Predictive Analytics

Customer Segmentation and Personality Analysis

Group:8

Submitted To

Ambatipudi Vamsidhar

(Department of Management)

Department of Management

BITS-Pilani, Pilani Campus

2021-2023

Date:10-05-2022

Submitted by:

Himani Singh [2021H1540822P]

Dhandu Sai Pratap[2021H1540835P]

Vishva Bhalodiya[2021H1540833P]

Mitta Naveen Kumar Reddy[2021H1540843P]

Chunduri Venkata Sri Sai Phani Sarma [2021H1540840P]

Customer Segmentation and Personality Analysis

Introduction:

Every customer is unique, and each customer journey is unique as well. Understanding a consumer is critical for better tailoring marketing strategies and meeting customer needs. Customer segmentation is a useful method for accomplishing this.

Customer segmentation is the process of dividing a company's customers into groups based on their shared characteristics. The purpose of customer segmentation is to determine how to relate to customers in each segment in order to optimize each customer's value to the company. It has the ability to enable marketers to reach out to each customer in the most efficient manner possible.

Here, we are doing Customer Segmentation and depending on the data we are doing their Personality Analysis. We are doing segmentation with the help of Clustering

This Customer Personality Analysis is a detailed analysis of a company’s ideal customers. It helps a business to modify its product based on its target customers from different types of customer segments. For example, instead of spending money to market a new product to every customer in the company’s database, a company can analyze which customer segment is most likely to buy the product and then market the product only on that particular segment.

Problem Statement: Grouping of customers into various meaningful groups.

Objective:

The objective is to cluster the customers into certain meaningful groups , which helps in targeting and better serving them. We intend to divide heterogeneous customer database into an optimum number of multiple homogeneous groups based on demographics, economical factors, personality characteristics and their spending habits.

Objective addressing through the solution to this problem:

Segmentation of customers is required because it helps business to better understand their target audience.It helps businesses to drive dynamic content and personalization tactics for timelier, relevant and more effective marketing communications. Their marketing campaign budgets and resources can be optimized if they knew their target audience beforehand. This is the area we‘re addressing after making effective clusters from crude customer database.

Dataset:

People

- ID: Customer's Identification Number
- Year_Birth: Customer's Year of Birth
- Education: Customer's Level of Education
- Marital_Status: Customer's marital status
- Income: Customer's annual household income
- Kidhome: Number of children in the customer's household
- Teenhome: Number of teenagers in the customer's household
- Dt_Customer: Customer's enrollment Date with the company
- Recency: Days since the customer last purchased
- Complain: 1 for complaint filed by customer in the last 2 years, 0 otherwise

Products

- MntWines: Amount spent on the wine in last 2 years
- MntFruits: Amount spent on the fruits in last 2 years
- MntMeatProducts: Amount spent on the meat in last 2 years
- MntFishProducts: Amount spent on the fish in last 2 years
- MntSweetProducts: Amount spent on the sweets in last 2 years
- MntGoldProds: Amount spent on the gold in last 2 years

Promotion

- NumDealsPurchases: Number of purchases made with a discount
- AcceptedCmp1: 1 for customer accepting the offer in the 1st campaign , 0 otherwise
- AcceptedCmp2: 1 for customer accepting the offer in the 2nd campaign , 0 otherwise
- AcceptedCmp3: 1 for customer accepting the offer in the 3rd campaign , 0 otherwise
- AcceptedCmp4: 1 for customer accepting the offer in the 4th campaign , 0 otherwise
- AcceptedCmp5: 1 for customer accepting the offer in the 5th campaign , 0 otherwise
- Response: 1 for customers accepting the offer in the last campaign , 0 otherwise.

Place

- NumWebPurchases: Number of purchases made through the company’s website
- NumCatalogPurchases: Number of purchases made using a catalogue
- NumStorePurchases: Number of purchases made directly in stores
- NumWebVisitsMonth: Number of visits to the company's website in the last month.

Approach and Analysis:

Data Loading:

```
#Loading data file.

data = pd.read_excel("C:/Users/HP/Downloads/marketing_campaign.xlsx")
print("Data loaded with", len(data), "rows")
data.head()
```

Data is loaded into the python framework for further analysis of the customer segmentation.

Customer segmentation data is a secondary data collected from

Data Wrangling:

- **Data Cleaning:**

  As part of data cleaning , we performed following steps:

1. Drop unneeded columns - we dropped unnecessary columns such as “ID”, “Z_CostContact”,

   “Z_CostContact”, “Z_Revenue”.

2. There’s a 24 null value available for the Income column in data. As the 24 is very less and not affecting the column , we approached to drop it.
3. Changed the format of the column Dt_Customer to a **DateTime** format.

```
#drop un-needed columns
data= data.drop(["ID","Z_CostContact", "Z_Revenue"], axis=1)
#drop rows with missing values
data.isna().sum()
data= data.dropna()
#change date format
data["Dt_Customer"]= pd.to_datetime(data["Dt_Customer"])
```

- **Preprocessing:**

  In order to work on any kind of data, it is necessary to Preprocess it. Preprocessing includes theses basic steps:

  1. Data Transformation,

  2. Outlier Removal

  3. Data Visualization

  4. Encoding Categorical Features

  5. Feature Scaling

  6. Feature Extraction

1.Data Transformation:

There's two categorical variables : Education and Marital_status. These both variables contain redundant categories. So we approached it to reduce it to a smaller set of values.

```
#examining categorical data
print("Education Values: ", data["Education"].unique())
print("Marital_Status Values:", data["Marital_Status"].unique())
```

```
Education Values:  ['Graduation' 'PhD' 'Master' 'Basic' '2n Cycle']
Marital_Status Values: ['Single' 'Together' 'Married' 'Divorced' 'Widow' 'Alone' 'Absurd' 'YOLO']
```

```
#give each feature a smaller set of values
#reducing the # of category
edu= {"Basic": "Undergraduate", "2n Cycle": "Undergraduate", "Graduation": "Graduate", "Master": "Postgraduate", "PhD": "Postgraduate"}
data["Education"]= data["Education"].replace(edu)

status= {"YOLO": "Single", "Absurd": "Single", "Alone": "Single", "Widow": "Single", "Divorced": "Single", "Together": "Taken", "Married": "Taken"}
data["Marital_Status"]= data["Marital_Status"].replace(status)

#updated categorical variables
print("Education Values: ", data["Education"].unique())
print("Marital_Status Values:", data["Marital_Status"].unique())
```

```
Education Values:  ['Graduate' 'Postgraduate' 'Undergraduate']
Marital_Status Values: ['Single' 'Taken']
```

As per requirement of data, we have performed the feature transformation and created some relevant columns out of it.

Age =current year - Year_Birth

Children_Count= Kidhome + Teenhome

Family_size = Children_Count + Marital_Status

Spendings = MntWines + MntFruits + MntFishProducts + MntMeatProducts + MntSweetProducts + MntGoldProds

Purchases = NumDealsPurchases + NumWebPurchases + NumCatalogPurchases + NumStorePurchases

Accepted_Campaigns = AcceptedCmp1 + AcceptedCmp2 + AcceptedCmp3 + AcceptedCmp4 + AcceptedCmp5

```
#finding customer age
data["Age"]= datetime.now().year-data["Year_Birth"]

#finding family size and number of children
data["Children_Count"]= data["Kidhome"]+data["Teenhome"]
data["Family_Size"]= 1+data["Children_Count"]+data["Marital_Status"].replace({"Taken": 1, "Single": 0})

#finding number of days since person became a customer
data["Customer_For"]= (datetime.now()-data["Dt_Customer"]).dt.days

#finding total spendings of customer
data["Spendings"]= data["MntWines"]+data["MntFruits"]+data["MntFishProducts"]+data["MntMeatProducts"]+data["MntSweetProducts"]+data["MntGoldProds"]

#finding total number of purchases of customer
data["Purchases"]= data["NumDealsPurchases"]+data["NumWebPurchases"]+data["NumCatalogPurchases"]+data["NumStorePurchases"]

#finding total number of accepted campaigns
data["Accepted_Campaigns"]= data["AcceptedCmp1"]+data["AcceptedCmp2"]+data["AcceptedCmp3"]+data["AcceptedCmp4"]+data["AcceptedCmp5"]
```

After the creation of new columns , We dropped the unneeded columns.

```
#dropping un-needed columns
data= data.drop(["Year_Birth","Dt_Customer"],axis=1)
data= data.drop(["Kidhome","Teenhome"],axis=1)
```

```
data.describe().T
```

	count	mean	std	min	25%	50%	75%	max
Income	2216.0	52247.251354	25173.076661	1730.0	35303.0	51381.5	68522.00	666666.0
Recency	2216.0	49.012635	28.948352	0.0	24.0	49.0	74.00	99.0
MntWines	2216.0	305.091606	337.327920	0.0	24.0	174.5	505.00	1493.0
MntFruits	2216.0	26.356047	39.793917	0.0	2.0	8.0	33.00	199.0
MntMeatProducts	2216.0	166.995939	224.283273	0.0	16.0	68.0	232.25	1725.0
MntFishProducts	2216.0	37.637635	54.752082	0.0	3.0	12.0	50.00	259.0
MntSweetProducts	2216.0	27.028881	41.072046	0.0	1.0	8.0	33.00	262.0
MntGoldProds	2216.0	43.965253	51.815414	0.0	9.0	24.5	56.00	321.0
NumDealsPurchases	2216.0	2.323556	1.923716	0.0	1.0	2.0	3.00	15.0
NumWebPurchases	2216.0	4.085289	2.740951	0.0	2.0	4.0	6.00	27.0
NumCatalogPurchases	2216.0	2.671029	2.926734	0.0	0.0	2.0	4.00	28.0
NumStorePurchases	2216.0	5.800993	3.250785	0.0	3.0	5.0	8.00	13.0
NumWebVisitsMonth	2216.0	5.319043	2.425359	0.0	3.0	6.0	7.00	20.0
AcceptedCmp3	2216.0	0.073556	0.261106	0.0	0.0	0.0	0.00	1.0
AcceptedCmp4	2216.0	0.074007	0.261842	0.0	0.0	0.0	0.00	1.0
AcceptedCmp5	2216.0	0.073105	0.260367	0.0	0.0	0.0	0.00	1.0
AcceptedCmp1	2216.0	0.064079	0.244950	0.0	0.0	0.0	0.00	1.0
AcceptedCmp2	2216.0	0.013538	0.115588	0.0	0.0	0.0	0.00	1.0
Complain	2216.0	0.009477	0.096907	0.0	0.0	0.0	0.00	1.0
Response	2216.0	0.150271	0.357417	0.0	0.0	0.0	0.00	1.0
Age	2216.0	53.179603	11.985554	26.0	45.0	52.0	63.00	129.0
Children_Count	2216.0	0.947202	0.749062	0.0	0.0	1.0	1.00	3.0
Family_Size	2216.0	2.592509	0.905722	1.0	2.0	3.0	3.00	5.0
Customer_For	2216.0	3218.006318	232.469034	2706.0	3046.0	3219.0	3392.00	3769.0
Spendings	2216.0	607.075361	602.900476	5.0	69.0	396.5	1048.00	2525.0
Purchases	2216.0	14.880866	7.670957	0.0	8.0	15.0	21.00	44.0
Accepted_Campaigns	2216.0	0.298285	0.679209	0.0	0.0	0.0	0.00	4.0

```
data.describe(include=["O"])
```

	Education	Marital_Status
count	2216	2216
unique	3	2
top	Graduate	Taken
freq	1116	1430

Checked for skewness and kurtosis

```
data.skew()#symmetricity of the data
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_
  """Entry point for launching an IPython kernel.
Income                       6.763487
Recency                      0.001648
MntWines                     1.170720
MntFruits                    2.101658
MntMeatProducts              2.025577
MntFishProducts              1.916369
MntSweetProducts             2.103328
MntGoldProds                 1.839231
NumDealsPurchases            2.415272
NumWebPurchases              1.197037
NumCatalogPurchases          1.881075
NumStorePurchases            0.701826
NumWebVisitsMonth            0.218043
AcceptedCmp3                 3.269397
AcceptedCmp4                 3.256758
AcceptedCmp5                 3.282143
AcceptedCmp1                 3.562482
AcceptedCmp2                 8.424753
Complain                    10.132737
Response                     1.958748
Age                          0.353661
Children_Count               0.408748
Family_Size                  0.086495
Customer_For                 0.006096
Spendings                    0.858055
Purchases                    0.250936
Accepted_Campaigns           2.725357
dtype: float64
```

```
data.kurtosis() # The deviation of the data from normality
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1:
  """Entry point for launching an IPython kernel.
Income                     159.636700
Recency                     -1.199777
MntWines                     0.582741
MntFruits                    4.054082
MntMeatProducts              5.055477
MntFishProducts              3.076476
MntSweetProducts             4.106141
MntGoldProds                 3.156342
NumDealsPurchases            8.974490
NumWebPurchases              4.072137
NumCatalogPurchases          8.067126
NumStorePurchases           -0.626462
NumWebVisitsMonth            1.852577
AcceptedCmp3                 8.696805
AcceptedCmp4                 8.614248
AcceptedCmp5                 8.780388
AcceptedCmp1                10.700936
AcceptedCmp2                69.038772
Complain                   100.763293
Response                     1.838352
Age                          0.734670
Children_Count              -0.259835
Family_Size                 -0.352669
Customer_For                -0.645940
Spendings                   -0.346535
Purchases                   -0.890765
Accepted_Campaigns           8.016841
dtype: float64
```

Encoding categorical features :

A. Creating the dummy variable for both categorical columns.

```
cat_columns = data.select_dtypes(include='object').columns
num_columns = data.select_dtypes(exclude='object').columns

# Display the Frequency of each of the categories in the cateogircal columns
for var in cat_columns:
    print("Name of the Category---------", var)
    print(data[var].value_counts())
```

```
Name of the Category--------- Education
Graduate         1116
Postgraduate      846
Undergraduate     254
Name: Education, dtype: int64
Name of the Category--------- Marital_Status
Taken     1430
Single     786
Name: Marital_Status, dtype: int64
```

B. Cross-Validation of the Categorical Variables.

```
# Cross Tabulations of the categorical variables
for var1 in cat_columns:
    for var2 in cat_columns:
        print(pd.crosstab(data[var1], data[var2], normalize=True))
```

```
Education      Graduate  Postgraduate  Undergraduate
Education
Graduate        0.50361      0.000000       0.000000
Postgraduate    0.00000      0.381769       0.000000
Undergraduate   0.00000      0.000000       0.114621
Marital_Status    Single     Taken
Education
Graduate        0.181408  0.322282
Postgraduate    0.135379  0.246390
Undergraduate   0.037906  0.076715
Education      Graduate  Postgraduate  Undergraduate
Marital_Status
Single          0.181408      0.135379       0.037906
Taken           0.322282      0.246390       0.076715
Marital_Status    Single     Taken
Marital_Status
Single          0.354693  0.000000
Taken           0.000000  0.645307
```

```
from itertools import product
cat1 = data[cat_columns]
cat2 = data[cat_columns]
cat_var_prod = list(product(cat1,cat2, repeat = 1))
#cat_var_prod

import scipy.stats as ss
result = []
for i in cat_var_prod:
    if i[0] != i[1]:
        result.append((i[0],i[1],list(ss.chi2_contingency(pd.crosstab(
                        data[i[0]], data[i[1]])))[1]))
result
chi_test_output = pd.DataFrame(result, columns = ["var1", "var2", "p-value"])
chi_test_output
```

	var1	var2	p-value
0	Education	Marital_Status	0.674668
1	Marital_Status	Education	0.674668

2.Outlier Removal:

```
#examine the boxplots of different features
features= ["Age", "Income", "Customer_For", "Spendings", "Purchases"]

#create plots
fig, axs = plt.subplots(ncols=len(features),figsize=(6*len(features),8))
for i in range(len(features)):
  sns.boxplot(data=data[features[i]],
              showfliers=True,
              ax=axs[i],
              palette=[palette[i]]
              ).set(xlabel=features[i])
  sns.despine()
```

We can see some outliers in the age, income and purchases. Those in the age and income will be removed as they are very few data points. The ones in the purchases will be kept as the range of values is considerable.

3.Data Visualization:

Histogram plot of each variable.

```
cat_columns = data.select_dtypes(include='object').columns
num_columns = data.select_dtypes(exclude='object').columns
for var in num_columns:
    plt.figure()
    sns.histplot(data = data, x = var, kde = True, color='teal', alpha=0.6)
```

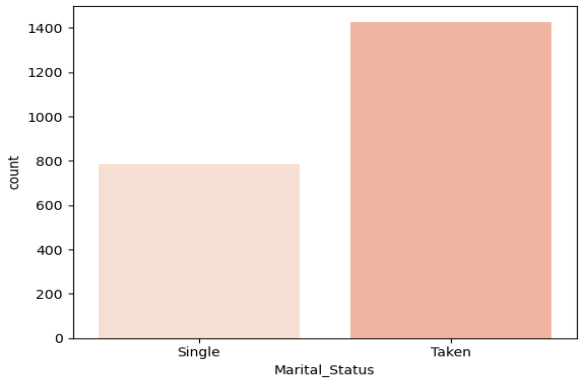
Data shown in Histograms are not showing any abnormality in them.

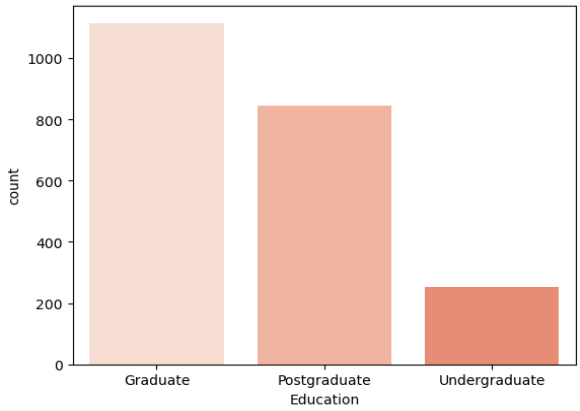
Histogram with Density Plot

```
# Numerical Columns - Univariate - Histogram with Density Plot
for var in num_columns:
    plt.figure()
    sns.kdeplot(data = data, x = var, color='teal', alpha=0.6)
```

Density Plots are also not showing any abnormality in them.

Categorical Column-Simple Frequency Charts





Simple Frequency Charts are also not showing any abnormality in them.

4. Encoding Categorical Features

The 2 categorial features we have can be encoded as follows:

Education: using a label encoder since it's considered ordinal

Marital_Status: using one hot encoding since it's considered nominal

5. Feature Scaling

Assuming that our features follows normal distribution, we have used the standard scaler.

```
#scale features
scaler= StandardScaler()
data = pd.DataFrame(scaler.fit_transform(data),columns = data.columns)

#check
data.head()
```

6. Features Extraction

A . Correlation matrix.

```
cor_data=data.corr()
cor_data
```

```
# Convert correlation matrix to 1-D Series and sort
sorted_mat = cor_data.unstack().sort_values()
sorted_mat[(abs(sorted_mat)>0.8) & (abs(sorted_mat)<1)]
```

```
Purchases          NumStorePurchases    0.822210
NumStorePurchases  Purchases            0.822210
MntMeatProducts    Spendings            0.845543
Spendings          MntMeatProducts      0.845543
Family_Size        Children_Count       0.849574
Children_Count     Family_Size          0.849574
Spendings          MntWines             0.892996
MntWines           Spendings            0.892996
dtype: float64
```

A lot of features have high correlation values. PCA will be used to reduce the dimensions while keeping 95% of the variations.

B. PCA

```
#95% variations
pca = PCA(n_components = 0.99)
pca.fit(data)
reduced_data = pd.DataFrame(pca.transform(data))

print("Current number of features= ",len(reduced_data.columns))
```

```
Current number of features=  24
```

By keeping 95% of the variations, the number of features dropped from 32 to 20, but we want to reduce that even more to reduce complexity, so we will compromise more variations.

```
#75% variations
pca = PCA(n_components = 0.75)
pca.fit(data)
reduced_data = pd.DataFrame(pca.transform(data))
print("Current number of features= ",len(reduced_data.columns))
```

```
Current number of features=  10
```

Data Clustering:

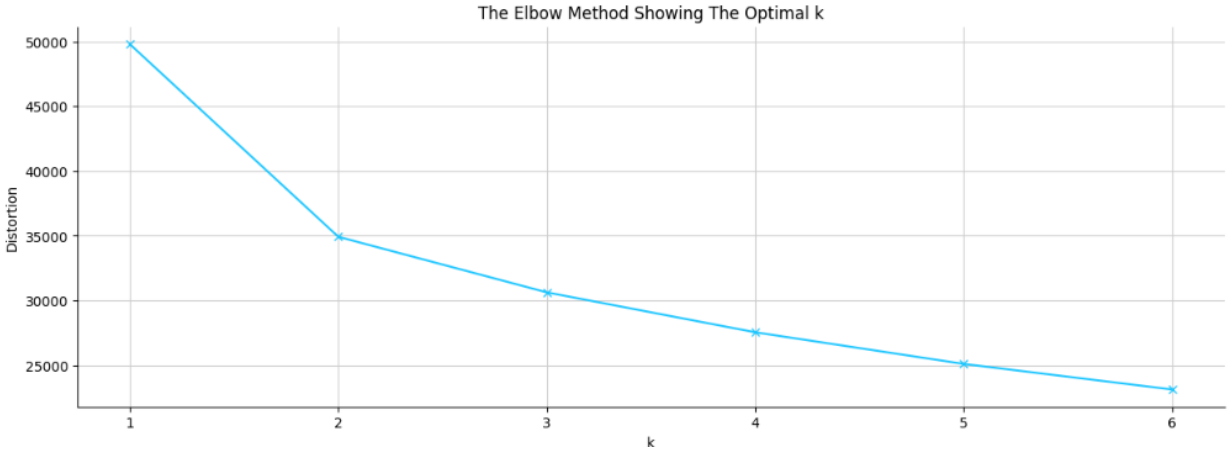
1.K-Means Clustering

Optimal number of clusters.

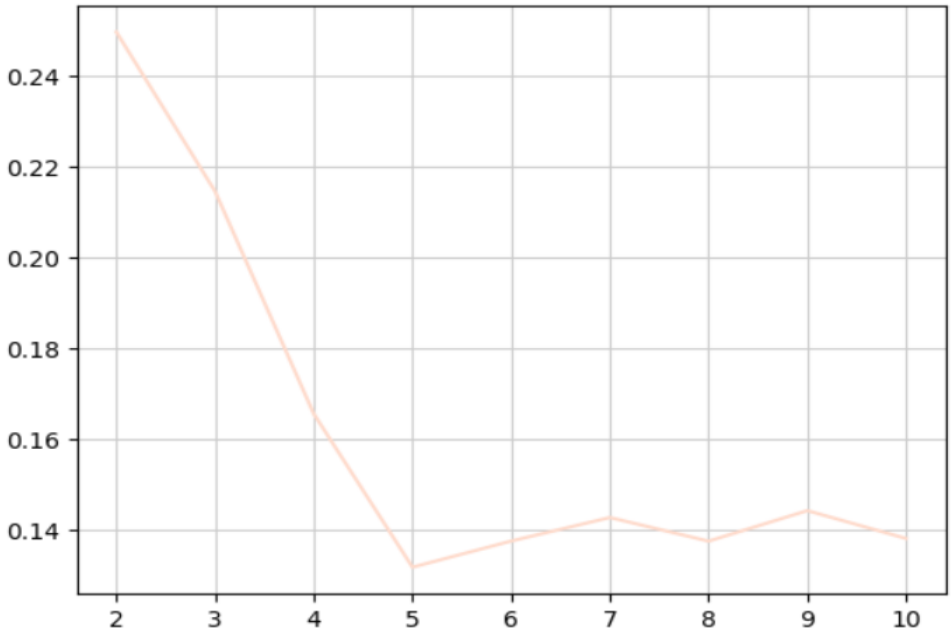
Firstly , we need to have a good sense of how many clusters are in our dataset. To determine this, we will use the elbow method.

```
#calculate distortions for different values of k (number of clusters)
distortions = []
K = range(1,7)
for k in K:
    kmeanModel = KMeans(n_clusters=k)
    kmeanModel.fit(reduced_data)
    distortions.append(kmeanModel.inertia_)
```

```
#plot elbow graph
plt.figure(figsize=(15,5))
plt.rcParams.update({'axes.grid': True})
plt.plot(K, distortions, 'bx-', color=palette[-1])
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method Showing The Optimal k')
sns.despine()
plt.show()
```



From the above graph we can infer that at k=2, the graph reaches an optimum minimum value (Loss function is reaching a minimum value). Even though the within-cluster distance decreases after 2, we would be doing more computations. Which is just analogous to the law of diminishing returns. Therefore, we choose a value of 2 as the optimum number of clusters. The reason it is named the elbow method is that the optimum number of clusters would represent an elbow joint.



The dissimilarity would not be defined for a single cluster, thus, minimum number of clusters should be 2.

1.The silhouette score falls within the range [-1, 1].

2.The silhouette score of 1 means that the clusters are very dense and nicely separated. The score of 0 means that clusters are overlapping. The score of less than 0 means that data belonging to clusters may be wrong/incorrect.

3. The silhouette plots can be used to select the most optimal value of the K (no. of cluster) in K-means clustering.

4. Here we got the silhouette score is maximum at 2, inferring 2 as the optimal number of clusters.

Clustering Algorithm:

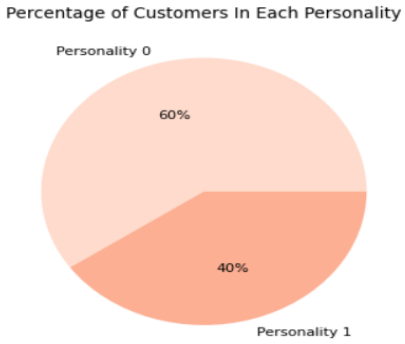
Clustering will be performed using the K-means algorithm, which assumes the clusters are somehow spherical (globular) in shape. Further there are a very low number of outliers (which have been handled).The data is also densely populated data. These factors support the process of choosing the K-means algorithm for clustering.

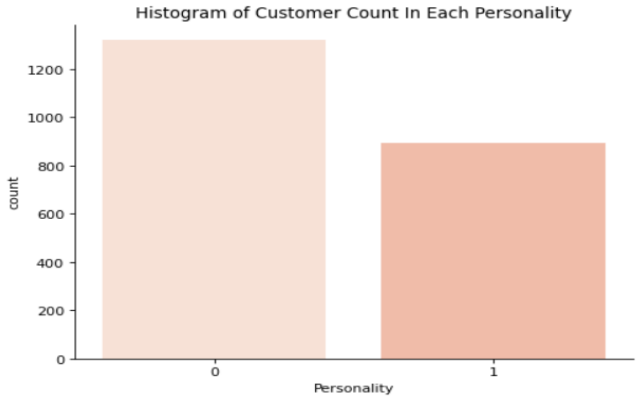
A lower within-cluster variation is an indicator of a good compactness (i.e., a good clustering). The different indices for evaluating the compactness of clusters are based on distance measures such as the cluster-wise within average/median distances between observations.

	Education	Income	Recency	MntWines	MntFruits	MntMeatProducts	MntFishProducts	MntSweetProducts	MntGoldProds	NumDealsPurchases	...
0	-0.893586	0.287105	0.310353	0.977660	1.552041	1.690293	2.453472	1.483713	0.852576	0.351030	...
1	-0.893586	-0.260882	-0.380813	-0.872618	-0.637461	-0.718230	-0.651004	-0.634019	-0.733642	-0.168701	...
2	-0.893586	0.913196	-0.795514	0.357935	0.570540	-0.178542	1.339513	-0.147184	-0.037254	-0.688432	...
3	-0.893586	-1.176114	-0.795514	-0.872618	-0.561961	-0.655787	-0.504911	-0.585335	-0.752987	-0.168701	...
4	0.571657	0.294307	1.554453	-0.392257	0.419540	-0.218684	0.152508	-0.001133	-0.559545	1.390492	...

Algorithm Evaluation:

Since we have chosen the number of clusters to be 2 , Whole data is grouped into 2 clusters. Analysis of each individual cluster helps in better understanding the customer characteristics and further enhances the process of targeting the customers through various campaigns.





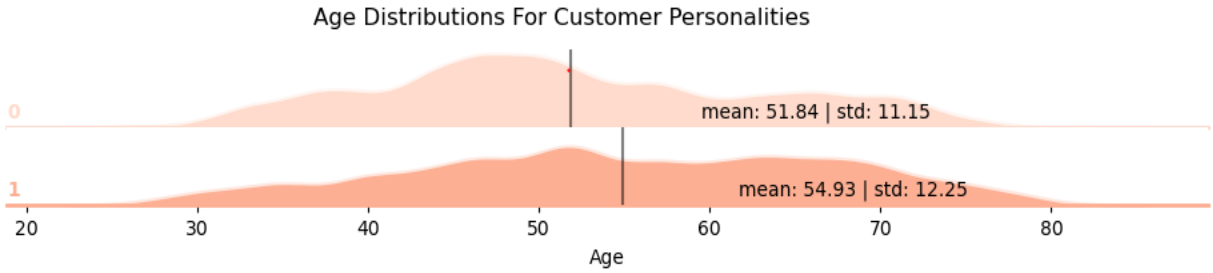
From the plots we can see that around 60% of customers fall under personality 0 and rest in personality 1.We can note that there is no sharp variation between the proportion of customers under both personalities.

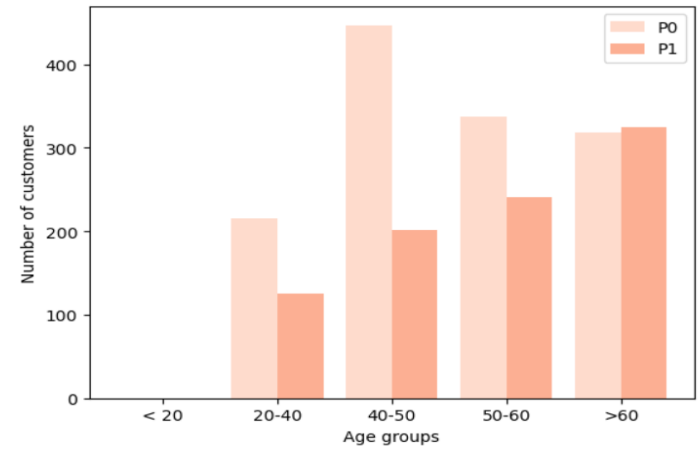
Data Analysis:

Inorder to better understand the variation between two clusters aka two personalities, we can further perform data analysis at a more granular level, i.e analysis of each feature.

Demographics

Age : To study the demographics of the different personality groups, we'll first take a look on their age distributions





Inference:

Personality 0:

- Has a wide and nearly even distribution, which shows a diverse variation in age groups.
- From the bar graph we can infer that 40-50 & 50-60 age groups constitute this kind of personality.

Personality 1:

- mainly consists of older people(>60 age) compared to other personalities

To validate this, let's calculate some stats.

```
#personality 0 age stats
d= dataCopy[dataCopy["Personality"]==0]
perc= (len(d[d["Age"]<55])/len(d))*100
print("Percentage of customer below 55 in personality 0= {:.2f}% ".format(perc))

#personality 1 age stats
d= dataCopy[dataCopy["Personality"]==1]
perc= (len(d[d["Age"]>50])/len(d))*100
print("Percentage of customer above 50 in personality 1= {:.2f}% ".format(perc))
```

```
Percentage of customer below 55 in personality 0= 49.44%
Percentage of customer above 50 in personality 1= 49.70%
```

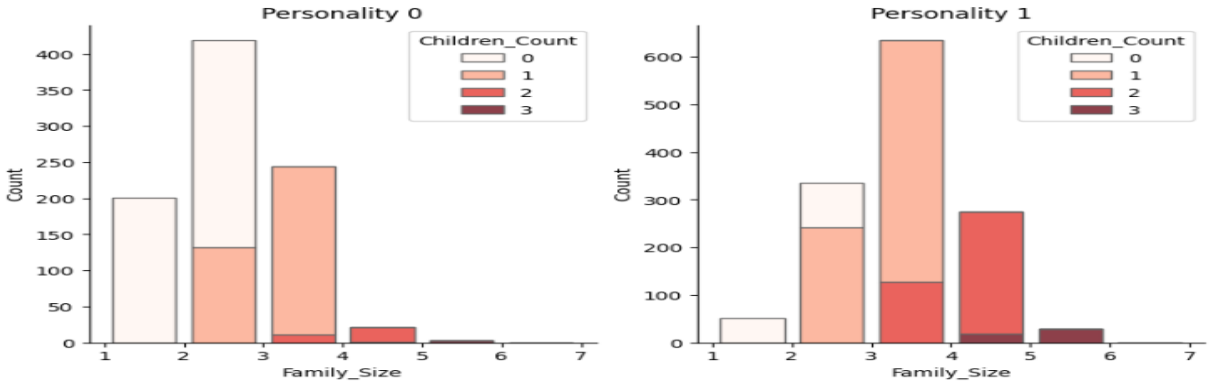
Comparative Study :

Age Group	Personality 0	Personality 1	Difference
< 20	0	0	0%
'20-40'	16.36	14.109	1.8%
,'40-50'	33.88	22.50	11 %
'50-60’	25.625	26.987	1.2%
'>60'	24.109	36.39	12%

We can see that both personalities differ at two major points , Personality 0 is more dominated by middle aged people, whereas Personality 1 is dominated by older people.

Family Breakdown

To analyze based on family shape, we'll consider 2 aspects; the family size and the number of children.



Inference:

Personality 0:

- The majority of customers have 1-2 kids and a partner.
- A considerable percentage are single parents.

Personality 1:

- The vast majority have 0 kids, followed by 1 kid.
- A considerable percentage are living alone.
- A considerable percentage are single parents.

Time for some stats.

```
#personality 0 kids stats
d= dataCopy[dataCopy["Personality"]==0]
perc= ((len(d[d["Children_Count"]==1]) + len(d[d["Children_Count"]==2]))/len(d))*100
print("Percentage of customers having 1-2 kids in personality 0 = {:.2f}% ".format(perc))

#personality 1 kids stats
d= dataCopy[dataCopy["Personality"]==1]
perc= ((len(d[d["Children_Count"]==0]))/len(d))*100
print("Percentage of customers having 0 kids in personality 1 = {:.2f}% ".format(perc))
perc= ((len(d[d["Children_Count"]==1]))/len(d))*100
print("Percentage of customers having 1 kids in personality 1 = {:.2f}% ".format(perc))
perc= ((len(d[d["Family_Size"]==1]))/len(d))*100
print("Percentage of customers living alone personality 1= {:.2f}% ".format(perc))
```

- Percentage of customers having 1-2 kids in personality 0 = 44.59%
- Percentage of customers having 0 kids in personality 1 = 10.88%
- Percentage of customers having 1 kids in personality 1 = 56.57%
- Percentage of customers living alone personality 1= 3.85%

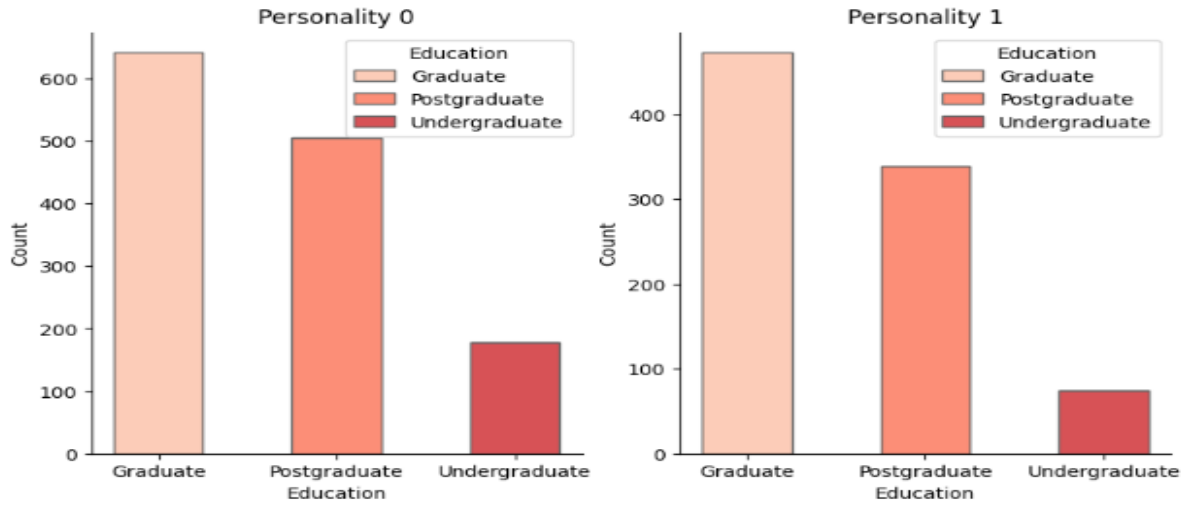
```
#single parents in personality 0
d= dataCopy[dataCopy["Personality"]==0]
perc= (len(d[d["Marital_Status"]=="Single"])/len(dataCopy[dataCopy["Marital_Status"]=="Single"]))*100
print("Percentage of single parents belonging to personality 0= {:.2f}% ".format(perc))

#single parents in personality 1
d= dataCopy[dataCopy["Personality"]==1]
perc= (len(d[d["Marital_Status"]=="Single"])/len(dataCopy[dataCopy["Marital_Status"]=="Single"]))*100
print("Percentage of single parents belonging to personality 1= {:.2f}% ".format(perc))
```

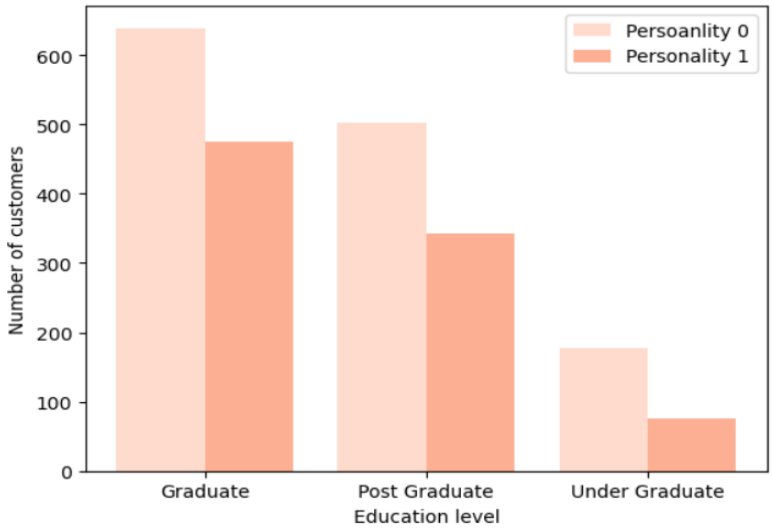
- Percentage of single parents belonging to personality 0= 44.01%
- Percentage of single parents belonging to personality 1= 55.99%

Education

Next, we will examine the educational level of the different customer personalities using a histogram.




```
[48.44579227 38.13495072 13.41925701]
[53.30347144 38.29787234  8.39865622]
```



Age Group	Personality 0	Personality 1	Difference
Graduate	48.44	53.303	4.86%
Post graduate	38.13	38.29	0.16%
Undergraduate	13.41	8.39	5.02%

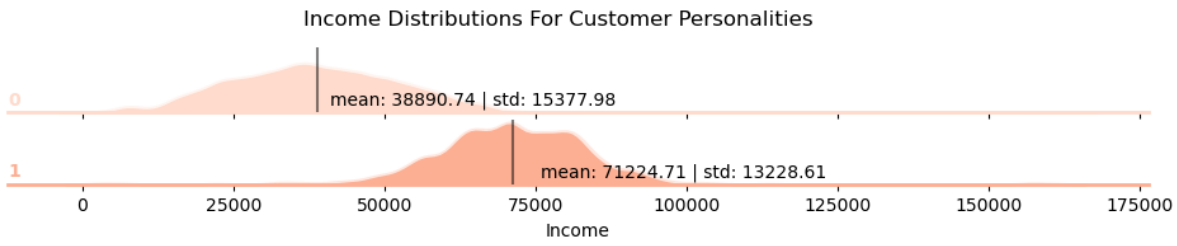
We can see that both personalities are almost equal in the case of Post graduation. Personality 1 is dominating in Graduation whereas Personality 0 is dominating in Undergraduation.

Inference:

- Personalities 0 & 1 have very similar education distributions.
- The education distributions don't really help in differentiating between the different segments.

Income

Now we'll visualize the income distributions of the different segments.

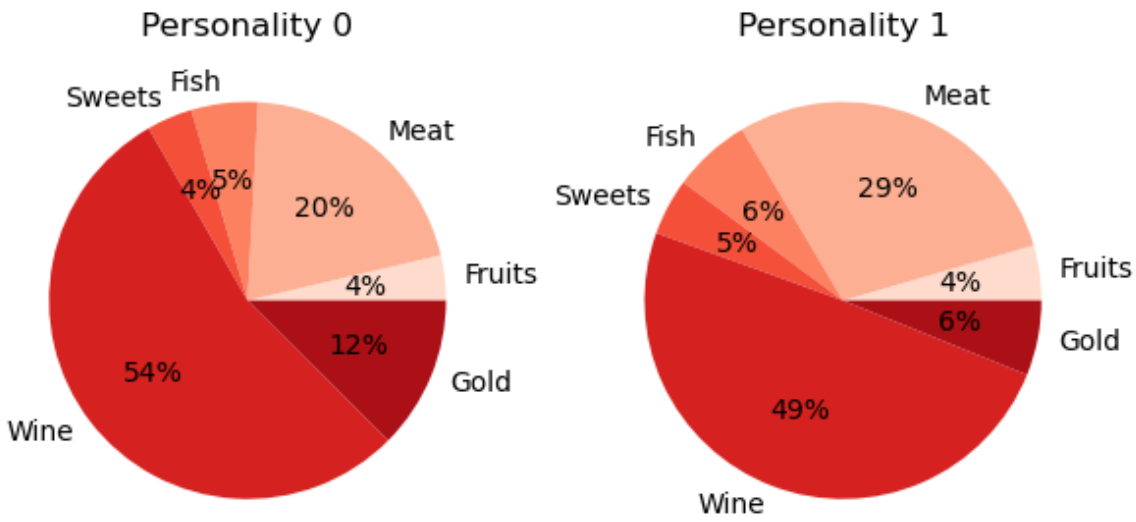


Inference:

- Personality 0 : Low average income group
- Personality 1 : High average income group

Business Related
We will focus here on the business-related aspect of the customer's personality to get more insights.

Products Breakdown
We will use a simple pie chart to visualize the types of products frequently bought by the customers.



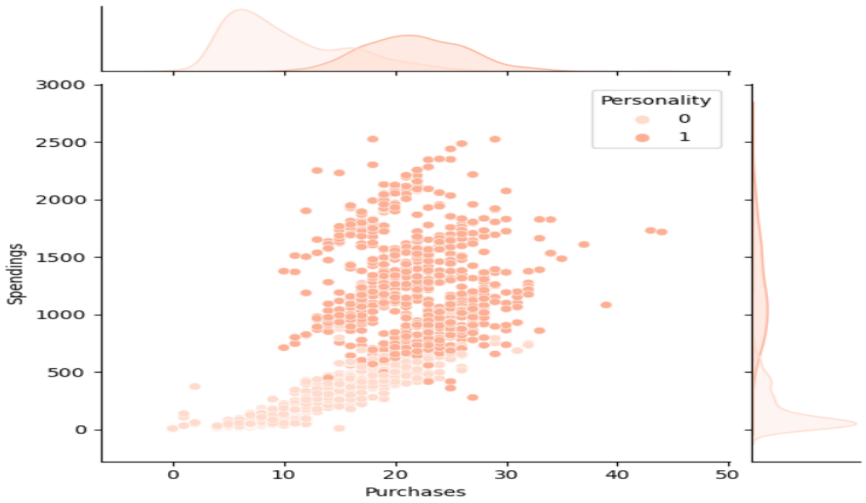
From the above plots, we can see that:

The percentage of spendings on primary goods such as fruits, fish and sweets is very close across all segments. Irrespective of wine (which is a common spending), we can infer that :

- Personality 0 spends relatively more on gold.
- Personality 1 spends relatively more on meat.

That means Personality 0 can be targeted for more premium products.

Spendings
To examine the value added by the customers, we will plot the purchases against the spendings for each segment.

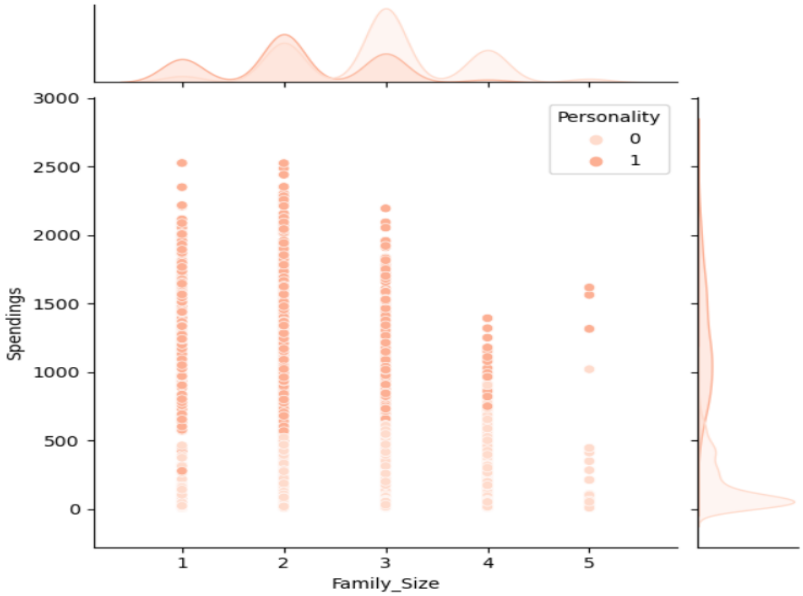


Inference:

From the above plots we can infer that , for Personality 0,more points are concentrated on the lower half of the cartesian plane , supporting the fact of low purchases and low spendings.

- Personality 0: Few purchases (ranges between 0-20), low spendings (0-500$).
- Personality 1: Relatively higher spendings(500-2000$), meaning that they buy more expensive products.

We will also plot the family size against the spendings for each segment.

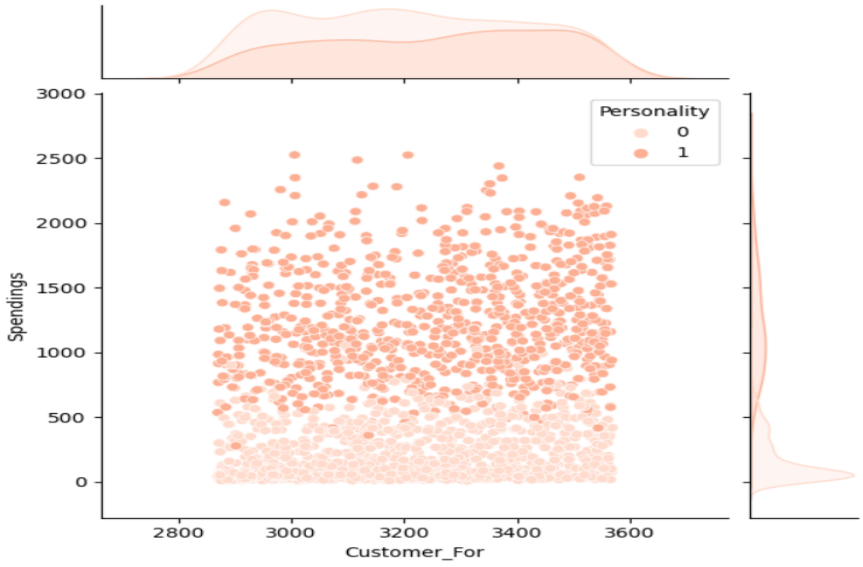


From the above plots we can infer that , for Personality 0,more points are concentrated on the lower half of the cartesian plane , supporting the fact that small families have low spendings.

Inference:

- Personality 0: Family_size 3 has more spendings.
- Personality 1: Family_size 1 & 2 has more spendings

We will now plot the customer_for against the spendings for each segment.



From the above plots we can infer that , for Personality 0,more points are concentrated on the lower half of the cartesian plane, supporting the fact that people with 0 personality have low spendings.

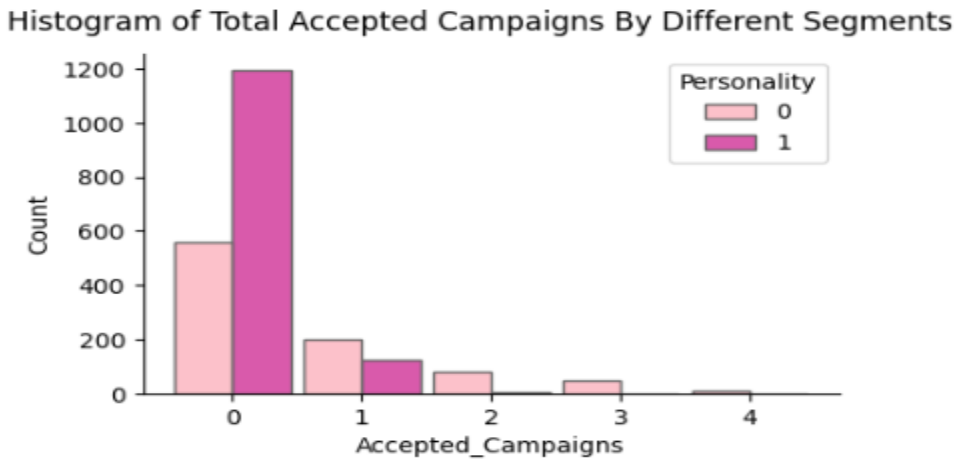
Inference:

- Personality 0: Spent is almost constant.
- Personality 1: Spent more as they stay for more time.

Behavioral:-

Response To Campaigns:-

Finally, we will examine the customer's reactions to the company's campaigns in terms of how many campaigns have they accepted.



Inference:

- Personalities 0 & 1: Majority accepted 0 campaigns. Very few accepted only 1.
- Personality 1: Some accepted 3 or 4.

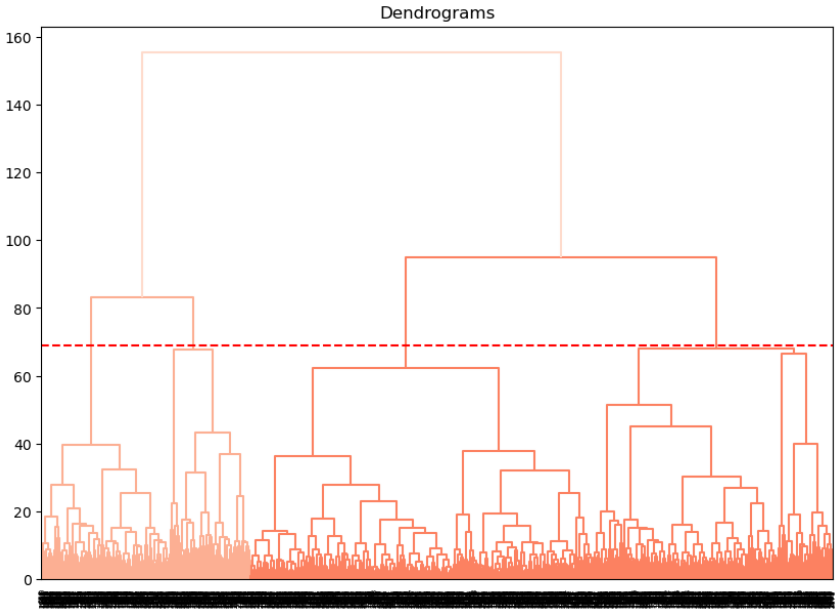
Summary:

Personality 0	**Personality 1**
covers 60% of all people	covers 40% of the people
34.6% people are in the Age group of 40-50	38% of people are Senior citizens.
27% have 1-2 kids	28% have 1-2 kids
9% are single parents	80% are single parents
48% are Graduate	55.5% are Graduate
Lower income and spending	Higher Income and Spending

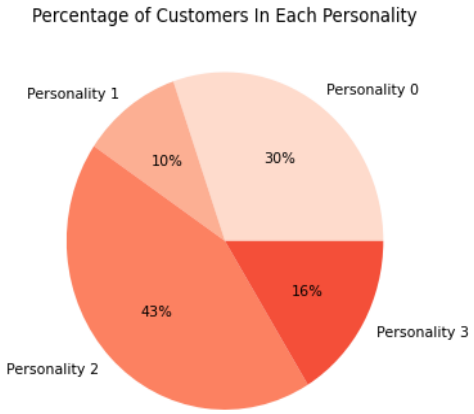
Personality 0 cluster is covering 60% of total no of persons from the age group of 40-50 with only 9% single parents which is highly less in total proportion of single parents, and differentiable lower income and spending as compared to personality 1 cluster which shows that this cluster are distinct from each other.

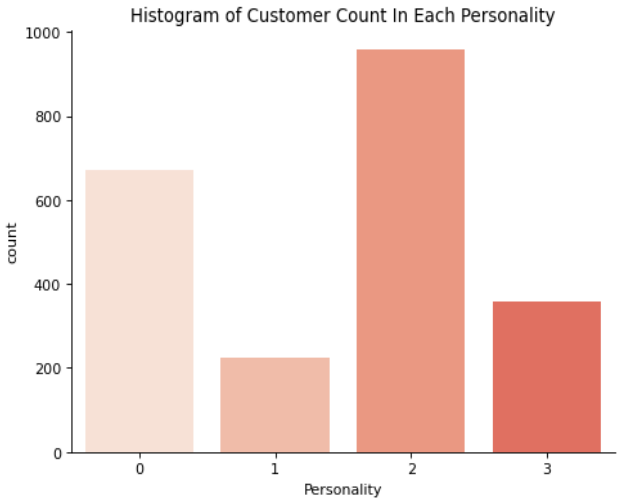
2.Agglomerative Hierarchical Clustering

Agglomerative is a bottom up approach which starts with many small clusters and merges into bigger clusters. There's no assumption about the number of clusters. Dendrogram is a visual presentation of the history of grouping.The number of clusters is the vertical line passing through the horizontal line.



Algorithm evolution





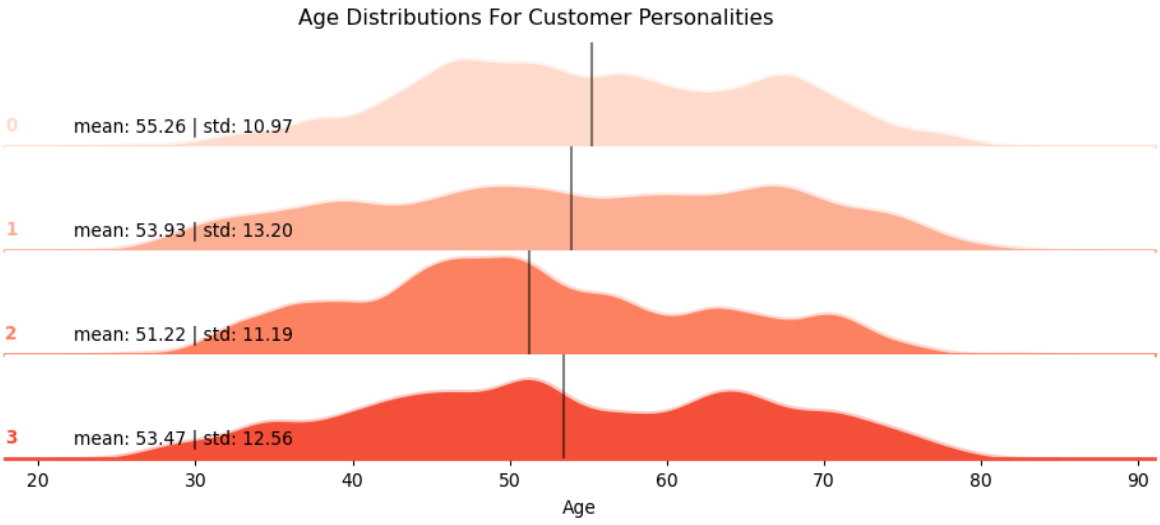
From the above , it is quite clear that we have chosen 4 clusters and distribution of data into 4 clusters is as shown in the above pie chart. Personality 2 has the highest proportion (43%), whereas Personality 1 has least with 10%.

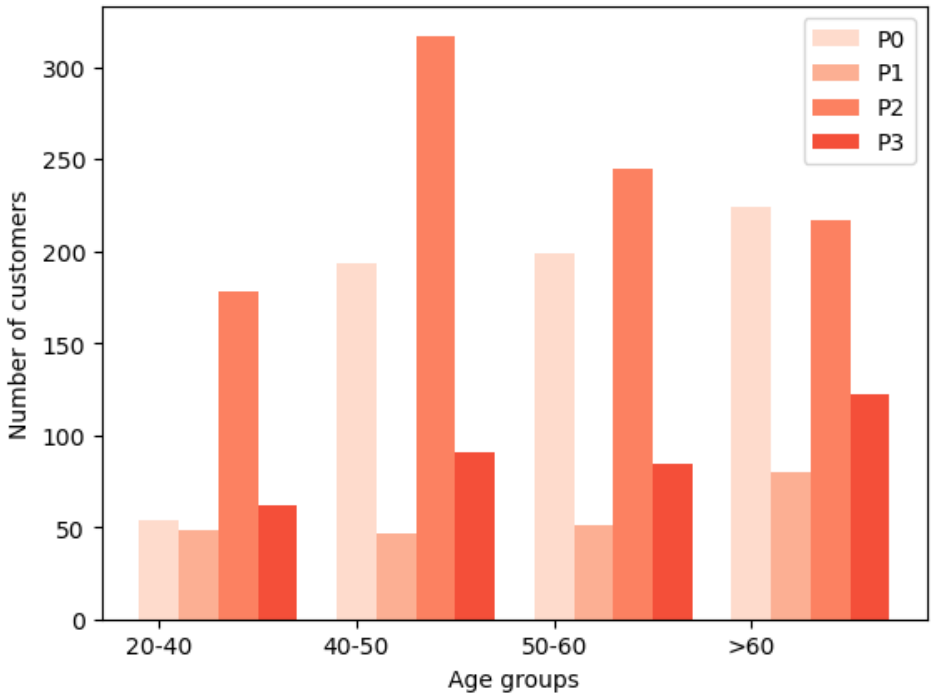
Data Analysis

To do characteristic analysis of each cluster, we'll study their demographic and behavior against each cluster.

Demographic Analysis:

Age: Age distribution of each cluster.





Inference:

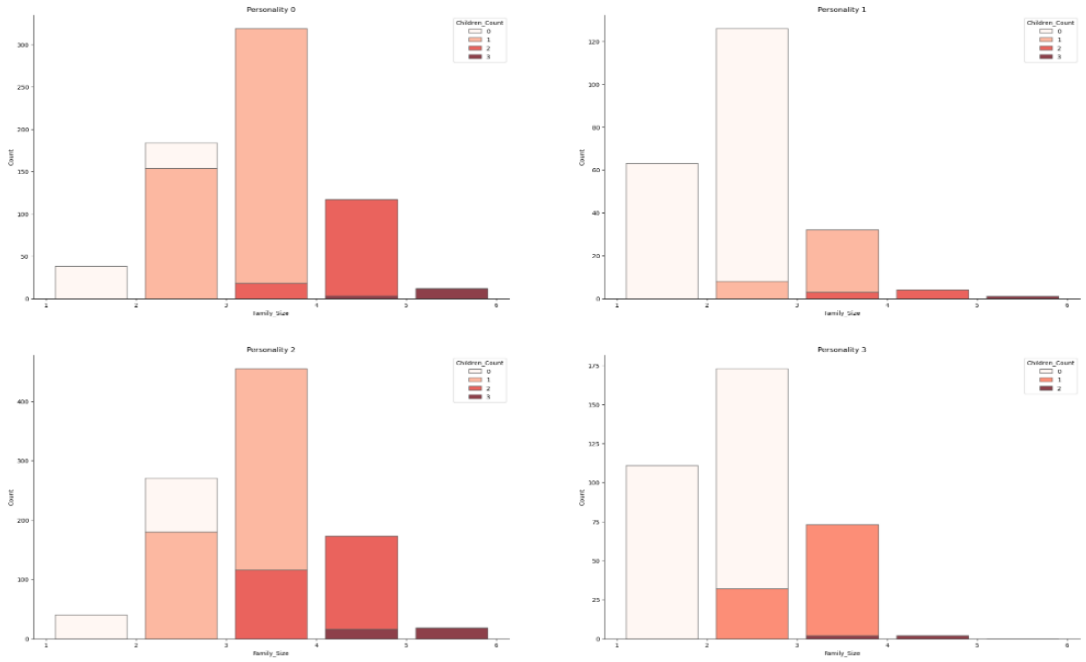
Comparative Study : (Percentages of Age wise population within personality types)

Age Group	Personality 0	Personality 1	Personality 2	Personality 3
< 20	0	0	0	0
'20-40'	8	21.23	18.59979101	17.27019499
'40-50'	28.8	20.79	33.12434692	25.34818942
'50-60’	29.7	22.56	25.60083595	23.39832869
'>60'	33	35.39	22.67502612	33.98328691
Comments	Overall it has 30% of customers. More Middle aged population	Overall it has only 10% of customers. More Old aged population	Overall it has 43% of customers. More middle aged population	Overall it has 16% of customers. More old aged population

When the number of clusters has increased to 4, the difference of age distribution among various clusters is not that significant. All 4 clusters are having mean age around 53-55 with more or less 10-12 standard deviation. This might indicate that targeting customers based on their age among clusters may not result in a fruitful manner.

Family Breakdown :

Family size comparison [family size and no. of children included].



Inference:

Personality 0:

- Majority of customers have 1 kid.
- Has good proportion of customers have family size 3.
- It has very less proportion with family_size 2 and no kids.
- Has good proportion of single parents.

Personality 1:

- Has good proportion of singles and customers have family size 2 with no kids.
- Single parents are very less.

Personality 2:

- Majority of customers have 1 kid.
- Has good proportion of customers have family size 3.
- Has good proportion of single parents with 2 kids.

Personality 3:

- Has good proportion of singles and customers have family size 2 with no kids.

Personality 1 Family statistics:-

- Percentage of customers having 1 kid in personality 0 = 67.91%
- Percentage of customers having family size 3 in personality 0= 47.61%
- Percentage of customers having family_size 2 and no kids in personality 0= 4.48%
- Percentage of single parents in personality 0= 26.12%

Personality 2 Family statistics:-

- Percentage of singles and customers have family size 2 with no kids in personality 1= 56.19%
- Percentage of single parents in personality 1= 4.87%

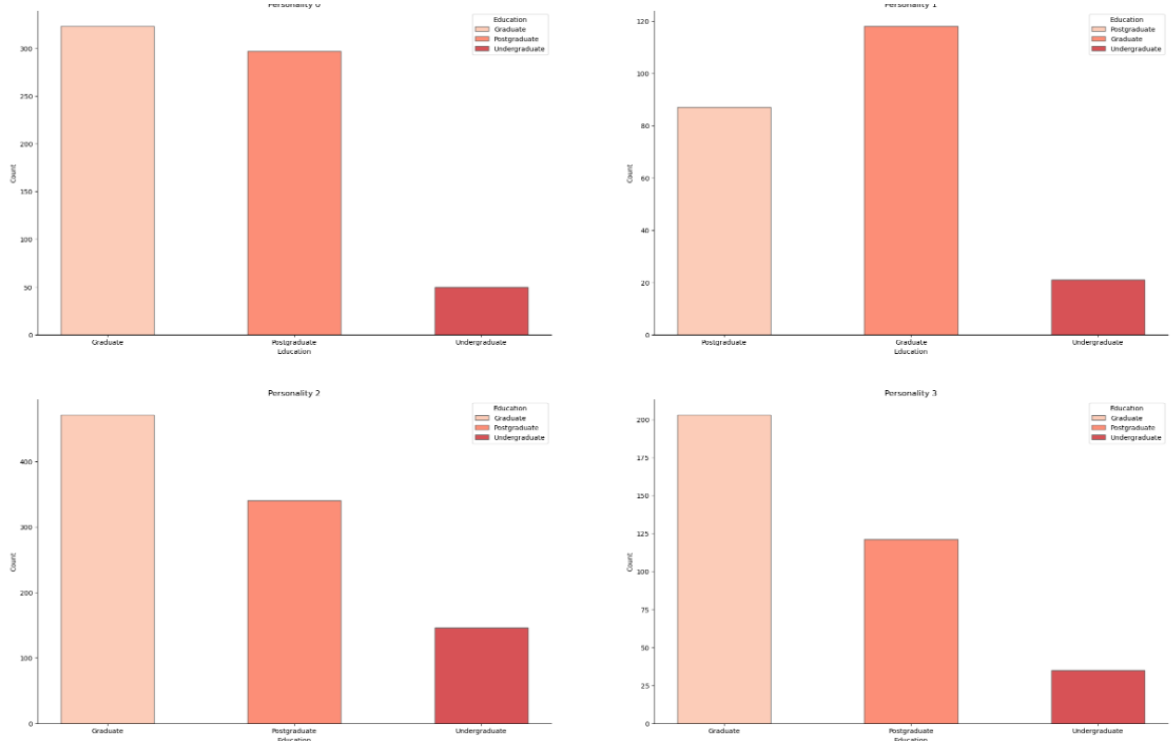
Personality 3 Family statistics:-

- Percentage of customers having 1 kid in personality 3 = 54.23%
- Percentage of customers having family size 3 in personality 3 = 47.54%
- Percentage of customers having family_size 2 and no kids in personality 3 = 9.51%
- Percentage of single parents in personality 3 = 32.60%

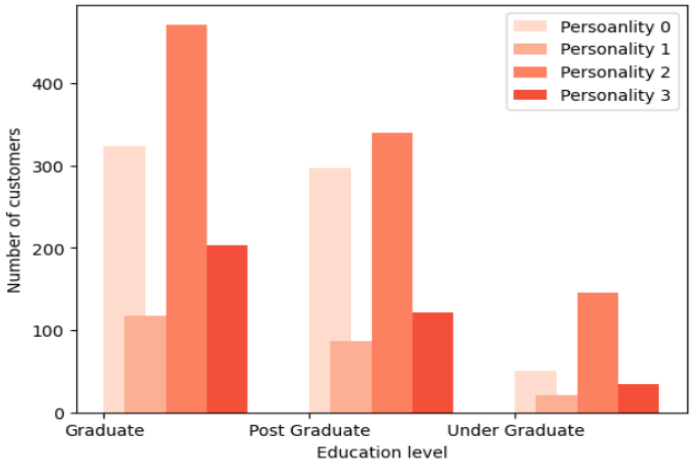
Personality 4 Family statistics:-

- Percentage of singles and customers have family size 2 with no kids in personality 3= 66.57%
- Percentage of single parents in personality 3= 9.47%

Education: The educational level of the different customer personalities using a histogram.

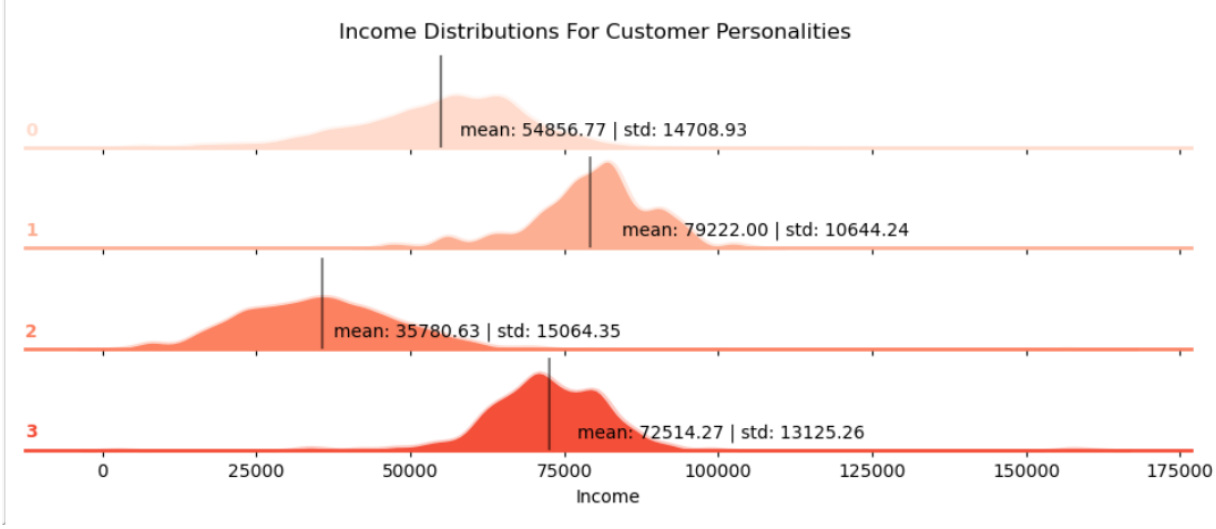


We can note that Graduates are more in Personality 1 than other three. In the rest of the personalities we can see the similar distribution of customers (Higher graduates followed by postgraduate and then by undergraduate) in Personalities 0,2,4.



Inference: Most of the graduates are in personality 2 cluster, personality 1 cluster has lower education level as compared to other clusters.

Income: the income distributions of the different segments.



Inference:

- Personality 0: Average income
- Personality 1 & 3: High Income
- Personality 2: Low Income.

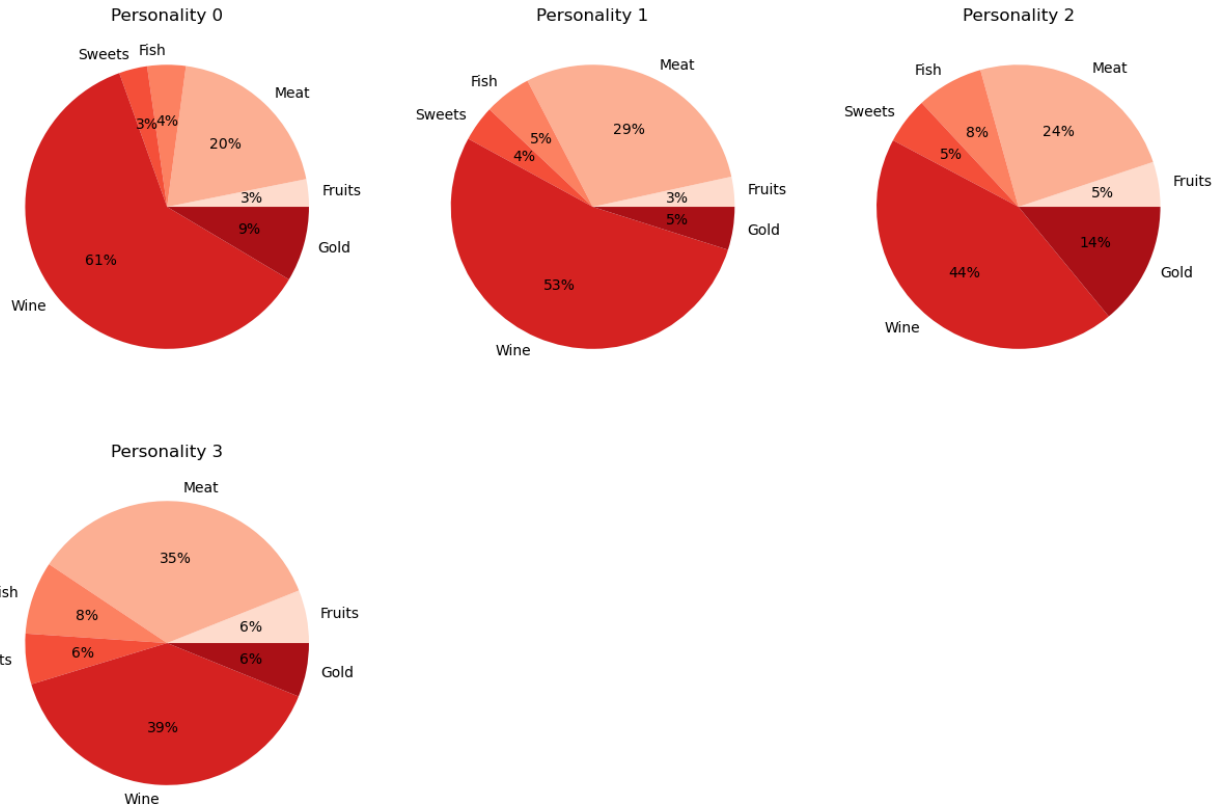
That implies that , we can target Personality 1, 3 for premium / luxury products whereas we can target Personality 2 for low cost products.

Business Related:

The business-related aspect of the customer's personality to get more insights.

Product Breakdown:

Personality comparison based on the types of product purchased.



Inference:

From the above plots, we can see that,

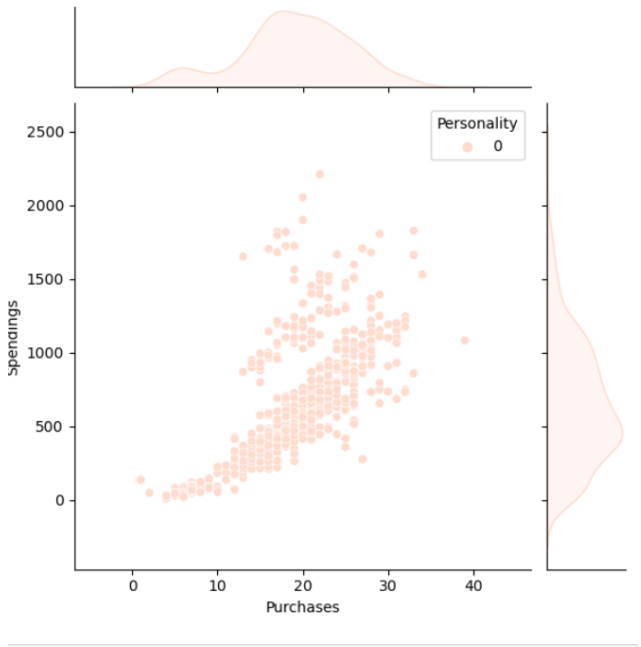
Wine is the most purchased product irrespective of Personality types.

- The percentage of spendings on primary goods such as fruits, fish and sweets is very close across all segments.
- Personality 2 spends relatively more on gold.
- Personality 1 & 3 spends relatively more on meat.
- Personality 0 spends relatively more on Wine.

Spendings :

1.To examine the value added by the customers, we will plot the Spendings against each Personality type.

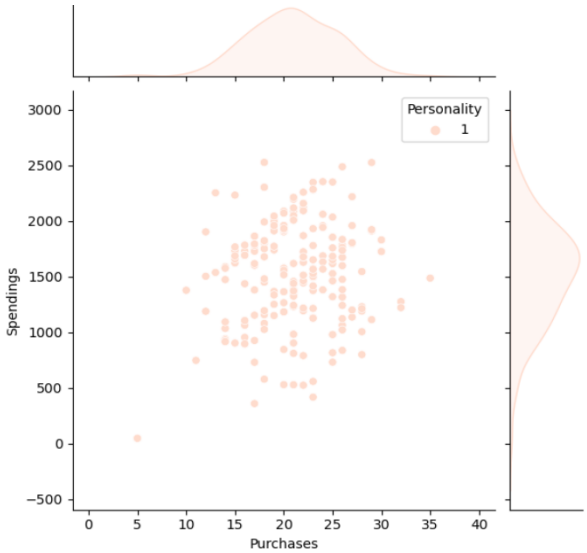
Personality 0 :



Inference:

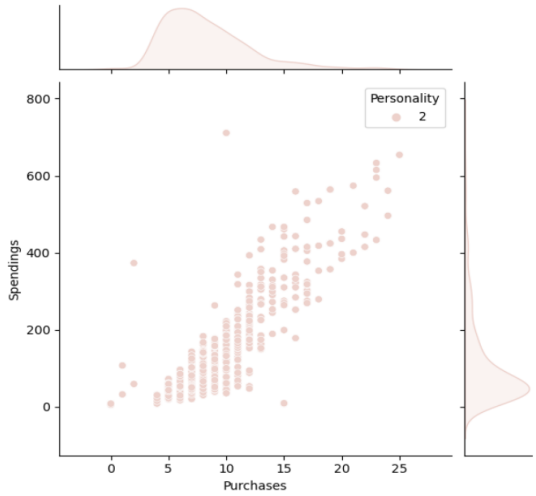
Personality 0: Average number of purchases, Average value of spendings(ranging from 0-1000$) .

Personality 1:



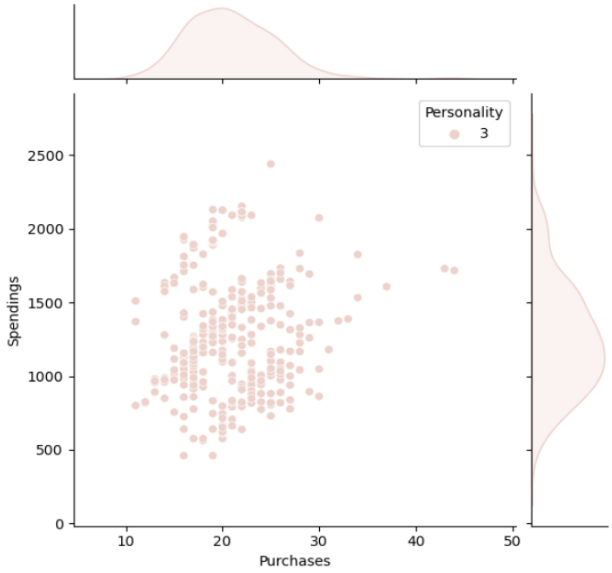
Inference: High number of purchases, Also high value spendings

Personality 2:



Inference: Average number of purchases, Also Low value spendings.

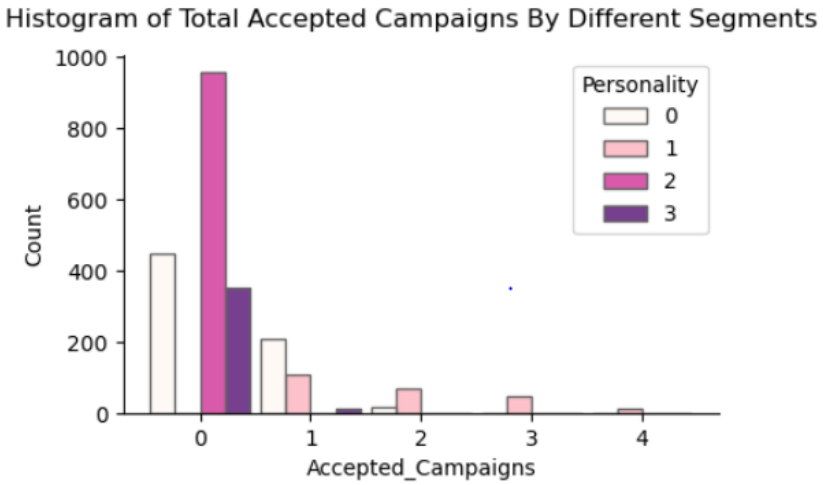
Personality 3:



Inference: Lower number of purchases but comparatively high value of spendings.

Behavioral:

Response To Campaigns : The customer's reactions to the company's campaigns in terms of how many campaigns have they accepted.



Inference:

- Personalities 0 & 1: Majority accepted 0 campaigns. Very few accepted only 1.
- Personality 1: Some accepted 3 or 4.
- Personality 3 : Majority accepted none, while few accepted 2.

Consolidated Information :

Personality 0	**Personality 1**	**Personality 2**	**Personality 3**
Overall it has 30% of customers.	Overall it has only 10% of customers.	Overall it has 43% of customers.	Overall it has 16% of customers.
More Middle aged population	More Old aged population	More middle aged population	More old aged population
Majority have 1 kid.	Majority have no kids.	Majority have 1 kid.	Has a good proportion of singles.
Average family size is 3	Average family size is 2	Average family size is 3	Average family size is 2
Average Income	Higher Income	Low income	High Income

Inferences :

- From the above table , we can infer that Personality 0 and 2 have average family size 3 with 1 kid, so they can be targeted for kid/child products and household products.
- Personality 1,3 are high earning groups with relatively lesser family size with no kids. They can be targeted for high end products and luxurious products and holiday packages (considering the fact that these groups are old age dominated).
- We also need to keep in mind that Personality 0, 2 are middle aged dominated and lower to average income groups, so they might be willing to react to campaigns and loyalty programs.
- It is quite evident that there is a lot of untapped opportunity in targeting Personality 1,2,3 for various campaigns (Especially Personality 2 , since it represents 43% of total customer base). Hence efforts in that direction is quite necessary.

3.Clustering using DBSCAN Clustering:

DBSCAN is Density based clustering algorithm with noise. Finds core samples of high density and expands clusters from them. Good for data which contains clusters of similar density.

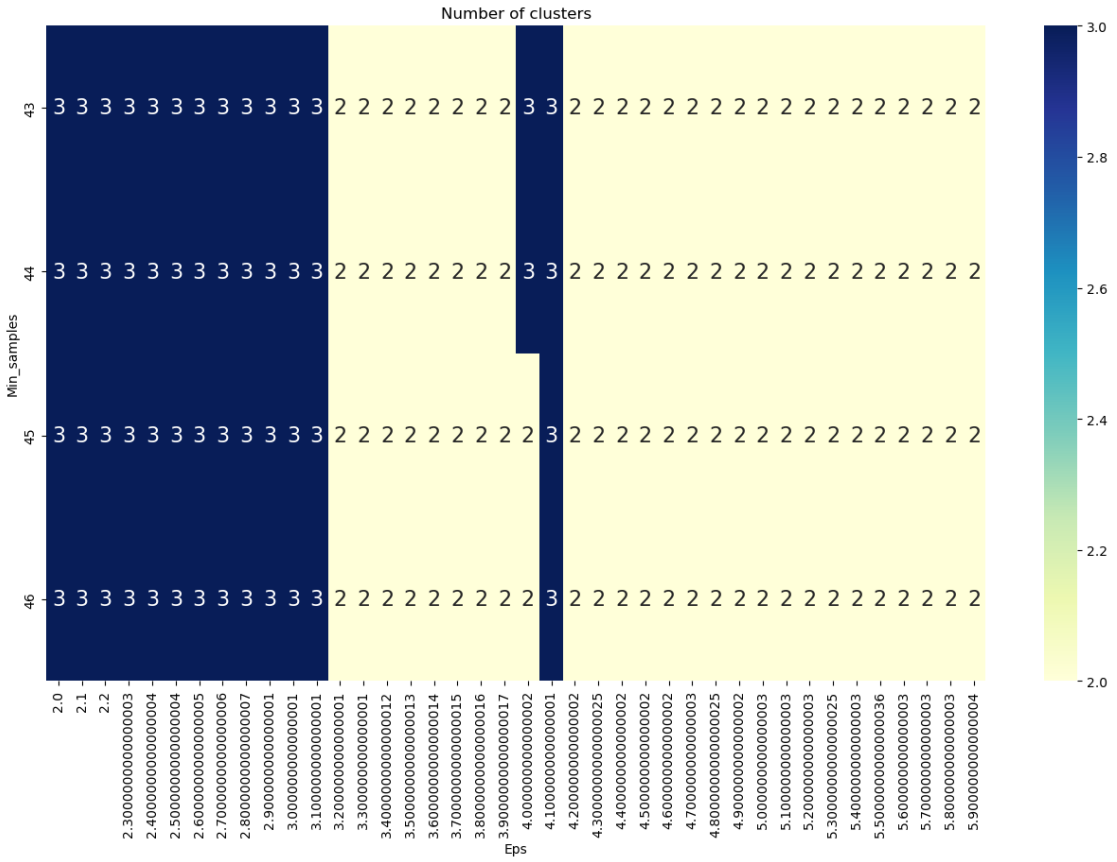
```
from itertools import product

#from itertools import product

eps_values = np.arange(2,6,0.1) # eps values is mean distance of whole dataframe
min_samples = np.arange(43,47) # min_sample is root of sample numbers
DBSCAN_params = list(product(eps_values, min_samples))
```

```
from sklearn.metrics import silhouette_score
no_of_clusters = []
sil_score = []

for p in DBSCAN_params:
    DBS_clustering = DBSCAN(eps=p[0], min_samples=p[1]).fit(reduced_data)
    no_of_clusters.append(len(np.unique(DBS_clustering.labels_)))
    sil_score.append(silhouette_score(reduced_data, DBS_clustering.labels_))
```



```
DBS_clustering = DBSCAN(eps=4.1, min_samples=43).fit(reduced_data)

DBSCAN_clustered = reduced_data.copy()
DBSCAN_clustered.loc[:,'Cluster'] = DBS_clustering.labels_
```

```
DBSCAN_clust_sizes = DBSCAN_clustered.groupby('Cluster').size().to_frame()
DBSCAN_clust_sizes.columns = ["DBSCAN_size"]
DBSCAN_clust_sizes
```

Cluster	DBSCAN_size
-1	479
0	1681
1	52

METRICS:

- 21.65 % data is a noise
- 75.99 % data is in cluster 0
- only 2.36 % data is in cluster 1

Inference : After performing the DBSCAN clustering we got three clusters including the noise cluster. Cluster 0 is formed by 75.99% of total observation which indicates some inaccuracy in results. From the proportion of clusters, we can conclude that data is showing some similarity in density and cause can be the high dimensional data. To get improved results, we need to reduce the number of dimensions of the data but as per business requirements, we're including all the necessary feature variables. Hence clustering using DBSCAN is not fruitful in this specific case.

Conclusion :

After performing data cleaning and data transformations we implemented clustering using K-Means algorithm. We found K=2 as the optimum number of clusters using the “Elbow method”. We achieved 2 clusters with well diversified personality and behavioral characteristics, facilitating for effective customer segmentation and targeting.

Then we thought of having more segments for customers to achieve more number of diversified groups.

We plotted dendogram to visualize the distances between the clusters and the history of grouping clusters.

We chose to cut the dendogram at a distance of 75 because if we decrease it further the number of clusters were increasing from 4 to 8.

We used Agglomerative clustering with the number of clusters = 4 and did personality analysis of each cluster.

We found that there was not a significant difference between clusters with respect to attributes like age and education.

If a company’s requirements best suit with a lesser number of clusters, they can utilize the clusters formed by K-means clustering.

In a similar way, if a company has a wide array of products, and requires a good number of customer segments, they can utilize the results of agglomerative clustering.

Final Outcomes Which can be used for targeting customers :-

Customer Segmentation: K-MEANS Clustering (K=2)

Personality 0	**Personality 1**
covers 60% of all people	covers 40% of the people
34.6% people are in the Age group of 40-50	38% of people are Senior citizens.
27% have 1-2 kids	28% have 1-2 kids
9% are single parents	80% are single parents
48% are Graduate	55.5% are Graduate
Lower income and spending	Higher Income and Spending

Suggestions:-

- Personality 1 has 80% of single parents so this might be a good target for companies focused on these type of customers.
- Personality 1 has high income and high spending.They can be targeted for high end products and luxurious products and holiday packages (considering the fact that this group is old age dominated).

Customer Segmentation: Agglomerative Hierarchical Clustering(N=4)

Personality 0	**Personality 1**	**Personality 2**	**Personality 3**
Overall it has 30% of customers.	Overall it has only 10% of customers.	Overall it has 43% of customers.	Overall it has 16% of customers.
More Middle aged population	More Old aged population	More middle aged population	More old aged population
Majority have 1 kid.	Majority have no kids.	Majority have 1 kid.	Has a good proportion of singles.
Average family size is 3	Average family size is 2	Average family size is 3	Average family size is 2
Average Income	Higher Income	Low income	High Income

Suggestions:-

- Personality 0 and 2 have average family size 3 with 1 kid, so they can be targeted for kid/child products and household products.
- Personality 1,3 are high earning groups with relatively lesser family size with no kids. They can be targeted for high end products and luxurious products and holiday packages (considering the fact that these groups are old age dominated).
- Personality 0, 2 are middle aged dominated and lower to average income groups, so they might be willing to react to campaigns and loyalty programs.