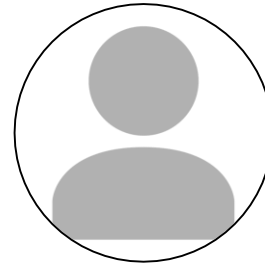


Predict Customer Personality to boost marketing campaign by using Machine Learning



Created by:

Abie bachtiar

Your Email

Your linkedIn Profile

A bachelor of economics from the Islamic University of Indonesia. I am an adaptive, life learner person and like to try new things. Familiar with processing and analysis of data using SQL, Python, google colabs and others.

“Sebuah perusahaan dapat berkembang dengan pesat saat mengetahui perilaku customer personality nya, sehingga dapat memberikan layanan serta manfaat lebih baik kepada customers yang berpotensi menjadi loyal customers. Dengan mengolah data historical marketing campaign guna menaikkan performa dan menyasar customers yang tepat agar dapat bertransaksi di platform perusahaan, dari insight data tersebut fokus kita adalah membuat sebuah model prediksi kluster sehingga memudahkan perusahaan dalam membuat keputusan ”

- Lakukan Feature Engineering dengan menghitung conversion rate dengan definisi ($\text{\#response} / \text{\#visit}$). Tidak hanya conversion rate, namun juga cari feature lain yang representatif, contohnya seperti umur, jumlah anak, total pengeluaran, total transaksi, dll.
- Tulislah **Exploration Data Analysis** (EDA) yang sudah kamu lakukan, mulai dari plot yang kamu buat hingga analisis interpretasinya. Tuliskan pula insight yang dapat dijadikan rekomendasi (jika ada).
- **Source code** yang sudah kamu buat, dapat ditampilkan dan berikan link untuk mengakses file tersebut. Contohnya seperti di pojok kanan bawah.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 30 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Unnamed: 0            2240 non-null  int64
1   ID                    2240 non-null  int64
2   Year_Birth            2240 non-null  int64
3   Education             2240 non-null  object
4   Marital_Status       2240 non-null  object
5   Income               2216 non-null  float64
6   Kidhome              2240 non-null  int64
7   Teenhome             2240 non-null  int64
8   Dt_Customer          2240 non-null  object
9   Recency              2240 non-null  int64
10  MntCoke               2240 non-null  int64
11  MntFruits             2240 non-null  int64
12  MntMeatProducts       2240 non-null  int64
13  MntFishProducts       2240 non-null  int64
14  MntSweetProducts      2240 non-null  int64
15  MntGoldProds          2240 non-null  int64
16  NumDealsPurchases     2240 non-null  int64
17  NumWebPurchases       2240 non-null  int64
18  NumCatalogPurchases   2240 non-null  int64
19  NumStorePurchases     2240 non-null  int64
20  NumWebVisitsMonth     2240 non-null  int64
21  AcceptedCmp3          2240 non-null  int64
22  AcceptedCmp4          2240 non-null  int64
23  AcceptedCmp5          2240 non-null  int64
24  AcceptedCmp1          2240 non-null  int64
25  AcceptedCmp2          2240 non-null  int64
26  Complain              2240 non-null  int64
27  Z_CostContact         2240 non-null  int64
28  Z_Revenue             2240 non-null  int64
29  Response              2240 non-null  int64
dtypes: float64(1), int64(26), object(3)
memory usage: 525.1+ KB
```

- Dataset merupakan data marketing campaign yang digunakan untuk melihat respon pelanggan terhadap penawaran yang telah diberikan
- Dataset berisi data transaksi pelanggan dan produk apa saja yang di beli yang dibeli.
- Dataset terdiri dari 2240 baris dan 30 kolom dengan tipe data int64,float dan object.

```
[ ] 1 #total acc cmp  
2 df['Total_Acc_Cmp'] = df['AcceptedCmp1']+df['AcceptedCmp2']+df['AcceptedCmp3']+df['AcceptedCmp4']+df['AcceptedCmp5']
```

```
[ ] 1 # Total Purchases  
2 df['total_purchases'] = ( df['NumDealsPurchases'] + df['NumWebPurchases']  
3                          + df['NumCatalogPurchases'] + df['NumStorePurchases'])
```

```
▶ 1 df['conversion_rate'] = np.round(df['Response'] / df['NumWebVisitsMonth'], 2)  
2 df.sample(5)
```

```
[ ] 1 # total childs  
2 df['total_children'] = df['Kidhome'] + df['Teenhome']
```

```
[ ] 1 # total spend  
2 df['Total_spend'] = df[['MntCoke', 'MntFruits', 'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts', 'MntGoldProds']].sum(axis=1)
```

- Total_Acc_Cmp adalah akumulasi terhdapt campaign
- Total_Purchase adalah akumulasi dari bebrapa pembeliandari berbagai media
- Convernasion_rate dalah repon dibagi kunjungan
- Total_children adalah akumulasi dari total anak yang dimiliki
- Total_sepend adalh total belanja terhdap beberapa produk

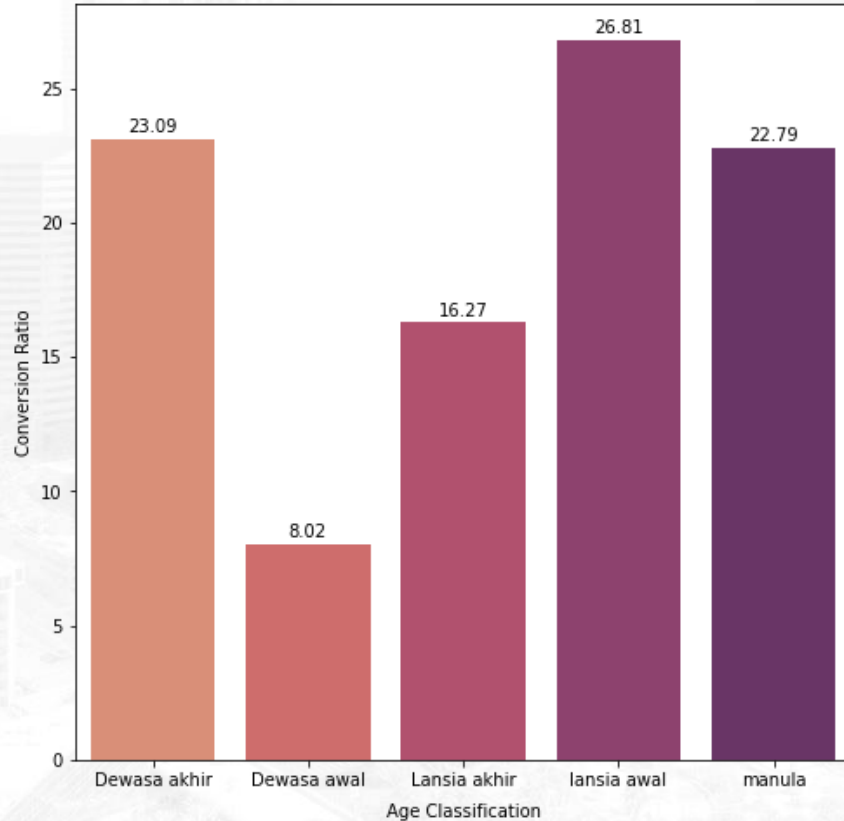

```
1 # umur
2 time = datetime.now().year
3 df['age'] = time - df['Year_Birth']
4 # grup age
5 age_list=[]
6 for i in df['age']:
7     if i >= 0 and i <= 1:
8         group = 'Infant'
9     elif i >= 2 and i <= 4:
10        group = 'balita'
11    elif i >= 5 and i <= 12:
12        group = 'Anak-anak'
13    elif i >= 13 and i <= 16:
14        group = 'Remaja awal'
15    elif i >= 17 and i <= 25:
16        group = 'Remaja akhir'
17    elif i >= 26 and i <= 35:
18        group = 'Dewasa awal'
19    elif i >= 36 and i <= 45:
20        group = 'Dewasa akhir'
21    elif i >= 46 and i <= 55:
22        group = "lansia awal"
23    elif i >= 56 and i <= 65 :
24        group = "Lansia akhir"
25    else:
26        group = 'manula'
27    age_list.append(group)
28
29 df['Age_Group'] = age_list
```

Variabel umur didapat dengan mengurangi tahun sekarang dengan tahun lahir yang kemudian dikelompokkan menjadi beberapa kelompok yaitu bayi, balita, remaja dewasa sampai lansia

	index	Age_Group	conversion_rate
1	1	Dewasa awal	8.02
0	0	Dewasa akhir	23.09
3	3	lansia awal	26.81
2	2	Lansia akhir	16.27
4	4	manula	22.79

Kemudian menghitung coversion rate berdasarkan kategory umur pelanggan

**Total of Customer Conversion Ratio
Based on Age**



Berdasarkan visualisasi disamping dapat kita lihat bahwa pelanggan kategori lansia awal (46-55 tahun) adalah pelanggan dengan conversion rate tertinggi yaitu 26,81% dan terendah adalah pelanggan kategori dewasa awal (26-35 tahun) dengan conversion rate 8,02%.

Descriptive statistics

```
1 df1.shape
(2240, 12)

1 # cek duplikat
2 df1.duplicated().sum()
183

1 #cek missing value
2 df1.isna().sum()

Education      0
Marital_Status 0
Income         24
Recency        0
Complain       0
Total_Acc_Cmp  0
total_purchases 0
conversion_rate 11
age            0
Age_Group      0
total_children 0
Total_spend    0
dtype: int64
```

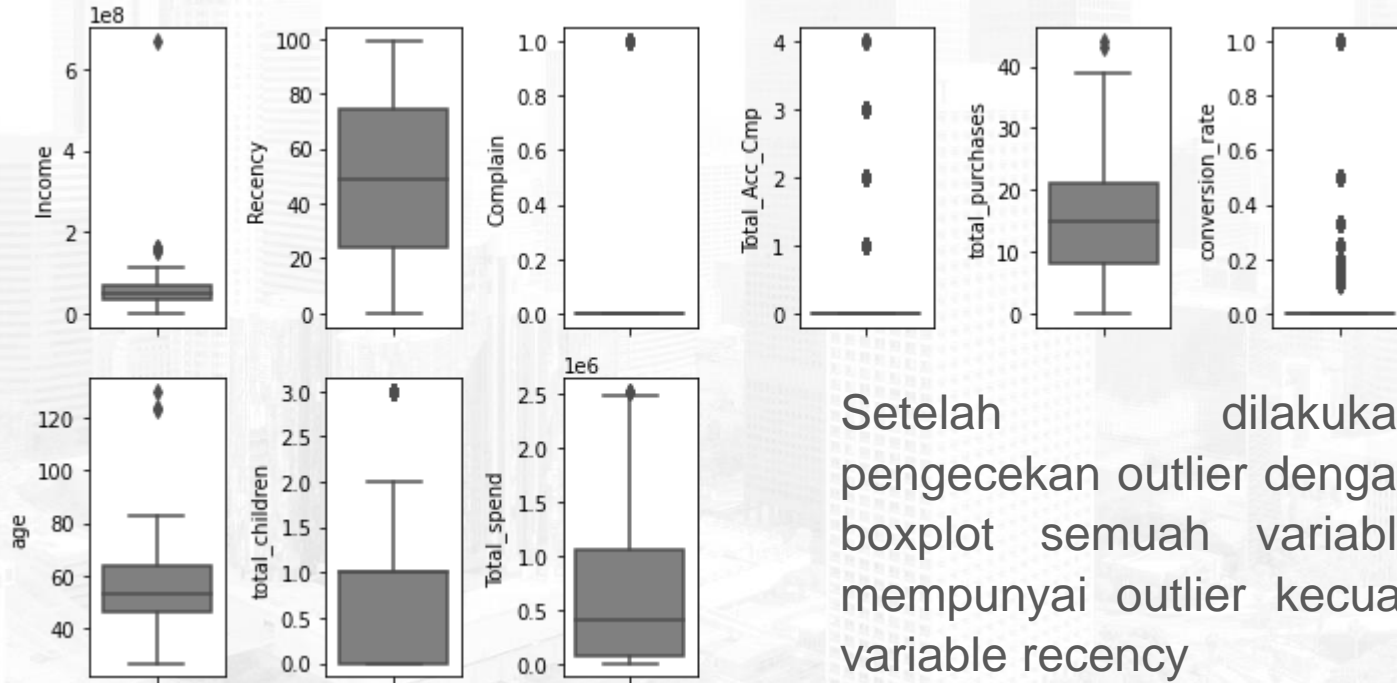
	Income	Recency	Complain	Total_Acc_Cmp	total_purchases	conversion_rate	age	total_children	Total_spend
count	2.216000e+03	2240.000000	2240.000000	2240.000000	2240.000000	2229.000000	2240.000000	2240.000000	2.240000e+03
mean	5.224725e+07	49.109375	0.009375	0.297768	14.862054	0.043508	54.194196	0.950446	6.057982e+05
std	2.517308e+07	28.962453	0.096391	0.678381	7.677173	0.142827	11.984069	0.751803	6.022493e+05
min	1.730000e+06	0.000000	0.000000	0.000000	0.000000	0.000000	27.000000	0.000000	5.000000e+03
25%	3.530300e+07	24.000000	0.000000	0.000000	8.000000	0.000000	46.000000	0.000000	6.875000e+04
50%	5.138150e+07	49.000000	0.000000	0.000000	15.000000	0.000000	53.000000	1.000000	3.960000e+05
75%	6.852200e+07	74.000000	0.000000	0.000000	21.000000	0.000000	64.000000	1.000000	1.045500e+06
max	6.666660e+08	99.000000	1.000000	4.000000	44.000000	1.000000	130.000000	3.000000	2.525000e+06

Setelah dilakukan pengecekan terhadap data terhadap duplikat sebanyak 183 dan missing value pada kolom Income(24) dan age (11)

```
1 #summary statistic
2 df[cat].describe()
```

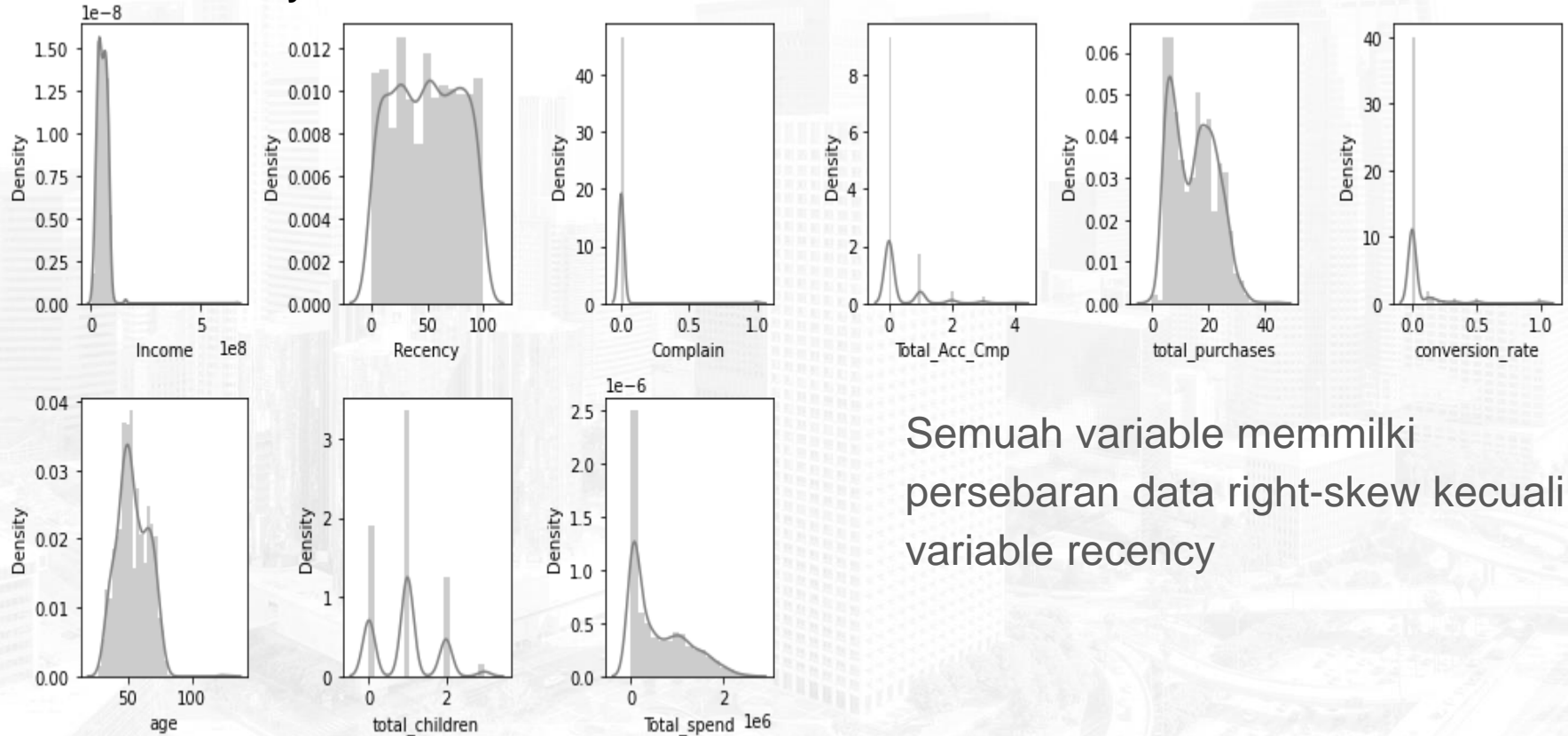
	Education	Marital_Status	Age_Group
count	2240	2240	2240
unique	5	6	5
top	S1	Menikah	lansia awal
freq	1127	864	732

Univariate Analysis



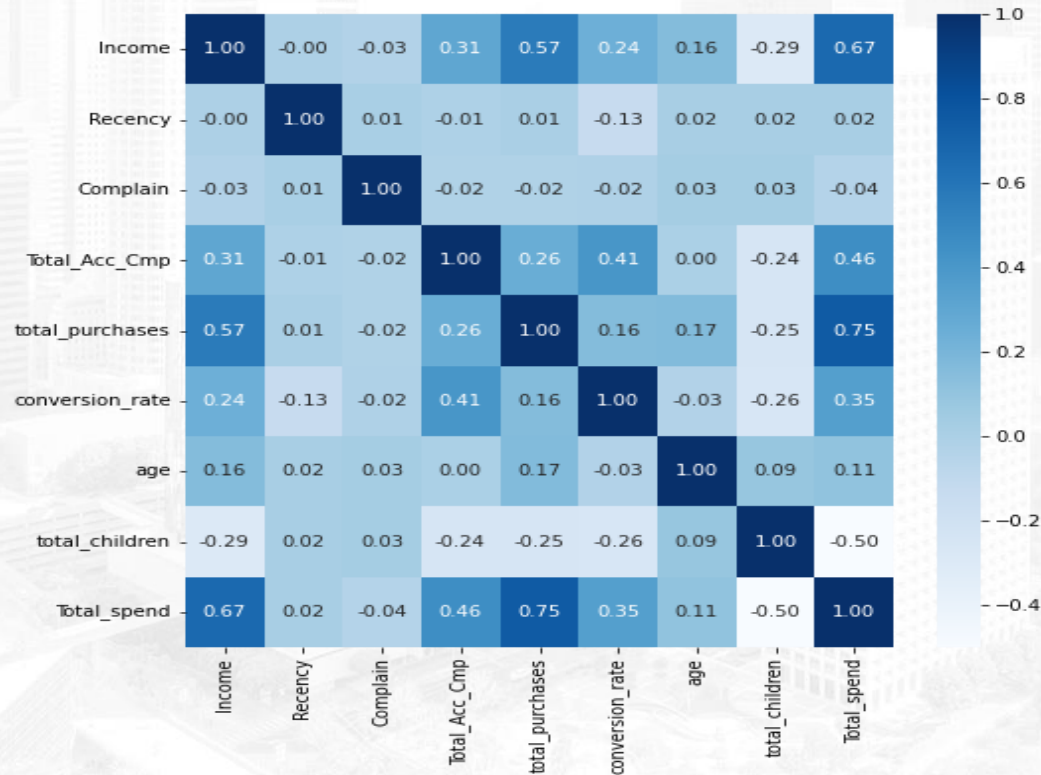
Setelah dilakukan pengecekan outlier dengan boxplot semua variable mempunyai outlier kecuali variable recency

Univariate Analysis



Semua variable memiliki persebaran data right-skew kecuali variable recency

Multivariate Analysis



```
1 #handle missing value
2 df1.dropna()
3 #handle duplicate
4 df1.drop_duplicates()
```

Menghapus data null dan duplikat

```
1 #hadnle outlier
2 from scipy import stats
3 print(f'Jumlah baris sebelum memfilter outlier: {len(df1)}')
4
5 filtered_entries = np.array([True] * len(df1))
6
7 for col in ['Income', 'Recency', 'Complain', 'Total_Acc_Cmp', 'total_purch
8
9     Q1 = df[col].quantile(0.25)
10    Q3 = df[col].quantile(0.75)
11    IQR = Q3 - Q1
12    low_limit = Q1 - (IQR * 1.5)
13    high_limit = Q3 + (IQR * 1.5)
14
15    filtered_entries = ((df1[col] >= low_limit) & (df1[col] <= high_
16
17 df1 = df1[filtered_entries]
18
19 print(f'Jumlah baris setelah memfilter outlier: {len(df1)}')
```

```
Jumlah baris sebelum memfilter outlier: 2240
Jumlah baris setelah memfilter outlier: 1534
```

Menghapus outlier dengan IQR

```
1 # label
2 mapping_pendidikan = {
3     'D3' : 0,
4     'SMA' : 1,
5     'S1' : 2,
6     'S2' : 3,
7     'S3' : 4
8 }
9 map_age = {
10     'Dewasa awal': 1,
11     'Dewasa akhir' : 2,
12     'lansia awal' : 3,
13     'Lansia akhir' : 4,
14     'manula' : 5
15 }
```

```
1 # ohe
2 for i in ['Marital_Status']:
3     onehots = pd.get_dummies(df1[i], prefix=i)
4     df1 = df1.join(onehots)
```

```
1 df1['pendidikan'] = df1['Education'].map(mapping_pendidikan)
2 df1['cat_usia'] = df1['Age_Group'].map(map_age)
```

Pelabelan pada kolom education dan Age_group dan one hot encoding pada kolom Marital_Status kemudian disimpan pada kolom baru pendidikan dan cat_usia

feature selection

Recency : Date of Last of Purchases : Recency

Frequency : Total Number of Orders : Total_Purchases

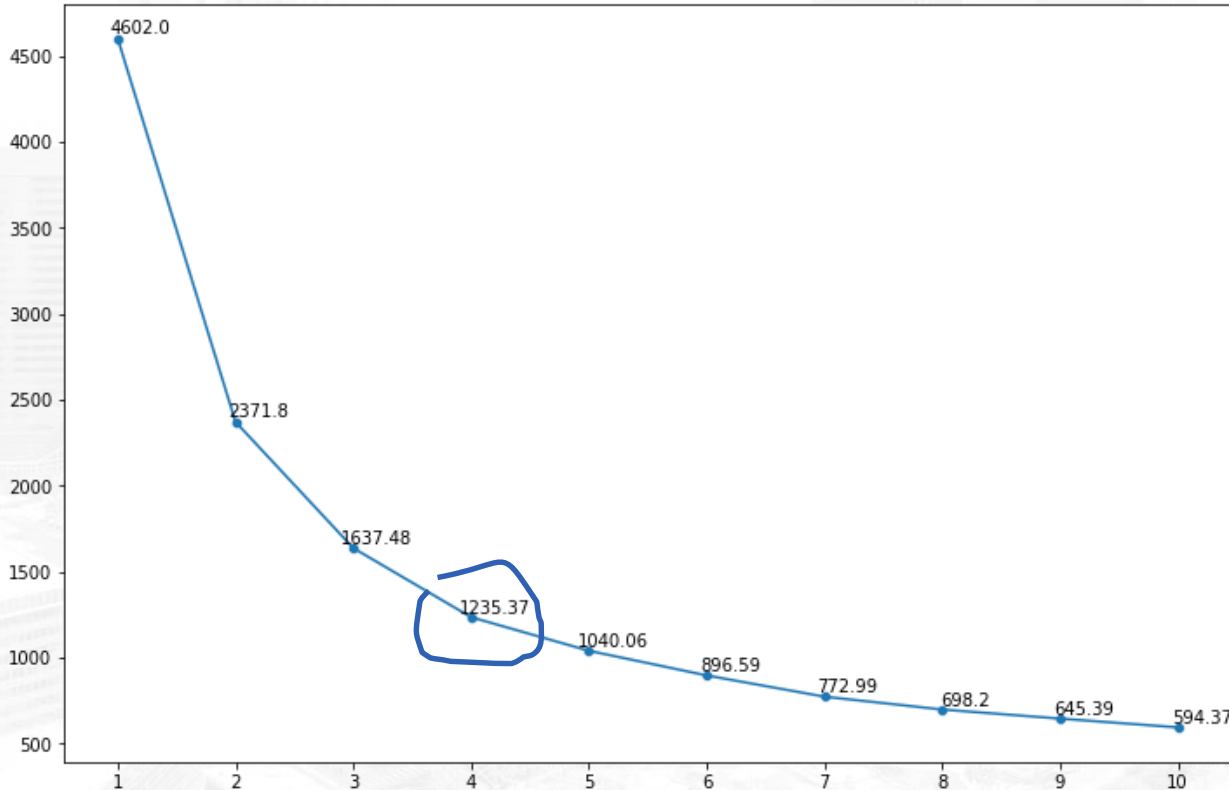
Monetization : Total order value : Total_spend

Loyalty : Total campaign accepted : Total_accepted_cmp

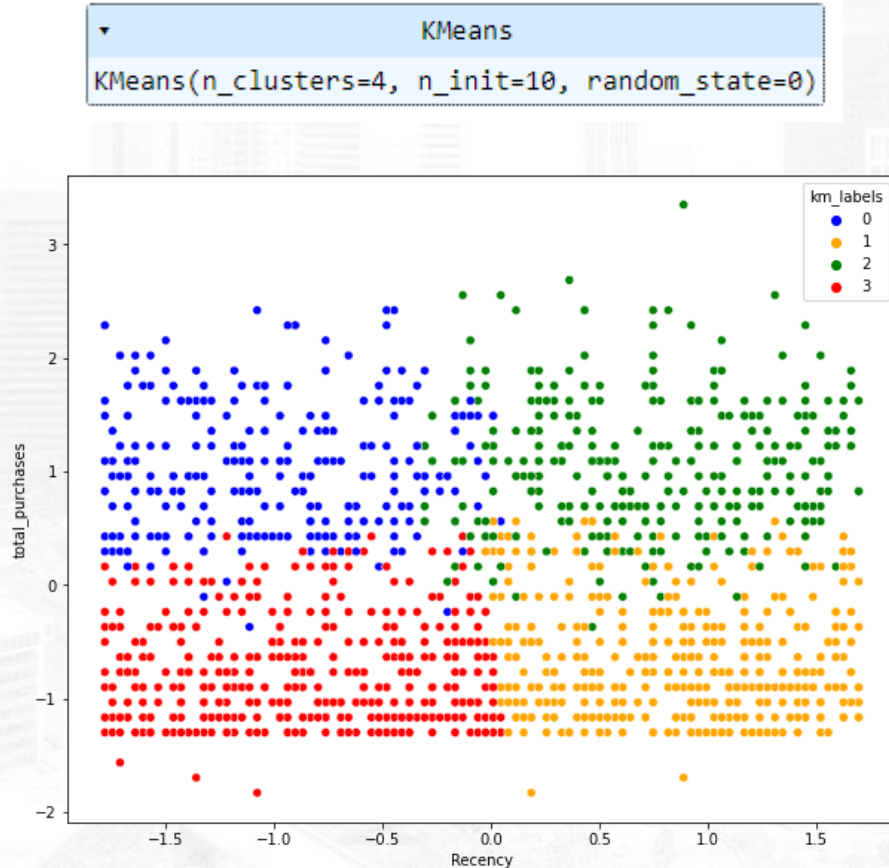
```
1 feature = ['Recency', 'Total_Acc_Cmp', 'total_purchases', 'Total_spend']  
2 x = df1[feature].values  
3 from sklearn.preprocessing import StandardScaler  
4 x_std = StandardScaler().fit_transform(x)  
5 new_df = pd.DataFrame(data= x_std, columns=feature)  
6 new_df.sample(5)
```

Standarisasi data
menggunakan
standardscaler

Elbow Method



Untuk mencari jumlah calaster kita dapat melihat pada grafik elbow method, berdasarkan visualisasi grafik disamping titik patahan menunjukan pada angka 4 yang berate jumlah cluster yang di perlukan sebanyak 4



setelah diketahui jumlah cluster yang diperlukan selanjutnya malukan visualisasi persebaran cluster tersebut. Berdasarkan grafik di samaping kita dapat melihat persebaran cluster menjadi 4 yaitu cluster 0(biru), cluster 1 (kuning), cluster 2 (hijau), cluster 3 (merah).

```
1 silhouette_score(new_df, kmeans.labels_)
0.5799481090429165
```

Nilai silhouette score 0.57994 yang berate mendekati 1 maka menunjukan bahwa cluster tidak tumpah tindih dan score yang postif menunjukan bahwa sampel telah ditugaskan ke cluster yang benar, karena masing masing cluster tidak mirip

	cluster	Recency	total_purchases	Total_spend	potensial_score
0	0	6255	5845	239883000	30.88
1	1	34794	4083	59836000	24.25
2	2	24431	7419	362205000	50.41
3	3	12290	3775	51902000	14.47

potensial_score adalah kalkulasi dari presentase **Recency+total_Purchases+Total_spend**

berdasarkan grafik di atas dapat kita lihat bahwa customer cluster 2 adalah cluster dengan score tertinggi yaitu 50.41 point dan yang terendah adalah customer dengan cluster 3 dengan 14.47 point. Sehingga dapat kita simpulkan bahwa untuk marketing campaign yang akan dilakukan alangkah lebih baik untuk berfokus pada customer cluster 2 memiliki kemungkinan lebih besar untuk campaign untuk diterima atau di respon

Rekomendasi

1. membuat membership berdasarkan total pengeluaran, membership dibedakan menjadi beberapa tingkatan seperti Platinum, Gold, Silver, Bronze dimana semakin tinggi tingkat pengeluaran maka semakin tinggi tingkat membership.
2. setiap membership memiliki keuntungan masing-masing dimana tingkatan tertinggi memiliki lebih banyak penawaran diskon dan promo dari pada tingkatan terbawah
3. dorong terus customer untuk meningkatkan membership mereka dengan cara memberikan penawaran menarik sehingga mereka meningkatkan pengeluaran mereka untuk lebih banyak belanja.
4. fokus campaign kepada customer cluster 2
5. membuat event-event yang menarik yang dimana membership yang dimiliki customer dapat digunakan seperti flash sale atau membership ditingkatan tertinggi berhak mendapatkan tiket konser musik, bioskop, potongan belanja S