

Table of Contents

I. PROJECT OVERVIEW

Customer Segmentation is the process of division of customer base into several groups of individuals that share a similarity in different ways that are relevant to marketing such as gender, age, interests, and miscellaneous spending habits.

Companies that deploy customer segmentation are under the notion that every customer has different requirements and require a specific marketing effort to address them appropriately. Companies aim to gain a deeper approach of the customer they are targeting. Therefore, their aim has to be specific and should be tailored to address the requirements of each and every individual customer. Furthermore, through the data collected, companies can gain a deeper understanding of customer preferences as well as the requirements for discovering valuable segments that would reap them maximum profit. This way, they can strategize their marketing techniques more efficiently and minimize the possibility of risk to their investment.

The technique of customer segmentation is dependent on several key differentiators that divide customers into groups to be targeted. Data related to demographics, geography, economic status as well as behavioural patterns play a crucial role in determining the company direction towards addressing the various segments.

II. NEEDS

- We shall require skill sets in Machine Learning with Python programming.
- We shall require designing a business case.
- We need to collect and prepare the Data.
- Performing Segmentation using Machine Learning.
- Tuning the optimal hyperparameters for the model.
- Visualization of the Results using "Power Pi "and build a Dashboard.
- We shall require access to computers to develop this system.
- We shall require guidance and consultations with our project supervisor.

III. ISSUES

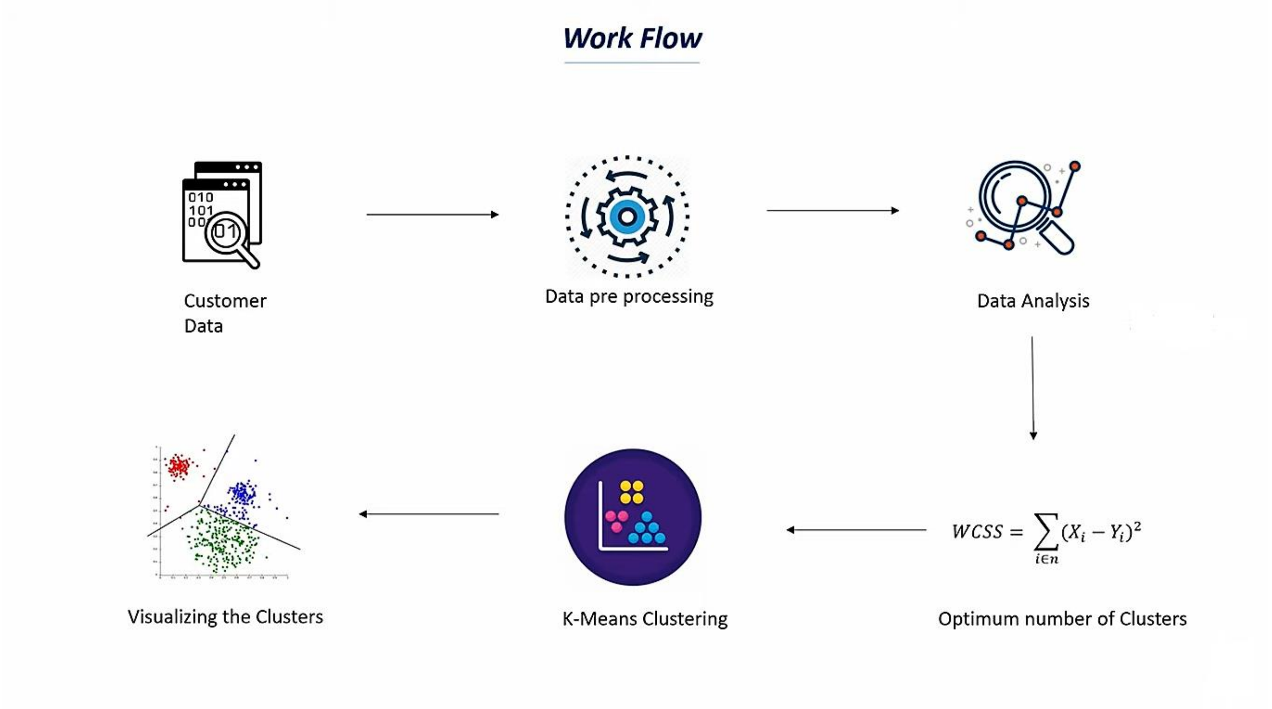
- Finding a dataset that's best suited for our business case maybe a little challenging.

IV. OBJECTIVES

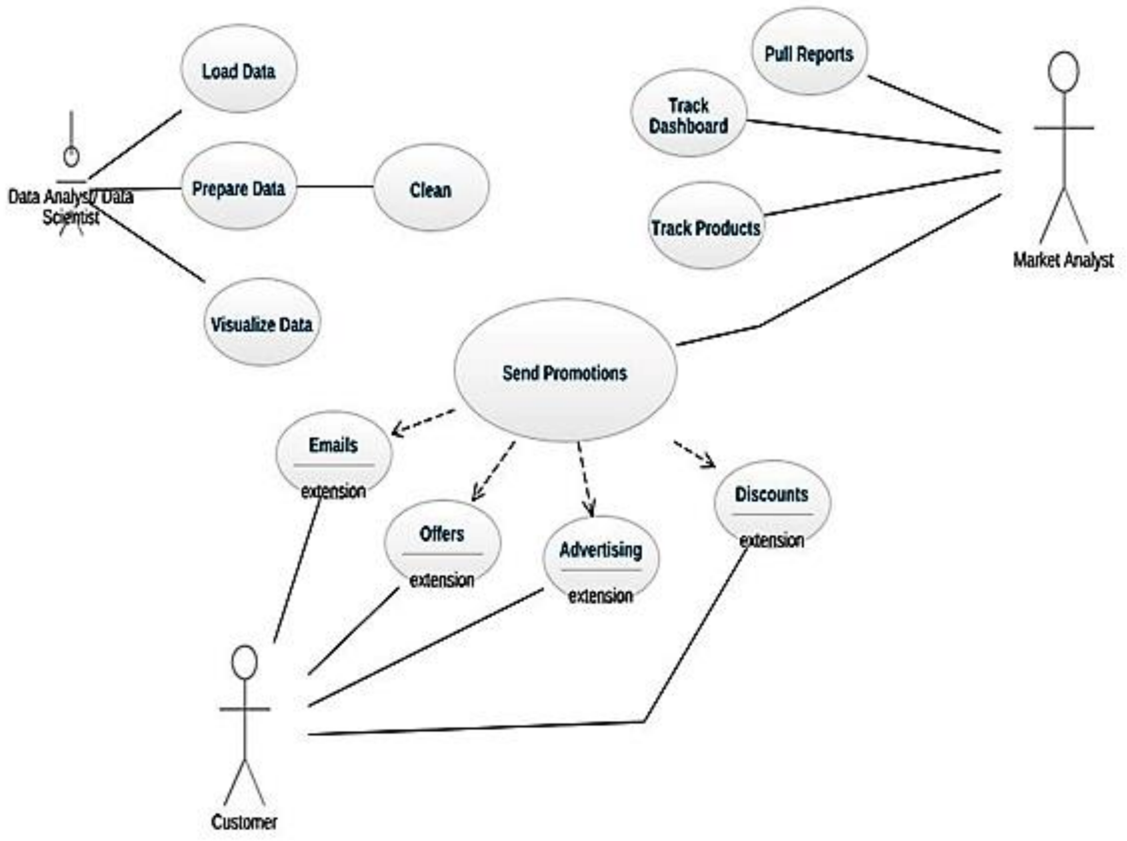
- To identify the shopping behaviours of customers in order to provide targeted advertisement during time periods and specific months of the year. To also identify the appropriate country locations to stock goods in warehouses (this is helpful to giant e-commerce companies like Amazon

V. *SCOPE OF WORK*

We are building a model to aid companies/organizations to understand their customers and identify their loyal ones using **RFM Analysis and K-means algorithm**; this process lets a company understand its customers and consequently decide on marketing, sales and financial decisions. We will start by applying RFM Analysis; the idea is to segment customers based on when their last purchases, how often they purchased in the past and how much they spent overall. These three questions simply explain "RFM which means Recency, Frequency and Monetary" After applying this analysis and calculating the RFM Score for each customer, we shall then apply unsupervised Machine Learning algorithm "K-Means" to group these customers into different clusters based on their RFM Score.



This Use-Case is describing the usage of the model we created, it is an important business intelligence tool that combines two roles: the first phase for the data analyst or the data scientist and the second phase for the marketing analyst.



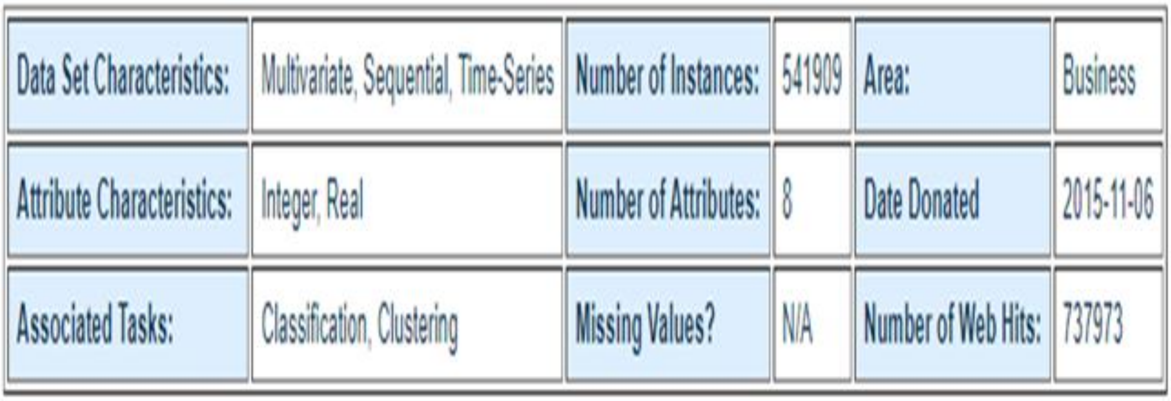
Steps we Performed:

1. **Description of Data**

Ecommerce dataset are hard to find among publicly available data; however, UCI Machine Learning Repository has made the dataset containing transactions from 1/12/2010 to 9/12/2011 available; it's for a UK based online Retail store.

Dataset Information:

This a transactional dataset which contains all the transactions that happened between 1/12/2010 to 9/12/2011 "1 Year" for UK based and registered non-online retail; the company mainly sells unique all-occasion gifts; many customers of the organization are wholesalers.



Data Set Characteristics:	Multivariate, Sequential, Time-Series	Number of Instances:	541909	Area:	Business
Attribute Characteristics:	Integer, Real	Number of Attributes:	8	Date Donated	2015-11-06
Associated Tasks:	Classification, Clustering	Missing Values?	N/A	Number of Web Hits:	737973

Attribute Information:

Invoice No. column: It is the invoice number of the transaction, Nominal, consist of 6-digit integral number uniquely assigned to the transaction, if the code begins with "C", it's a cancelation.

Stock Code Column: It's the product code, Nominal, consist of 5-digit integral number uniquely assigned to each product.

Description Column: The product name, string.

Quantity Column: Number of quantities of each product per transaction, Numeric.

Invoice date Column: The invoice date and time, Numeric. The day and time where each transaction occurred.

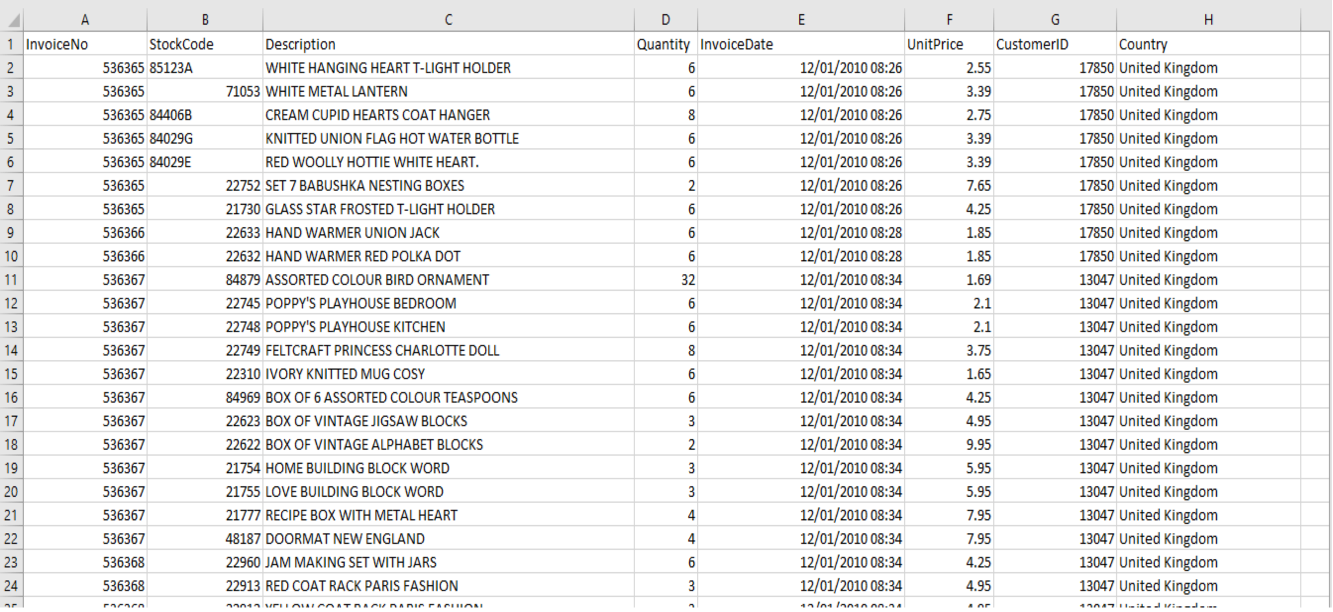
Unit Price Column: The price of product, Numeric, Product price unit per Pound sterling.

Customer ID Column: The customer ID, Nominal. A 5-digit integral uniquely assigned to each customer.

Country Column: Nominal, It's the name of the country where each customer resides.

2. Data Collection

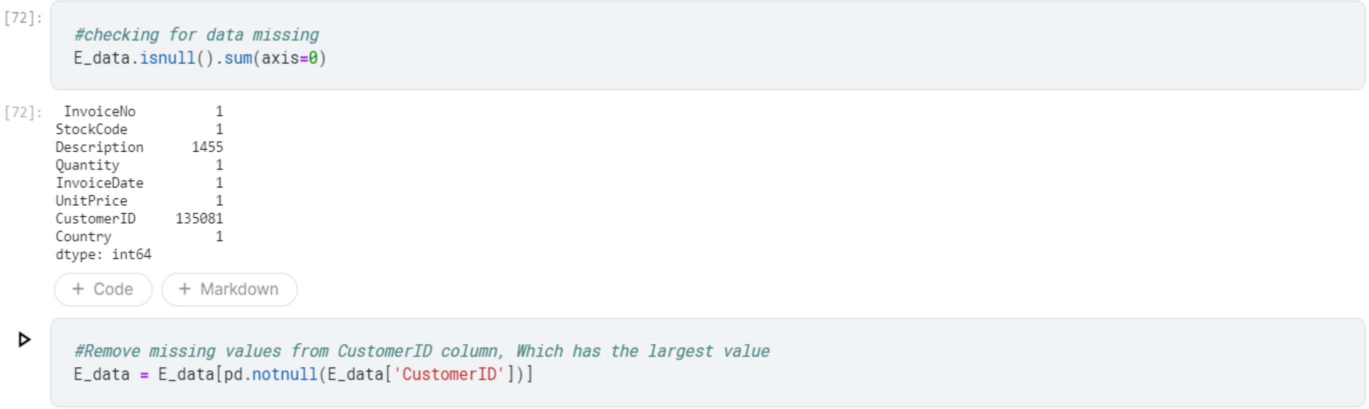
We intend to apply RFM Analysis and K-means Clustering to it.



	A	B	C	D	E	F	G	H
1	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
2	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/01/2010 08:26	2.55	17850	United Kingdom
3	536365	71053	WHITE METAL LANTERN	6	12/01/2010 08:26	3.39	17850	United Kingdom
4	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/01/2010 08:26	2.75	17850	United Kingdom
5	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/01/2010 08:26	3.39	17850	United Kingdom
6	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/01/2010 08:26	3.39	17850	United Kingdom
7	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/01/2010 08:26	7.65	17850	United Kingdom
8	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	12/01/2010 08:26	4.25	17850	United Kingdom
9	536366	22633	HAND WARMER UNION JACK	6	12/01/2010 08:28	1.85	17850	United Kingdom
10	536366	22632	HAND WARMER RED POLKA DOT	6	12/01/2010 08:28	1.85	17850	United Kingdom
11	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/01/2010 08:34	1.69	13047	United Kingdom
12	536367	22745	POPPY'S PLAYHOUSE BEDROOM	6	12/01/2010 08:34	2.1	13047	United Kingdom
13	536367	22748	POPPY'S PLAYHOUSE KITCHEN	6	12/01/2010 08:34	2.1	13047	United Kingdom
14	536367	22749	FELTCRAFT PRINCESS CHARLOTTE DOLL	8	12/01/2010 08:34	3.75	13047	United Kingdom
15	536367	22310	IVORY KNITTED MUG COSY	6	12/01/2010 08:34	1.65	13047	United Kingdom
16	536367	84969	BOX OF 6 ASSORTED COLOUR TEASPOONS	6	12/01/2010 08:34	4.25	13047	United Kingdom
17	536367	22623	BOX OF VINTAGE JIGSAW BLOCKS	3	12/01/2010 08:34	4.95	13047	United Kingdom
18	536367	22622	BOX OF VINTAGE ALPHABET BLOCKS	2	12/01/2010 08:34	9.95	13047	United Kingdom
19	536367	21754	HOME BUILDING BLOCK WORD	3	12/01/2010 08:34	5.95	13047	United Kingdom
20	536367	21755	LOVE BUILDING BLOCK WORD	3	12/01/2010 08:34	5.95	13047	United Kingdom
21	536367	21777	RECIPE BOX WITH METAL HEART	4	12/01/2010 08:34	7.95	13047	United Kingdom
22	536367	48187	DOORMAT NEW ENGLAND	4	12/01/2010 08:34	7.95	13047	United Kingdom
23	536368	22960	JAM MAKING SET WITH JARS	6	12/01/2010 08:34	4.25	13047	United Kingdom
24	536368	22913	RED COAT RACK PARIS FASHION	3	12/01/2010 08:34	4.95	13047	United Kingdom

3. Data Pre-processing using python

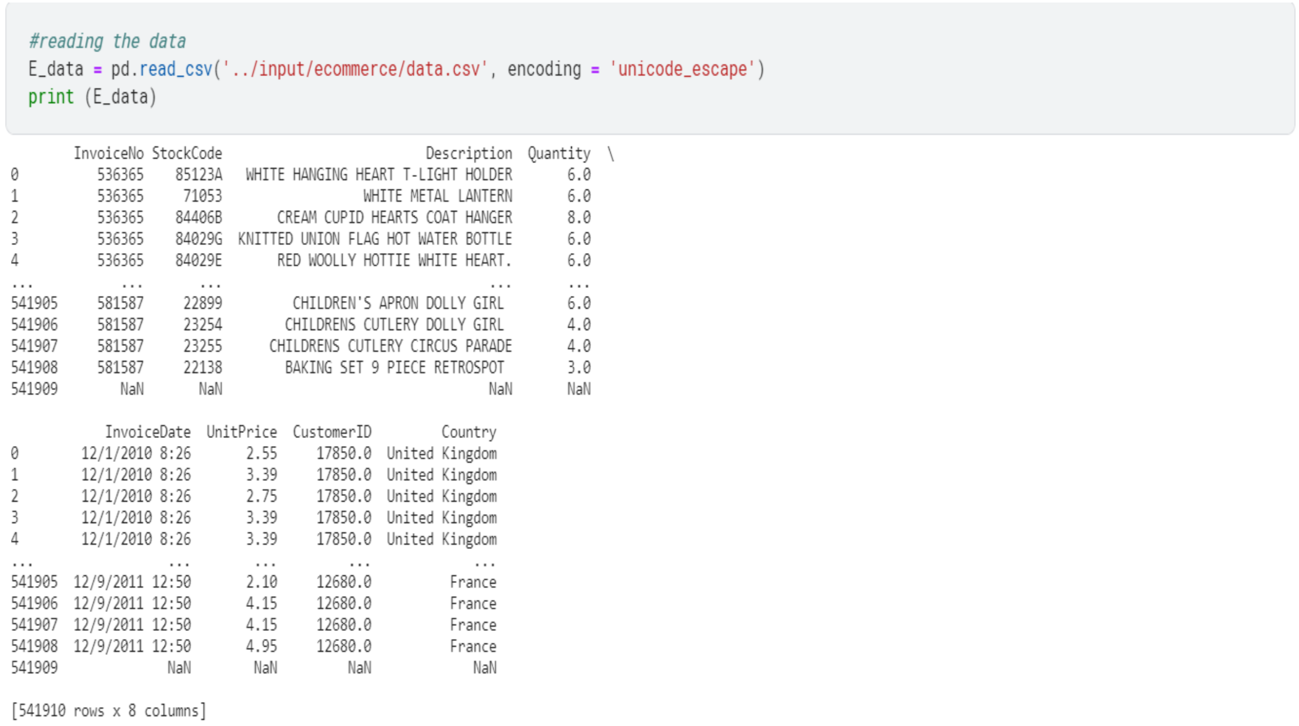
We shall read the data using pandas library; it contains [541910 rows x 8 columns] Then we complete the process by checking missing values, and negative values, duplicating ones.



```
[72]: #checking for data missing
      E_data.isnull().sum(axis=0)
```

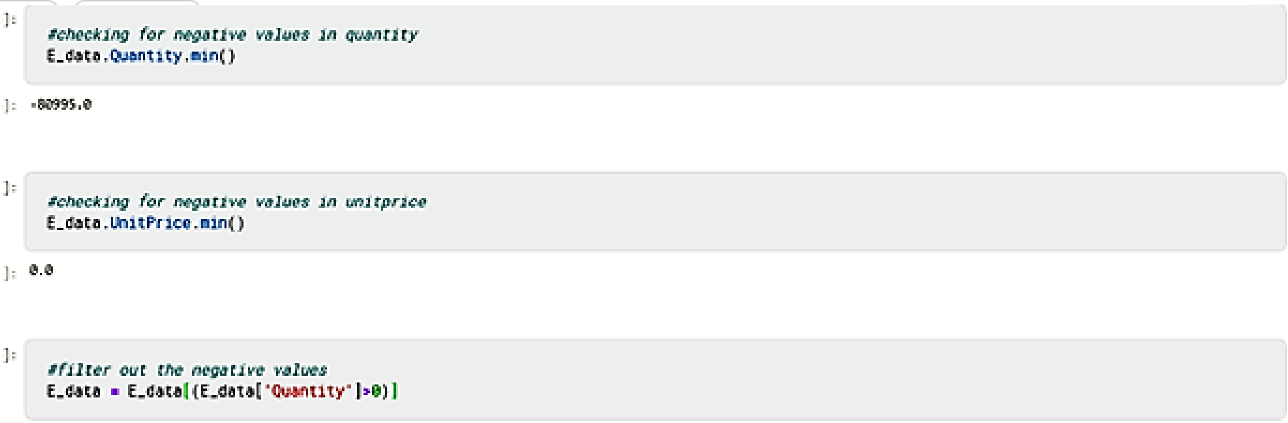
```
[72]:  InvoiceNo          1
StockCode          1
Description     1455
Quantity           1
InvoiceDate        1
UnitPrice          1
CustomerID    135081
Country            1
dtype: int64
```

```
#Remove missing values from CustomerID column, Which has the largest value
E_data = E_data[pd.notnull(E_data['CustomerID'])]
```



```
#reading the data
E_data = pd.read_csv('../input/ecommerce/data.csv', encoding = 'unicode_escape')
print (E_data)
```

```
       InvoiceNo StockCode                          Description  Quantity  \
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER       6.0
1         536365     71053                  WHITE METAL LANTERN       6.0
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER       8.0
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE       6.0
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.       6.0
...          ...       ...                                  ...       ...
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL        6.0
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL        4.0
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE       4.0
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT        3.0
541909       NaN       NaN                                  NaN       NaN

           InvoiceDate  UnitPrice  CustomerID         Country
0        12/1/2010 8:26       2.55     17850.0  United Kingdom
1        12/1/2010 8:26       3.39     17850.0  United Kingdom
2        12/1/2010 8:26       2.75     17850.0  United Kingdom
3        12/1/2010 8:26       3.39     17850.0  United Kingdom
4        12/1/2010 8:26       3.39     17850.0  United Kingdom
...                 ...        ...         ...             ...
541905  12/9/2011 12:50       2.10     12680.0          France
541906  12/9/2011 12:50       4.15     12680.0          France
541907  12/9/2011 12:50       4.15     12680.0          France
541908  12/9/2011 12:50       4.95     12680.0          France
541909              NaN        NaN         NaN             NaN

[541910 rows x 8 columns]
```



```
]: #checking for negative values in quantity
   E_data.Quantity.min()
```

```
]: -80995.0
```

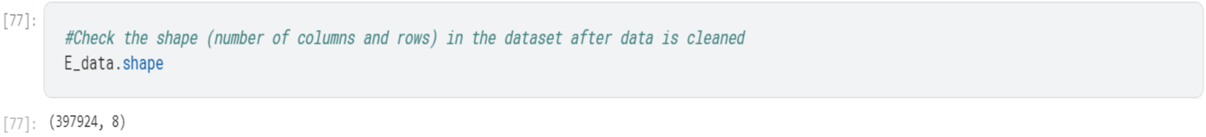
```
]: #checking for negative values in unitprice
   E_data.UnitPrice.min()
```

```
]: 0.0
```

```
]: #filter out the negative values
   E_data = E_data[(E_data['Quantity']>0)]
```

4. Data shape after Pre-Processing

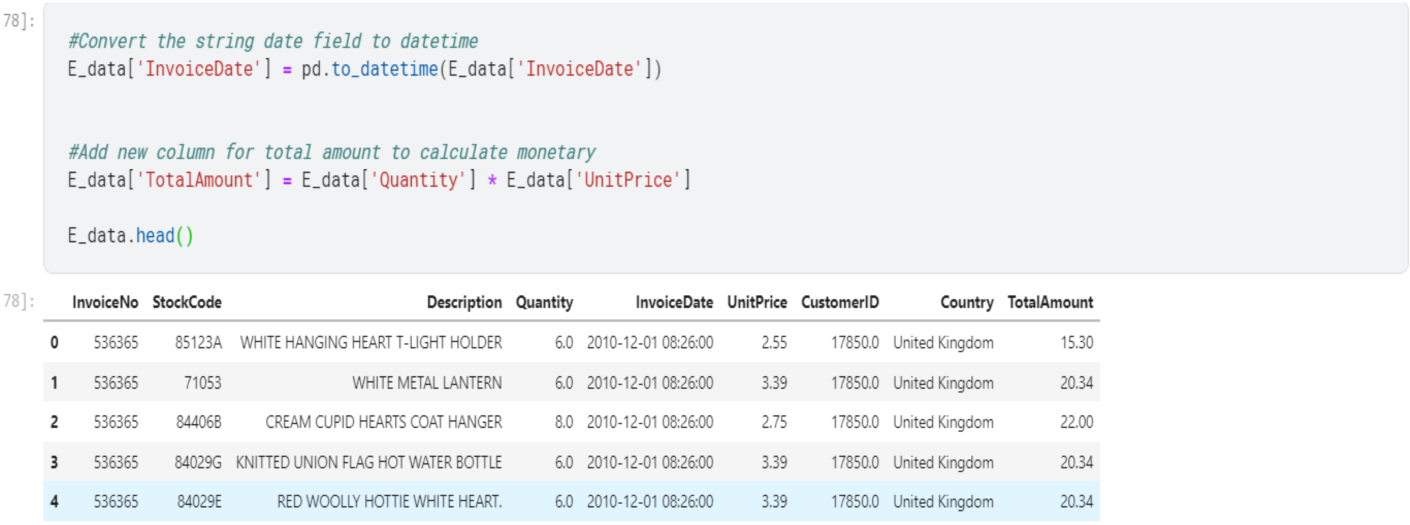
After filtering the data, it then contained [397924 rows x 8 columns]



```
[77]: #Check the shape (number of columns and rows) in the dataset after data is cleaned
      E_data.shape

[77]: (397924, 8)
```

5. Prepare the data for the RFM Analysis

We prepare the data by converting the date from string format into datetime format so that we can apply calculations on it, and by creating a new column “Total Amount” to calculate the Monetary.

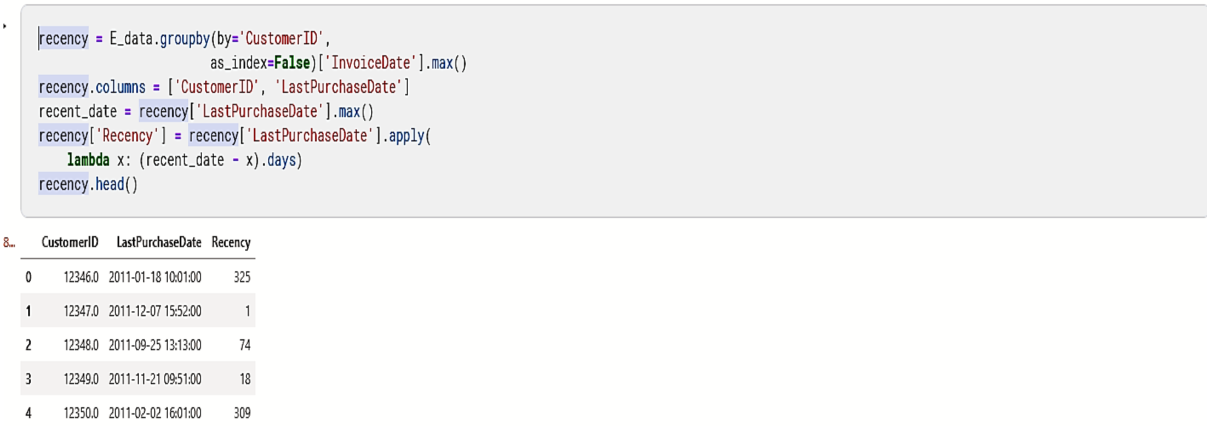


```
78]: #Convert the string date field to datetime
     E_data['InvoiceDate'] = pd.to_datetime(E_data['InvoiceDate'])


     #Add new column for total amount to calculate monetary
     E_data['TotalAmount'] = E_data['Quantity'] * E_data['UnitPrice']

     E_data.head()
```

78]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	TotalAmount
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6.0	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	15.30
1	536365	71053	WHITE METAL LANTERN	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	20.34
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8.0	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	22.00
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	20.34
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	20.34

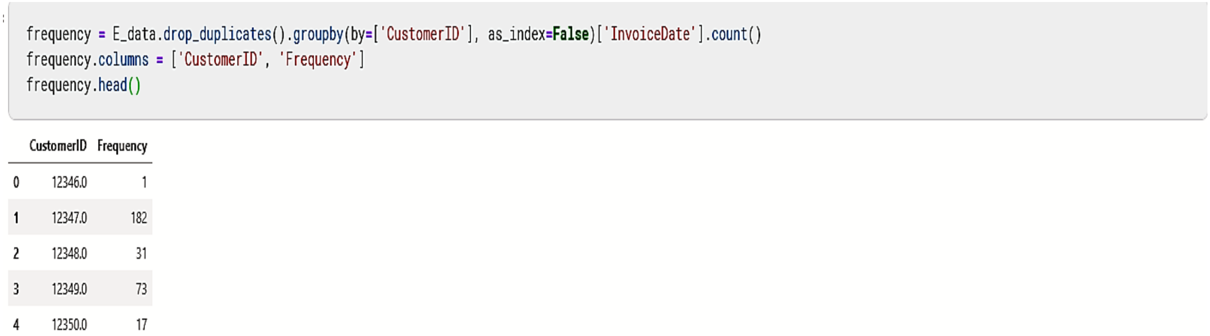
6. RFM Calculations



```
recency = E_data.groupby(by='CustomerID',
                         as_index=False)['InvoiceDate'].max()
recency.columns = ['CustomerID', 'LastPurchaseDate']
recent_date = recency['LastPurchaseDate'].max()
recency['Recency'] = recency['LastPurchaseDate'].apply(
    lambda x: (recent_date - x).days)
recency.head()
```

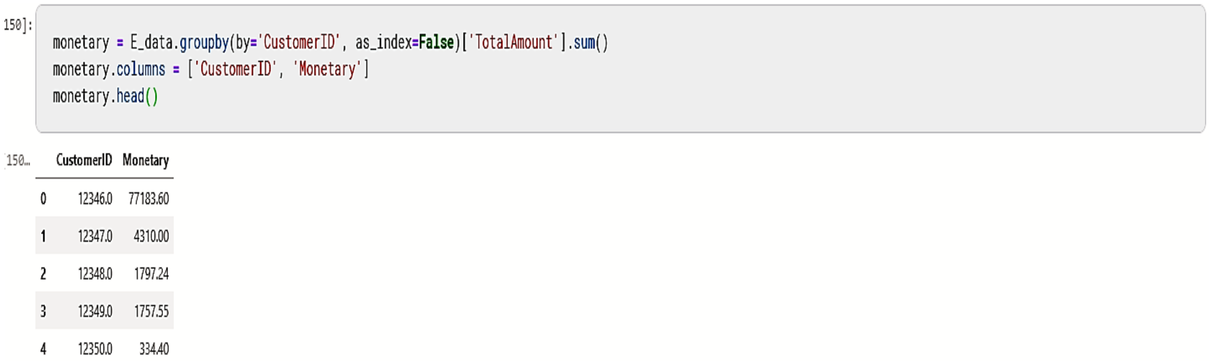
8...

	CustomerID	LastPurchaseDate	Recency
0	12346.0	2011-01-18 10:01:00	325
1	12347.0	2011-12-07 15:52:00	1
2	12348.0	2011-09-25 13:13:00	74
3	12349.0	2011-11-21 09:51:00	18
4	12350.0	2011-02-02 16:01:00	309



```
frequency = E_data.drop_duplicates().groupby(by=['CustomerID'], as_index=False)['InvoiceDate'].count()
frequency.columns = ['CustomerID', 'Frequency']
frequency.head()
```

	CustomerID	Frequency
0	12346.0	1
1	12347.0	182
2	12348.0	31
3	12349.0	73
4	12350.0	17



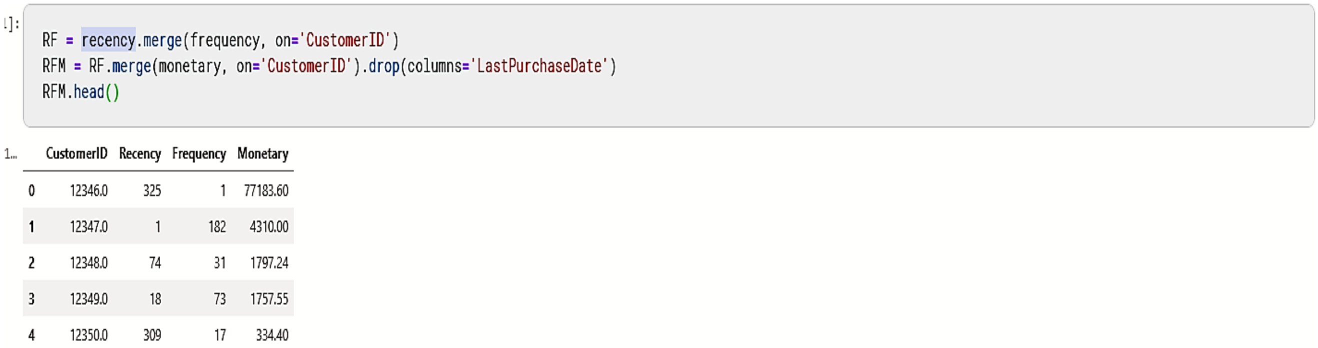
150]:

```
monetary = E_data.groupby(by='CustomerID', as_index=False)['TotalAmount'].sum()
monetary.columns = ['CustomerID', 'Monetary']
monetary.head()
```

150...

	CustomerID	Monetary
0	12346.0	77183.60
1	12347.0	4310.00
2	12348.0	1797.24
3	12349.0	1757.55
4	12350.0	334.40

7. RFM Outputs



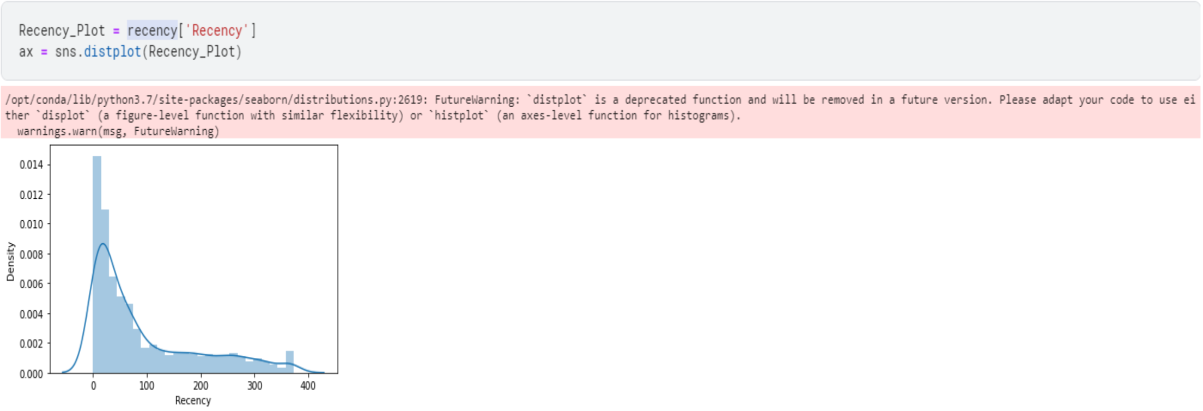
]:

```
RF = recency.merge(frequency, on='CustomerID')
RFM = RF.merge(monetary, on='CustomerID').drop(columns='LastPurchaseDate')
RFM.head()
```

1...

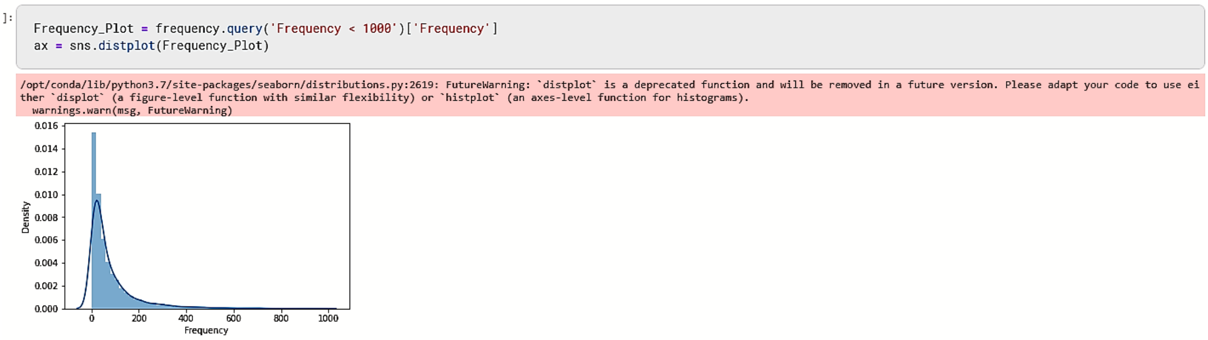
	CustomerID	Recency	Frequency	Monetary
0	12346.0	325	1	77183.60
1	12347.0	1	182	4310.00
2	12348.0	74	31	1797.24
3	12349.0	18	73	1757.55
4	12350.0	309	17	334.40

8. RFM Visualization



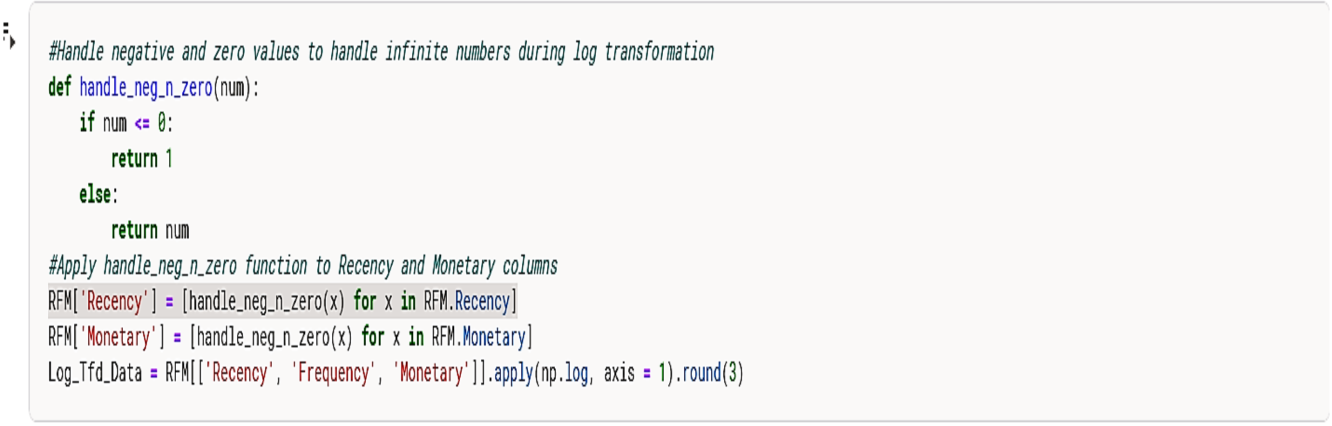
```
Recency_Plot = recency['Recency']
ax = sns.distplot(Recency_Plot)
```

```
/opt/conda/lib/python3.7/site-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use ei
ther `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

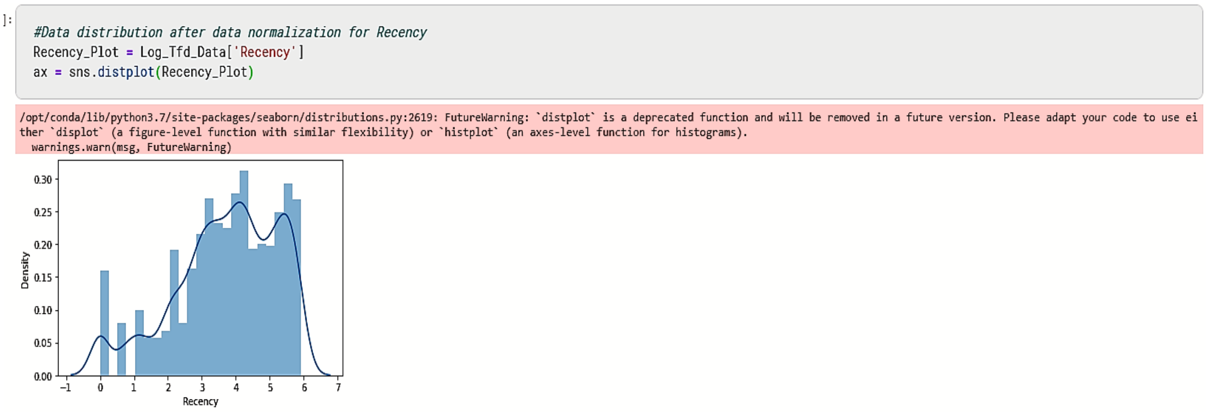


```
Frequency_Plot = frequency.query('Frequency < 1000')['Frequency']
ax = sns.distplot(Frequency_Plot)
```

```
/opt/conda/lib/python3.7/site-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use ei
ther `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

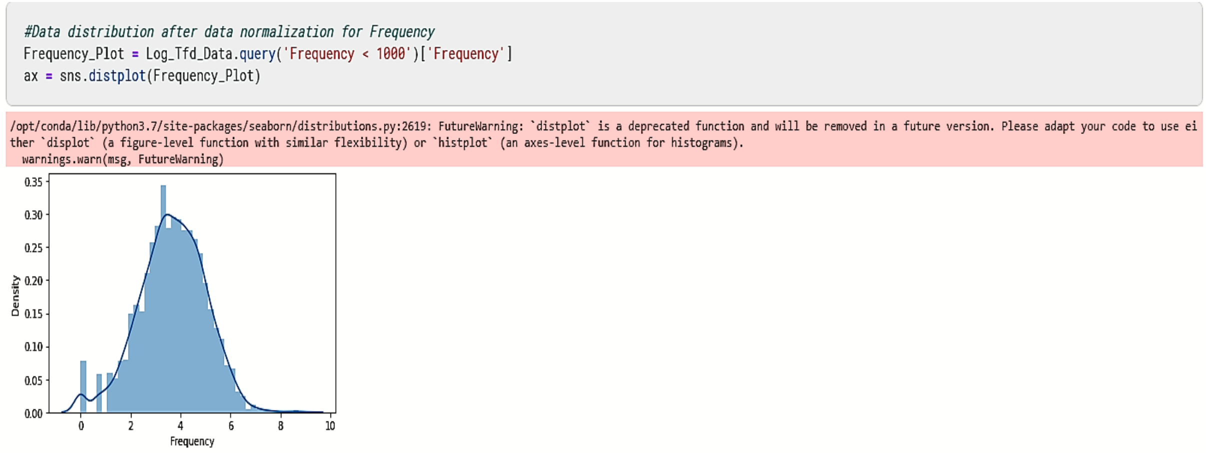


```
#Handle negative and zero values to handle infinite numbers during log transformation
def handle_neg_n_zero(num):
    if num <= 0:
        return 1
    else:
        return num
#Apply handle_neg_n_zero function to Recency and Monetary columns
RFM['Recency'] = [handle_neg_n_zero(x) for x in RFM.Recency]
RFM['Monetary'] = [handle_neg_n_zero(x) for x in RFM.Monetary]
Log_Tfd_Data = RFM[['Recency', 'Frequency', 'Monetary']].apply(np.log, axis = 1).round(3)
```



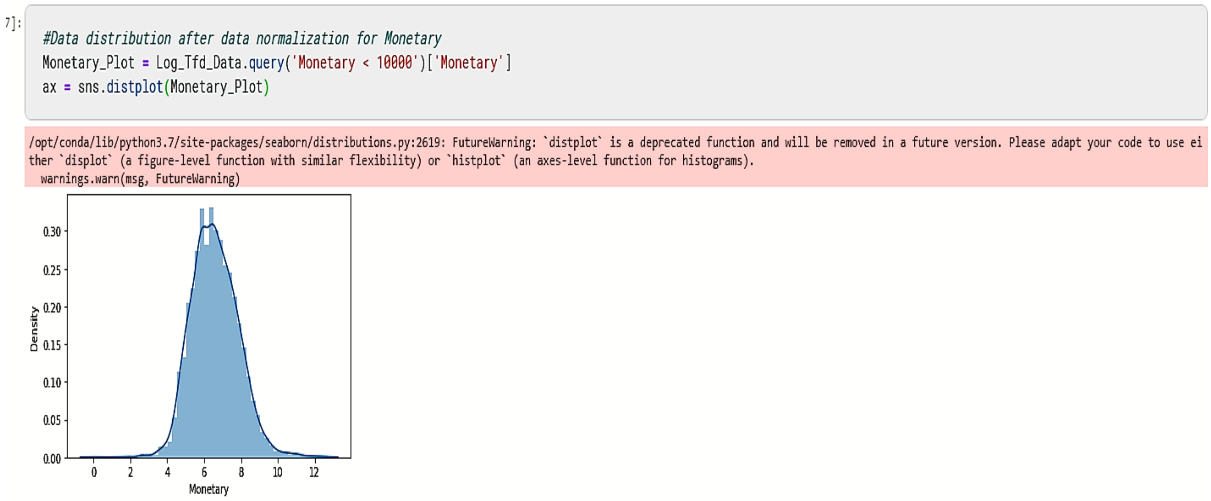
```
#Data distribution after data normalization for Recency
Recency_Plot = Log_Tfd_Data['Recency']
ax = sns.distplot(Recency_Plot)
```

```
/opt/conda/lib/python3.7/site-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use ei
ther `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



```
#Data distribution after data normalization for Frequency
Frequency_Plot = Log_Tfd_Data.query('Frequency < 1000')['Frequency']
ax = sns.distplot(Frequency_Plot)
```

```
/opt/conda/lib/python3.7/site-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use ei
ther `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

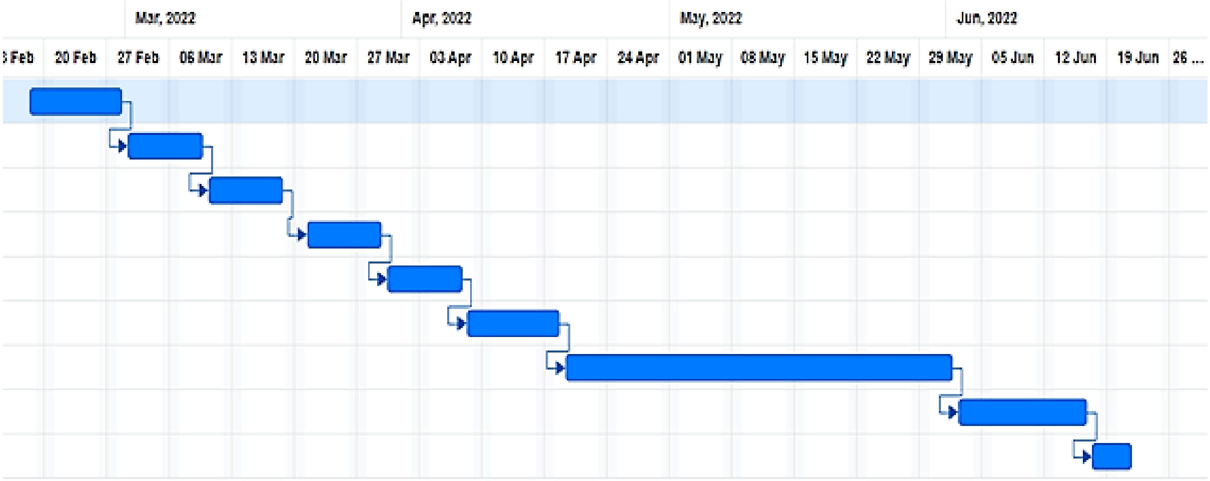


```
7]:
#Data distribution after data normalization for Monetary
Monetary_Plot = Log_Tfd_Data.query('Monetary < 10000')['Monetary']
ax = sns.distplot(Monetary_Plot)
```

```
/opt/conda/lib/python3.7/site-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use ei
ther `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

9. **Phase two**
10. **Identify optimal number of K.**
11. **Apply K-means algorithm.**
12. **Calculate the accuracy of the algorithm.**
13. **Visualize Plots.**
14. **Combine all these plots in a dashboard.**
15. **Writing the insights that we get from the dashboard.**
16. **Test our model with other datasets and visualize them.**
17. **Deploy our model so it can be used in web applications.**

VI. PROJECT TIME FRAME



ID	Name	Start Date	End Date
1	Searching for Ecommerce data	Feb 18, 2022	Feb 28, 2022
2	Visualize the data	Mar 01, 2022	Mar 09, 2022
3	Studying Customer Segmentation	Mar 10, 2022	Mar 18, 2022
4	Studying Unsupervised ML	Mar 21, 2022	Mar 29, 2022
5	choosing K-Means Algorithm	Mar 30, 2022	Apr 07, 2022
6	Studying RFM Analysis	Apr 08, 2022	Apr 18, 2022
7	Pre-processing the data	Apr 19, 2022	Jun 01, 2022
8	Applying RFM Analysis	Jun 02, 2022	Jun 16, 2022
9	Document what we did	Jun 17, 2022	Jun 20, 2022

VII. *ACTIVITIES*

Below are the activities and sequence upon which we intend to implement the project.

- We shall import customer data set and process it. This shall involve data cleaning such as eliminating rows which have cells with missing values.
- Next shall be data analysis which shall involve knowing the number of rows and columns in the data set, knowing the type of data under each column, et cetera.
- We shall then perform RFM analysis.
- Our next step shall involve finding the optimum number of clusters K by using a parameter WCSS (Within Clusters Sum of Squares).
- We shall then fit the data to K-Means clustering.
- We shall finally visualize the clusters on a scatter plot so that insights can then be derived from the dataset.

VIII. *PROJECT'S EXPECTED OUTPUT*

Below are the expected output of the model:

- Build a robust and efficient machine learning model to segment customer data.
- Build a dash board to aid in gathering of insights from the data visualized.

The above outputs shall aid to provide outcomes such as:

- Optimized and effective marketing campaigns.
- Improved customer satisfaction.
- Right decisions on management, expansion, et cetera.

IX. *BENEFICIARIES.*

Data scientists, Data analysts, Business analysts and business owners shall be the main beneficiaries of this model.

X. *RELATED DOCUMENTS.*

https://www.analyticsvidhya.com/blog/2021/06/kmodes-clustering-algorithm-for-categorical-data/#:~:text=KModes%20clustering%20is%20one%20of,similar%20our%20data%20points%20are.
https://www.onlinegantt.com/#/gantt
https://archive.ics.uci.edu/ml/datasets/online+retail