CISC 886: Project Part B – P2 – Customer Segmentation

DEBI – UNDER SUPERVISION OF DR. ANWAR HOSSAIN

GROUP 9
ABOELELA – 22398556
ELREEDY – 20398548
IBRAHIM – 20398554
MORSY – 20398551
SAYED – 20398048

Queen's University - DEBI

Contents

Problem specification:

Customer Segmentation: *Use Spark to analyze customer data and segment customers based on their behavior, demographics, and other characteristics. You can use this information to personalize marketing campaigns and improve customer retention.*

Customer segmentation is one of the most important process that businesses depend on. It allows the business to better understand its customers' behavior, needs and desires, in order to deliver the most profitable products and services. This project aims to check and segment customers on an online retail data.

Data Collection:

Our Dataset is a consolidated data for an international online retail purchase platform. It consists of different types of data which is related to both the product and the customers.
The data contains the customers' countries, their time of purchase and their customer ID. It also contains the invoices paid, the unit cost for each purchase, description of the product, the product's stock number and the quantity bought by the customers.
The dataset is in CSV format in order to be easily processed using Spark.

Sample from the Dataset:

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850	United Kingdom
536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850	United Kingdom
536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850	United Kingdom
536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850	United Kingdom
536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850	United Kingdom
536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/1/2010 8:26	7.65	17850	United Kingdom
536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	12/1/2010 8:26	4.25	17850	United Kingdom

Table. 1

Data Preparation:

1. The dataset was examined to check if there are any missing values across all columns

   a. This is what was found

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	1454	0	0	0	135080	0

Table. 2

   b. The nulls were dropped from both columns

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	0	0	0	0	0	0

Table. 3

2. The dataset was examined to check if any outliers were found

Country	count
United Kingdom	495478
Germany	9495
France	8557
EIRE	8196
Spain	2533
Netherlands	2371
Belgium	2069
Switzerland	2002
Portugal	1519
Australia	1259

only showing top 10 rows

Table. 4

This table shows that the United Kingdom has the most count, thus, the data will be biased for the United Kingdom

Thus, it was decided to use only data related to United Kingdom to avoid bias.

```
+---------+---------+--------------------+--------+-------------+---------+----------+--------------+
|InvoiceNo|StockCode|         Description|Quantity|  InvoiceDate|UnitPrice|CustomerID|       Country|
+---------+---------+--------------------+--------+-------------+---------+----------+--------------+
|   536365|   85123A|WHITE HANGING HEA...|       6|12/1/2010 8:26|     2.55|     17850|United Kingdom|
|   536365|    71053| WHITE METAL LANTERN|       6|12/1/2010 8:26|     3.39|     17850|United Kingdom|
|   536365|   84406B|CREAM CUPID HEART...|       8|12/1/2010 8:26|     2.75|     17850|United Kingdom|
|   536365|   84029G|KNITTED UNION FLA...|       6|12/1/2010 8:26|     3.39|     17850|United Kingdom|
|   536365|   84029E|RED WOOLLY HOTTIE...|       6|12/1/2010 8:26|     3.39|     17850|United Kingdom|
+---------+---------+--------------------+--------+-------------+---------+----------+--------------+
```

Table. 5

3. The dataset was examined in the numerical columns like the quantity to filter any negative value – if any as this doesn't make sense.

4. As both "unit price" and "quantity" can be combined into a "total price" column.

Feature Engineering:

The team went through some research to find a suitable method to help us in the customer segmentation and it was decided upon The Recency, Frequency & Monetary "RFM" model.

The new feature that are needed now are the three pillars of the model – below is each pillar separately and how it was achieved:

1. Recency:

   a. Dates for customers' purchases – according to the invoices – were grouped into the nearest date and the furthest date (minimum and maximum).

```python
print(df.InvoiceDate.min())
print(df.InvoiceDate.max())
df["InvoiceDate"]
```

```
1/10/2011 10:32
9/9/2011 9:52

0          12/1/2010 8:26
1          12/1/2010 8:26
2          12/1/2010 8:26
3          12/1/2010 8:26
4          12/1/2010 8:26
               ...
354340    12/9/2011 12:31
354341    12/9/2011 12:49
354342    12/9/2011 12:49
354343    12/9/2011 12:49
354344    12/9/2011 12:49
Name: InvoiceDate, Length: 354345, dtype: object
```

```python
LastDate=df.InvoiceDate.max()
print(LastDate)
print(pd.DateOffset(days=1))
print(df.InvoiceDate)
```

```
2011-12-09 12:49:00
<DateOffset: days=1>
0        2010-12-01 08:26:00
1        2010-12-01 08:26:00
2        2010-12-01 08:26:00
3        2010-12-01 08:26:00
4        2010-12-01 08:26:00
                 ...
354340   2011-12-09 12:31:00
354341   2011-12-09 12:49:00
354342   2011-12-09 12:49:00
354343   2011-12-09 12:49:00
354344   2011-12-09 12:49:00
Name: InvoiceDate, Length: 354345, dtype: datetime64[ns]
```

b. Then their format was adjusted to be able to calculate the Recency for each customer

```python
df["InvoiceDate"] = pd.to_datetime(df["InvoiceDate"])
df["InvoiceDate"]
```

```
0        2010-12-01 08:26:00
1        2010-12-01 08:26:00
2        2010-12-01 08:26:00
3        2010-12-01 08:26:00
4        2010-12-01 08:26:00
                 ...
354340   2011-12-09 12:31:00
354341   2011-12-09 12:49:00
354342   2011-12-09 12:49:00
354343   2011-12-09 12:49:00
354344   2011-12-09 12:49:00
Name: InvoiceDate, Length: 354345, dtype: datetime64[ns]
```

c. Recency calculation

```
#calculating our recency value
LastDate=df.InvoiceDate.max() #calculating the last date of InvoiceDate
LastDate = LastDate + pd.DateOffset(days=1)
df["Diff"] = LastDate - df.InvoiceDate
recency = df.groupby("CustomerID").Diff.min()
recency = recency.reset_index()
recency.head(10)
```

	CustomerID	Diff
0	12346	326 days 02:48:00
1	12747	2 days 22:15:00
2	12748	1 days 00:29:00
3	12749	4 days 02:53:00
4	12820	3 days 21:37:00
5	12821	214 days 20:58:00
6	12822	71 days 02:45:00
7	12823	75 days 05:14:00
8	12824	60 days 00:00:00
9	12826	3 days 02:24:00

Table. 6

2. Frequency:

The Frequency was calculated by getting the count of purchases (invoices) of each customer using “Groupby” and “Count”.

```
frequency=df.groupby("CustomerID").InvoiceNo.count()
frequency = frequency.reset_index()
frequency.head()
```

	CustomerID	InvoiceNo
0	12346	1
1	12747	103
2	12748	4596
3	12749	199
4	12820	59

Table. 7

3. <u>Monetary:</u>

The monetary was calculated using the total amount of each customer's purchases from the previously generated "Total Amount" column – stated in the previous section of data preprocessing- using "Groupby" and "Sum".

```
# calculating the monetary values
monetary =df.groupby("CustomerID").TotalAmount.sum()
monetary = monetary.reset_index()
monetary.head()
```

	CustomerID	TotalAmount
0	12346	77184.0
1	12747	4207.0
2	12748	33956.0
3	12749	4115.0
4	12820	946.0

Table. 8

4. <u>Final Shape:</u>

The three pillars were merged together with the CustomerID to generate our desired analytical dataset.

_c0	CustomerID	Monetary	Frequence	Recency
0	12346	77184.0	1	326
1	12747	4207.0	103	2
2	12748	33956.0	4596	1
3	12749	4115.0	199	4
4	12820	946.0	59	3
5	12821	93.0	6	214
6	12822	953.0	46	71
7	12823	1761.0	5	75
8	12824	399.0	25	60
9	12826	1498.0	91	3
10	12827	435.0	25	6

Table. 9

5. <u>Standardization and processing the new features:</u>

Data was visualized to check the shape of each pillar in order to preprocess it before going into the Machine Learning ML model creation

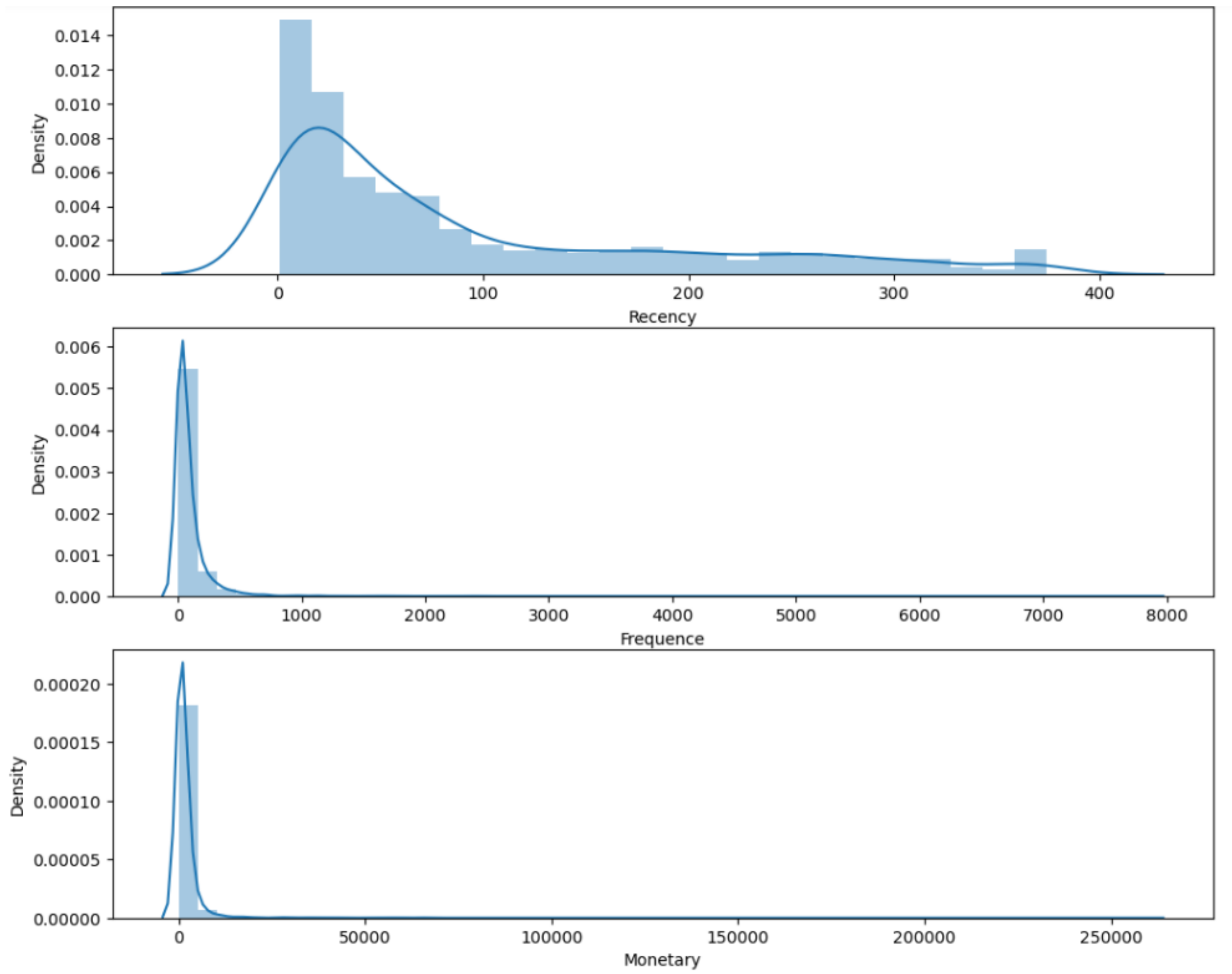

Fig.1

a. Negative and Zeros were removed
b. All features were vectorized and standardized as they were on different scale and this is not suitable for ML.

```
# vectorize all features
assembler = VectorAssembler(inputCols=features, outputCol="rfm_features")
assembled_data = assembler.transform(rfm_data)
assembled_data = assembled_data.select('CustomerID', 'rfm_features')
assembled_data.show(5)
```

```
+----------+-------------------+
|CustomerID|       rfm_features|
+----------+-------------------+
|     12346| [77184.0,1.0,326.0]|
|     12747|   [4207.0,103.0,2.0]|
|     12748|[33956.0,4596.0,1.0]|
|     12749|   [4115.0,199.0,4.0]|
|     12820|     [946.0,59.0,3.0]|
+----------+-------------------+
only showing top 5 rows
```

Table. 10

```
# Standardization
scaler = StandardScaler(inputCol='rfm_features', outputCol='rfm_standardized')
data_scale = scaler.fit(assembled_data)
scaled_data = data_scale.transform(assembled_data)
scaled_data.show(5)
```

```
+----------+-------------------+--------------------+
|CustomerID|       rfm_features|    rfm_standardized|
+----------+-------------------+--------------------+
|     12346| [77184.0,1.0,326.0]|[10.3115381937557...|
|     12747|   [4207.0,103.0,2.0]|[0.56204188926630...|
|     12748|[33956.0,4596.0,1.0]|[4.53641416494571...|
|     12749|   [4115.0,199.0,4.0]|[0.54975098034961...|
|     12820|     [946.0,59.0,3.0]|[0.12638260690418...|
+----------+-------------------+--------------------+
only showing top 5 rows
```

Table. 11

Model Selection:

The Model used in this project is a clustering unsupervised model "the K-means model" to cluster the customers as per the RMF model.

A "for loop" was applied to check different values of K and their costs in order to plot the "Elbow Curve" and choose our desired and suitable K for our segmentation process

1. Elbow Curve:

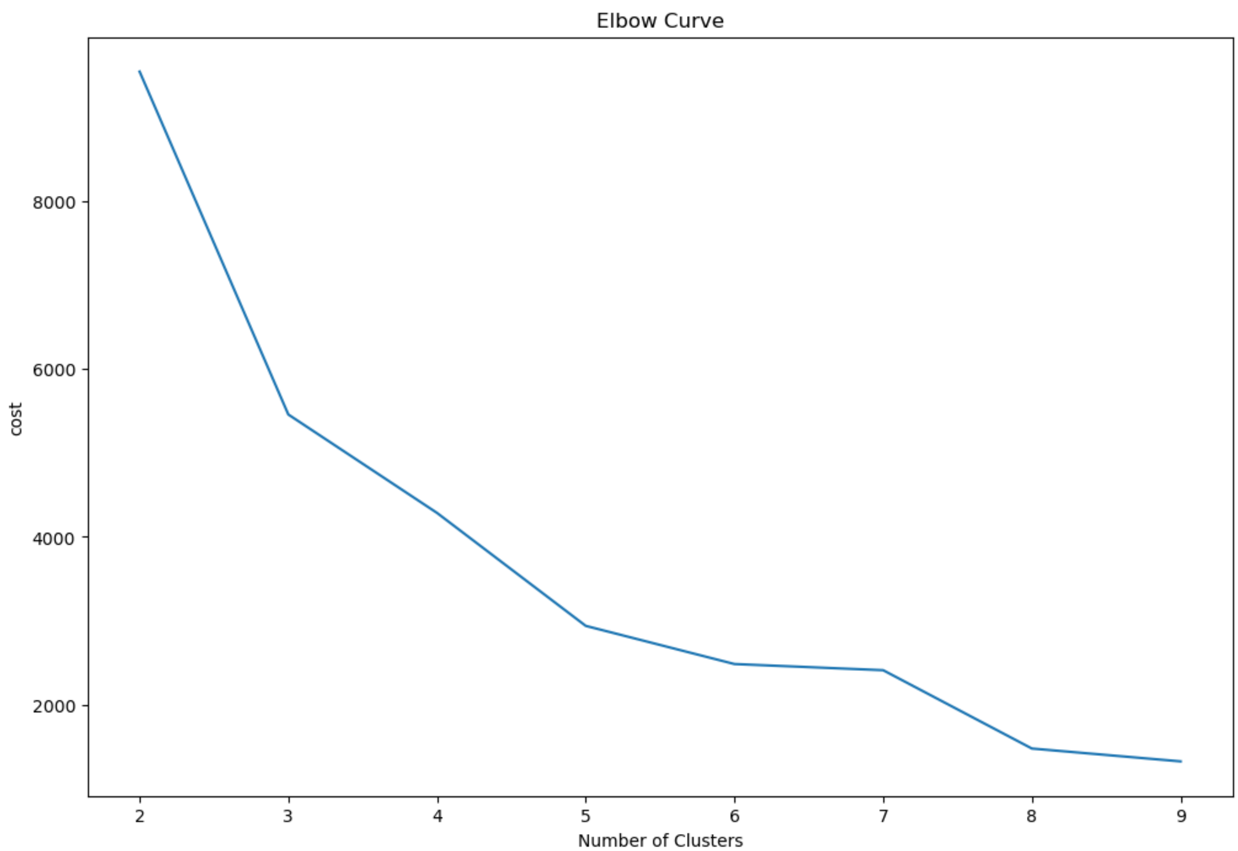

Fig.2

2. Cost table:

	cluster	cost
0	2	9548.640415
1	3	5459.652179
2	4	4286.082248
3	5	2939.182207
4	6	2484.076131
5	7	2410.453773
6	8	1476.536593
7	9	1323.310370

Table. 12

It was decided that according to our dataset and the required segmentation with respect to the cost of each K of clusters, K = 5 was chosen. Acceptable cost with respect to other K like 2 and an acceptable amount of segmentations for the customers.

3. Applying the model:

   As previously stated K was chosen to be 5 and below are the dataset including the cluster segmentation and the clusters plot.

   a. Dataset with segmentation as per each cluster number (predicted):

	CustomerID	prediction	Monetary	Frequence	Recency
0	12346	4	77184.0	1	326
1	12747	0	4207.0	103	2
2	12748	3	33956.0	4596	1
3	12749	0	4115.0	199	4
4	12820	0	946.0	59	3
...	...	...	...	...	...
3916	18280	2	182.0	10	278
3917	18281	2	81.0	7	181
3918	18282	0	181.0	12	8
3919	18283	4	2140.0	756	4
3920	18287	0	1836.0	70	43

Table. 13

b. Clusters plot:

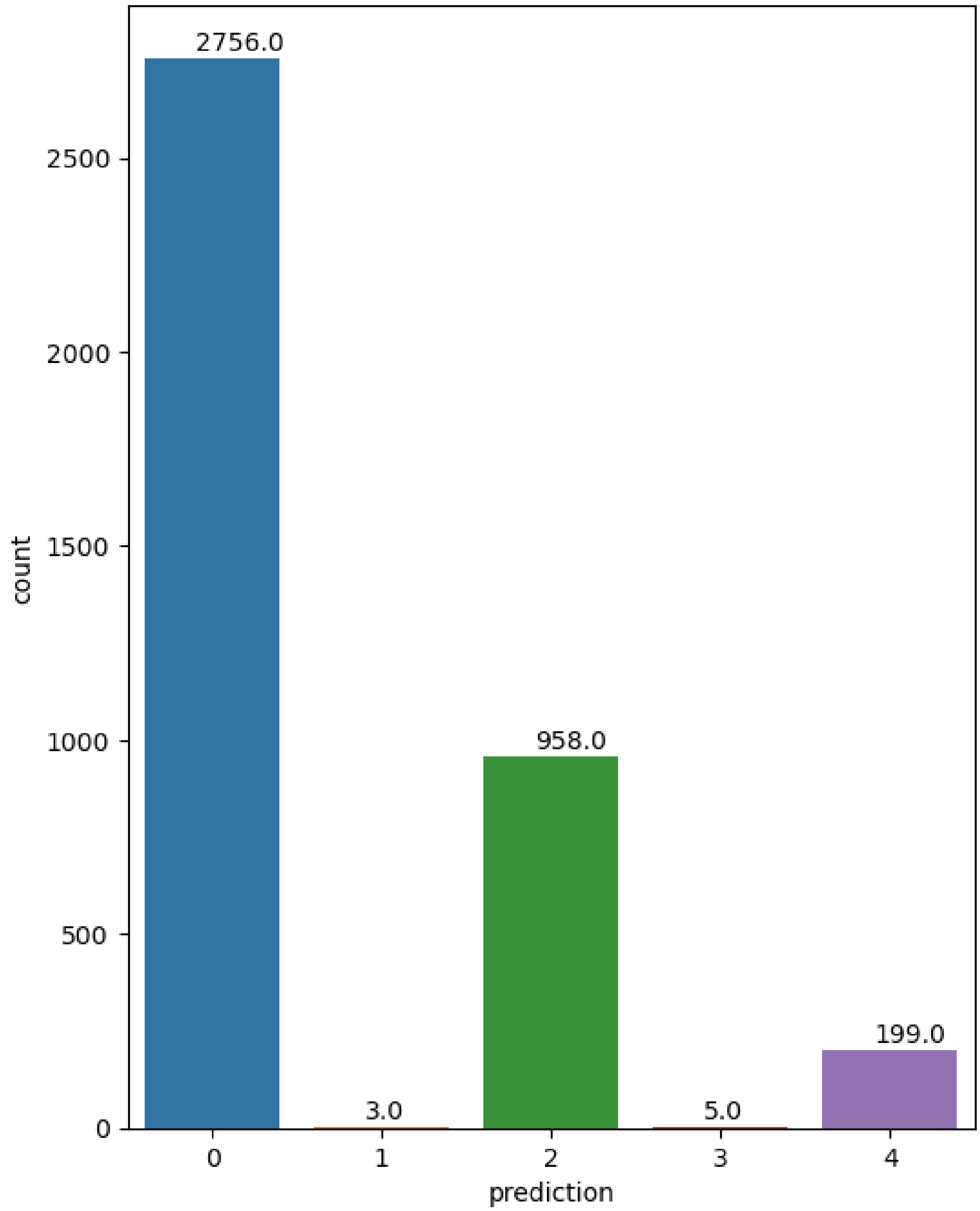

Fig.3

4. <u>Insights and analysis from the segmentation and the plot:</u>

- Highest count cluster is cluster 0 then 2 the 4.
- Lowest count cluster is cluster 1 then 3.

	Recency	Frequency	Monetary
Cluster 0	From 0 up to 150	From 0 up till around 1000	From 0 up till around 30000
Cluster 1	From 0 up to below 50	From 0 up till around 50	From 150000 up till around 300000
Cluster 2	From 150 till more than 350	From 0 up till around 100	From 0 up till around 10000
Cluster 3	From 0 up to 50	Scattered from 0 till 8000	Scattered between 0 and 50000
Cluster 4	From 0 up to 100	From 0 up till around 2000	From 30000 up till around 100000

Table. 14

Model Evaluation:

In order to evaluate the model, the three pillars were plotted and evaluated against each other with respect to all 5 clusters:

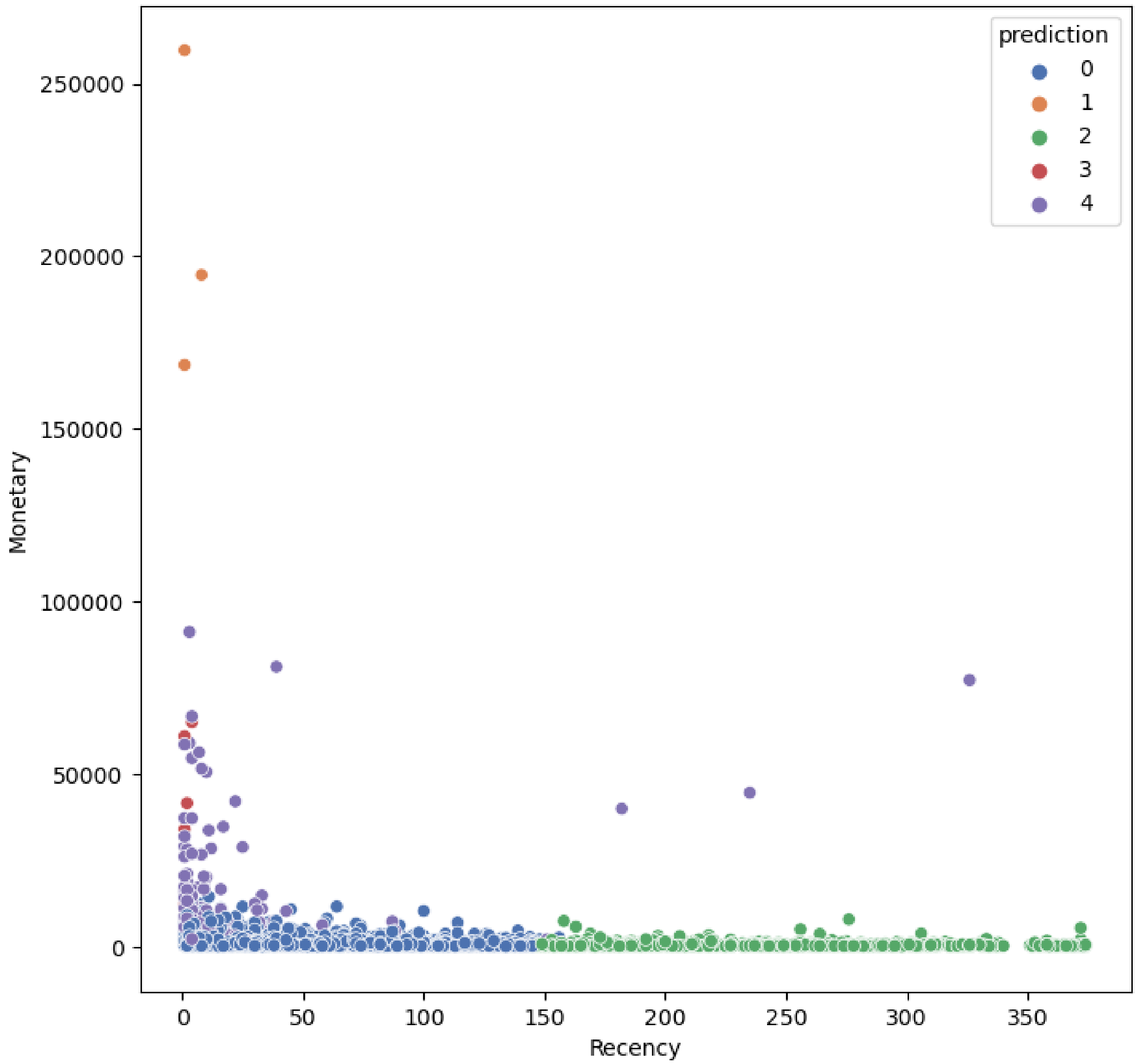

Fig.4

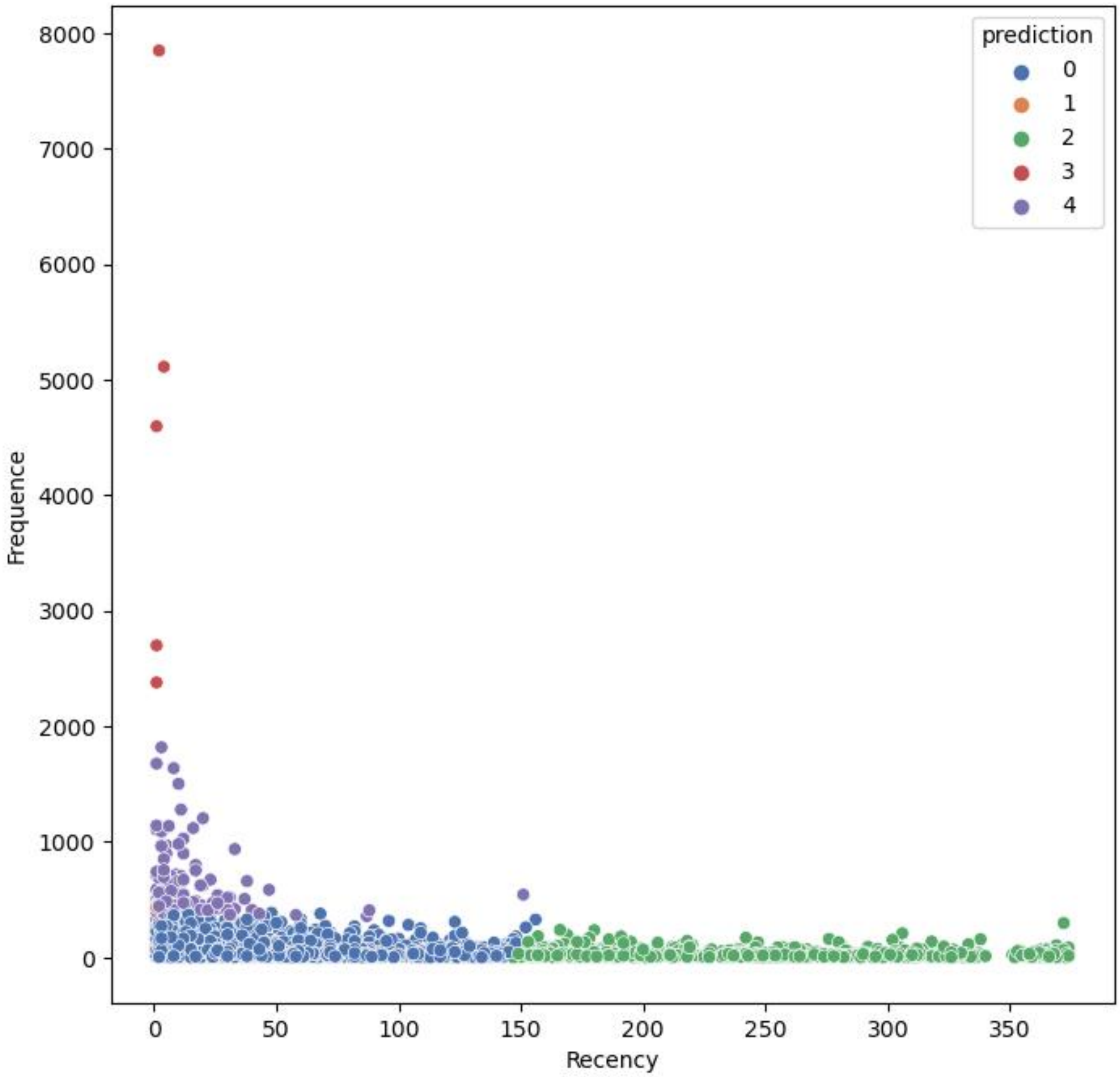
prediction
0
1
2
3
4
8000
7000
6000
5000
4000
3000
2000
1000
0
Frequence
0
50
100
150
200
250
300
350
Recency

Fig.5

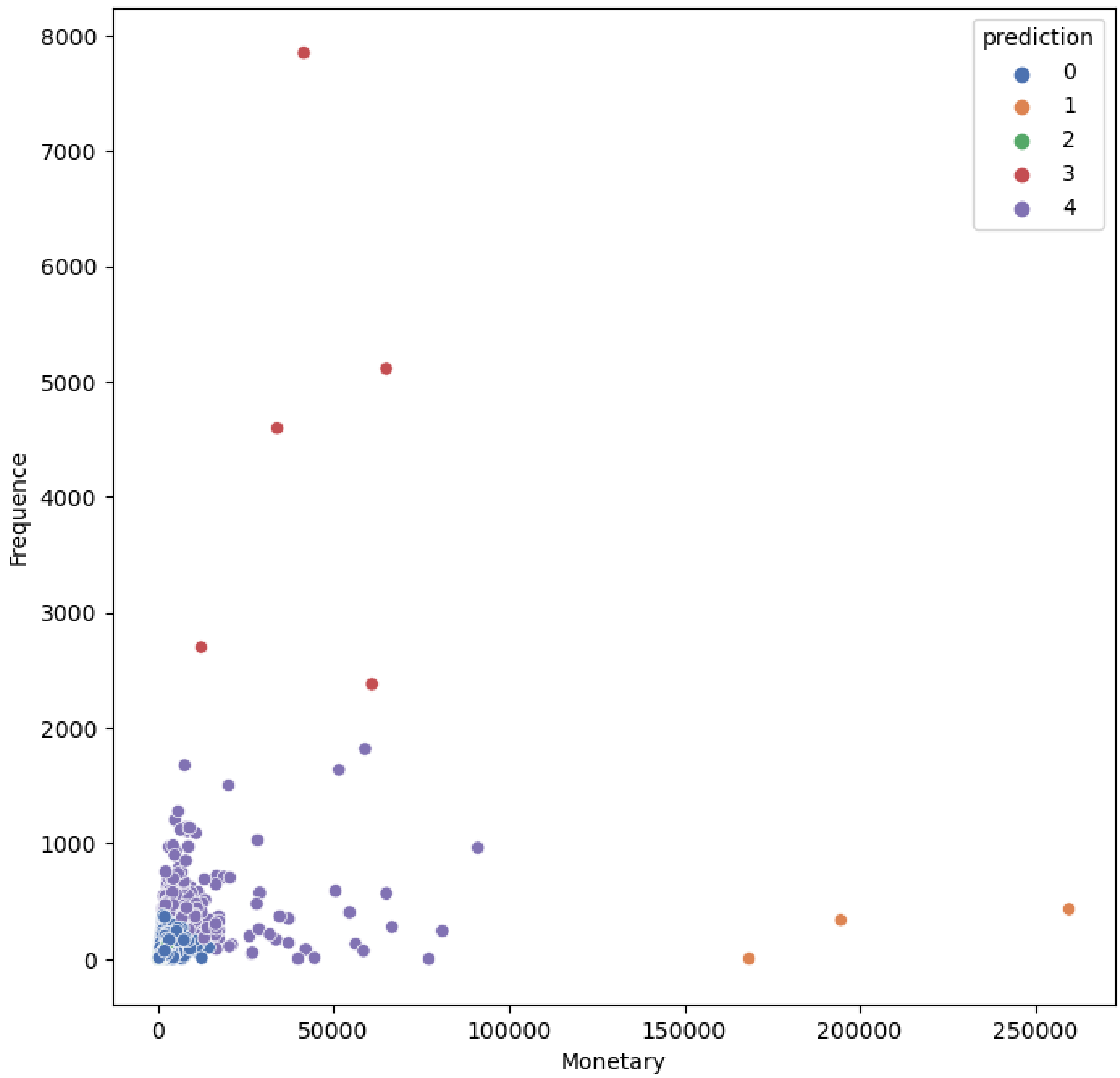

Fig.6

The three plots showed the same behavior for the 5 clusters:

- Cluster 1 and 3 are the rarest and scattered
- Cluster 0 is the highest
- All ranges for the three pillars are the same as table 14.

Segregation of Duties:

Duties	Names
Data Gathering and Project Research	All Team
Preprocessing	Ali Aboelela – Abdallah Ibrahim – Amr Sayed
Feature Engineering	Ali Aboelela – Abdallah Ibrahim – Amr Sayed
Model Selection	Bilal Morsy – Eslam Elreedy
Model Evaluation	Bilal Morsy – Eslam Elreedy
Report	All Team

Table. 15