Data Analysis - Machine Learning

Customer Clustering

Based on Behavior & Needs

Zijing XUE

DAFT NOV 2021

11 Feb 2022

Domain

Help the **company** to better understand their **customers** by separating customers into groups that reflect **similarities** among them.

Problem 1

ROI:

modify products based on target customers needs.

Problem 2

Customer satisfaction:

serve customers with different shopping behaviors by attractive campaigns and customer service.

Process

1/ Data Preparation

Define target, Find Dataset, Cleaning(Missing value, Outlier, Duplication), Encode, Scale

2/ EDA

Understanding of data, Correlation, Descriptive Analysis, Drop/ Merge/ Create Columns

3/ Clustering

PCA, Model Building & Evaluation, Unsupervised ML

4/ Insight

Patterns Analysis from Descrptive Statistics and Profiling

Step 1

Dataset

1/ Data Preparation

Data Map

People

ID
Year_Birth
Education
Marital_Status
Kidhome
Teenhome

Income

Date_Enrollment
Recency
Complain

Products

MntWines
MntFruits
MntMeatProducts
MntFishProducts
MntSweetProducts
MntGoldProds

Place

NumWebPurchases
NumCatalogPurchases
NumStorePurchases
NumWebVisitsMonth

Promotion

NumDealsPurchases

AcceptedCmp1
AcceptedCmp2
AcceptedCmp3
AcceptedCmp4
AcceptedCmp5
Response

1/ Data Preparation

Input Missing Values
& Remove Outliers

Step 2

Cleaning

```
data.isna().sum()
```

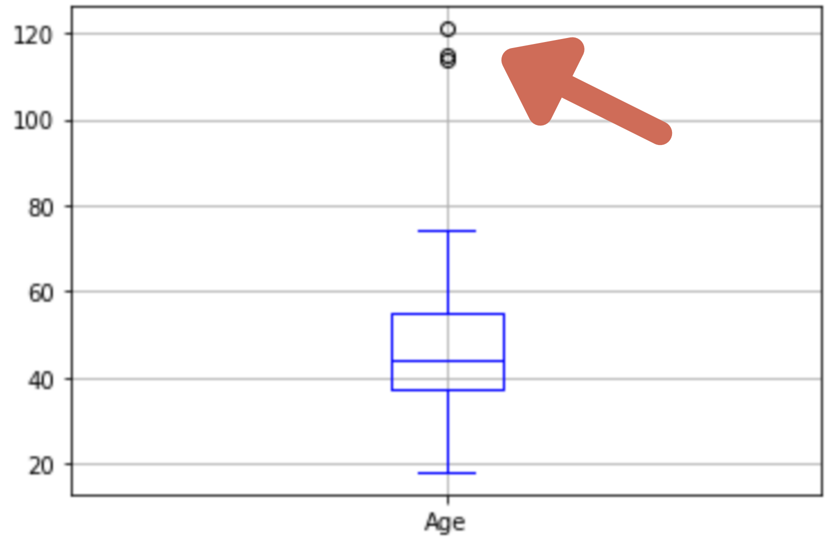
```
ID                 0
Year_Birth         0
Education          0
Marital_Status     0
Income            24
Kidhome            0
Teenhome           0
```

```
avg_sal_edu = round(data.groupby('Education')['Income'].agg('mean'),1)

for index, row in data[data.isna().any(axis=1)].iterrows():
    a = row['Education']
    data.loc[index,'Income']= avg_sal_edu[a]
```

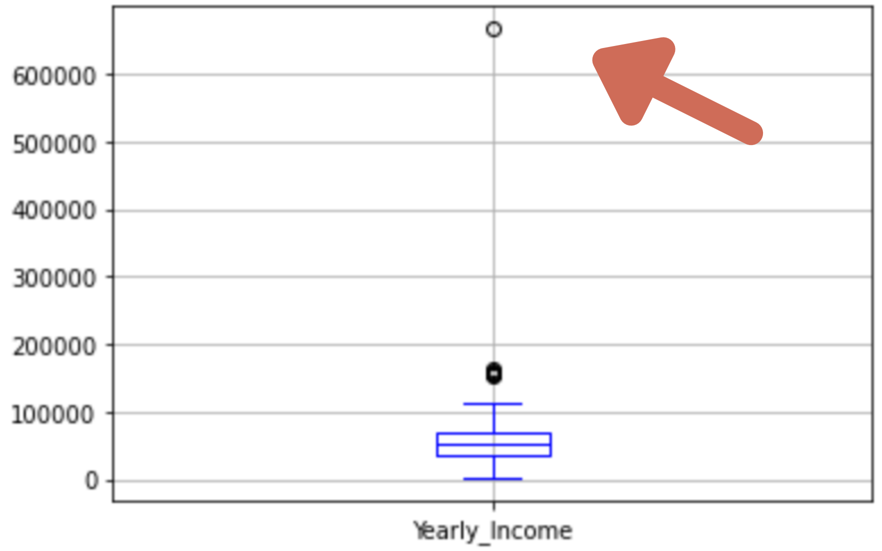
```
data.boxplot('Age', grid='true', color='blue')
```

<AxesSubplot:>




```
data.boxplot('Yearly_Income', grid='true', color='blue')
```

<AxesSubplot:>



1/ Data Preparation

Data Conversion
& Add up columns into total value

Year_Birth	Dt_Customer
1983	15-11-2013
1986	20-02-2013
1959	05-11-2013
1951	01-01-2014
1982	17-06-2013

Age	Client_since_(month)
42	7.6
30	21.8
44	28.7
28	28.9
40	24.1

Kidhome	Teenhome
0	0
1	1
0	0
1	0
1	0

Children
0
2
0
1
1

```python
# total spendings:
data['Spending']=data['Wines']+data['Fruits']+data['Meat']+data['Fish']+data['Sweets']+data['Gold']
```

Wines	Fruits	Meat	Fish	Sweets	Gold
635	88	546	172	88	88
11	1	6	2	1	6
426	49	127	111	21	42
11	4	20	10	3	5
173	43	118	46	27	15

Spending
1617
27
776
53
422

Step 3

Wrangling

1/ Data Preparation

Categorical Columns value grouping

Step 2

Cleaning

```
data['Education'].value_

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

```
Bachelor          1127
PhD                486
Master             370
Undergraduate      257
Name: Education, dtype: int64
```

```
data['Marital_Status'].value_

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

```
Married      864
Couple       578
Single       563
Divorced     231
Name: Marital_Status, dtype: int64
```

1/ Data Preparation

Encode Non-Numeric Columns

Using LabelEncoder

```
lable = LabelEncoder()

for x in ['Education','Marital_Status']:
    # to have a dict of class & encode
    lable.fit(data_encode[x])
    label_name_mapping = dict(zip(lable.classes_, lable.transform(lable.classes_)))
    print(label_name_mapping)

    #Encode class col
    data_encode[x]=lable.fit_transform(data_encode[x])
data_encode
```

```
{'Bachelor': 0, 'Master': 1, 'PhD': 2, 'Undergraduate': 3}
{'Couple': 0, 'Divorced': 1, 'Married': 2, 'Single': 3}
```

Education	Marital_Status
Bachelor	Single
Bachelor	Single
Bachelor	Couple
Bachelor	Couple
PhD	Married

Education	Marital_Status
0	3
0	3
0	0
0	0
2	2

1/ Data Preparation

Scale Data

Using StandardScaler

Education	Marital_Status	Yearly_Income	Recency_(days)	Wines	Fruits	Meat	Fish
0	3	58138.0	58	635	88	546	172
0	3	46344.0	38	11	1	6	2
0	0	71613.0	26	426	49	127	111
0	0	26646.0	26	11	4	20	10
2	2	58293.0	94	173	43	118	46

```
df = data.copy()

scaler = StandardScaler()
scaler.fit(df)
scaled_data = pd.DataFrame(scaler.transform(df), columns=df.columns)
scaled_data
```

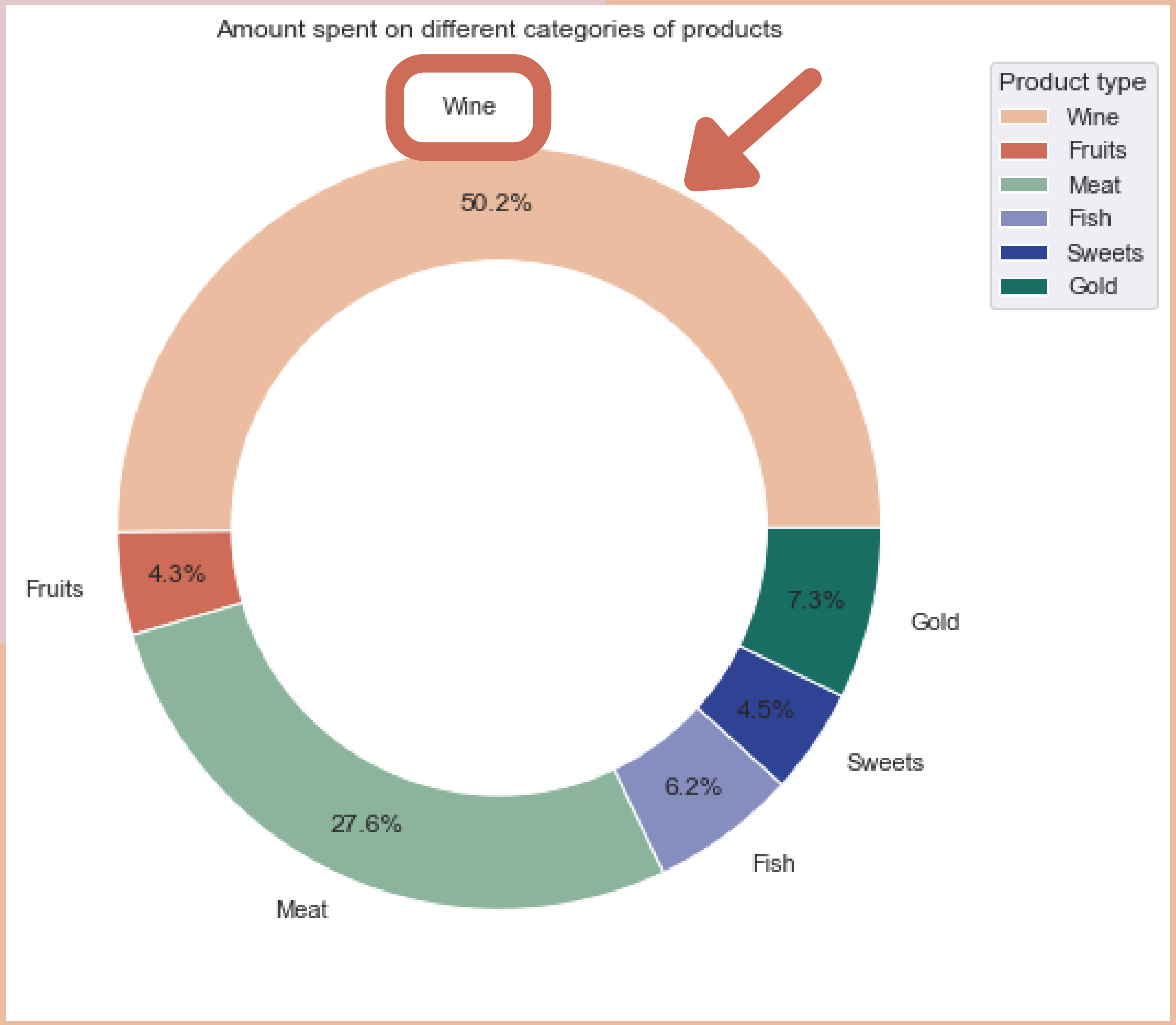
	Education	Marital_Status	Yearly_Income	Recency_(days)	Wines	Fruits	Meat	Fish
0	-0.869141	1.222432	0.288195	0.306856	0.983228	1.554170	1.679746	2.461068
1	-0.869141	1.222432	-0.262715	-0.383971	-0.871064	-0.636431	-0.713455	-0.650414
2	-0.869141	-1.457331	0.917627	-0.798467	0.362159	0.572177	-0.177201	1.344595
3	-0.869141	-1.457331	-1.182829	-0.798467	-0.871064	-0.560893	-0.651409	-0.503991
4	0.977319	0.329178	0.295435	1.550344	-0.389661	0.421101	-0.217088	0.154911

2/ EDA

Categories

The best-selling categories are:
Wine(50.2%)
Meat(27.6%)

The worst-selling categories are:
Fruits(4.3%)
Sweets(4.5%)

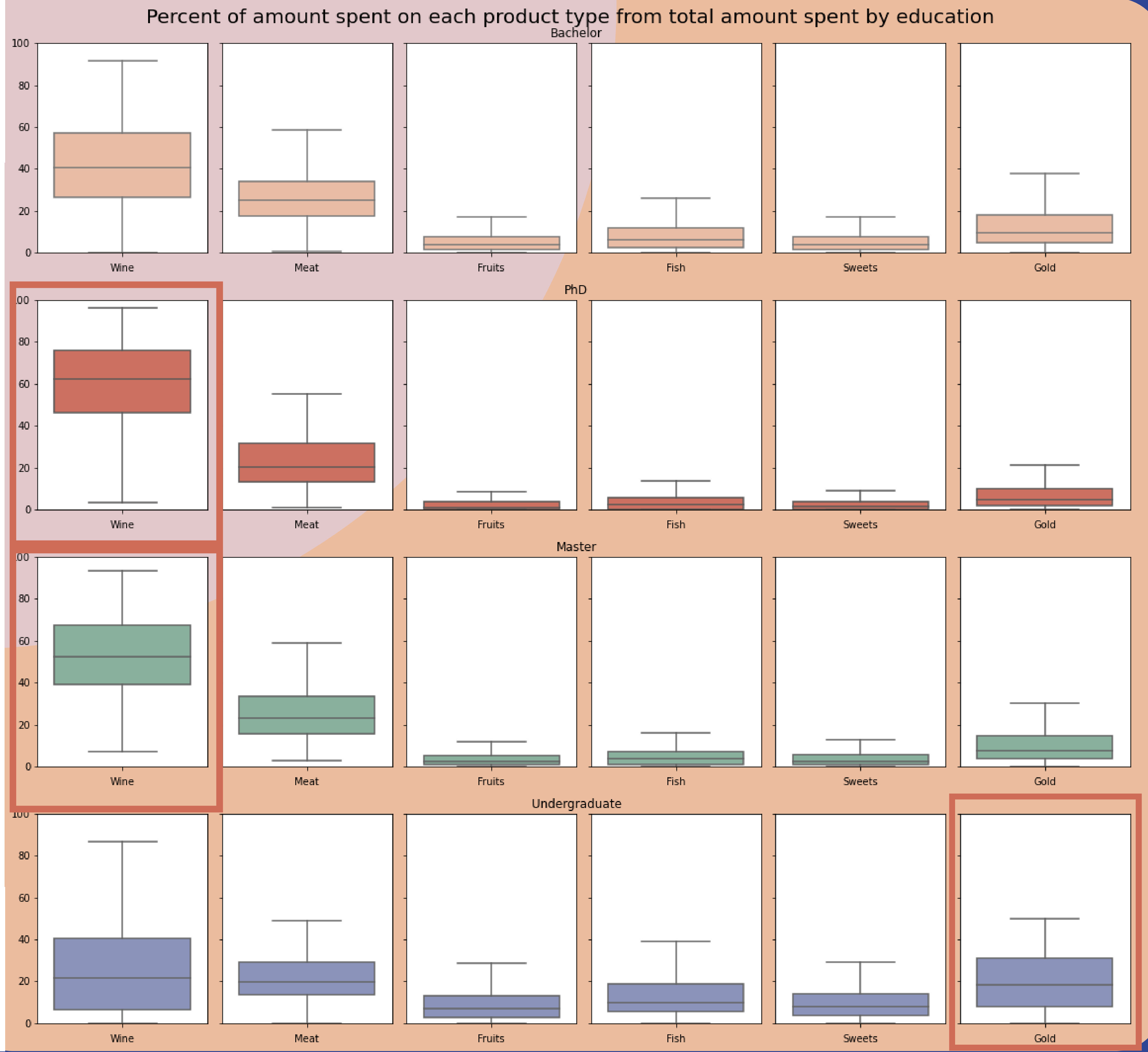


2/ EDA

Categories preference of customers with different Education level

PhD's & Master spend 50%(in median) of total spending on **Wine** categories.

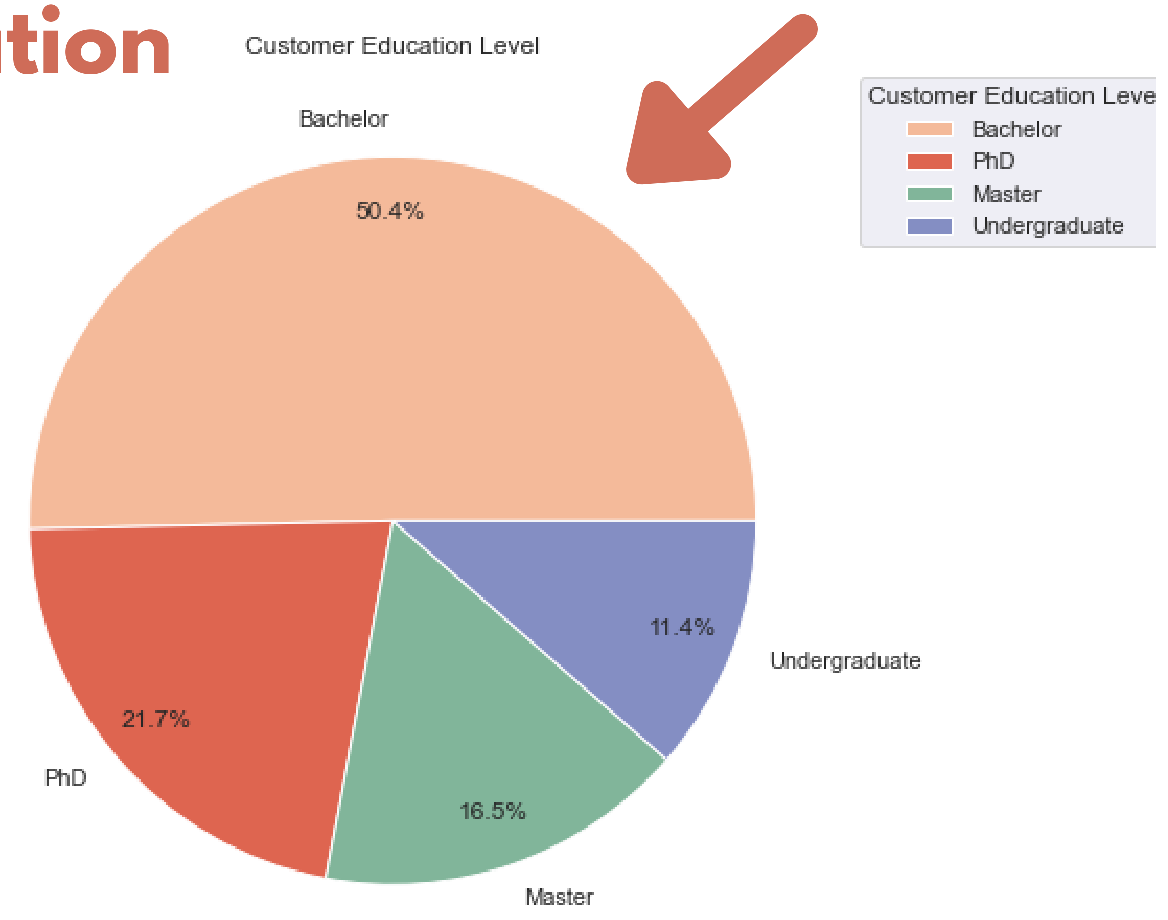
Undergraduates spend more of their budget(20%) on **Gold** than other customers.



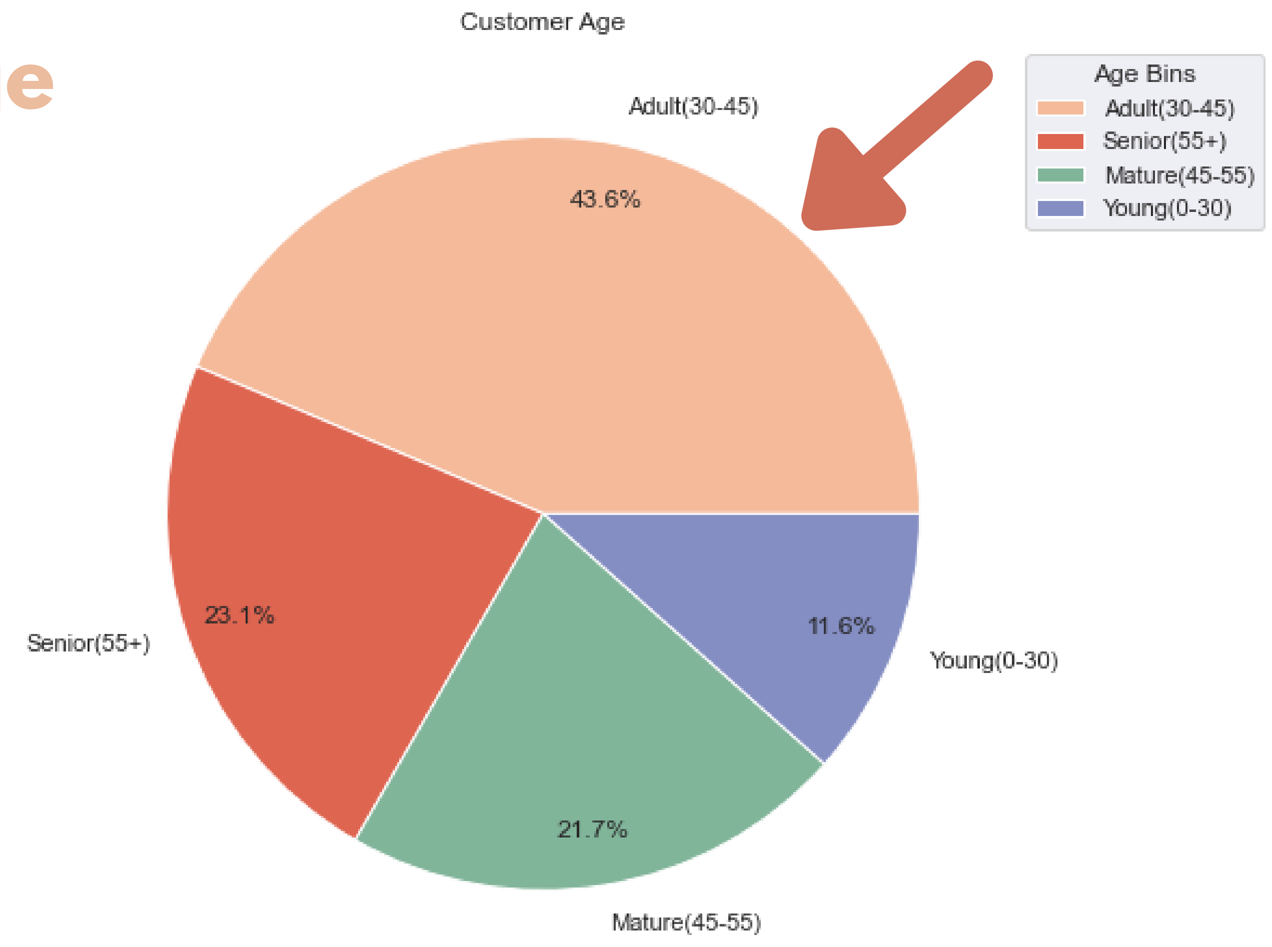
2/ EDA

Half of the customers are: Bachelor(50.4%)

Education



Age

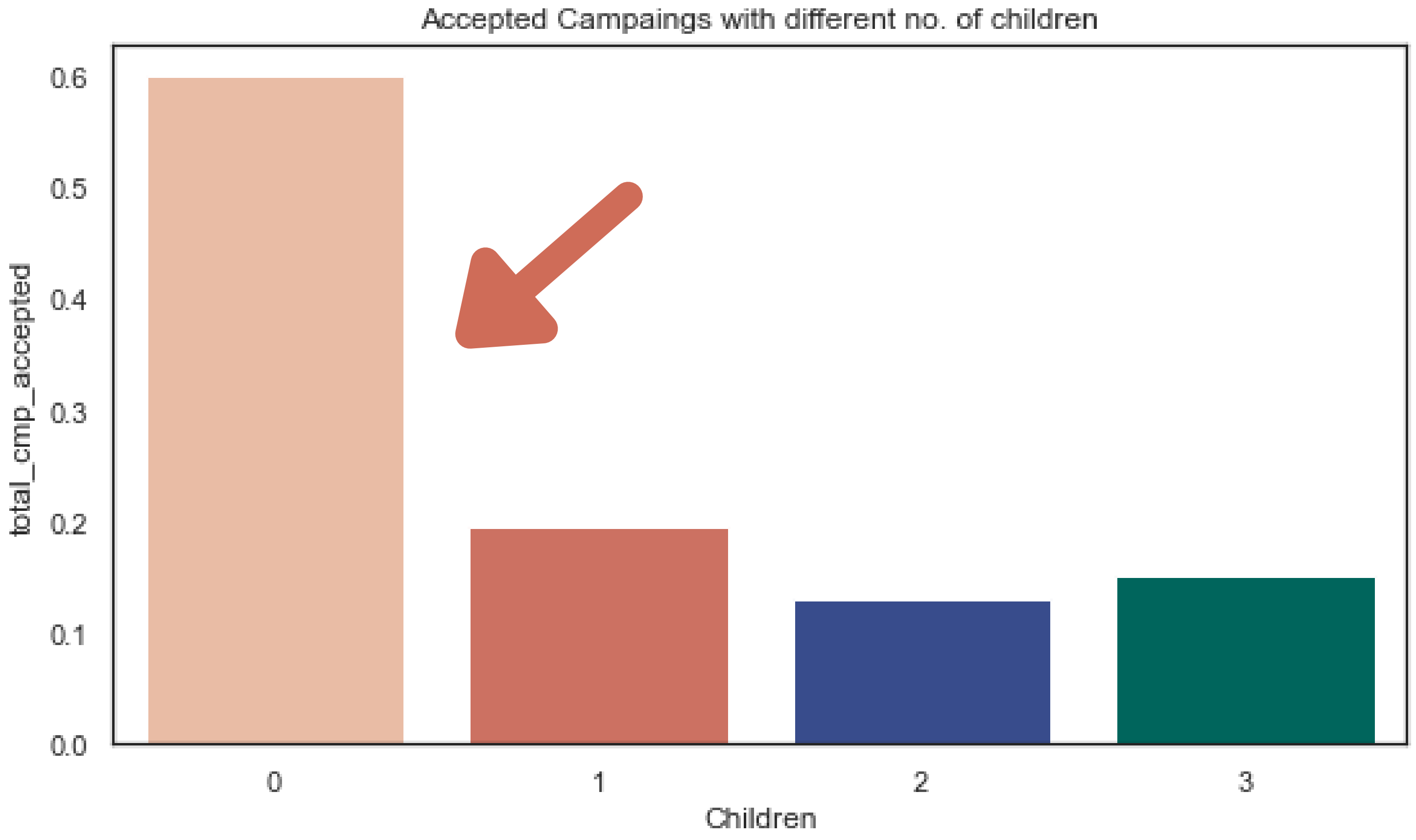


Biggest customers age group is: 30-45 years old (43.6%)

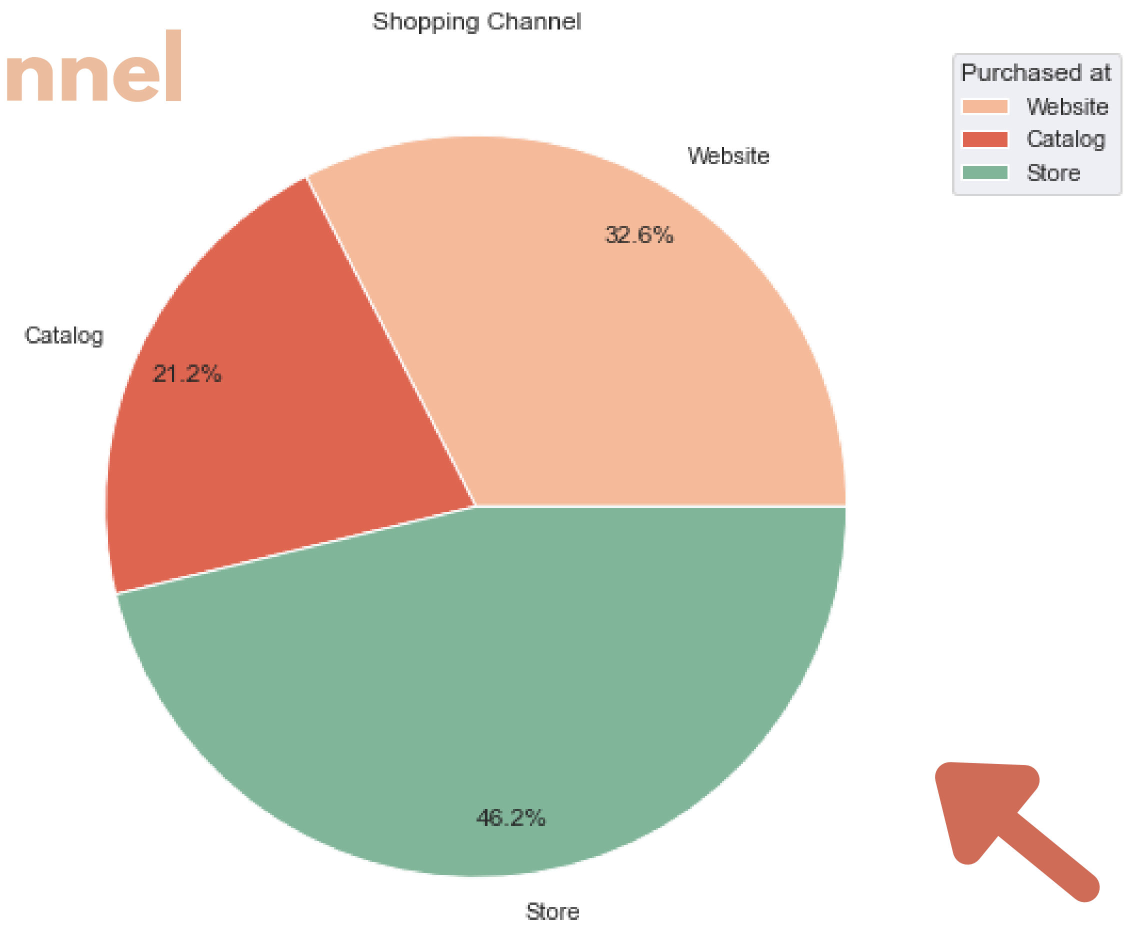
2/ EDA

Most of the customers(60%) don't have any child.

Children



Channel



The most popular Channel is: Store (46.2%)

3/ Clustering

Unsupervised ML

Using unsupervised machine learning to cluster customers who share some similarities, to better serve for future sales & marketing activities.

Step1

Build Model with Different Methods

Step 2

Evaluate Model: Silhouette Score

Step 3

Visulization with PCA

3/ Clustering

- Build Model with Different Methods.
- Compare the **Silhouette score**.
- **KMeans** & GMM got the best score, but KMeans seems more interesting given the distribution of customer in different clusters.

	Model	No. of Clusters	Silhouette Score
0	KMeans	4	0.524
1	Agglomerative Clustering	4	0.486
2	Gaussian Mixture	4	0.524
3	DBSCAN	3	0.058

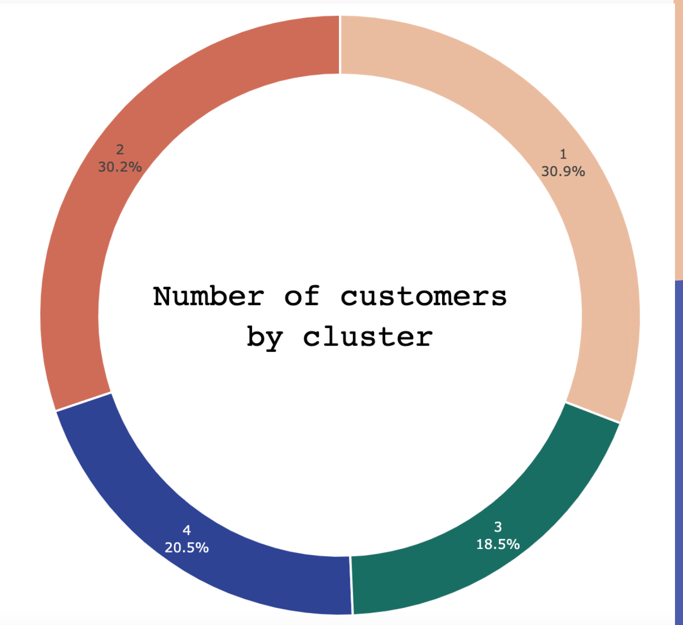
```
2      689
1      675
3      459
4      413
Name: cluster_Kmeans, dtype: int64
```

```
1     1036
2      768
0      401
3       31
Name: cluster_gmm, dtype: int64
```

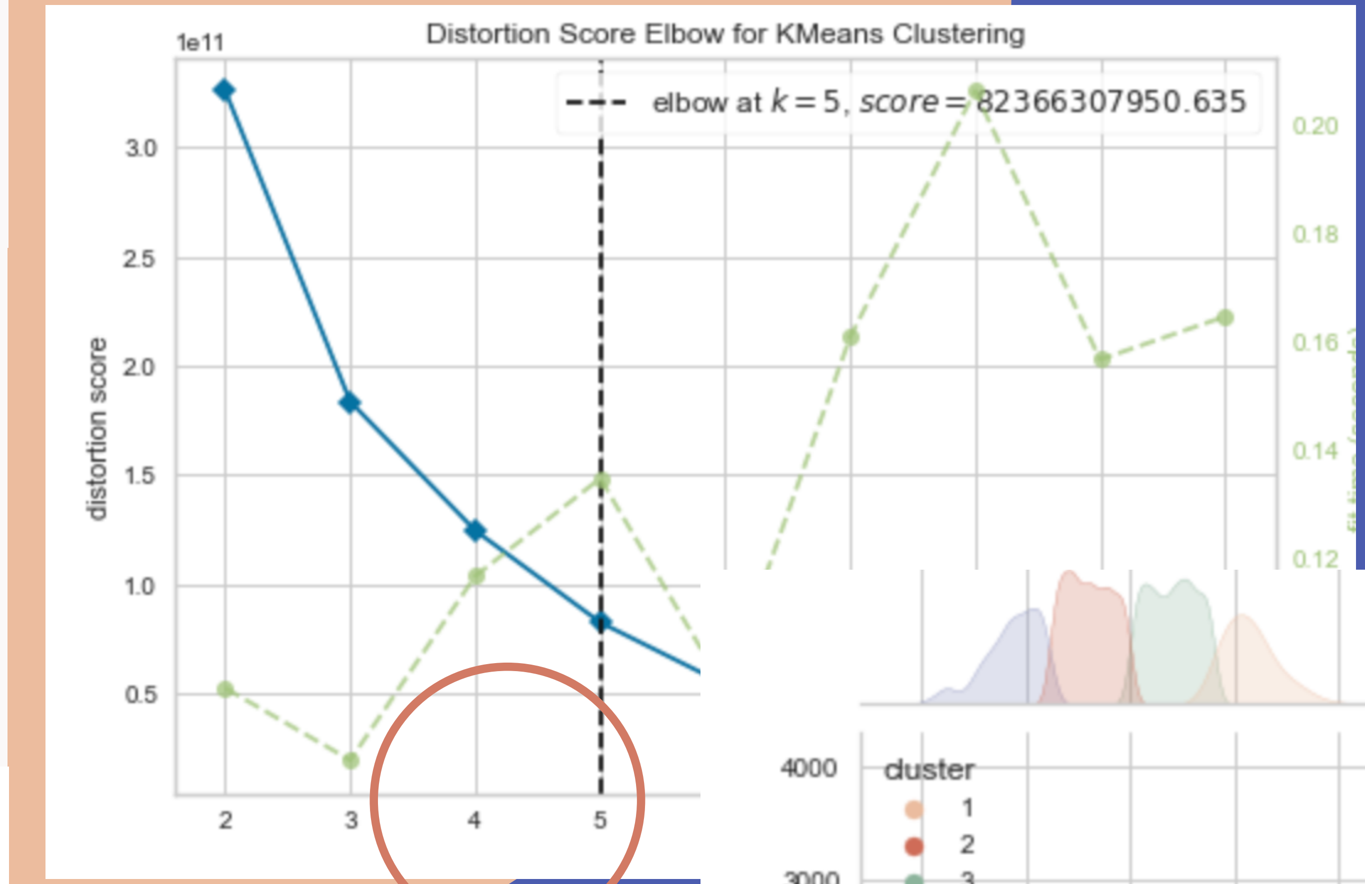
3/ Clustering

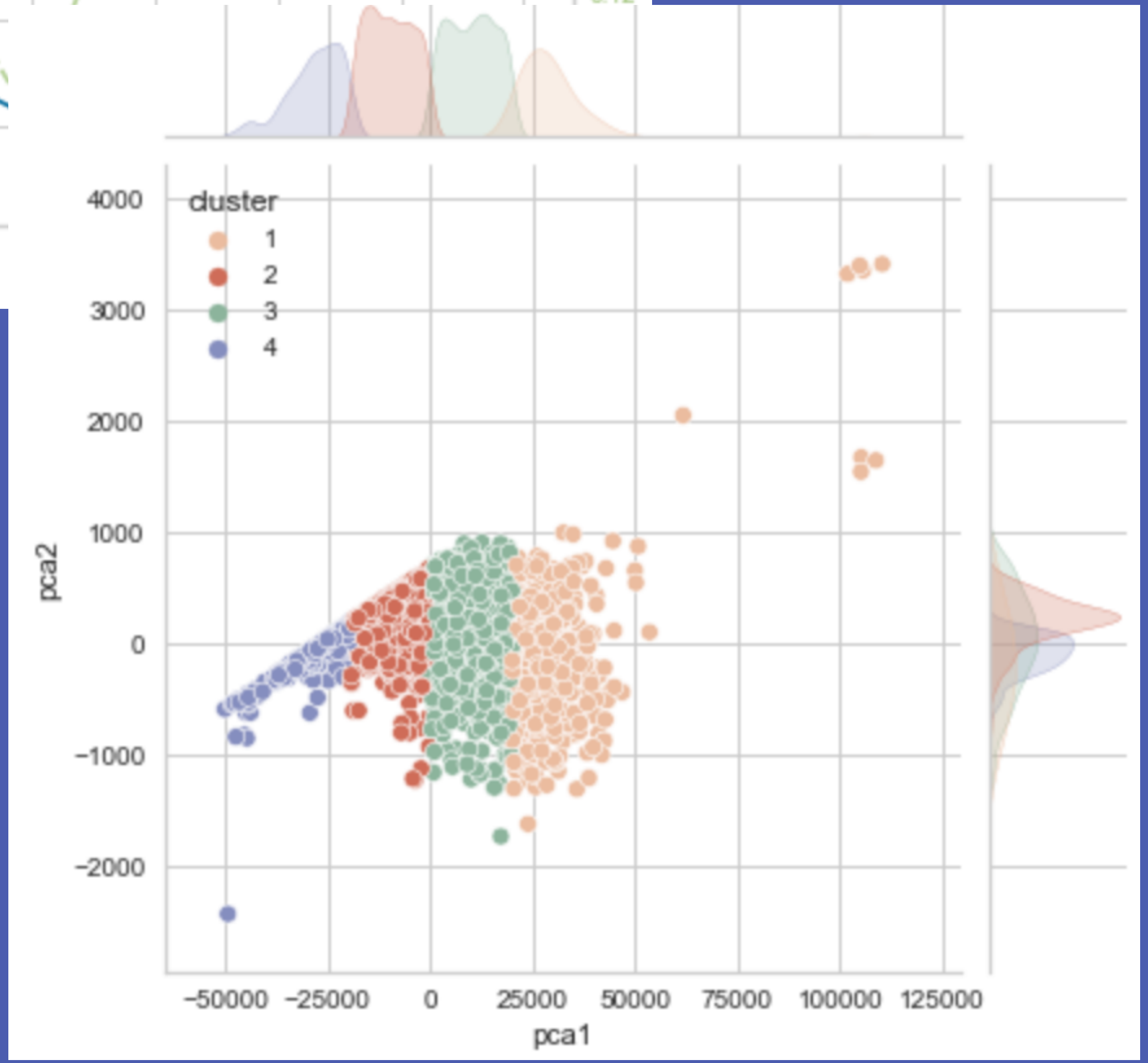
KMeans Model

- **KElbowVisualizer** to find optimal no. of clusters
- Train the model.



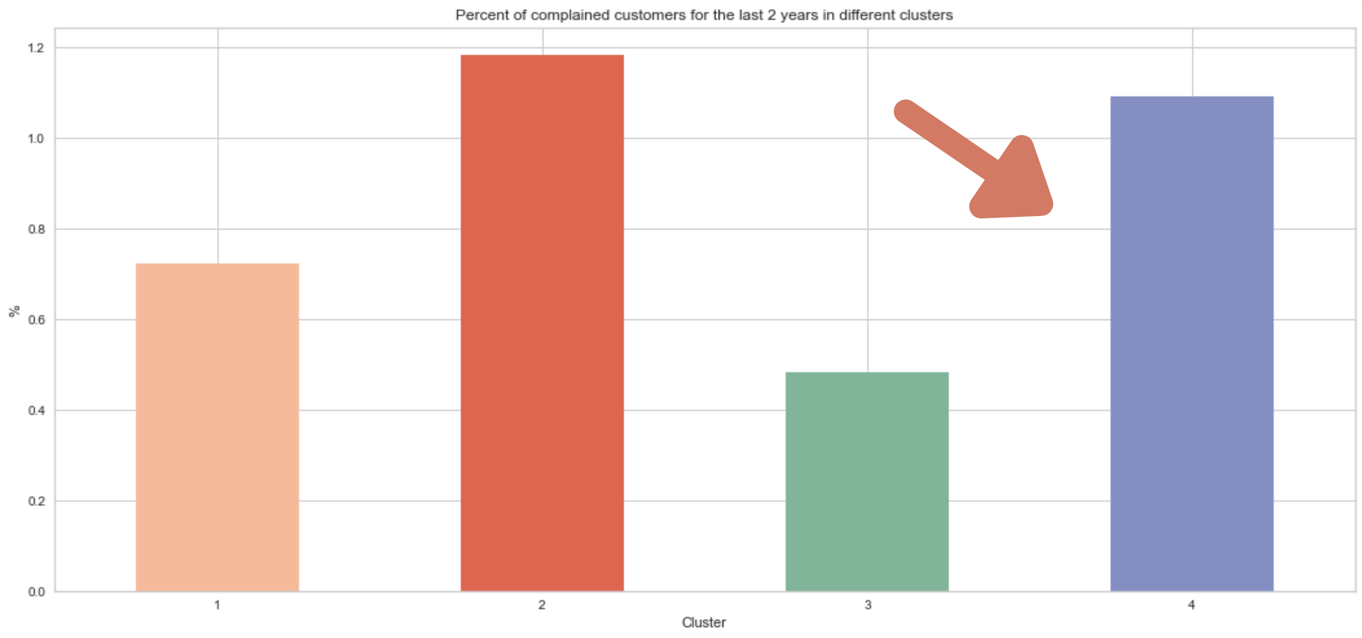
- **PCA** feature selection to visualize clusters



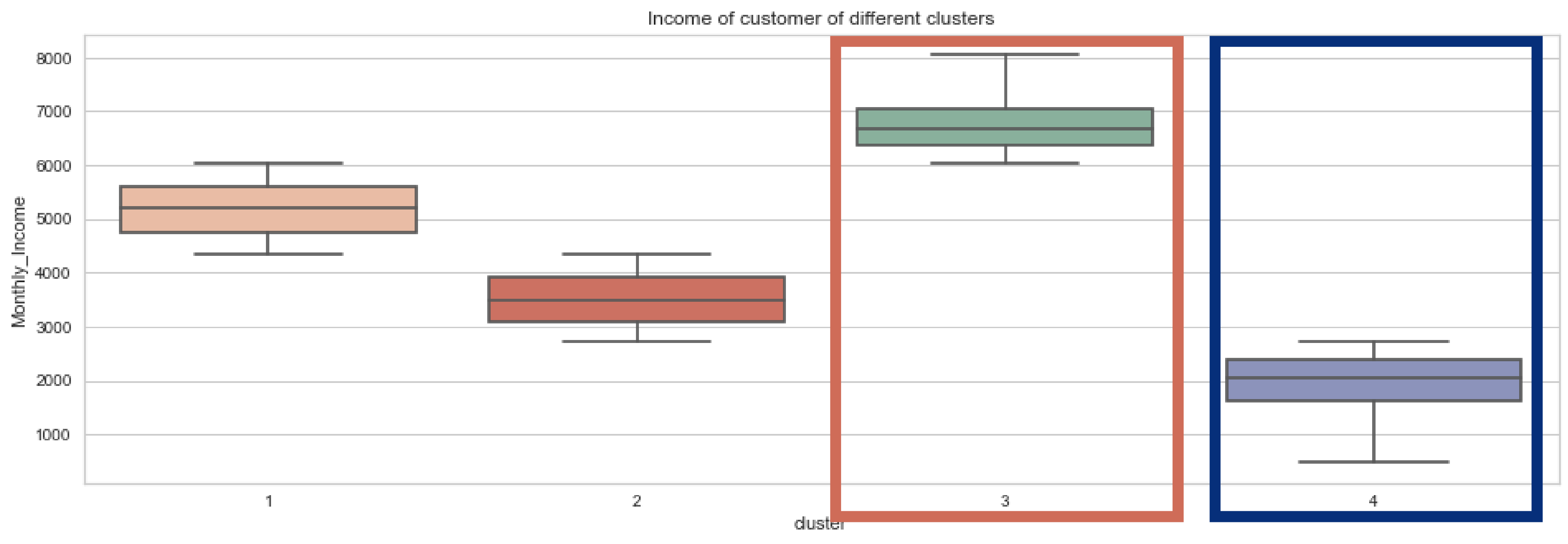


4/ Insight

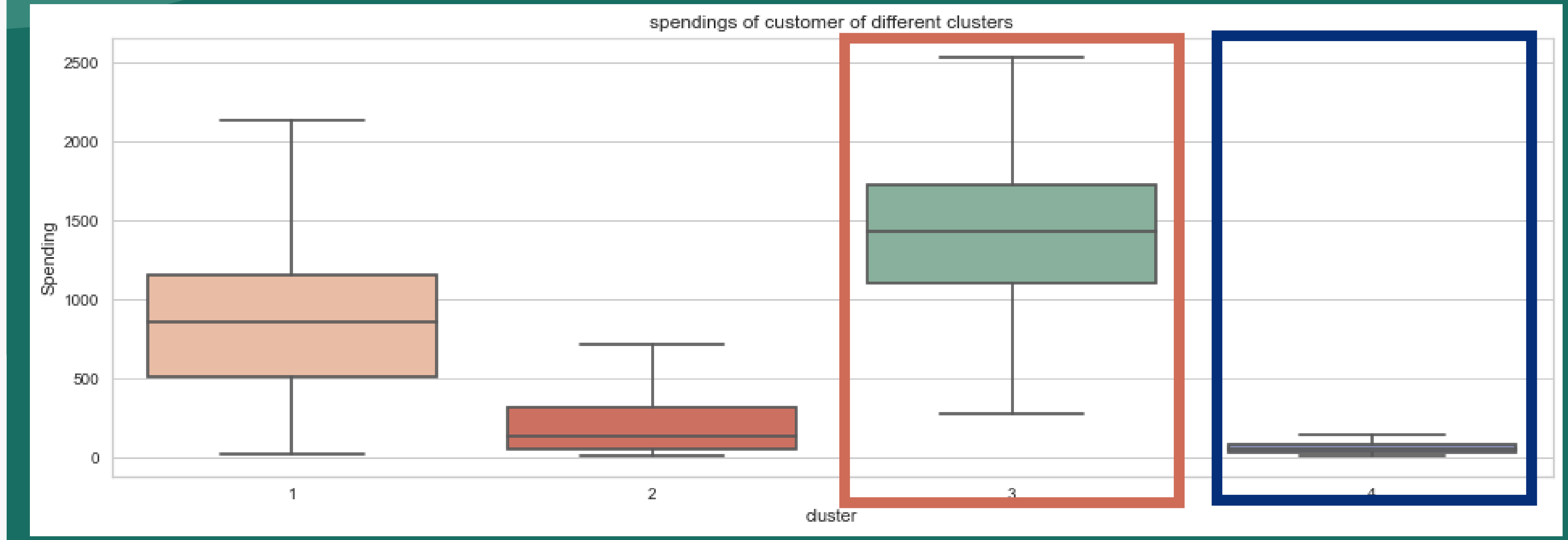
- Cluster1: Average Income x High Spending
- Cluster 2: Average Income x Low Spending
- Cluster 3: High Income x High Spending
- **Custer 4: Low Income x Low Spending, but they have the 2nd most complaints (in %)**



Income



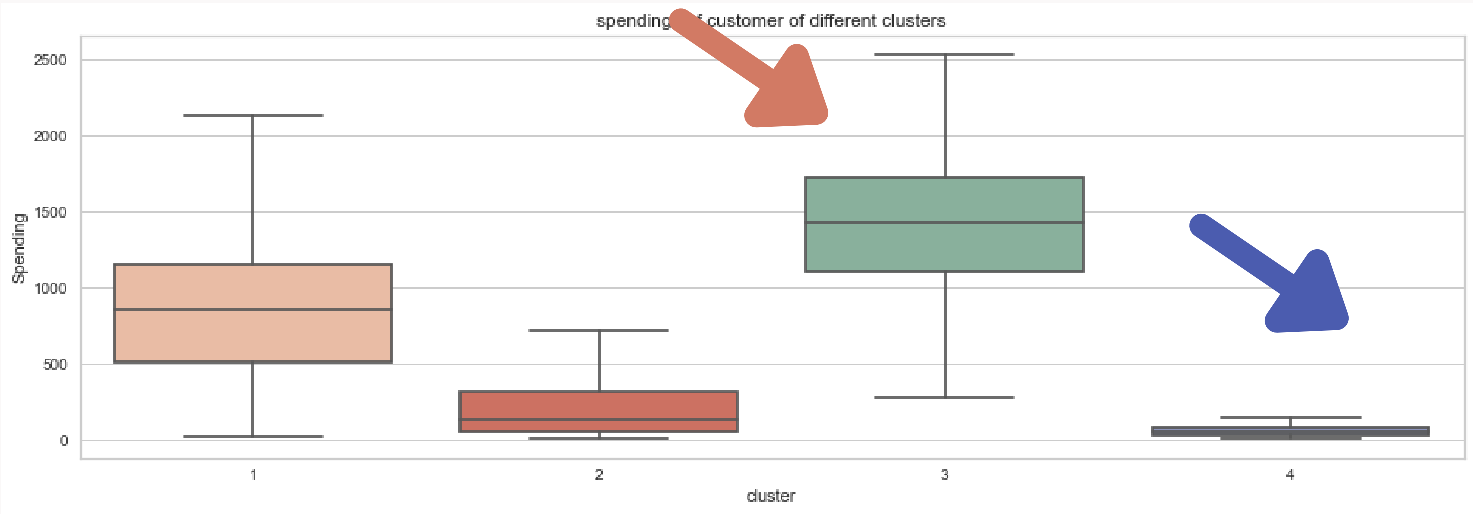
Spending



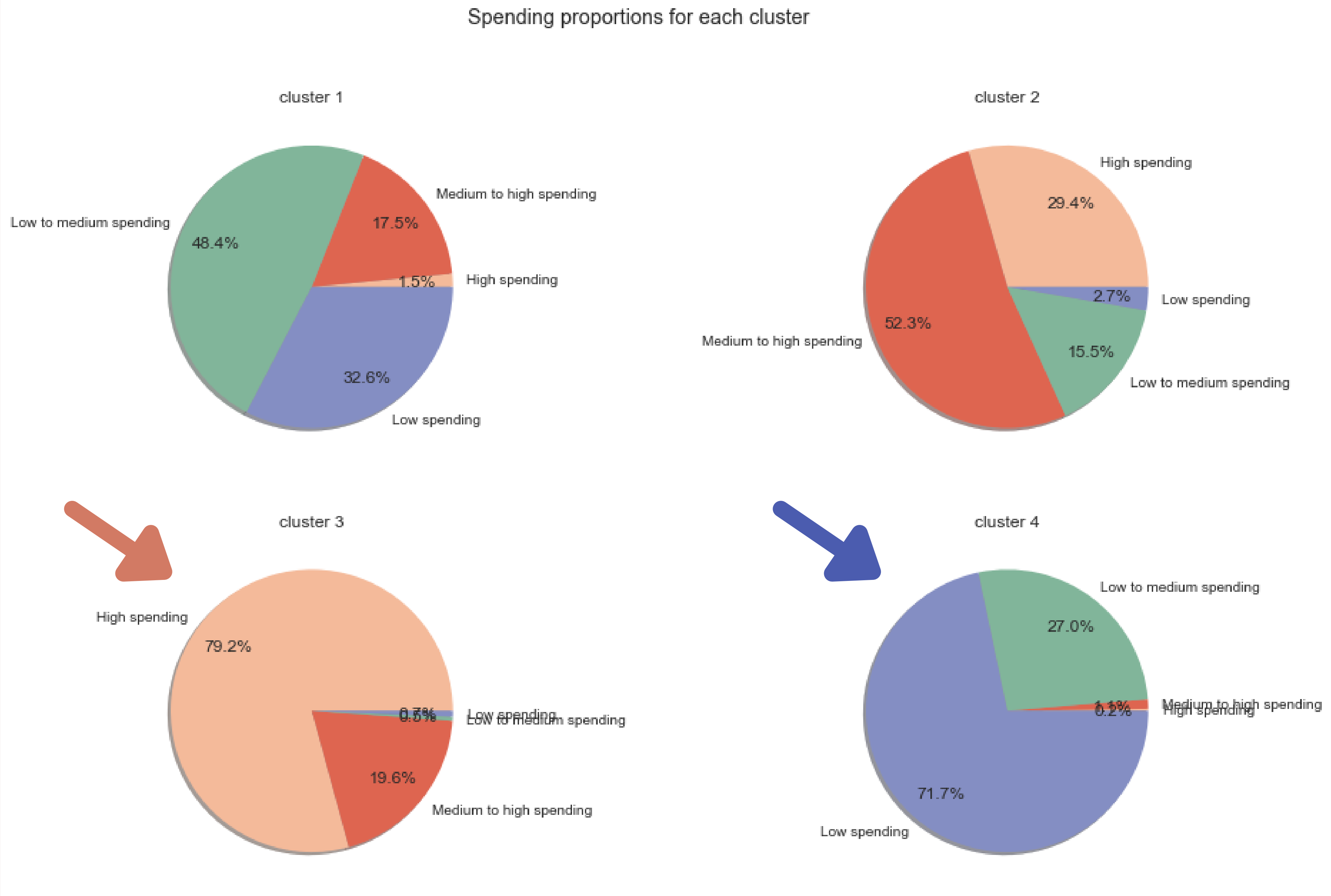
*I convert yearly income into monthly income, it's easier to understand for business scenario

4/ Insight

- Cluster1: Average Income x High Spending
- Cluster 2: Average Income x Low Spending
- **Cluster 3: High Income x High Spending**
- **Custer 4: Low Income x Low Spending, but they have the 2nd most complaints (in %)**



Spending



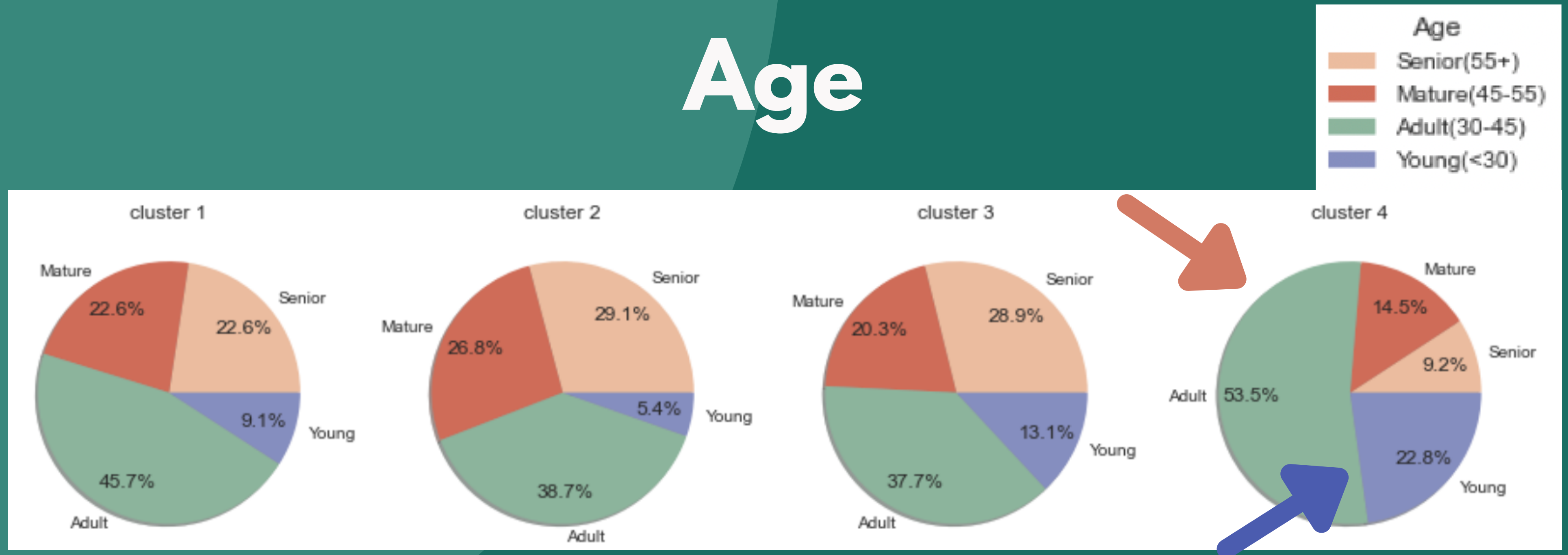
*I convert yearly income into monthly income, it's easier to understand for business scenario

4/ Insight

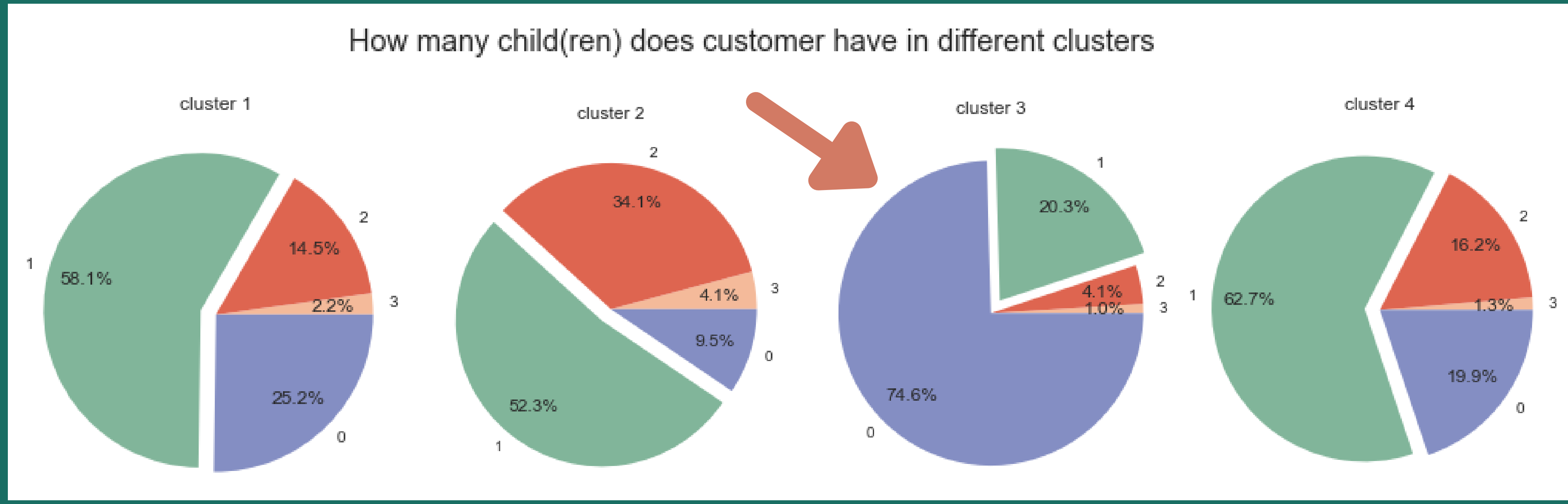
- Most customers in **each cluster** are among 30-45 years old
- Cluster 4 have 50% + of 30-45 year-old customers, and **most of the Young people** among all clusters

- There are mostly parents in 1st, 2nd, 4th clusters.
- Most customers in **3rd cluster** have **no child**.

Age



Children



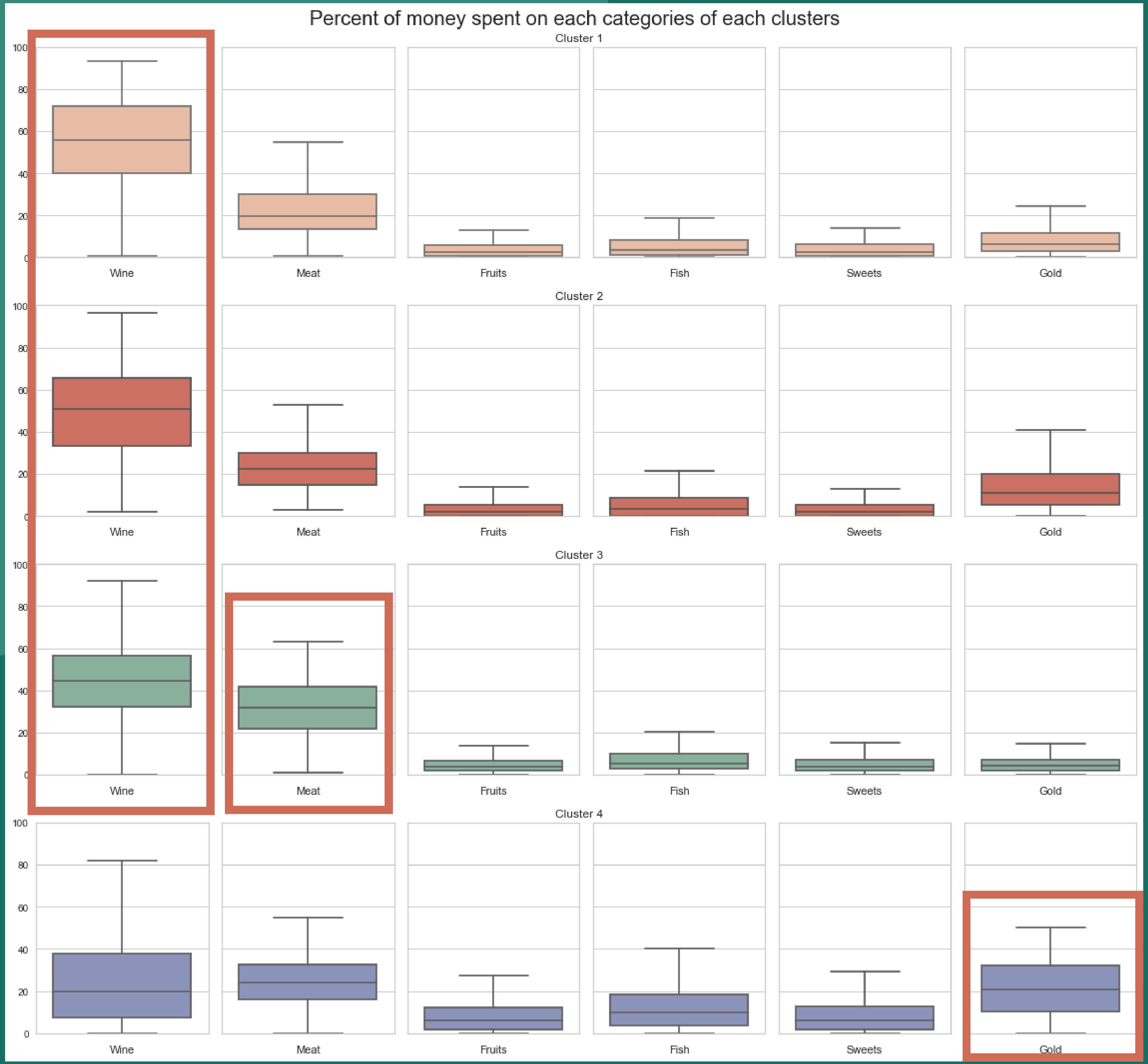
4/ Insight

Category wise:

- Popular products types are the same in all clusters : **Wine & Meat**

Cluster wise:

- **Cluster 1&2&3** spend around **50%** on Wine products (median, in %)
- **Cluster 3** buy **Meat** more than others
- **Cluster 4** spend the most (in %) on **Gold** among all clusters



Channel

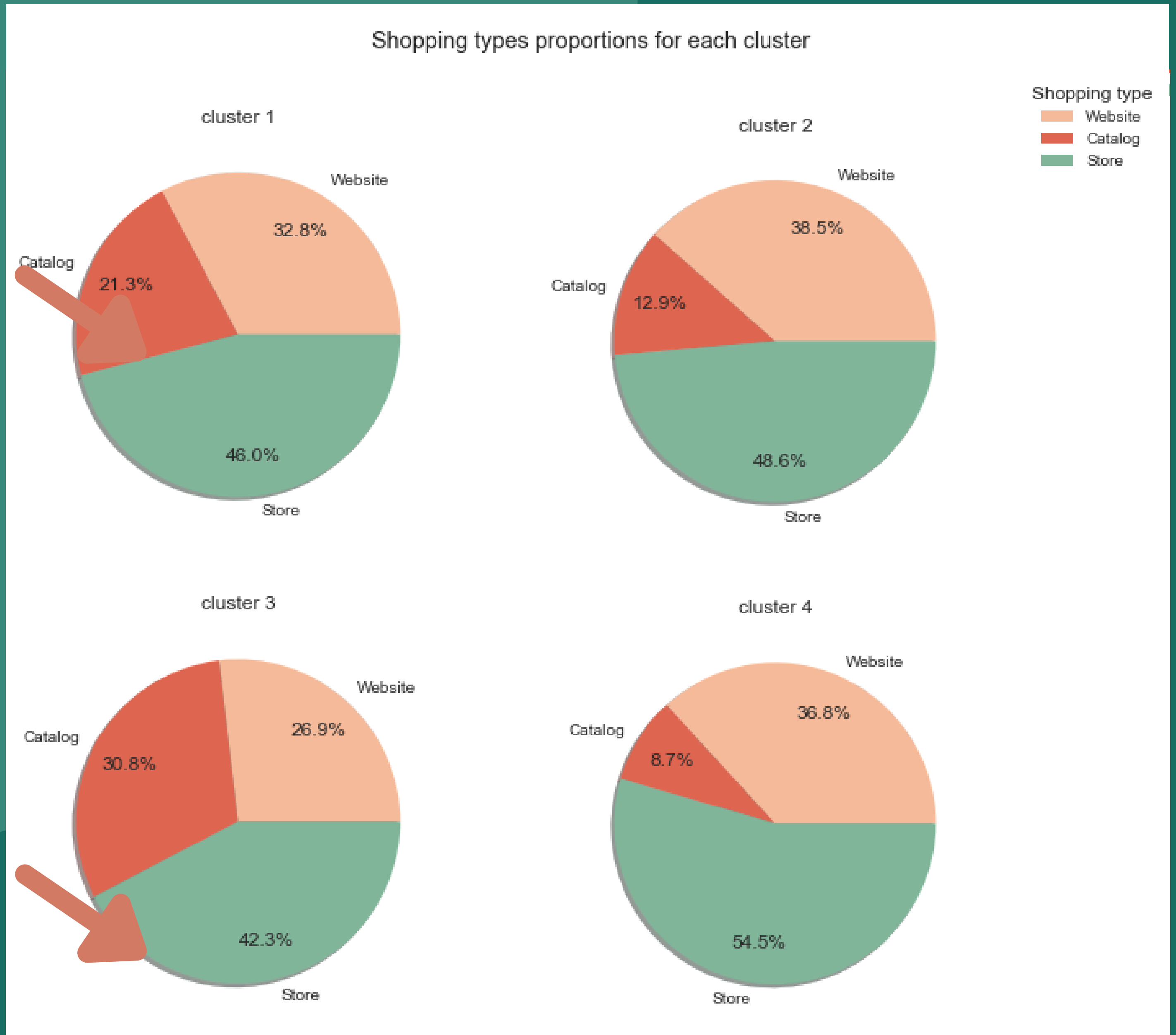
4/ Insight

Channel wise:

- Mustomers in each cluster mostly bought from **Stores**.

Cluster wise:

- Customers from **Cluster 1&3** clusters bought more from **Catalog*** more than other clusters, better to target them for Catalog Channel
- **Cluster 1 & 4** visit more company's **website** than others



Website Visit

*Retailers provide product information to consumers through mail

4/ Insight

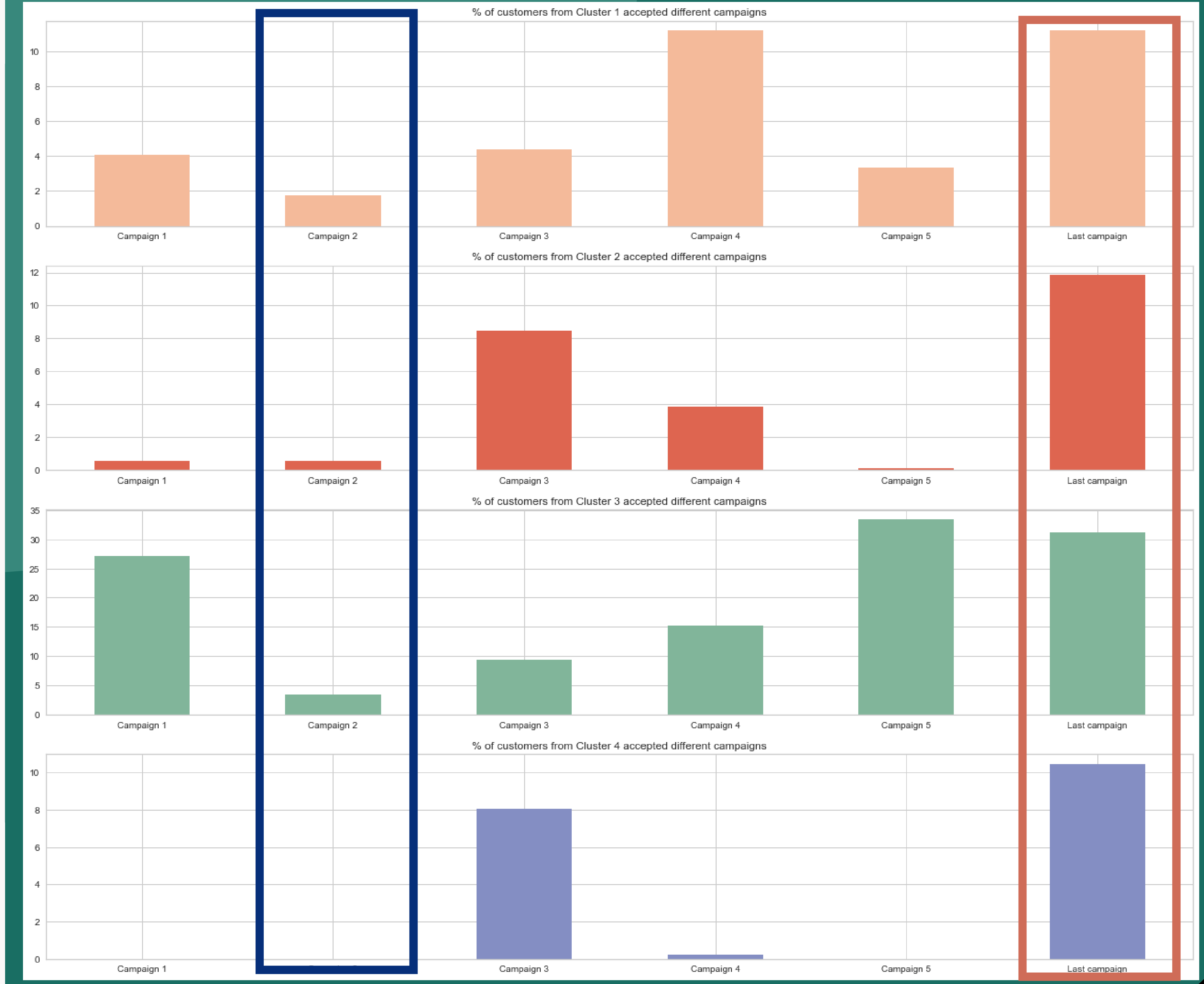
Campaign wise:

- **Last campaign** was the most successful, **Campaign 2** is the least accepted,
- The biggest interest in Campaign 5: Cluster 3

Cluster wise:

- **Cluster 1 & 3** accepted more campaigns than other clusters
- Interesting that **wealthy people** are more chasing for **sales**

Campaign



Conclusion

Gold

Cluster 1

Avg Income x High Spending

Most of them are parent.

Wine, Meat

Store, Catalog

Sensitive to Camp

4 & last Camp works well

Silver

Cluster 2

Avg Income x Low Spending

Most of them are parent.

Wine, Meat, Gold

Store, Website

Not quite sensitive to Camp

3 & last Camp works well

Platinum

Cluster 3

High Income x High Spending

Most of them have no child

Wine, Meat

Store, Catalog

Sensitive to Camp

1 & 5 & last Camp works well

Highest complain rate

Bronze

Cluster 4

Low Income x Low Spending

Most of them are parent.

Have most of customers <30 y/o

Meat, Gold, Wine

Store, Website

Not sensitive to Camp

3 & last Camp works well

2nd highest complain rate

Customer Clustering
Based on Behavior & Needs

Thank you :)

Zijing XUE

DAFT NOV 2021

Customer Clustering
Based on Behavior & Needs

Question?

Zijing XUE

DAFT NOV 2021