Provided By: Outliers Team
Coach By: Muhammad Anwar Sanusi

Customer Segmentation

Using K-Means Clustering

DigitalSkola

What will we talk about?

1. Business Understanding
2. Data Understanding (EDA)
3. Analytical Approach
4. Approach 1 & Recommendation
5. Approach 2 & Recommendation
6. Conclusion

Business Understanding

Gain Customer Behavior to Improve Customer Service Relationships

Problem Statement:

How an e-commerce company can utilize customer's behavioral data to improve Customer Relationship Management (CRM)

Objectives:

- Improve customer service relationships
- Assist in customer retention
- Drive sales growth

Mission:

- Analyze and gather information about customer's behavior
- Create Cluster based on customer's behavior
- Make or apply marketing strategy recommendation for customer

Data Understanding (EDA)

DATA UNDERSTANDING (EDA)

Dataset given was 18 unique attributes

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9800 entries, 0 to 9799
Data columns (total 18 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   row_id         9800 non-null   int64
 1   order_id       9800 non-null   object
 2   order_date     9800 non-null   object
 3   ship_date      9800 non-null   object
 4   ship_mode      9800 non-null   object
 5   customer_id    9800 non-null   object
 6   customer_name  9800 non-null   object
 7   segment        9800 non-null   object
 8   country        9800 non-null   object
 9   city           9800 non-null   object
 10  state          9800 non-null   object
 11  postal_code    9789 non-null   float64
 12  region         9800 non-null   object
 13  product_id     9800 non-null   object
 14  category       9800 non-null   object
 15  sub_category   9800 non-null   object
 16  product_name   9800 non-null   object
 17  sales          9800 non-null   float64
dtypes: float64(2), int64(1), object(15)
memory usage: 1.3+ MB
```

```
df.nunique()

row_id          9800
order_id        4922
order_date      1230
ship_date       1326
ship_mode          4
customer_id      793
customer_name    793
segment            3
country            1
city             529
state             49
postal_code      626
region             4
product_id      1861
category           3
sub_category      17
product_name    1849
sales           5757
dtype: int64
```

- Customer Profile such as Customer ID, Customer Name, and Segment
- Demography such as Country, City, State, Postal Code, and Region
- Order Details such as Order ID, Order Date, Ship Date, and Ship Mode
- Product Details such as Product ID, Category, Sub-Category and Product Name
- Sales
- 2 numerical columns, 16 categorical columns

DigitalSkola

Drop Unnecessary Features

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9800 entries, 0 to 9799
Data columns (total 18 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   row_id         9800 non-null   int64
 1   order_id       9800 non-null   object
 2   order_date     9800 non-null   object
 3   ship_date      9800 non-null   object
 4   ship_mode      9800 non-null   object
 5   customer_id    9800 non-null   object
 6   customer_name  9800 non-null   object
 7   segment        9800 non-null   object
 8   country        9800 non-null   object
 9   city           9800 non-null   object
 10  state          9800 non-null   object
 11  postal_code    9789 non-null   float64
 12  region         9800 non-null   object
 13  product_id     9800 non-null   object
 14  category       9800 non-null   object
 15  sub_category   9800 non-null   object
 16  product_name   9800 non-null   object
 17  sales          9800 non-null   float64
dtypes: float64(2), int64(1), object(15)
memory usage: 1.3+ MB
```

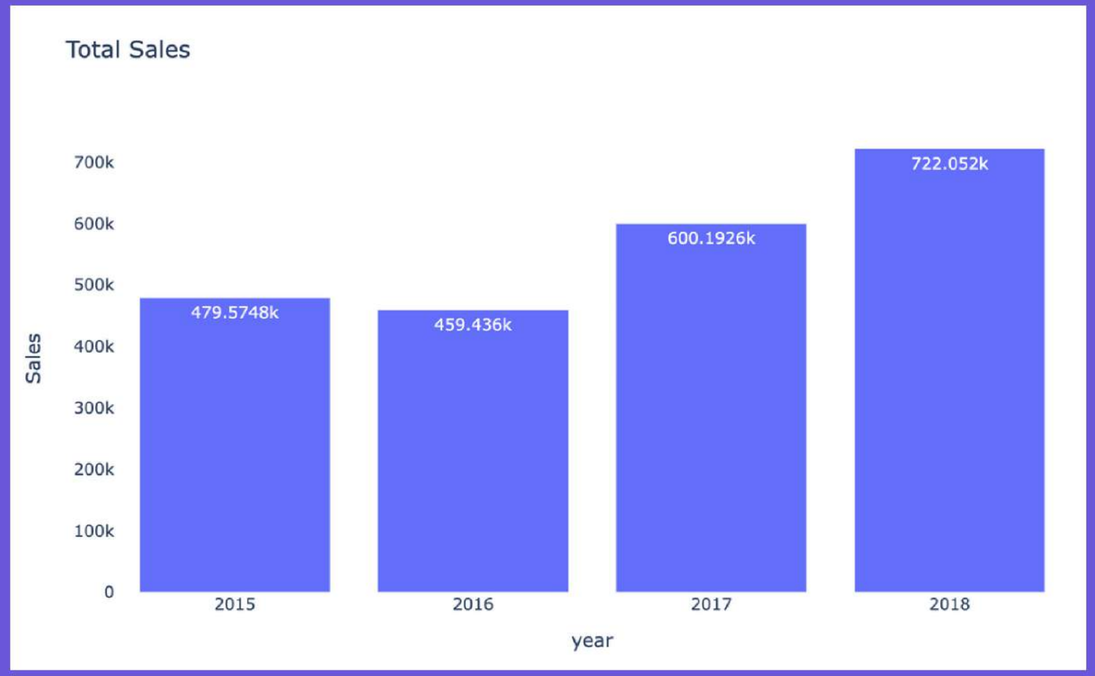
- Postal code attribute contains missing value.
- We considered to drop the entire column, because we can trace the postal code through city and state.
- Drop the row id so we can check duplicated value clearly

order_id object	order_date object	ship_date object	ship_mode object
US-2015-150119	23/04/2015	27/04/2015	Standard Class
US-2015-150119	23/04/2015	27/04/2015	Standard Class

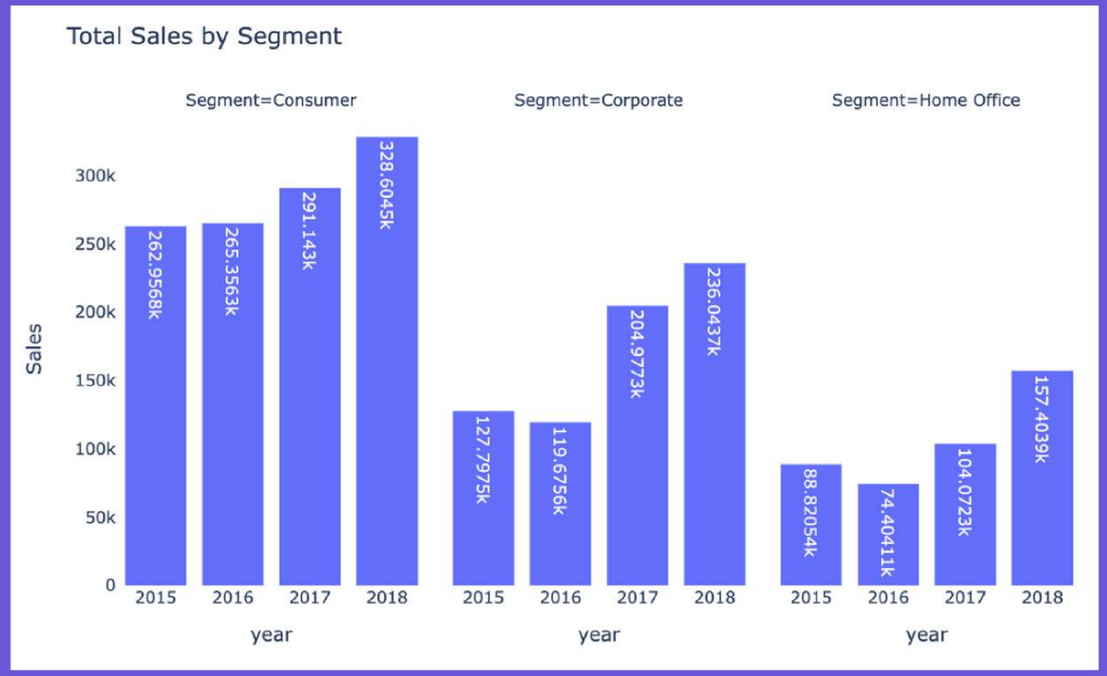
The dataset contains a duplicated data and we drop it.

Consumer segment is the only segment that is growing every year

Our sales slightly dropped in 2016, however we can manage to grow for the next 2 years.



The dropped coming from Corporate and Home Office ‘segment’.

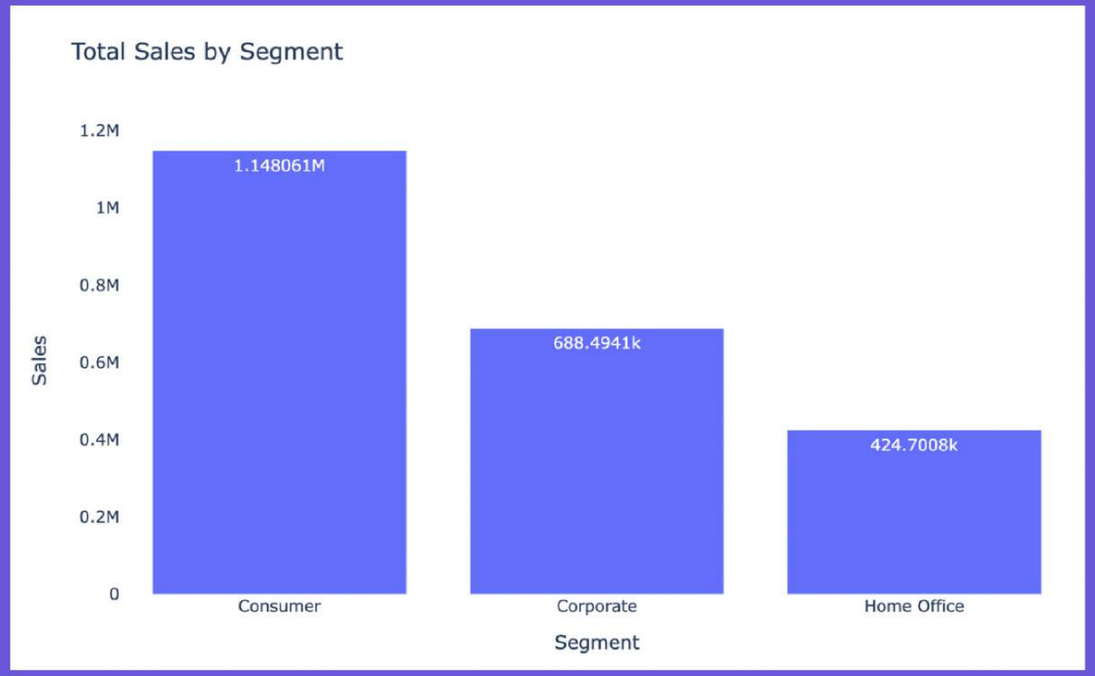


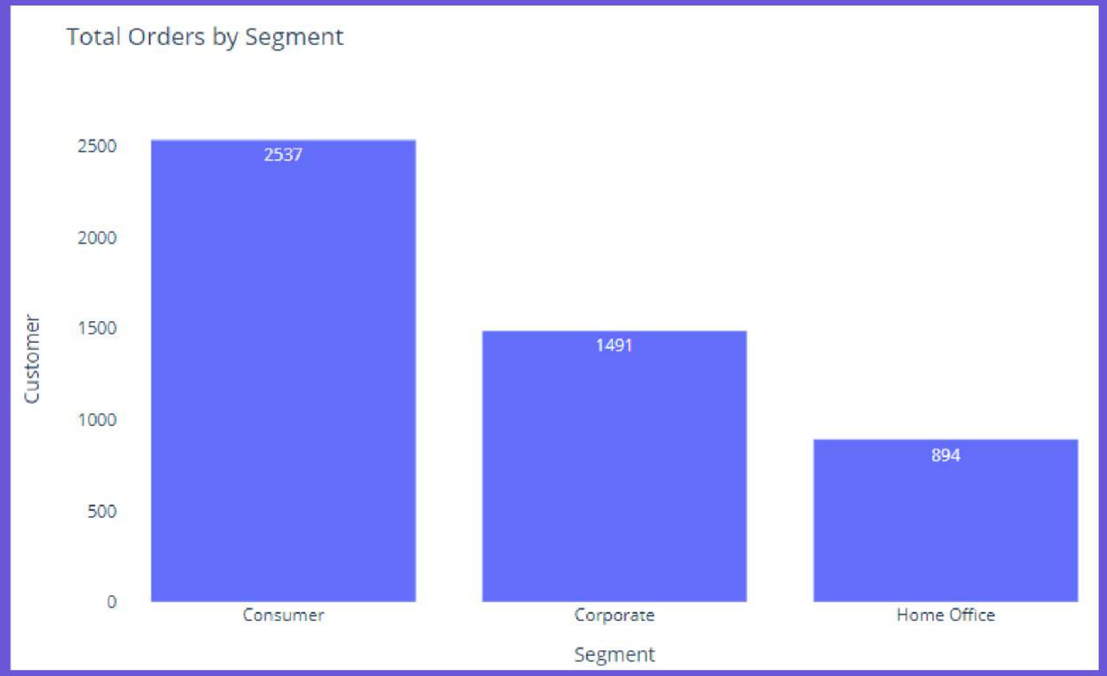
DigitalSkola

Consumer segment is the largest contribution of sales and transaction.

51% of sales coming from Consumer Segment

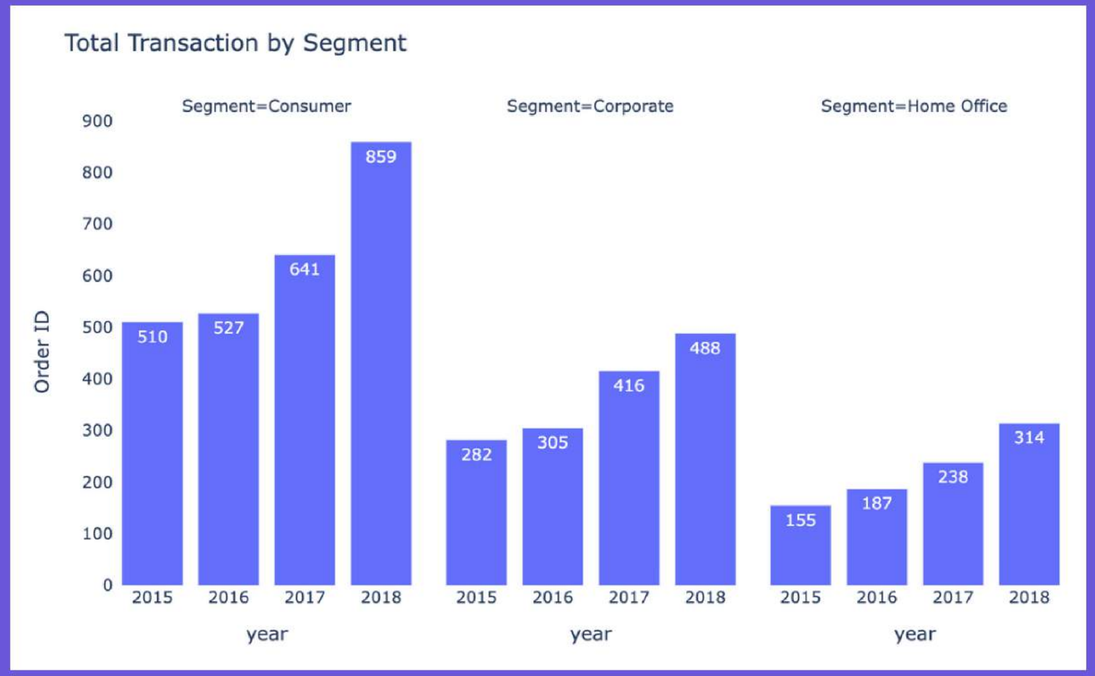
4.742 transaction made from total Customers.
49% comes from consumer segment.



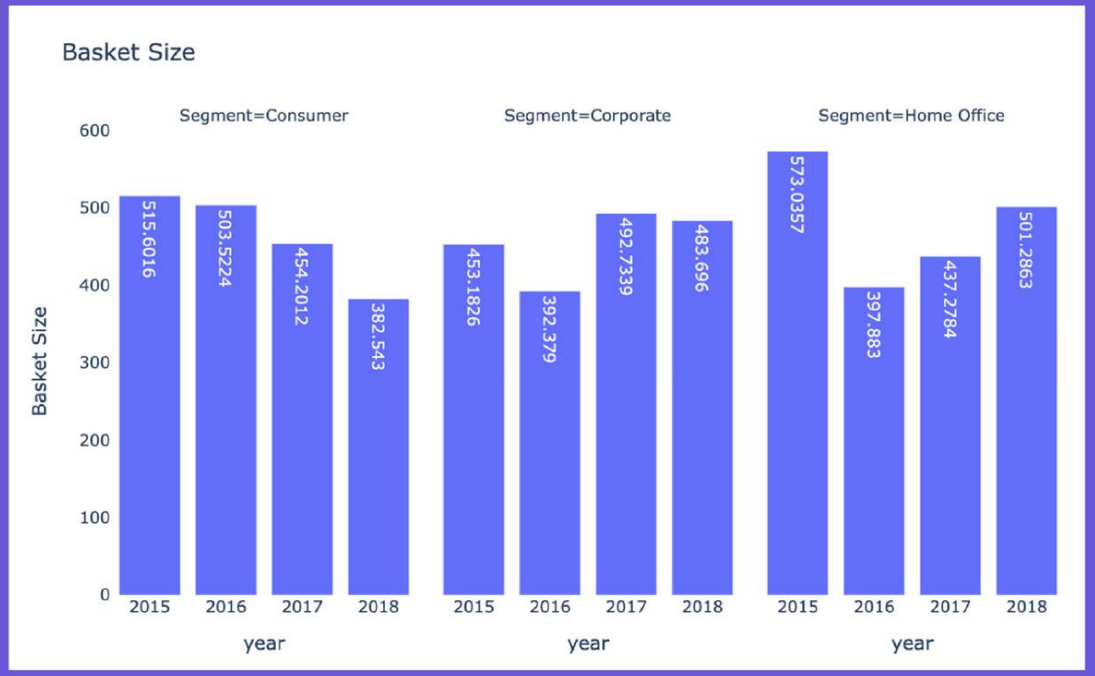


Customers are buying smaller basket size / cheaper product in Consumer Segment.

Transactions are growing across segments.



But it's not as expected that our Basket Size is declining, especially in Consumer Segment.

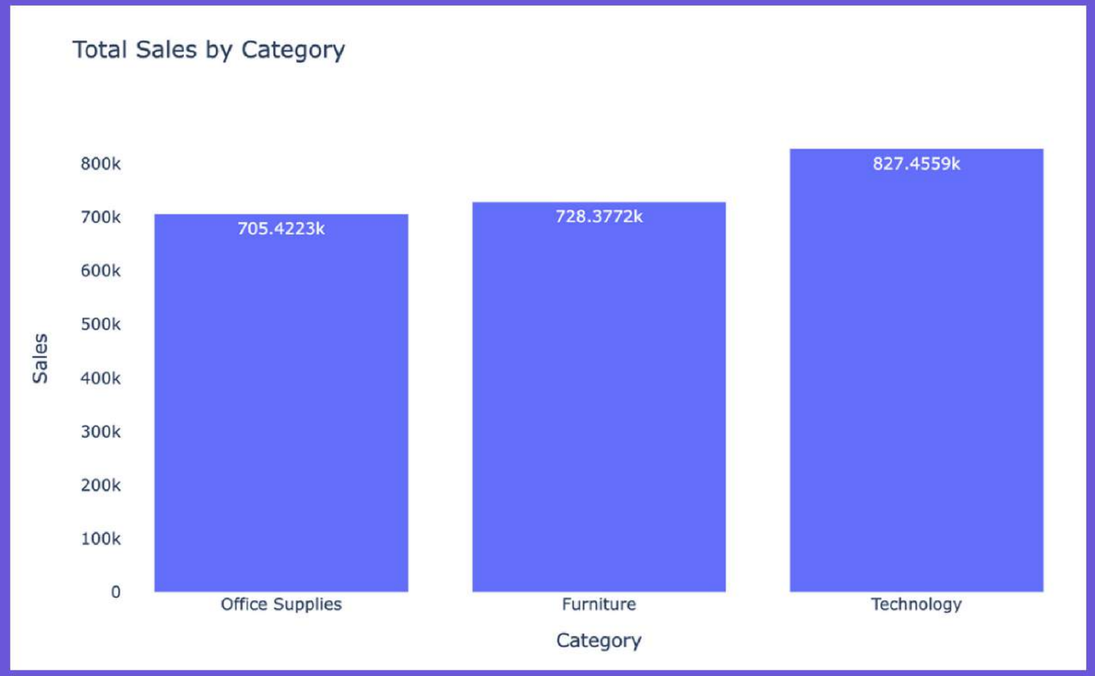


DigitalSkola

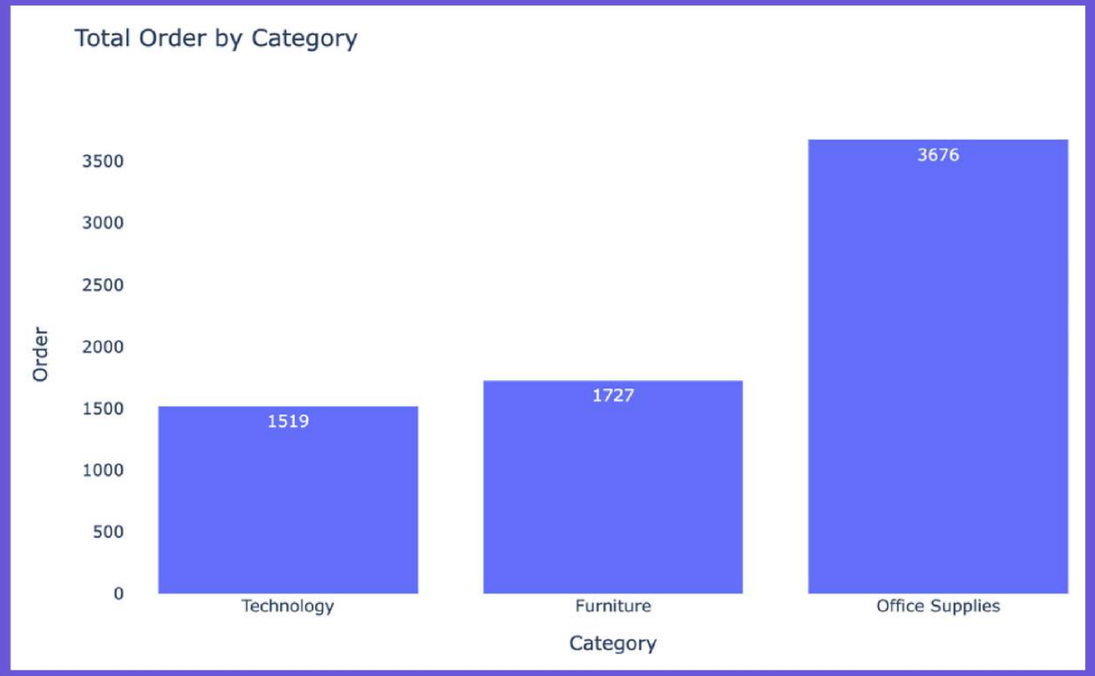
DATA UNDERSTANDING (EDA)

Technology is the biggest sales contribution while Office Supplies has the biggest transaction.

36% sales coming from Technology Category



53% orders included Office Supplies in the transaction.

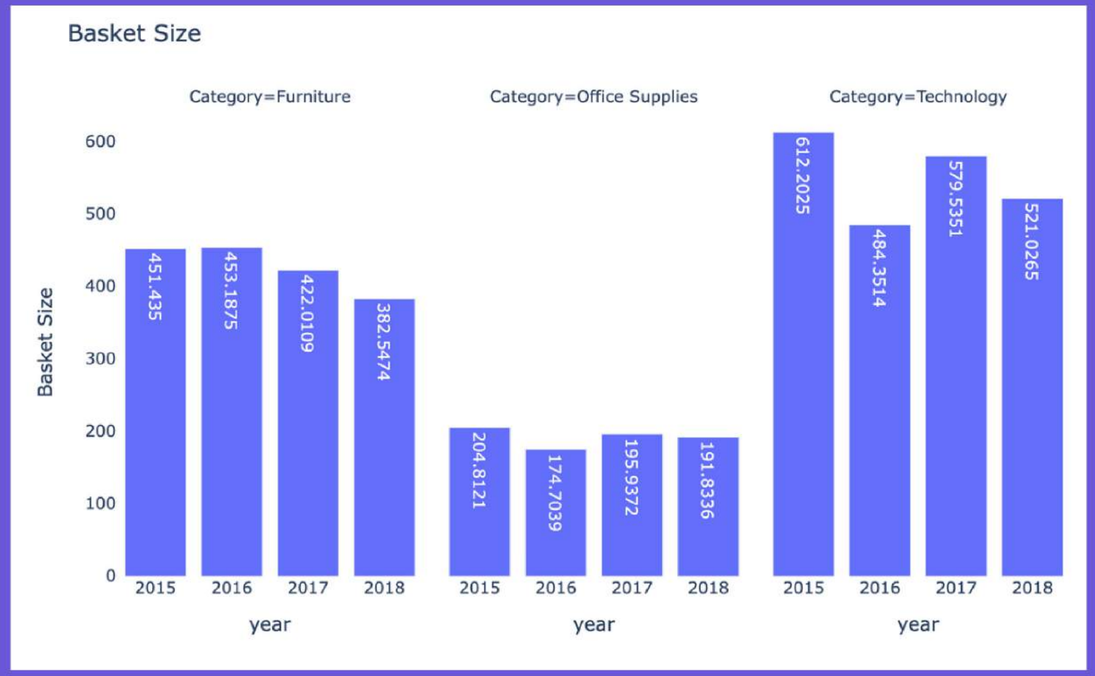


DigitalSkola

Transaction is growing across category, however the basket size doesn't follow.

Transaction by Category is also growing each year.

It's fluctuate in Technology but tend to stagnate in Office Supplies.



DigitalSkola

Analytical Approaches

We Use Two Analytical Approach:

1. K-Means Clustering with Initial Dataset
2. K-Means Clustering with RFM

Data Preparation (Approach 1)

DigitalSkola

Feature Selection

```
df_fe.head()
```

Visualize

	ship_mode object	segment object	region object	category object	sales float64
0	Second Class	Consumer	South	Furniture	261.96
1	Second Class	Consumer	South	Furniture	731.94
2	Second Class	Corporate	West	Office Supplies	14.62
3	Standard Class	Consumer	South	Furniture	957.5775
4	Standard Class	Consumer	South	Office Supplies	22.368

5 rows, showing 10 per page Page 1 of 1

Drop unnecessary columns

[73]

```
df_fe.drop(columns=['order_id', 'order_date', 'ship_date', 'customer_id', 'product_id', \
'sub_category', 'city', 'state', 'customer_name', 'country',  'product_name'], inplace=True)
```

DigitalSkola

Feature Encoding

One Hot Encoding for Region, Category and Segment.
Label Encoding for Ship_Mode

Visualize

	ship_mode i...	sales float64	region_Central ui...	region_East uint8	region_South uint8	region_West uint8	category_Furnit...
0	2	261.96	0	0	1	0	
1	2	731.94	0	0	1	0	
2	2	14.62	0	0	0	1	
3	3	957.5775	0	0	1	0	
4	3	22.368	0	0	1	0	

5 rows, showing 10 per page Page 1 of 1

```
df_encode.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9799 entries, 0 to 9799
Data columns (total 12 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   ship_mode                 9799 non-null   int64
 1   sales                     9799 non-null   float64
 2   region_Central            9799 non-null   uint8
 3   region_East               9799 non-null   uint8
 4   region_South              9799 non-null   uint8
 5   region_West               9799 non-null   uint8
 6   category_Furniture        9799 non-null   uint8
 7   category_Office Supplies  9799 non-null   uint8
 8   category_Technology       9799 non-null   uint8
 9   segment_Consumer          9799 non-null   uint8
 10  segment_Corporate         9799 non-null   uint8
 11  segment_Home Office       9799 non-null   uint8
dtypes: float64(1), int64(1), uint8(10)
memory usage: 583.4 KB
```

DigitalSkola

Drop Outliers

```
df_encode2 = df_encode[(df_encode['sales'] <= upper) & (df_encode['sales'] >= lower)]
df_encode2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8654 entries, 0 to 9799
Data columns (total 12 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   ship_mode                 8654 non-null   int64
 1   sales                     8654 non-null   float64
 2   region_Central            8654 non-null   uint8
 3   region_East               8654 non-null   uint8
 4   region_South              8654 non-null   uint8
 5   region_West               8654 non-null   uint8
 6   category_Furniture        8654 non-null   uint8
 7   category_Office Supplies  8654 non-null   uint8
 8   category_Technology       8654 non-null   uint8
 9   segment_Consumer          8654 non-null   uint8
 10  segment_Corporate         8654 non-null   uint8
 11  segment_Home Office       8654 non-null   uint8
dtypes: float64(1), int64(1), uint8(10)
memory usage: 287.3 KB
```

DigitalSkola

Scaling

X_mms

[123]

Visualize

	ship_mode float64 0.0 - 1.0	sales float64 0.0 - 1.000000000...	region_Central fl... 0.0 - 1.0	region_East float... 0.0 - 1.0	region_South flo... 0.0 - 1.0	region_West floa... 0.0 - 1.0	category_F 0.0 - 1.0
0	0.6666666666666666	0.5232454841575362	0.0	0.0	1.0	0.0	
1	0.6666666666666666	0.028363572337513705	0.0	0.0	0.0	1.0	
2	1.0	0.04386589728609272	0.0	0.0	1.0	0.0	
3	1.0	0.09687152358162131	0.0	0.0	0.0	1.0	
4	1.0	0.013677580452824753	0.0	0.0	0.0	1.0	
5	1.0	0.03613474297513385	0.0	0.0	0.0	1.0	
6	1.0	0.22900543421716063	0.0	0.0	0.0	1.0	
7	1.0	0.030228333159929253	0.0	0.0	1.0	0.0	
8	1.0	0.8153966818461933	0.0	0.0	0.0	1.0	
9	1.0	0.1367878094262459	1.0	0.0	0.0	0.0	

8654 rows, showing 10 per page

Page 1 of 866

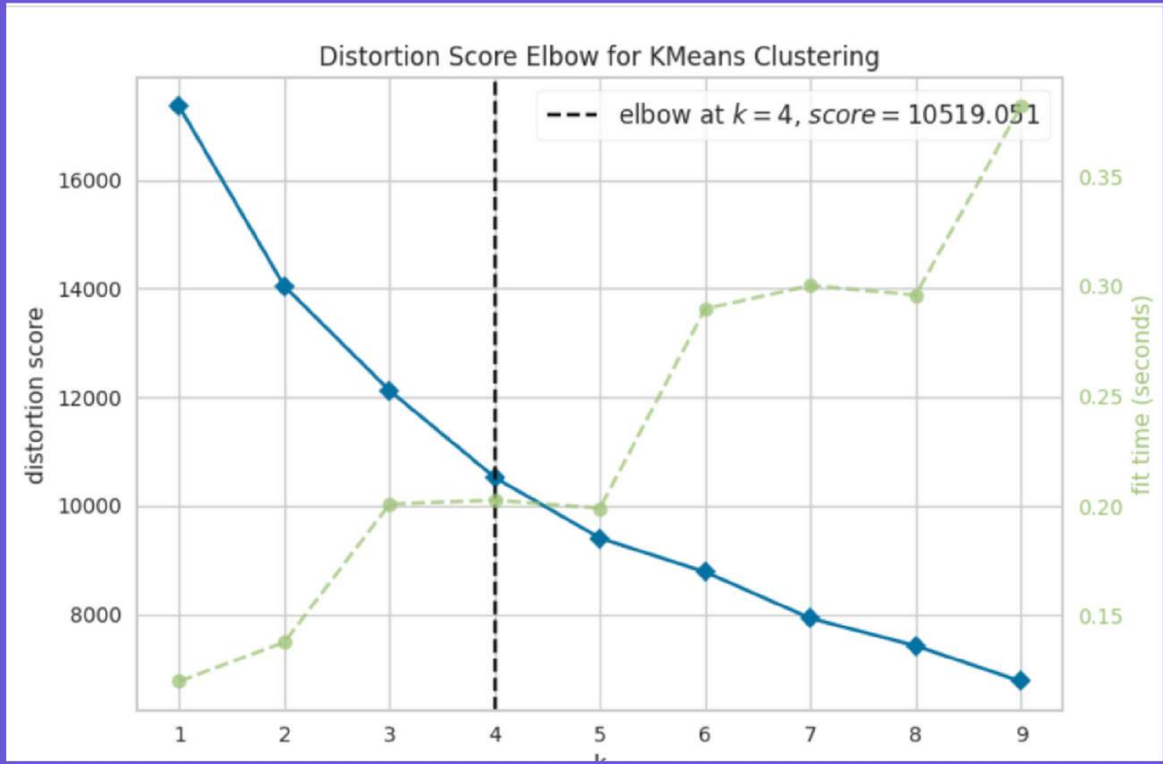
Scaling the data using MinMaxScaler

Modelling (Approach 1)

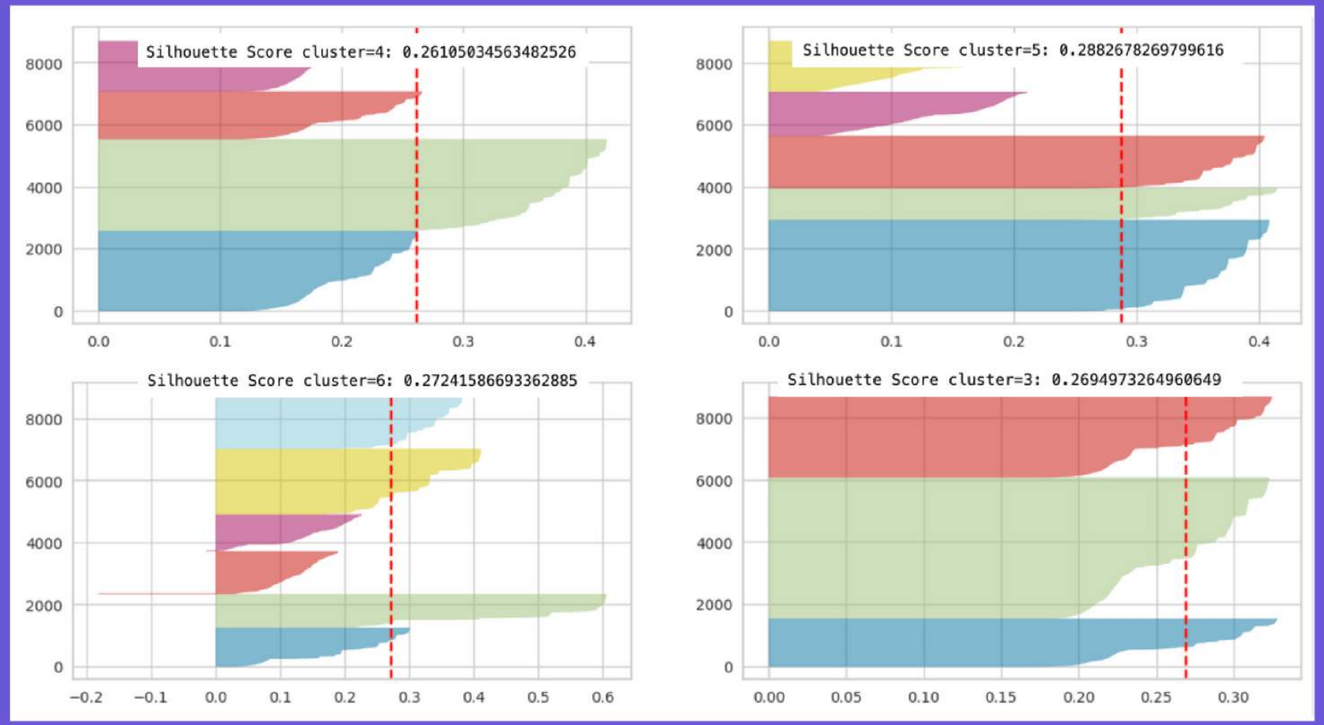
K-Means Modelling

We choose to have 5 clusters as evaluated in Silhouette Score

Optimum Elbow k = 4



Highest Silhouette Score = cluster 5

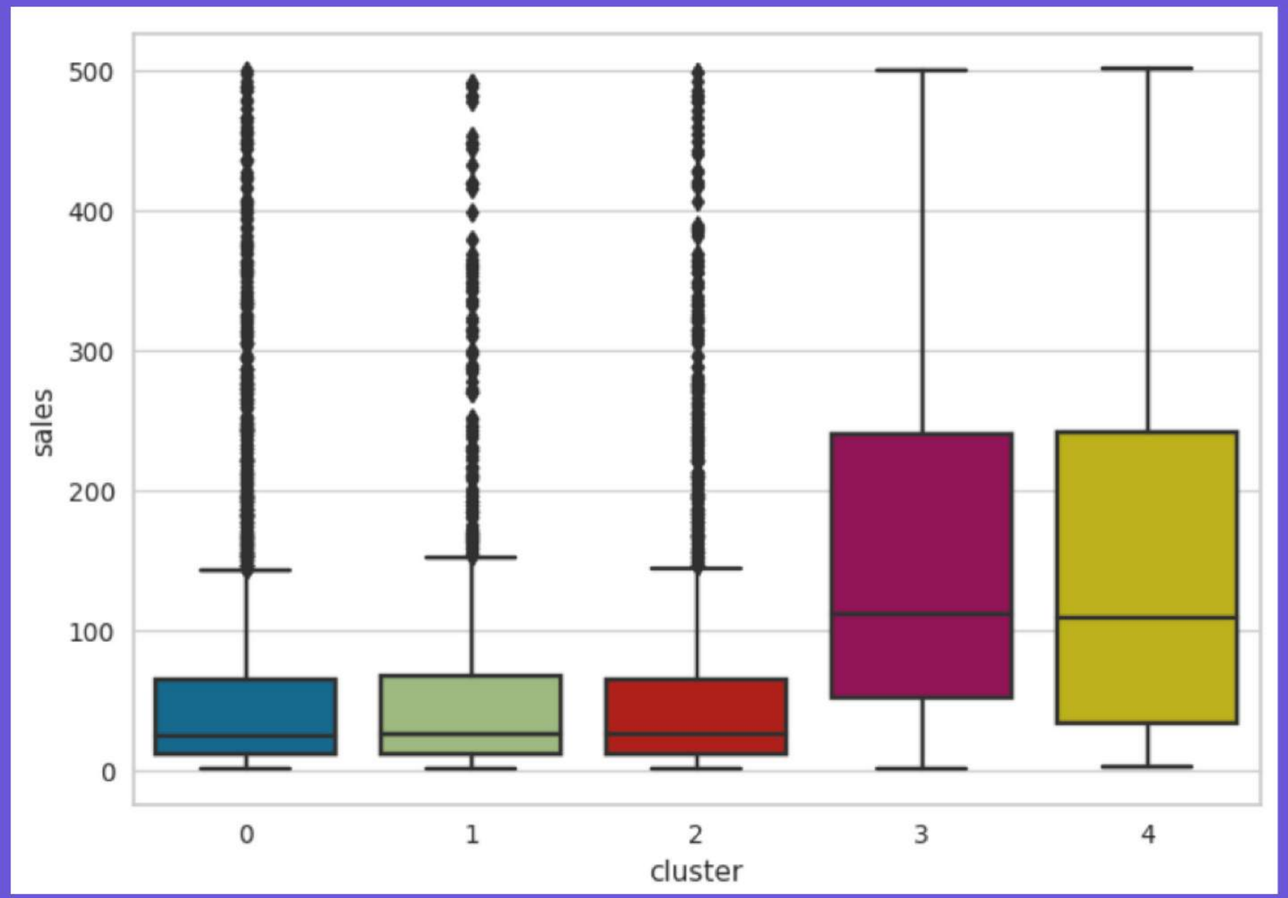


DigitalSkola

Interpretation

Customer Segmentation Method 1

Sales

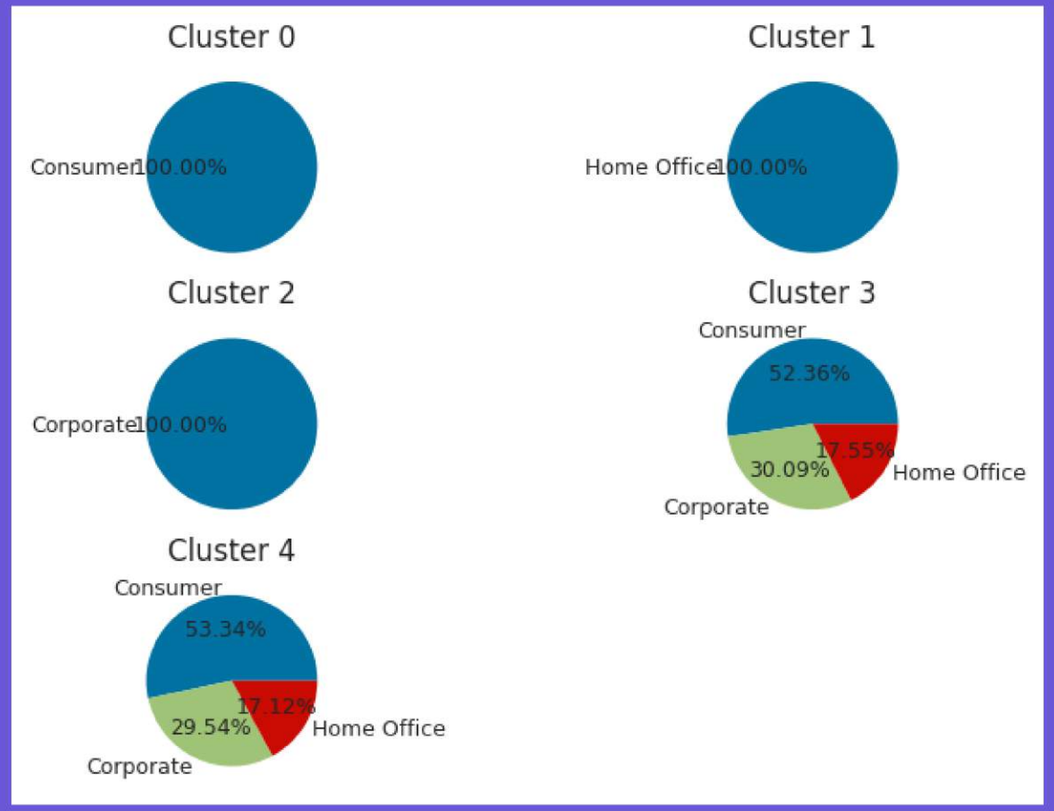


DigitalSkola

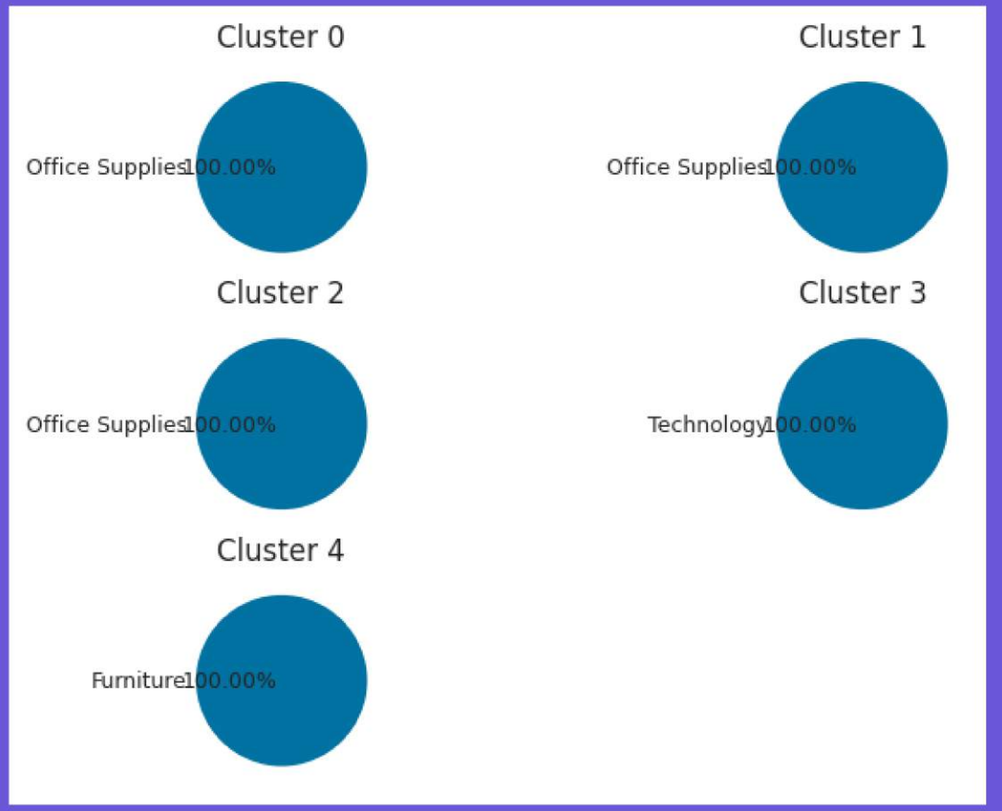
Interpretation

Customer Segmentation Method 1

Segment



Category

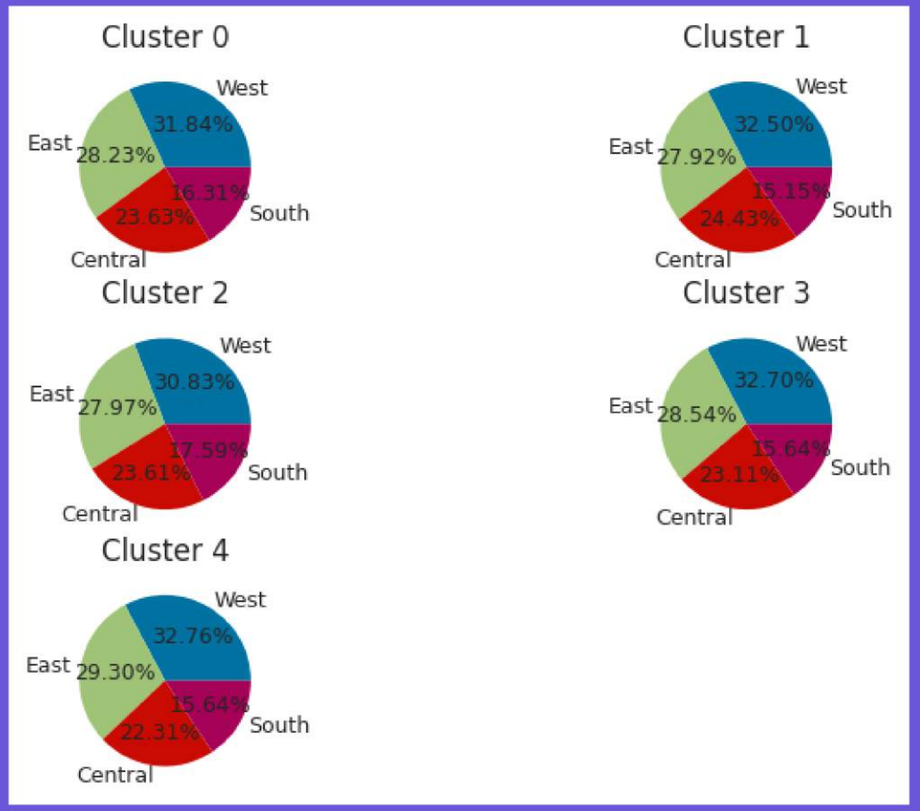


DigitalSkola

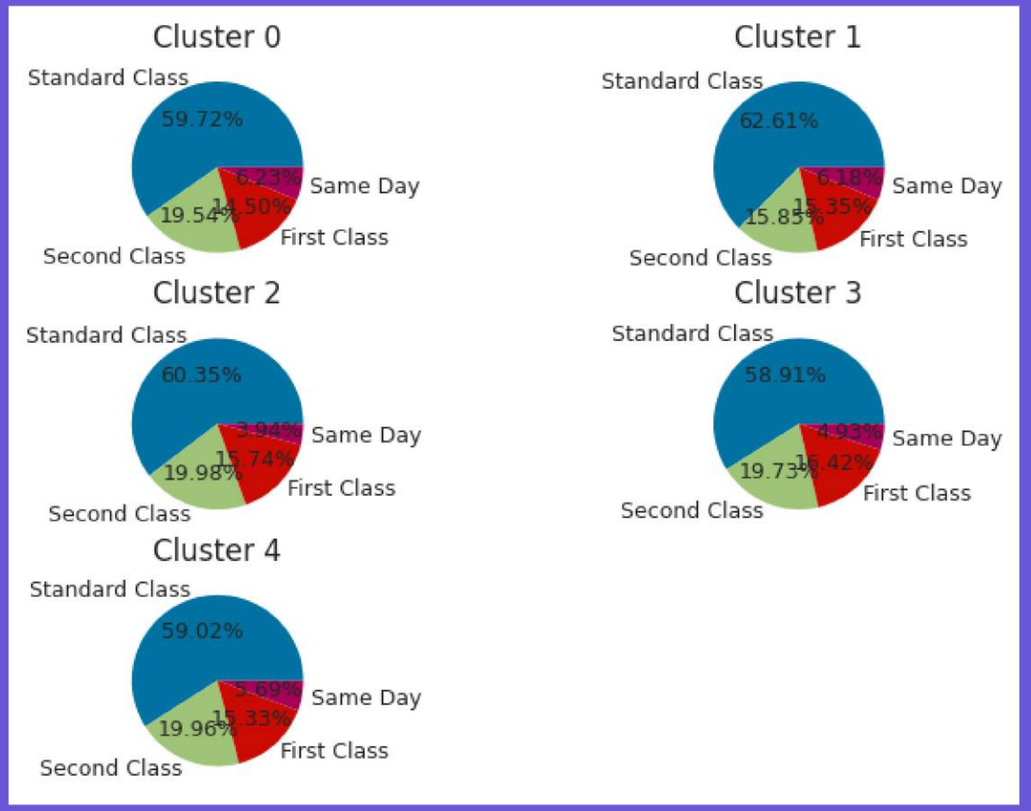
Interpretation

Customer Segmentation Method 1

Region



Ship Mode



DigitalSkola

Interpretation

Customer Segmentation Method 1

Cluster 0 - Consumer Segment, buying Office Supplies with low purchase power

Cluster 1 - Home Office Segment, buying Office Supplies with low purchase power

Cluster 2 - Corporate Segment, buying Office Supplies with low purchase power

Cluster 3 - Mixed Segment, buying Technology with high purchase power

Cluster 4 - Mixed Segment, buying Furniture with high purchase power

Customer Segmentation Method 1

Recommendation:
- Promoting other Personal Office Supplies product at low prices in the market or with discounts
- Shipping Discount

Recommendation:
- Promoting other Office Supplies product which have quite small dimensions and packages that have a discounted price instead of buying individually. So are shipping discounts

Recommendation:
- Promoting other Office Supplies product by applying wholesale prices to this cluster
- Shipping Discount

Recommendation:
- Promoting new Technology product
- Shipping Discount

Recommendation:
- Promoting new Furniture product
- Shipping Discount

RFM
(Approach 2)

DigitalSkola

Create RFM DataFrame

```
df_customers.head()
```

[64]

Visualize

	customer_id obj...	recency i...	frequency i...	monetary f...	r cat...	f cat...	m cat...	rfm obj...	rfm_score int...
0	CG-12520	338	3	1148.7800000000002	1	5	4	154	10
1	DV-13045	48	5	1119.4830000000002	4	4	4	444	12
2	SO-20335	112	6	2602.575500000001	3	3	3	333	9
3	BH-11710	171	8	6255.351000000001	2	2	1	221	5
4	AA-10480	259	4	1790.512	2	5	3	253	10

Drop unnecessary columns

```
rfm = pd.read_csv('customer_rfm.csv')
rfm.drop(['Unnamed: 0','recency_date','r','f','m','rfm','rfm_score','customer_id'],axis=1,inplace=True)
rfm.head()
```

Visualize

DigitalSkola

Features We Choose for K-Means Clustering

```
rfm.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 793 entries, 0 to 792
Data columns (total 3 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   recency    793 non-null    int64  
 1   frequency  793 non-null    int64  
 2   monetary   793 non-null    float64
dtypes: float64(1), int64(2)
memory usage: 18.7 KB
```

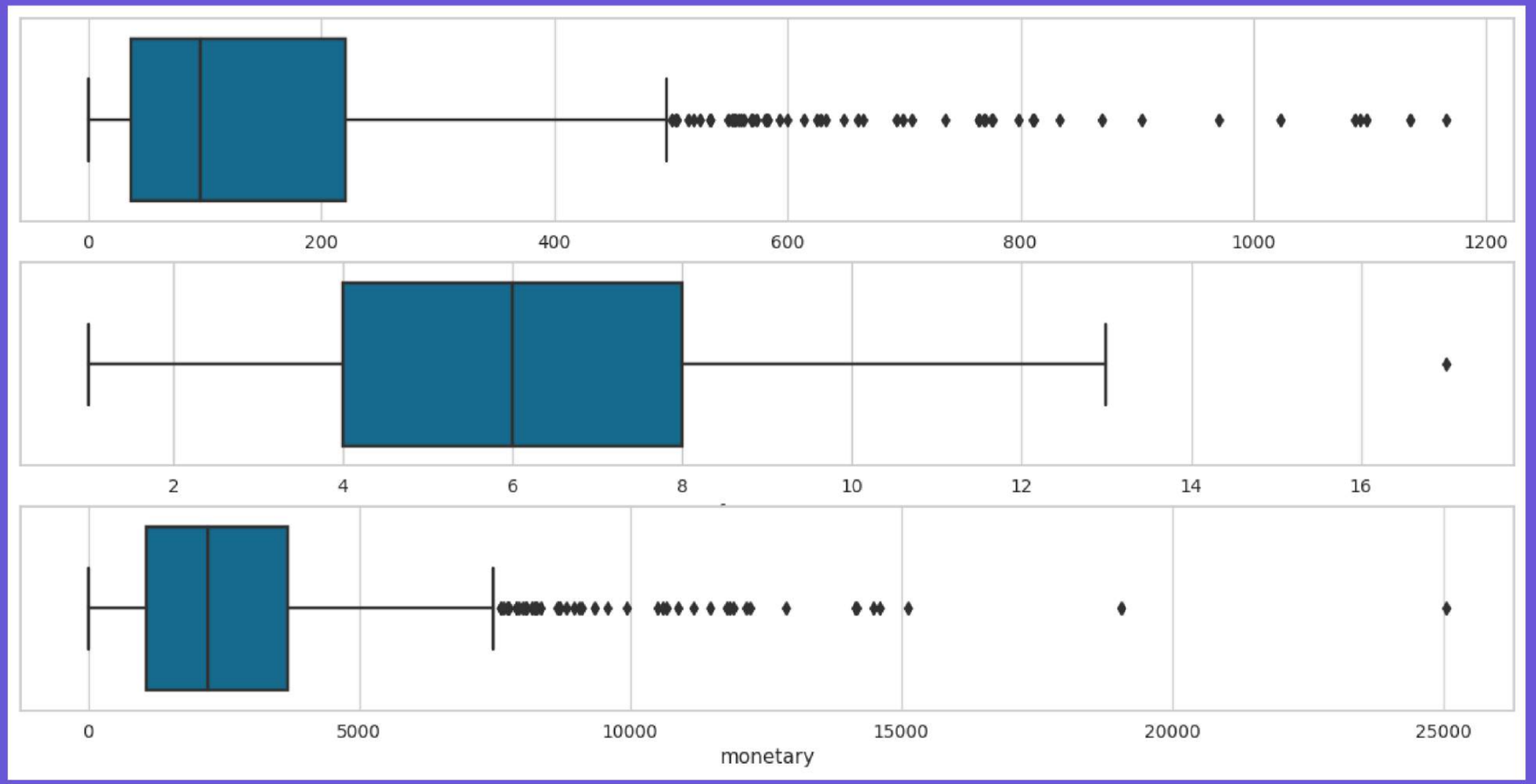
DigitalSkola

Check Outliers

Recency

Frequency

Monetary



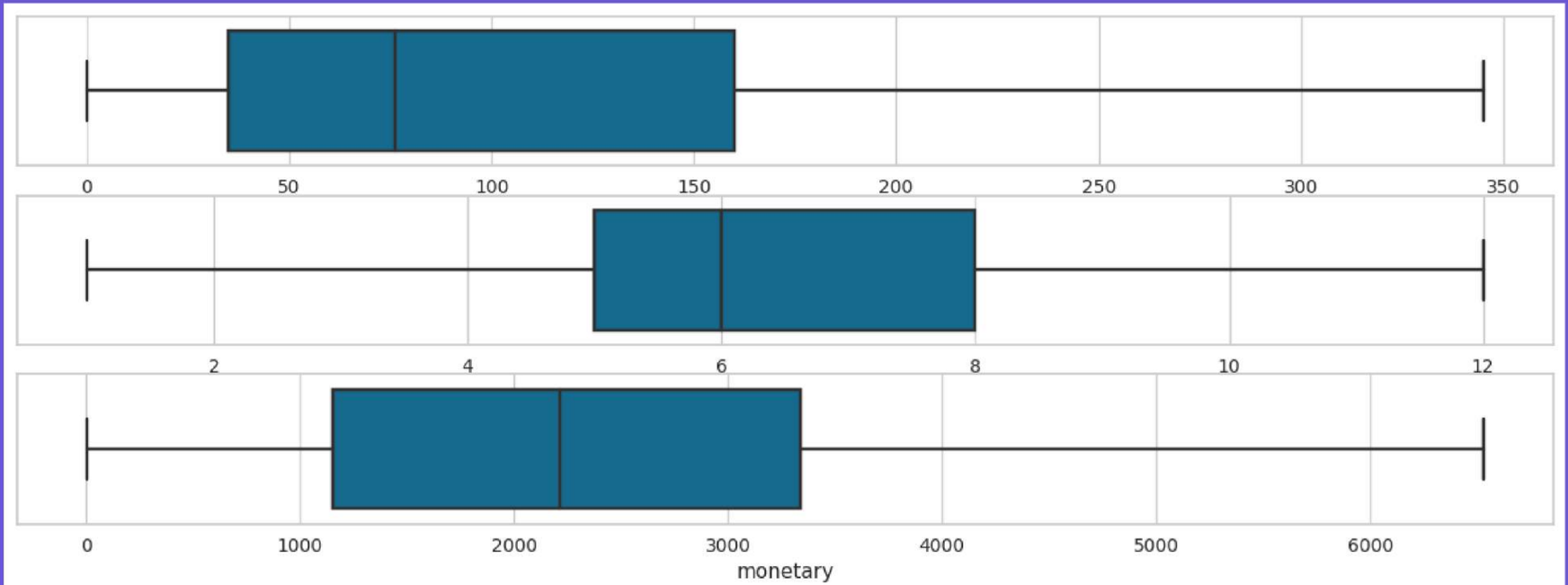
DigitalSkola

Drop Outliers

Recency

Frequency

Monetary



We Drop Outliers for 5 times to get data full without outliers

793 Instances → 627 Instances

DigitalSkola

Scaling

```
rfm_mms.head()
```

	recency float64	frequency float64	monetary float64
0	0.9797101449275363	0.1818181818181818	0.17536589781611817
1	0.1391304347826087	0.3636363636363636	0.17087469786689077
2	0.3246376811594027	0.4545454545454545	0.3982312518041372
3	0.4956521739130435	0.6363636363636364	0.9581979767295226
4	0.7507246376811594	0.2727272727272727	0.2737427529827764

5 rows, showing 10 per page Page 1

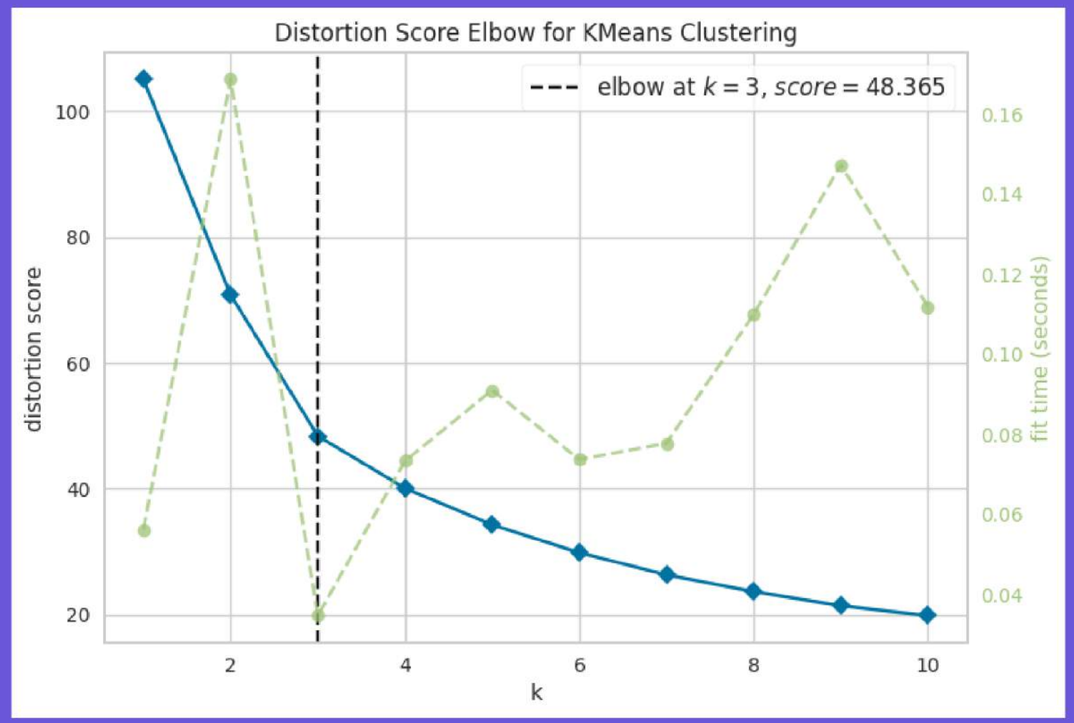
Scaling the data using MinMaxScaler

RFM Clustering (Approach 2)

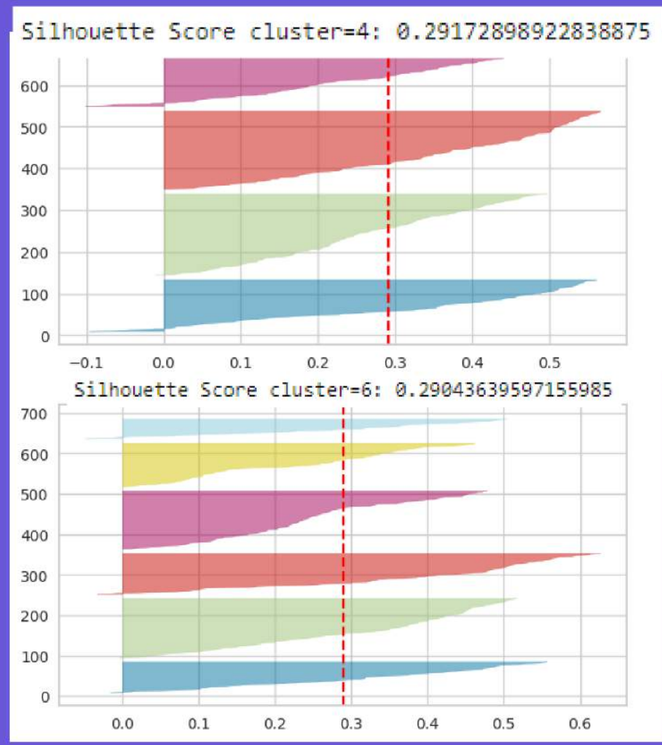
K-Means Modelling

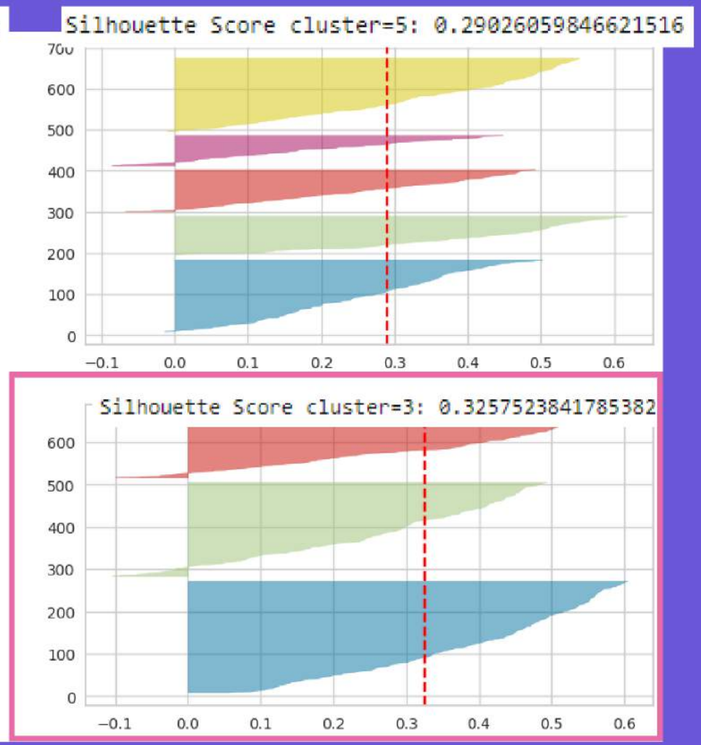
We choose to have 3 clusters as evaluated in Silhouette Score

Optimum Elbow k = 3



Highest Silhouette Score = cluster 3

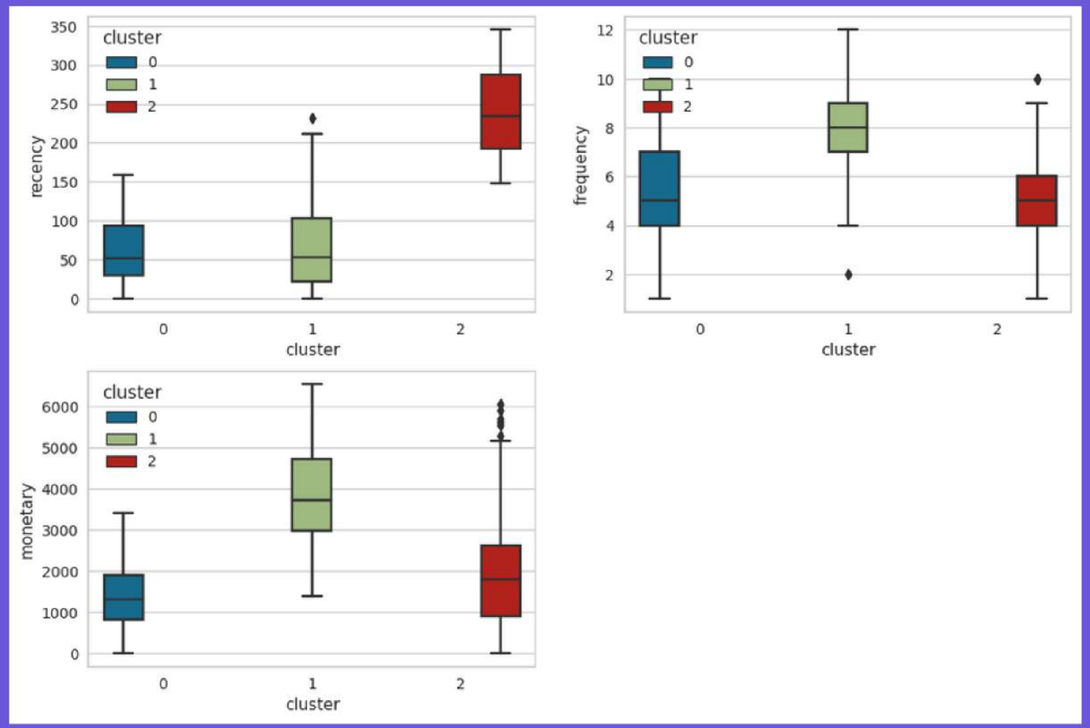


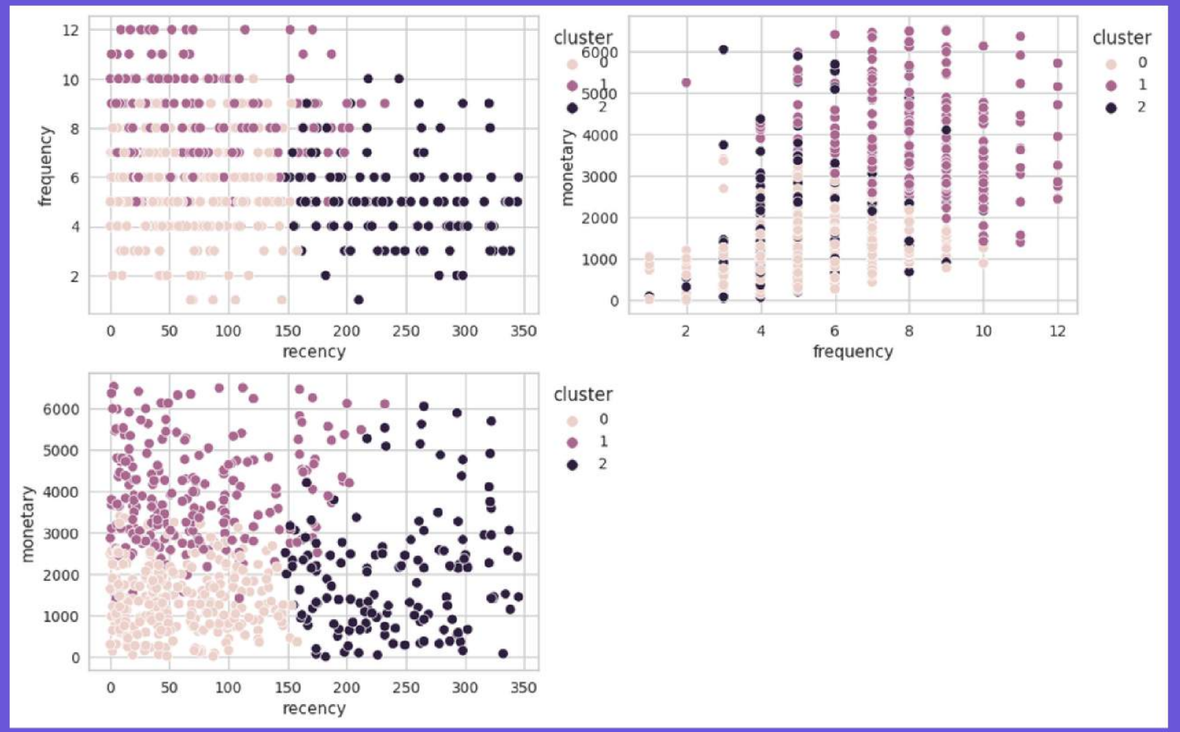


DigitalSkola

Interpretation

Customer Segmentation Method 2





DigitalSkola

Interpretation

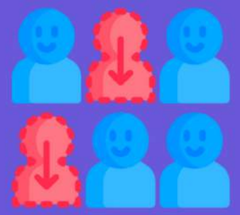
Customer Segmentation Method 2

Cluster 0 - Developing Customer

Characteristic : Low Monetary, New Recency, Low Frequency

Cluster 1 - Loyal Customer

Characteristic : High Monetary, New Recency, High Frequency



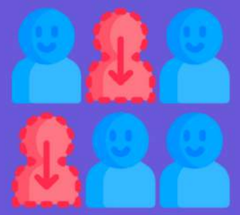
Cluster 2 - Churn Customer

Characteristic : Moderate Monetary, Old Recency, Moderate Frequency

Customer Segmentation Method 2

Recommendation :
- Giving discount after second/third purchase
- Giving free shipping
- Promote high value product to improve basket size

Recommendation :
- Develop Loyalty Program / Membership Program
- Promote new product development



Recommendation :
- Reactivation coupon for top product (to get them immediately purchase)
- Giving free shipping
- Improve products quality and service

Conclusion

Conclusion

1. Sales grow from the last 3 years, in the same time, our transaction across categories and segments also growing. However, our sales growth rate haven't catch up the growth rate of transaction hence our basket size is declining.
2. Cluster segmented from approach 1 is 5 clusters.
3. Cluster segmented from approach 2 is 3 clusters.

Our Team

MUHAMMAD FADHLI

ALWANDIA RIDWAN W

SALOMO HENDRIAN S

YOEAN OCTARHAIEZKY P

Thank you!

Provided By: Outliers Team