Customer Personality Analysis

Thi Phong - Quoc Trung

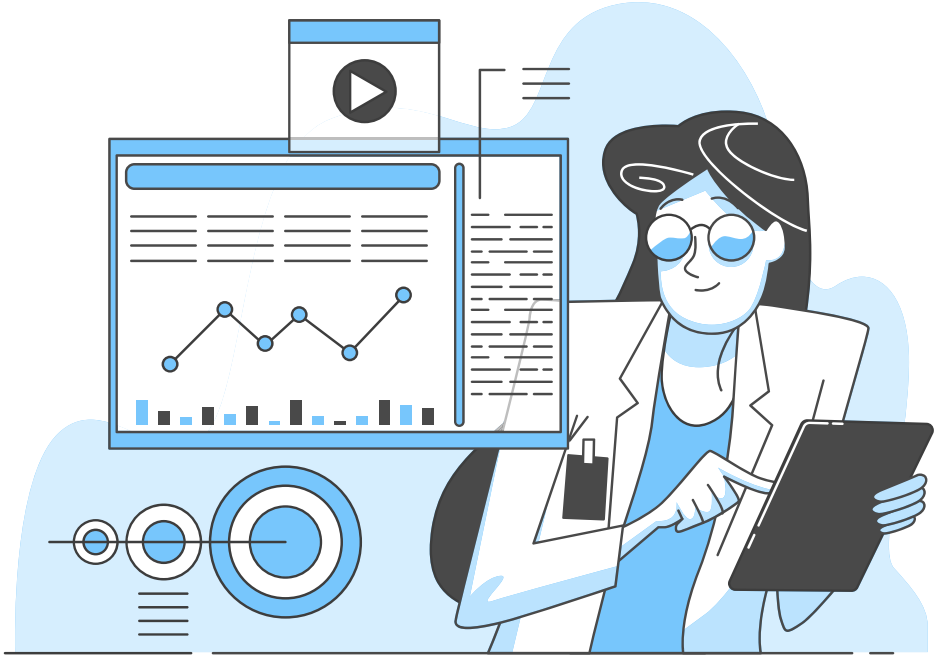
Table of Contents

01 Data Introduction

02 Data Handling & EDA

03 RFM Model

04 Propose Stategy



01

Data Introduction

Data source:

https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis

Tools:

- Pandas
- Matplotlib
- Power BI
- Figma

Target:

- Base on customer's behaviors and characters to **divide customers into group**.
- Find company's **ideal customers**.
- Helps a business to **better understand** its customers.
- Provide **marketing action** according to customers groups.

Data shape: 29 columns x 2240 rows

People

1. ID
2. Year_Birth
3. Education
4. Marital_Status
5. Income
6. Kidhome
7. Teenhome
8. Dt_Customer
9. Recency
10. Complain

Product

1. MntWines
2. MntFruits
3. MntMeatProducts
4. MntFishProducts
5. MntSweetProducts
6. MntGoldProds

Promotion

1. NumDealsPurchases
2. AcceptedCmp1
3. AcceptedCmp2
4. AcceptedCmp3
5. AcceptedCmp4
6. AcceptedCmp5
7. Response

Place

1. NumWebPurchases
2. NumCatalogPurchases
3. NumStorePurchases
4. NumWebVisitsMonth

02

Data Handling & EDA

Check null and Duplicate:

- No duplicate
- Have null values in Income column

```
# no duplicated
mc.duplicated().sum()
```

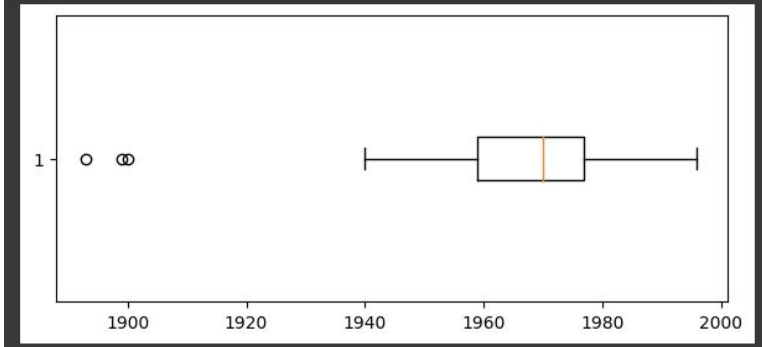
```
0
```

```
# 'Income' has null values
mc.isna().sum()
```

```
ID                     0
Year_Birth             0
Education              0
Marital_Status         0
Income                24
Kidhome                0
Teenhome               0
Dt_Customer            0
Recency                0
MntWines               0
MntFruits              0
MntMeatProducts        0
MntFishProducts        0
MntSweetProducts       0
MntGoldProds           0
NumDealsPurchases      0
NumWebPurchases        0
NumCatalogPurchases    0
NumStorePurchases      0
NumWebVisitsMonth      0
AcceptedCmp3           0
AcceptedCmp4           0
AcceptedCmp5           0
AcceptedCmp1           0
AcceptedCmp2           0
Complain               0
Z_CostContact          0
Z_Revenue              0
Response               0
dtype: int64
```

Year_Birth	- Outliers - Age groups
Education	
Marital_Status	
Income	
Kidhome & Teenhome	
Dt_Customer	
ID & Recency	

```python
plt.figure(figsize=(7, 3))
plt.boxplot(mc['Year_Birth'], vert=False)

plt.show()
```




```python
# considering that the data were incorrectly fill -- change to the median data
mc.loc[mc['Year_Birth'] <= 1900, 'Year_Birth'] = int(mc['Year_Birth'].median())
```

```python
# create birth group

def Birth_group(x):
  if x >= 45:
    return 'Senior'
  elif x >= 30:
    return 'Adult'
  elif x >= 18:
    return 'Young-adult'
  else:
    return 'Children'
```

```python
mc['Age group'] = (2014 - mc['Year_Birth']).apply(Birth_group)
```

Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	
Income	
Kidhome & Teenhome	
Dt_Customer	
ID & Recency	

```python
# count 'Education' values
mc['Education'].value_counts()
```

```
Graduation    1127
PhD            486
Master         370
2n Cycle       203
Basic           54
Name: Education, dtype: int64
```

```python
# Resize Education by 2 groups: Undergraduate and Postgraduate
edu_clean = {
    'Graduation': 'Postgraduate'
    , 'PhD'     : 'Postgraduate'
    , 'Master'  : 'Postgraduate'
    , 'Basic'   : 'Undergraduate'
    , '2n Cycle': 'Undergraduate'
}

mc['Education_clean'] = mc['Education'].replace(edu_clean)
```

Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	Re-groups
Income	
Kidhome & Teenhome	
Dt_Customer	
ID & Recency	

```python
# checking to re-group
mc['Marital_Status'].value_counts()
```

```
Married     864
Together    580
Single      480
Divorced    232
Widow        77
Alone         3
Absurd        2
YOLO          2
Name: Marital_Status, dtype: int64
```

```python
# Resize Marital by 2 groups: Couple and Single
marital_clean = {
      'Together'  : 'Couple'
    , 'Married'   : 'Couple'
    , 'Divorced'  : 'Single'
    , 'Widow'     : 'Single'
    , 'Alone'     : 'Single'
    , 'Absurd'    : 'Single'
    , 'YOLO'      : 'Single'
}

mc['Marital_clean'] = mc['Marital_Status'].replace(marital_clean)
```

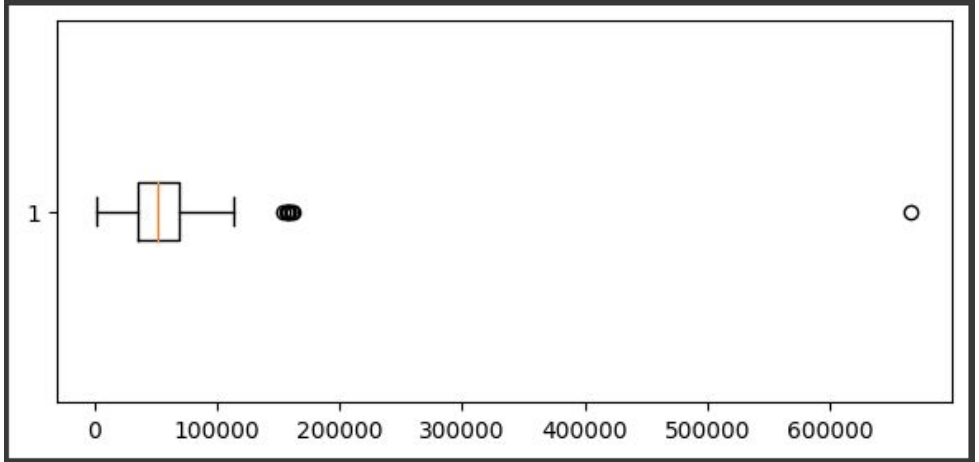
```python
mc['Marital_clean'].value_counts()
```

```
Couple    1443
Single     796
Name: Marital_clean, dtype: int64
```

Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	Re-groups
Income	- Outliers - Null values
Kidhome & Teenhome	
Dt_Customer	
ID & Recency	

```python
# column 'Income' has null values -- fill in values by the median of Income
mc = mc.fillna(mc['Income'].median())
```

```python
plt.figure(figsize=(7, 3))
plt.boxplot(mc['Income'], vert=False)

plt.show()
```




```python
# Drop the outliers in Income
mc = mc[mc['Income'] < 200000]
```

Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	Re-groups
Income	- Outliers - Null values
Kidhome & Teenhome	Sum up values
Dt_Customer	
ID & Recency	

```
# Children column
mc['Children'] = mc['Kidhome'] + mc['Teenhome']
```

Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	Re-groups
Income	- Outliers - Null values
Kidhome & Teenhome	Sum up values
Dt_Customer	Change datatype
ID & Recency	

```
mc['Dt_Customer'].info()

<class 'pandas.core.series.Series'>
RangeIndex: 2240 entries, 0 to 2239
Series name: Dt_Customer
Non-Null Count  Dtype
--------------  -----
2240 non-null   object
dtypes: object(1)
memory usage: 17.6+ KB
```

```
# date column 'Dt_Customer' is not in datatime datatype -- change datatype
mc['Dt_Customer'] = pd.to_datetime(mc['Dt_Customer'], dayfirst = True)
```

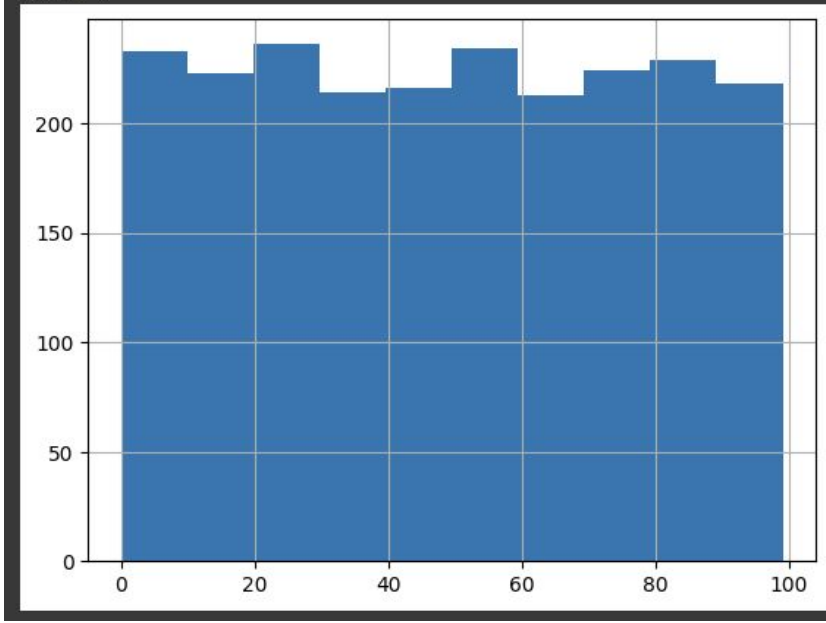
Year_Birth	- Outliers - Age groups
Education	Re-groups
Marital_Status	Re-groups
Income	- Outliers - Null values
Kidhome & Teenhome	Sum up values
Dt_Customer	Change datatype
ID & Recency	Clean

```
# Check duplicate in 'ID'
mc.duplicated('ID').sum()
```

```
0
```

```
mc['Recency'].hist()
```

```
<Axes: >
```



There are 2 columns **'Z_CostContact', 'Z_Revenue'** have only 1 values of 3 and 11. Do not have contribute to data meaning.

Solution is drop both columns.

```
mc[['Z_CostContact', 'Z_Revenue']].value_counts()

# there only 1 number in each group, there are not much values.
```

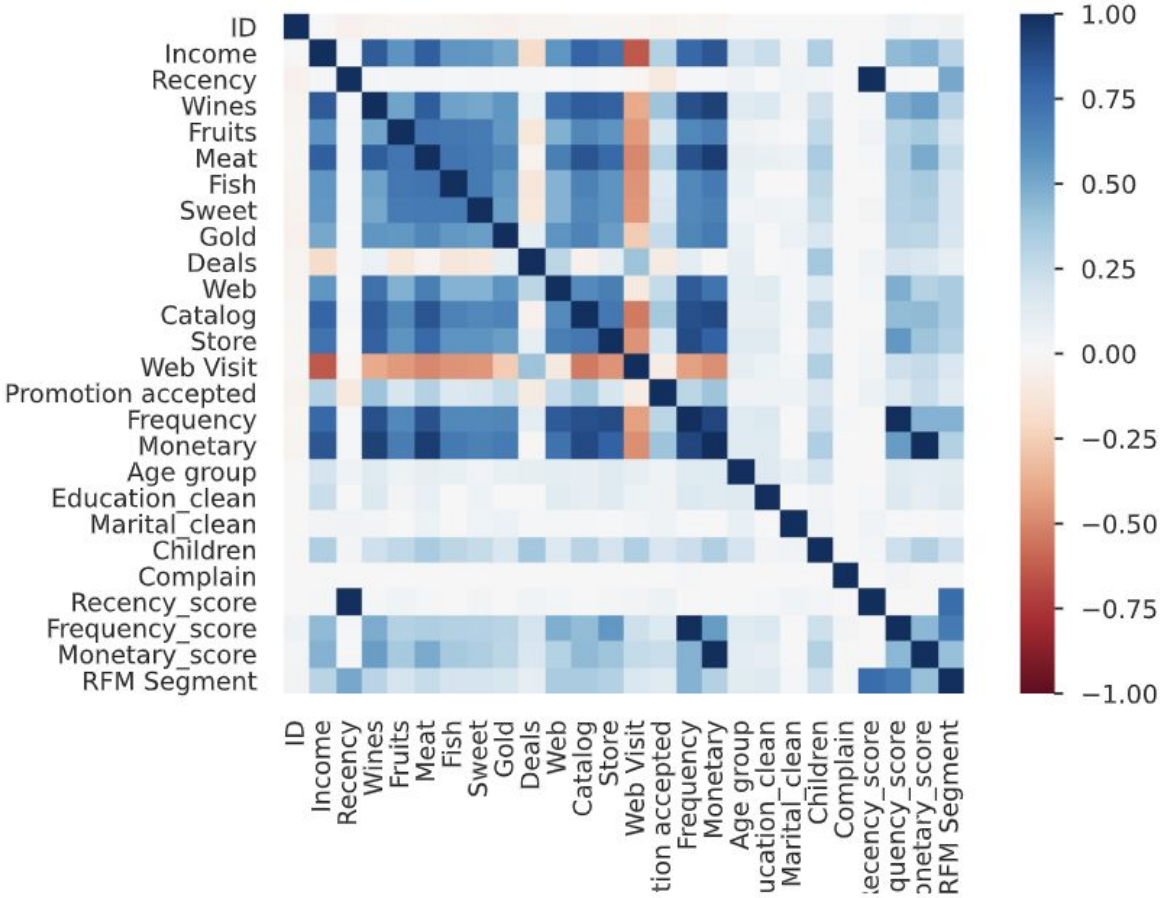
```
Z_CostContact  Z_Revenue
3              11          2240
dtype: int64
```

Drop other unnecessary columns and rearrange columns.

```
# Delete 2 columns 'Z_CostContact', 'Z_Revenue' and other unclean columns
mc = mc[['ID', 'Age group', 'Education_clean', 'Marital_clean', 'Children', 'Income', 'Dt_Customer', 'Recency',  # Customer anthropology
         'MntWines', 'MntFruits', 'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts', 'MntGoldProds',       # Products
         'NumDealsPurchases', 'NumWebPurchases', 'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth', # Purchase behaviors
         'Promotion accepted', 'Complain' # Responces
]]
```

Rename columns

```
# Change name columns

# rename columns for pbi
new_name = {  'MntWines'            : 'Wines'
            , 'MntFruits'           : 'Fruits'
            , 'MntMeatProducts'     : 'Meat'
            , 'MntFishProducts'     : 'Fish'
            , 'MntSweetProducts'    : 'Sweet'
            , 'MntGoldProds'        : 'Gold'
            , 'NumDealsPurchases'   : 'Deals'
            , 'NumWebPurchases'     : 'Web'
            , 'NumCatalogPurchases' : 'Catalog'
            , 'NumStorePurchases'   : 'Store'
            , 'NumWebVisitsMonth'   : 'Web Visit'
            , 'Dt_Customer'         : 'Enrollment date'}

mc = mc.rename(columns=new_name)

mc.columns
```



Positive interaction:

Income

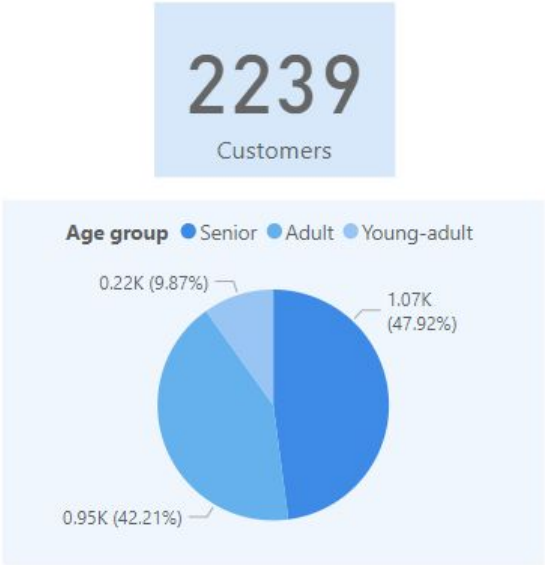
Products: Foods, Wines, Gold

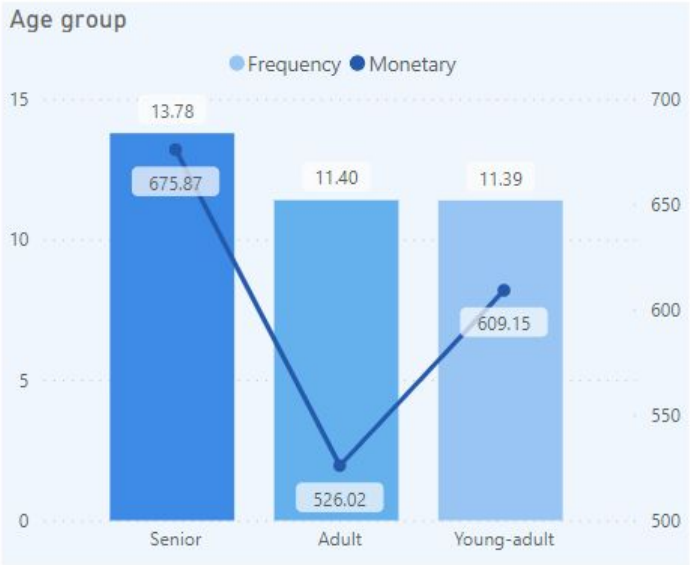
Purchase channels: Web, store, catalog

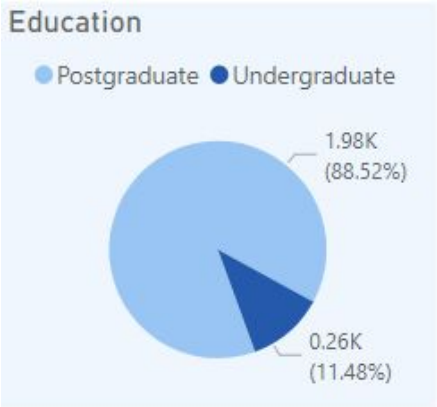
Negative interaction:

Web visit

Customer anthropologies







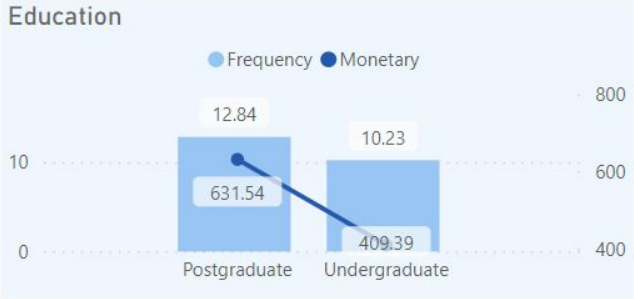
Most customer is Senior, following by Adult, more than 40%.

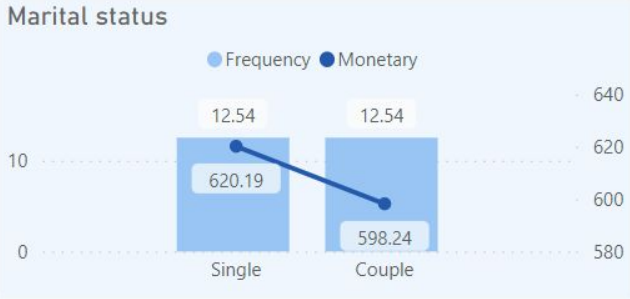
Senior also spend more and shopping more frequent.

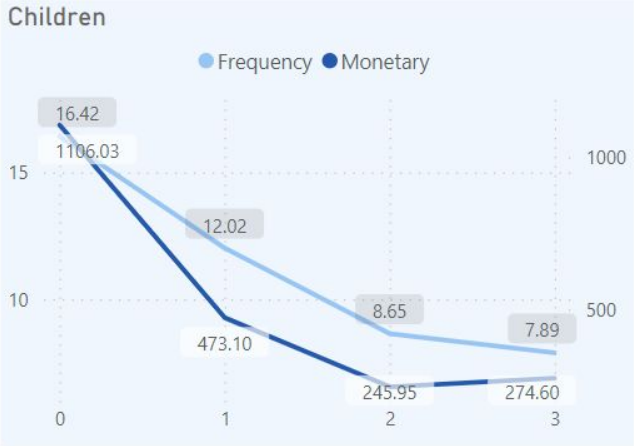
Most customer in group Postgraduate

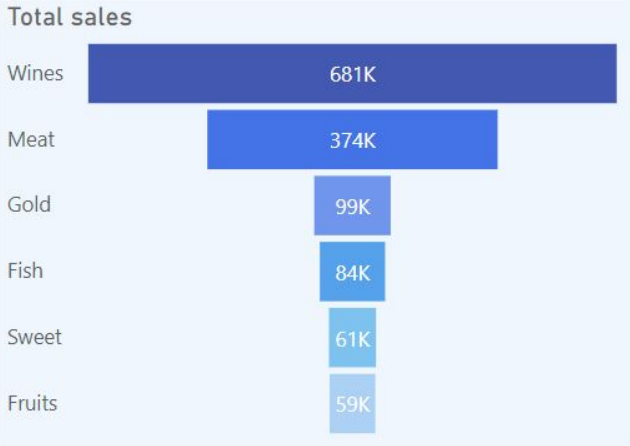
Customer behaviors

Postgraduate and Single spending more and shopping more.







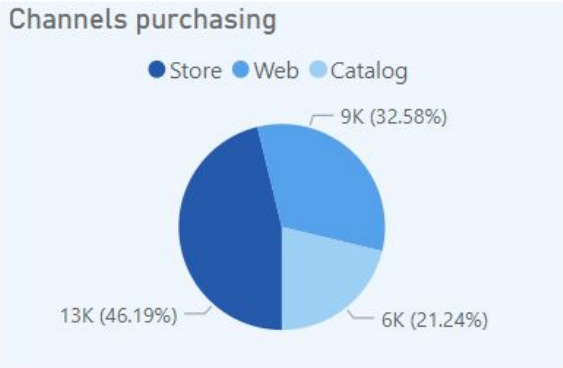


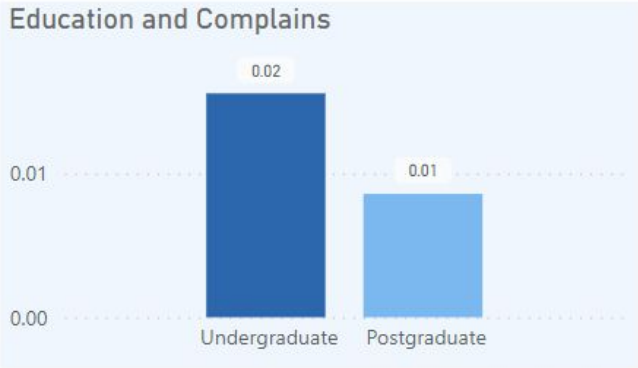
The more children customer have the less they shop and spend.

Wines and Meat spending are significant different than other products.

Customer behaviors

Customers mainly shopping at Store and Web.



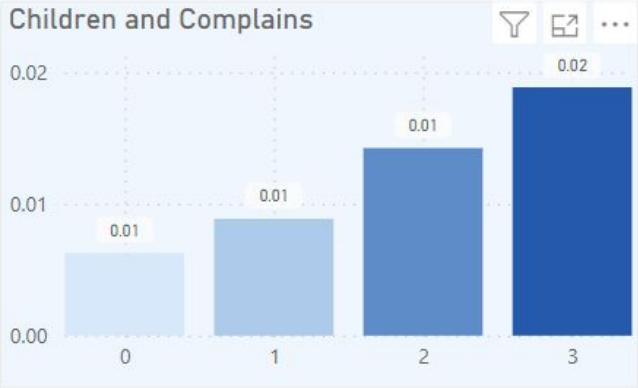


Undergraduate and customer with children complain more than others.

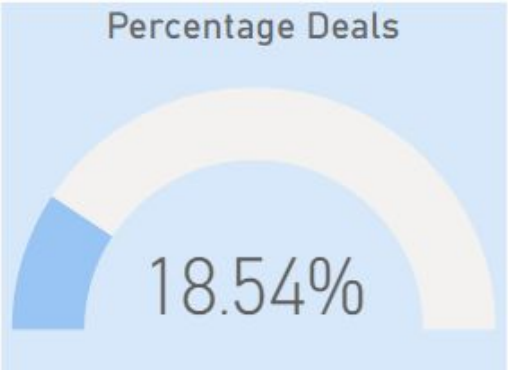
Complain ratio in total customer is low, not enough 1%.

21
Complains

0.94%
Percentage Complains



Promotion interaction



Percentage of purchase orders using discount only accounts for 18,5% of the total number of orders.

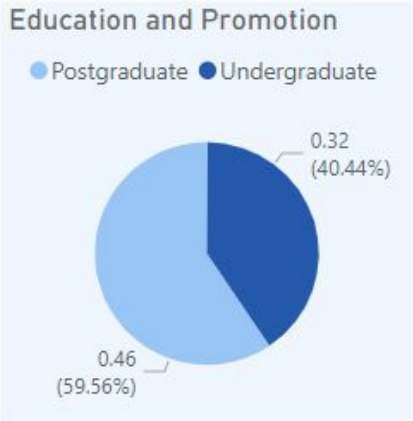
Income is inversely proportional to customer's promotion acceptance rate.

Promotion interaction

Single using Promotion more than Couple.

Young-adult using promotion more but spend less than Senior and Adult.

Postgraduate using promotion more than Undergraduate.



Customers with children pay less attention to about promotion.

Customer characteristics

Mostly over 30 years old. Over 45 buy more than other age groups.

Postgraduate, Single, customer with no kid spend more and use promotion more.

Customers spend a lot for Wines and Meat.

Purchases mostly make through Store and Web.

Customer not interact with promotion and service.

03

RFM model

RFM (**R**ecency, **F**requency, **M**onetary) is a marketing analysis technique used to evaluate customer behavior and segment customers based on three key metrics:

Recency

How recent the customer's last purchase

Frequency

Number of orders that customer has placed

Monetary

Amount of money customers has spent

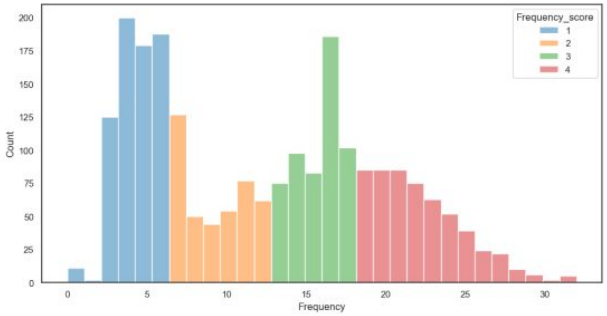
Turn RFM to score:

Apply pd.qcut() into 4 quantiles for Frequency, Recency and Monetary

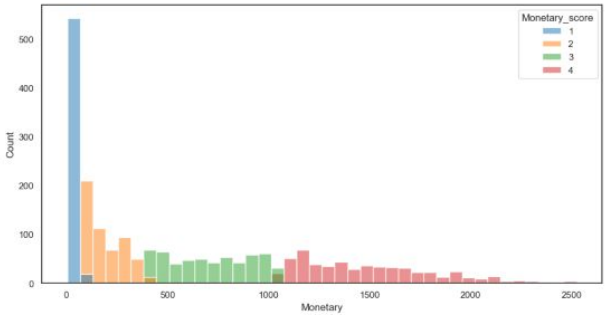
```python
# The lower the recency the better
rfm_data['Recency_score'] = pd.qcut(rfm_data['Recency'], q = 4, labels=[4,3,2,1])

# The higher the frequency and monetary the better
rfm_data['Frequency_score'] = pd.qcut(rfm_data['Frequency'], q = 4, labels=[1,2,3,4])
rfm_data['Monetary_score'] = pd.qcut(rfm_data['Monetary'], q = 4, labels=[1,2,3,4])
```

✓ 0.0s

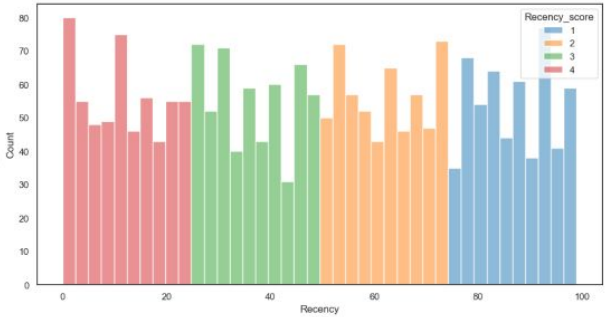
Frequency_score →



Monetary_score →



Recency_score →

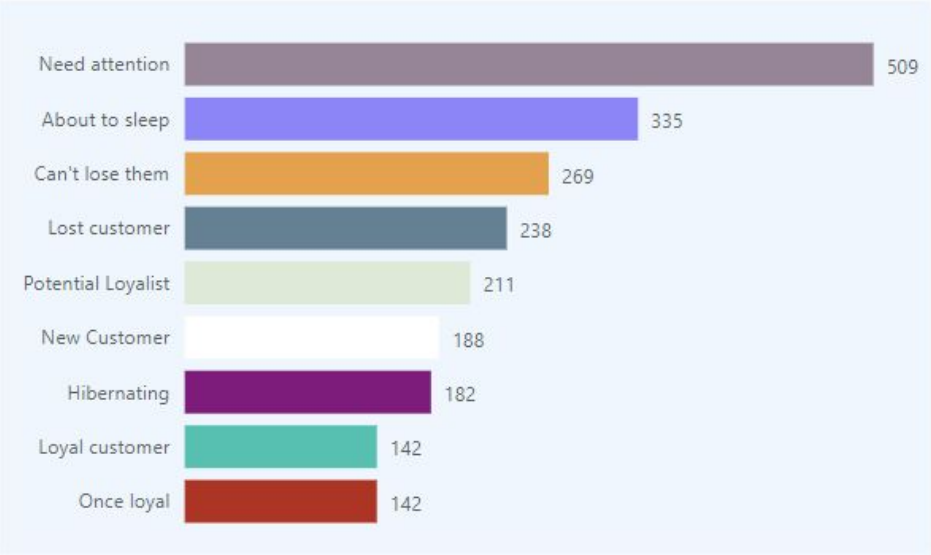


Recency-to-Frequency Matrix

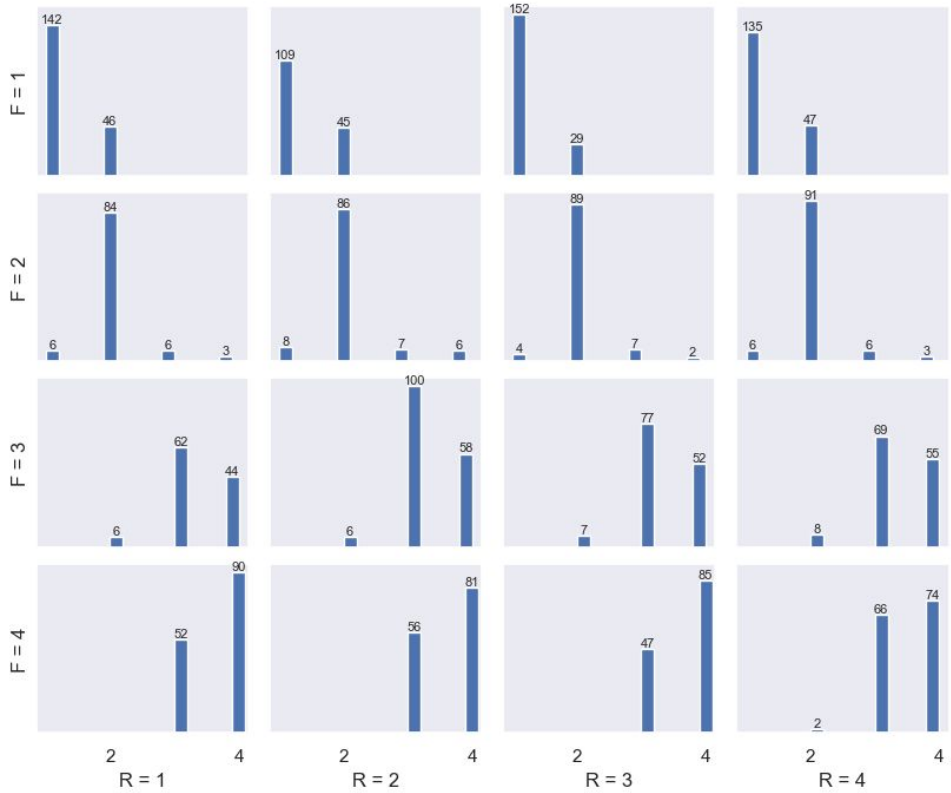
		Long time no see			Shopped recently
		R = 1	R = 2	R = 3	R = 4
Don't go much	F = 1	Try once then quit	Don't really need our service/ product	Don't really need our service/ product	New Customer
	F = 2	Lost customer	Only go when needed or already have alternatives	Only go when needed	We got what they need
	F = 3	Lost customer	Only go when needed or already have alternatives	Only go when needed	They like us
Go alot	F = 4	Lost loyal customer	Fix the problem quick or we lose them	Something made our favorite customer sad	Loyal customer

Recency-to-Frequency Customer Segments

	R = 1	R = [2,3]	R = 4
F = 1	Lost customer	About to sleep	New Customer
F = [2,3]	Hibernating	Need attention	Potential Loyalist
F = 4	Once loyal	Can't lose them	Loyal customer



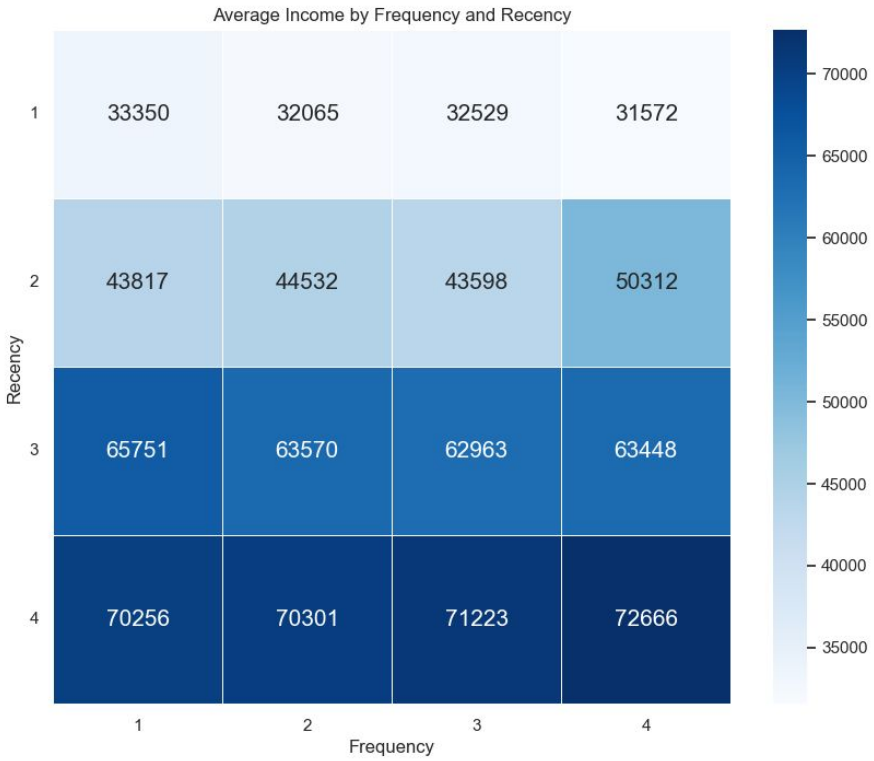
How Customer spend money each Segment?



Observation

There is not much different between Recency. A clear pattern that: customers with **higher Frequency**, meaning those who make more purchases, **achieve higher Monetary ranks** due to their accumulated spending.

How Customer spend money each Segment?

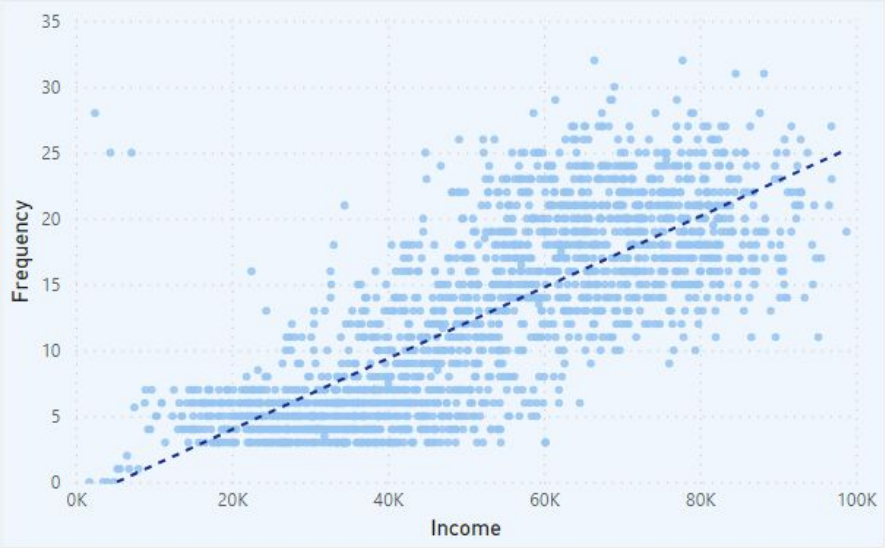


Observation

There is not much different between Recency. A clear pattern that: customers with **higher Frequency**, meaning those who make more purchases, **achieve higher Monetary ranks** due to their accumulated spending.

However, it is seen that customers who make high number of purchases have, on average, **twice the income of those who make least purchases**.

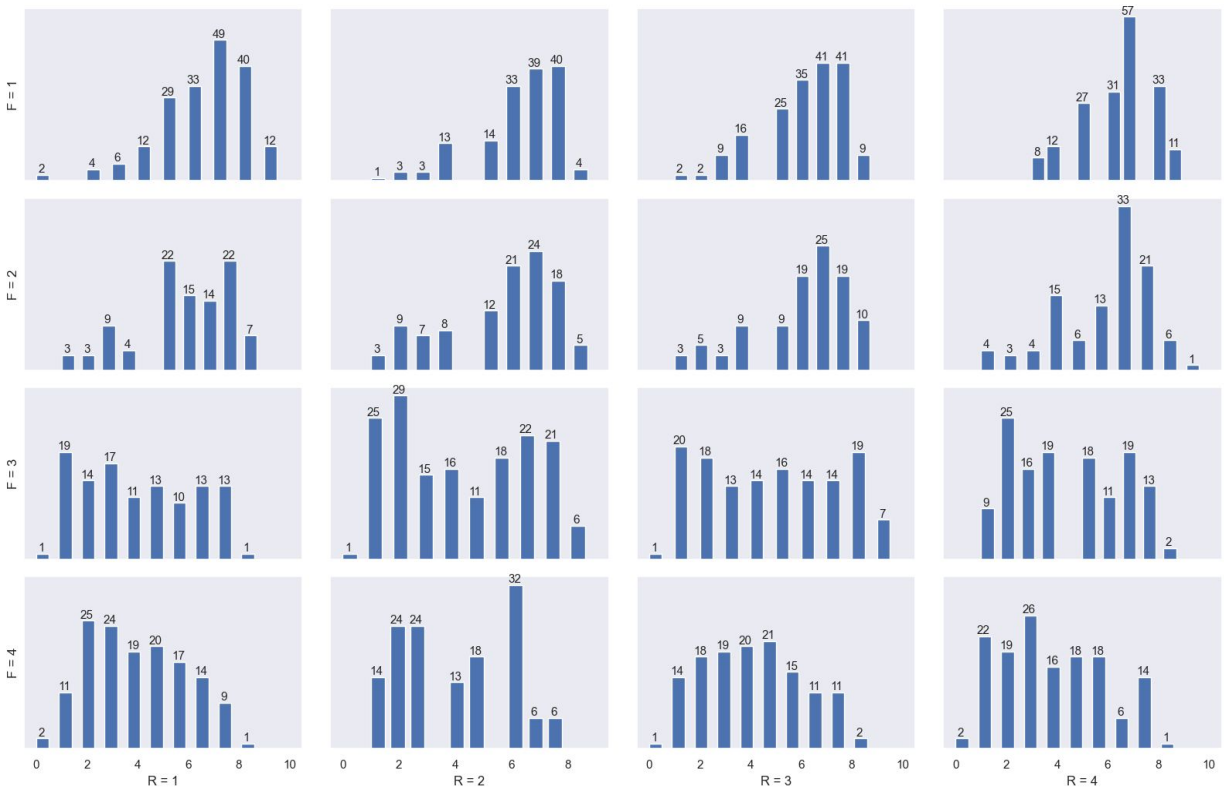
How Customer spend money each Segment?



Observation

As customer **income rises**, there is an **increasement in both the frequency of their purchases and the amount of money they spend**.

How often each Segment access Website?



Observation

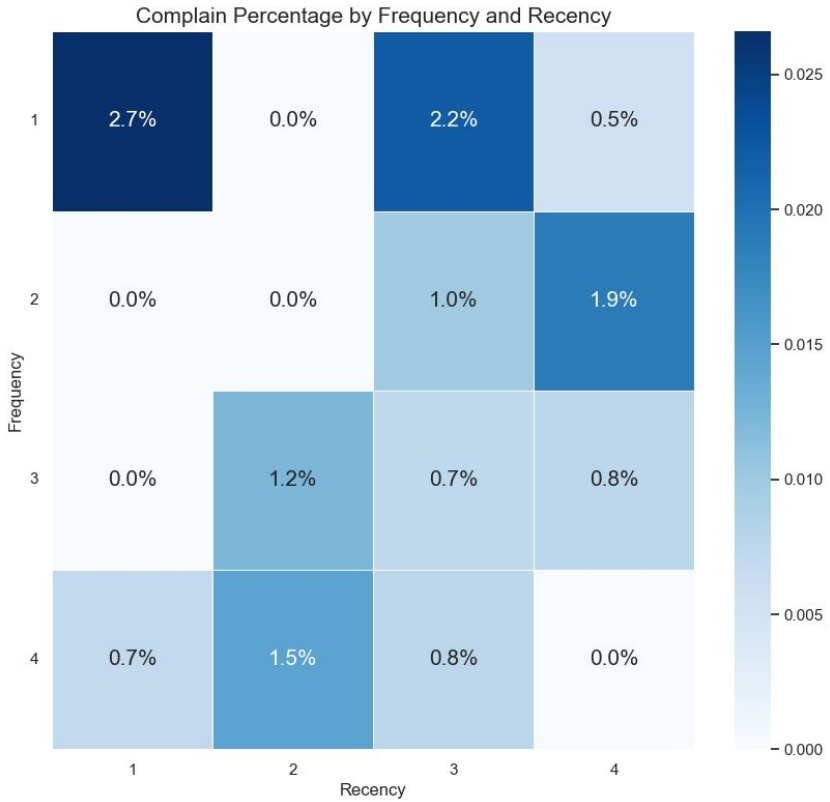
There is not much different between Recency.

=> It suggests that even **customers who have stopped buying continue to visit the company's website**, making it a valuable channel for communication with customers.

Customers who buy from the company less frequently tend to visit the website more often.

=> Customers might want to get more information on the product before buying.

Did customers stop due to dissatisfaction?



Observation

The number of Complain is very low and there is no clear relation from RFM to Complain rate.

So complain should not be the factor of customer stop buying products from company.

04

Propose Stategy

How to turn Customer to VIP?

	R = 1	R = [2,3]	R = 4
F = 1	Lost customer	About to sleep	New Customer
F = [2,3]	Hibernating	Need attention	Potential Loyalist
F = 4	Once loyal	Can't lose them	**Loyal customer**

Retention Campaigns
to Increase purchase volume

Re-engagement Campaigns
to Pull back Inactive Customer

Re-engagement Campaigns

Reactivation Messages

- send email/SMS messages to inactive customers
- personalized landing page with a time-limited coupon or offer

Abandoned Cart Recovery

- automated email reminder for customers who leave items in cart
- offer a time-sensitive discount or free shipping

Product Recommendations

- based on customer's previous purchase and web browsing activities
- promote products on sale or related campaigns

Community Engagement

- host community engagement events (local charities, social groups)
- Showcase the positive impact through communication channels

Retention Campaigns

Welcome and Onboarding

- personalized welcome email
- useful information, product guides
- special discount for new customer

Loyalty Program

- tiered loyalty where customers earn points for each purchase
- personalized offers based on their purchase history

Exclusive Content

- sneak peeks of new products
- add products based on high demand
- exclusive discount

Referral Program

- rewards or discounts for referring new customers
- more rewards if the referred customer purchase reach certain goal

Thanks!

Do you have any questions?

What we discussed

Data Introduction

Data Handling & EDA

RFM Model

Propose Stategy