Predict Customer Personality to boost marketing campaign by using Machine Learning

Overview

“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. Dalam project ini kami menggunakan dataset yang disediakan oleh Rakamin Academy. Seluruh teknis pengerjaan menggunakan bahasa pemrograman python.”

Source code is here

Table of Contents

01

Conversion Rate Analysis Based on Income, Spending and Age

FEATURE ENGINEERING

1

```
df['Age'] = 2022 - df['Year_Birth'] #add Age feature (age in 2022)
df['Total_Child'] = df['Kidhome'] + df['Teenhome'] #add Total Child feature
df['Total_Spend'] = df['MntCoke'] + df['MntFruits'] + df['MntMeatProducts'] + df['MntFishProducts'] + df['MntSweetProducts']
df['Total_Transaction'] = df['NumDealsPurchases'] + df['NumWebPurchases'] + df['NumCatalogPurchases'] +df['NumStorePurchases
df['Total_Acc_Campaign'] = df['AcceptedCmp1'] + df['AcceptedCmp2'] + df['AcceptedCmp3'] + df['AcceptedCmp4'] + df['AcceptedC
df['Is_Parents'] = np.where(df['Total_Child'] >0,1,0) #add is_parents feature

#add Conversion Rate feature
def conversion(x,y):
    if y == 0:
        return 0
    return x / y

df['Conversion_Rate'] = df.apply(lambda x: conversion(x['Total_Transaction'],x['NumWebVisitsMonth']), axis=1)
```

2

```
#grouping customers Age
age_group_list = []
for i, kolom in df.iterrows():
    prefix = kolom['Age']
    if prefix <= 45 :
        age = 'Young'
    elif prefix <= 60 :
        age = 'Middle'
    else :
        age = 'Elderly'
    age_group_list.append(age)
df['Age_Group'] = age_group_list
```

Pada data historical marketing campaign ini kami menambahkan beberapa kolom yaitu **Conversion_Rate** untuk melihat perbandingan antara total transaksi dengan jumlah visit customer. Terdapat penambahan kolom lain yaitu **Age, Total_Child, Total_Spend**, dan **Total_Transaction, Total_Acc_Campaign**, dan **Is_Parents**

Disini saya mengelompokkan usia customer ke dalam 3 kelompok yang berbeda yaitu Young, Middle, dan Elderly. Pengelompokan ini dilakukan dengan maksud untuk mempermudah analisis customer berdasarkan kategori usia.

DESCRIPTIVE STATISTICS

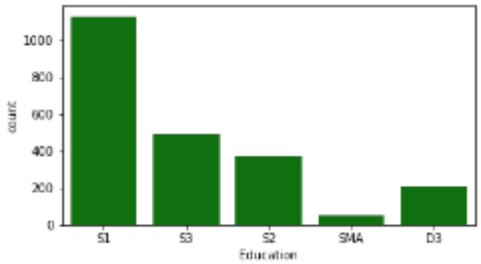
```
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 38 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Unnamed: 0           2240 non-null   int64
 1   ID                   2240 non-null   int64
 2   Year_Birth           2240 non-null   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
 30  Age                  2240 non-null   int64
 31  Total_Child          2240 non-null   int64
 32  Total_Spend          2240 non-null   int64
 33  Total_Transaction    2240 non-null   int64
 34  Total_Acc_Campaign   2240 non-null   int64
 35  Is_Parents           2240 non-null   int32
 36  Conversion_Rate      2240 non-null   float64
 37  Age_Group            2240 non-null   object
dtypes: float64(2), int32(1), int64(31), object(4)
```

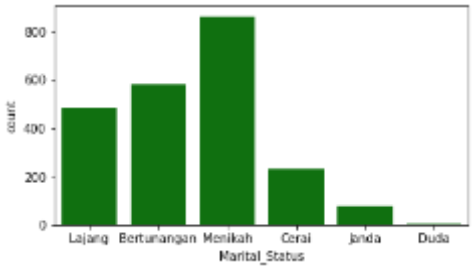
	Z_CostContact	Z_Revenue
count	2240.0	2240.0
mean	3.0	11.0
std	0.0	0.0
min	3.0	11.0
25%	3.0	11.0
50%	3.0	11.0
75%	3.0	11.0
max	3.0	11.0

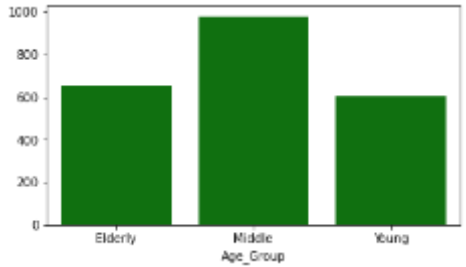
1 Pada dataset ini terdapat missing value yaitu pada kolom Income. Jumlah row missing value pada kolom Income cukup sedikit sehingga dapat di drop.

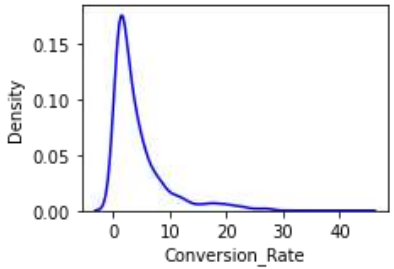
2 Kolom Z_CostContact dan Z_Revenue hanya memiliki 1 value. Terlihat dari nilai minimum dan nilai maksimum yang sama. Sehingga kolom ini tidak perlu terlalu diperhatikan dalam analisis maupun dalam modeling.

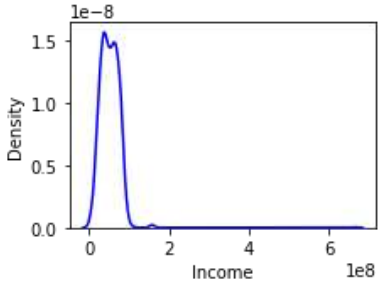
UNIVARIATE ANALYSIS

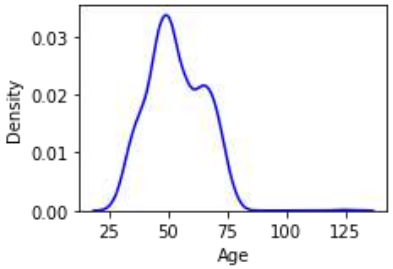


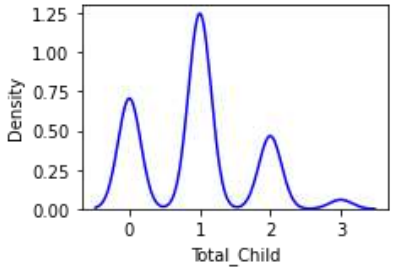


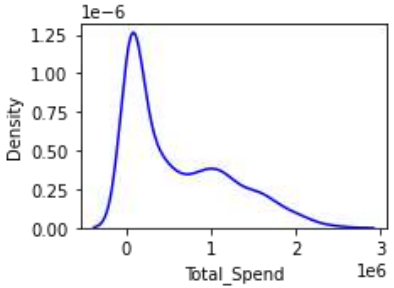


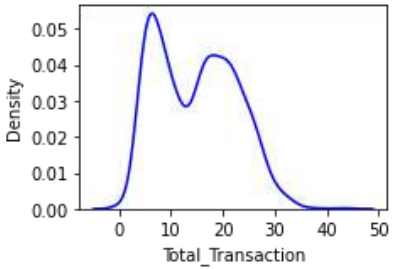










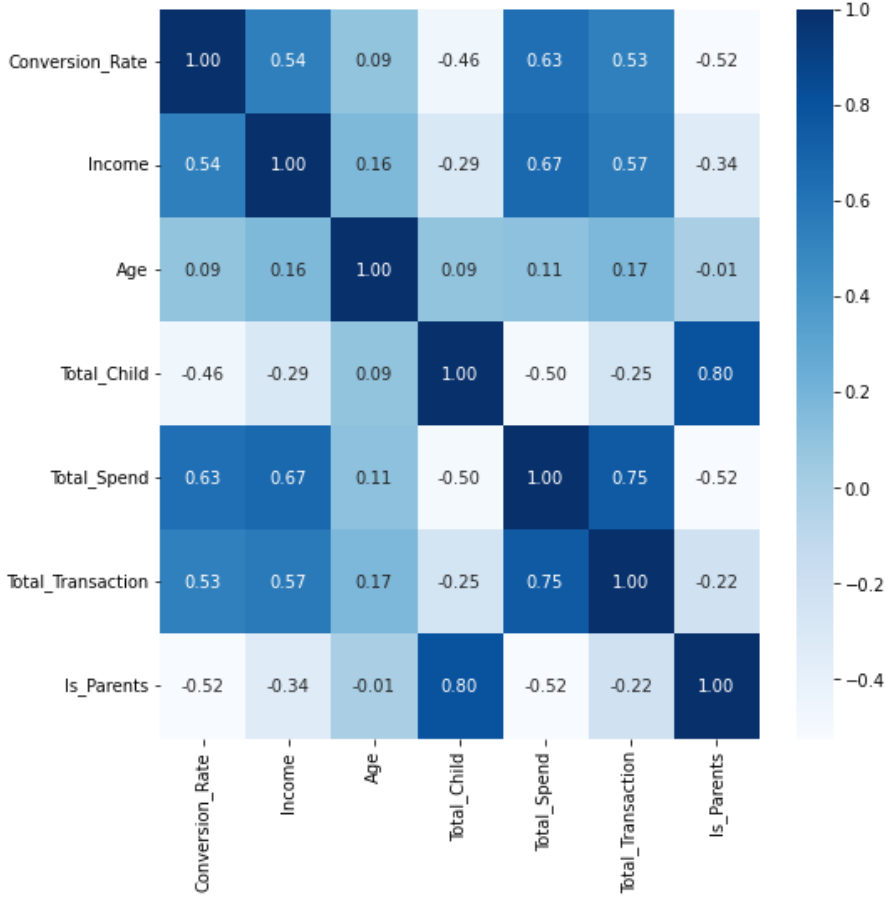


1 Pada countplot di atas menunjukkan jumlah customer berdasarkan pengelompokan pendidikan, status, dan kelompok usia.

2 Pada grafik distribusi Conversion Rate, density paling tinggi adalah pada Conversion Rate antara 0 – 10.

3 Distribusi conversion rate, income, age, total spend, dan total transaction cenderung skew.

BIVARIATE ANALYSIS

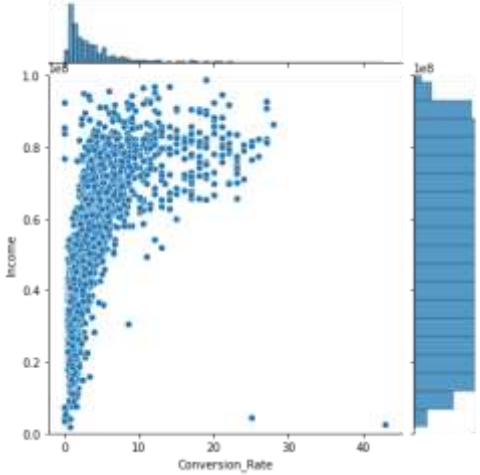


1 Pada *correlation heatmap* terlihat bahwa Conversion_Rate berkorelasi positif terhadap Income. Ini berarti bahwa customer yang memiliki **Income lebih tinggi cenderung meresponse campaign**.

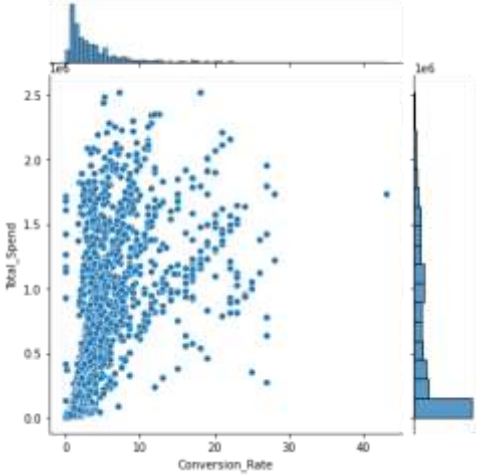
2 Conversion_Rate juga berkorelasi positif dengan Total_Spend, dan Total_Transaction. Hal ini berarti customer yang **spending lebih banyak dan lebih sering berbelanja cenderung meresponse campaign.**

3 Conversion_Rate berkorelasi negatif terhadap Total_Child. Hal ini berarti customer yang memiliki **lebih banyak anak cenderung tidak meresponse campaign.**

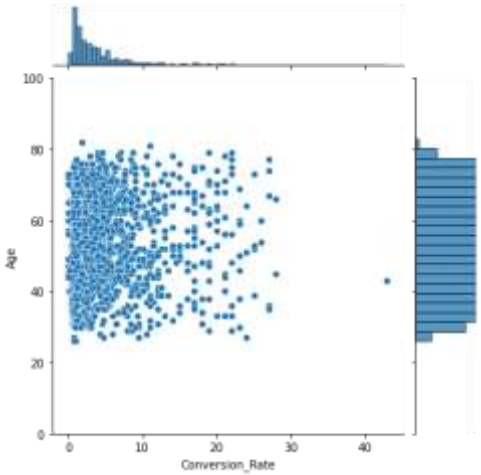
4 Conversion_Rate hampir tidak berkorelasi dengan usia. Ini berarti bahwa **usia tidak mempengaruhi apakah customer akan meresponse campaign atau tidak.**



Conversion_Rate vs Income



Conversion_Rate vs Total_Spend



Conversion_Rate vs Age

Untuk lebih jelasnya dapat dilihat pada scatter plot di atas. Dimana Conversion_Rate memiliki korelasi atau berbanding lurus dengan Income dan juga Total_Spend. Berbeda dengan hubungan antara Conversion_Rate dengan Age, dimana pada scatter plot di atas tidak terlihat adanya korelasi.

02

Data Cleaning & Preprocessing

DATA CLEANING

Sebelum masuk ke proses modeling, disini saya melakukan pembersihan data terlebih dahulu dengan mengecek apakah ada nilai null atau missing value dan duplicated data.

1

```
1  df.isnull().sum()
```

```
Unnamed: 0          0
ID                  0
Year_Birth          0
Education           0
Marital_Status      0
Income             24
```

```
2  df = df.dropna()
```

Terdapat 24 rows missing value pada kolom Income. Karena missing value tidak terlalu banyak jika dibandingkan dengan total row (1.07%) maka disini saya membuang missing value tersebut.

2

```
1  #check duplicated data
2  df.duplicated().any()
```

```
False
```

Pada dataset ini tidak terdapat data duplicate.

FEATURE ENCODING

1

```
#label encoding for Education feature
mapping_education = {
    'SMA' : 0,
    'D3' : 1,
    'S1' : 2,
    'S2' : 3,
    'S3' : 4
}

df['Education'] = df['Education'].map(mapping_education)
```

Education merupakan tipe data ordinal, sehingga untuk kolom ini dapat menggunakan metode label encoding. Disini saya memberikan nilai berdasarkan urutan jenjang pendidikan

2

```
# One hot encoding Marital_Status
for cat in ['Marital_Status']:
    onehots = pd.get_dummies(df[cat], prefix=cat)
    df = df.join(onehots)

# One hot encoding Age_Group
for cat in ['Age_Group']:
    onehots = pd.get_dummies(df[cat], prefix=cat)
    df = df.join(onehots)

# One hot encoding Is_Parents
for cat in ['Is_Parents']:
    onehots = pd.get_dummies(df[cat], prefix=cat)
    df = df.join(onehots)
```

Pada beberapa feature categorical lainnya menggunakan metode one hot encoding. Sehingga akan mengeluarkan kolom-kolom baru.

3

Marital_Status_Bertunangan	Marital_Status_Cerai	Marital_Status_Duda	Marital_Status_Janda	Marital_Status_Lajang	Marital_Status_Menikah
0	0	0	0	1	0
0	0	0	0	1	0

Age_Group_Elderly	Age_Group_Middle	Age_Group_Young	Is_Parents_0	Is_Parents_1
1	0	0	1	0
1	0	0	0	1

FEATURE TRANSFORMATION (STANDARDIZATION)

Pada proses sebelumnya terlihat bahwa distribusi kolom-kolom pada dataset ini tidak normal dan cenderung skew. Sebelum melakukan modeling, kita harus membuat kolom-kolom tersebut menjadi terdistribusi normal. Disini saya melakukan feature transformation menggunakan metode standardization pada kolom-kolom numerical.

```
#making a copy of dataframe
df_standard = df.copy()

#some features that will be transformed
standard_num = ['Income','Recency', 'MntCoke', 'MntFruits', 'MntMeatProducts',
       'MntFishProducts', 'MntSweetProducts', 'MntGoldProds',
       'NumDealsPurchases', 'NumWebPurchases', 'NumCatalogPurchases',
       'NumStorePurchases', 'NumWebVisitsMonth','Age','Total_Spend','Total_Transaction','Conversion_Rate']
```

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()

for col in standard_num:
    df_standard[col] = scaler.fit_transform(df_standard[[col]].values.reshape(len(df_standard),1))

df_standard.head()
```

Before Standardization

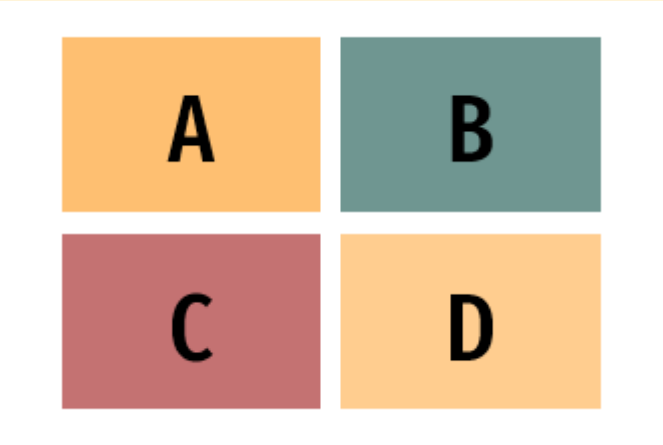
MntCoke	MntFruits	MntMeatProducts	MntFishProducts	MntSweetProducts	MntGoldProds	NumDealsPurchases	NumWebPurchases
2.240000e+03	2240.000000	2.240000e+03	2240.000000	2240.000000	2240.000000	2240.000000	2240.000000
3.039357e+05	26302.232143	1.669500e+05	37525.446429	27062.946429	44021.875000	2.325000	4.084821
3.365974e+05	39773.433765	2.257154e+05	54628.979403	41280.498488	52167.438915	1.932238	2.778714

After Standardization

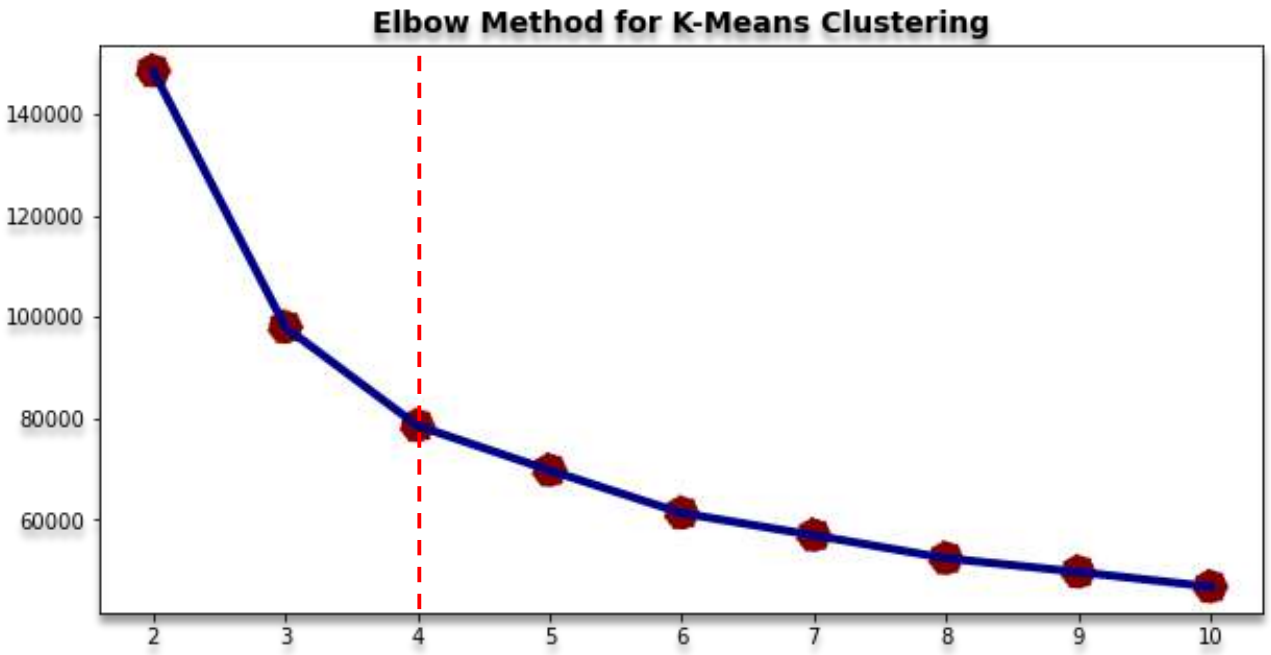
MntCoke	MntFruits	MntMeatProducts	MntFishProducts	MntSweetProducts	MntGoldProds	NumDealsPurchases	NumWebPurchases
0.978226	1.549429	1.690227	2.454568	1.484827	0.850031	0.351713	1.428553
-0.872024	-0.637328	-0.717986	-0.651038	-0.633880	-0.732867	-0.168231	-1.125881
0.358511	0.569159	-0.178368	1.340203	-0.146821	-0.037937	-0.688176	1.428553

03

Data Modeling (Clustering)



ELBOW METHOD



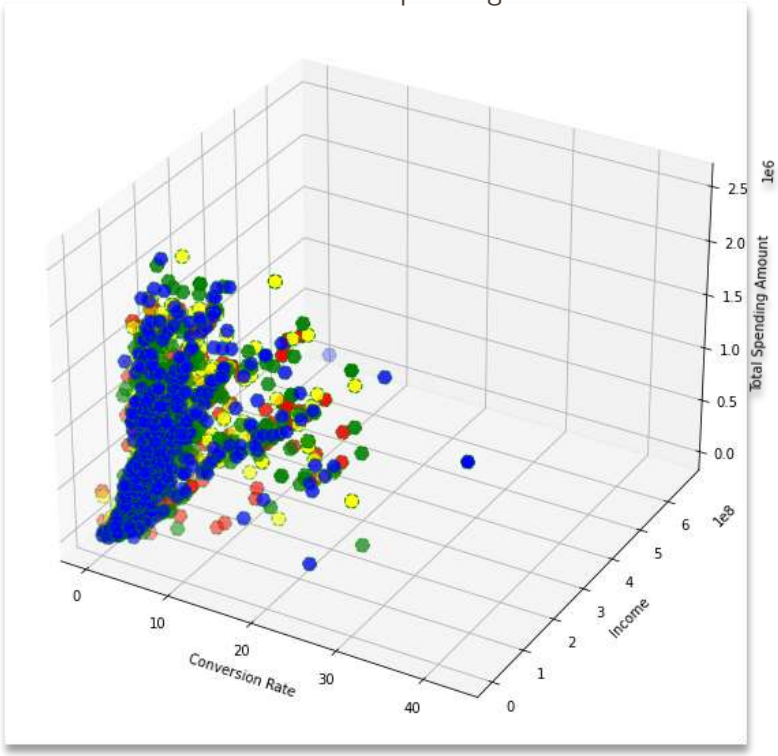
Sebelum melakukan clustering kita harus menentukan dahulu berapa jumlah clusternya. Disini saya menggunakan Elbow Method untuk menentukan berapa jumlah cluster yang optimal.

Dari grafik di samping jumlah clustering yang optimal untuk case ini adalah sebanyak **4 cluster**.

K-MEANS CLUSTERING

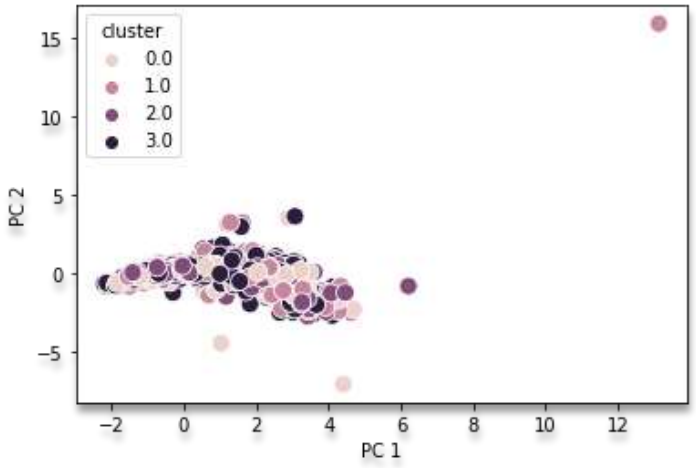
1

Berikut merupakan hasil dari clustering menggunakan metode K-Means Clustering dilihat dari 3 faktor yaitu Conversion Rate, Income, dan Total Spending.

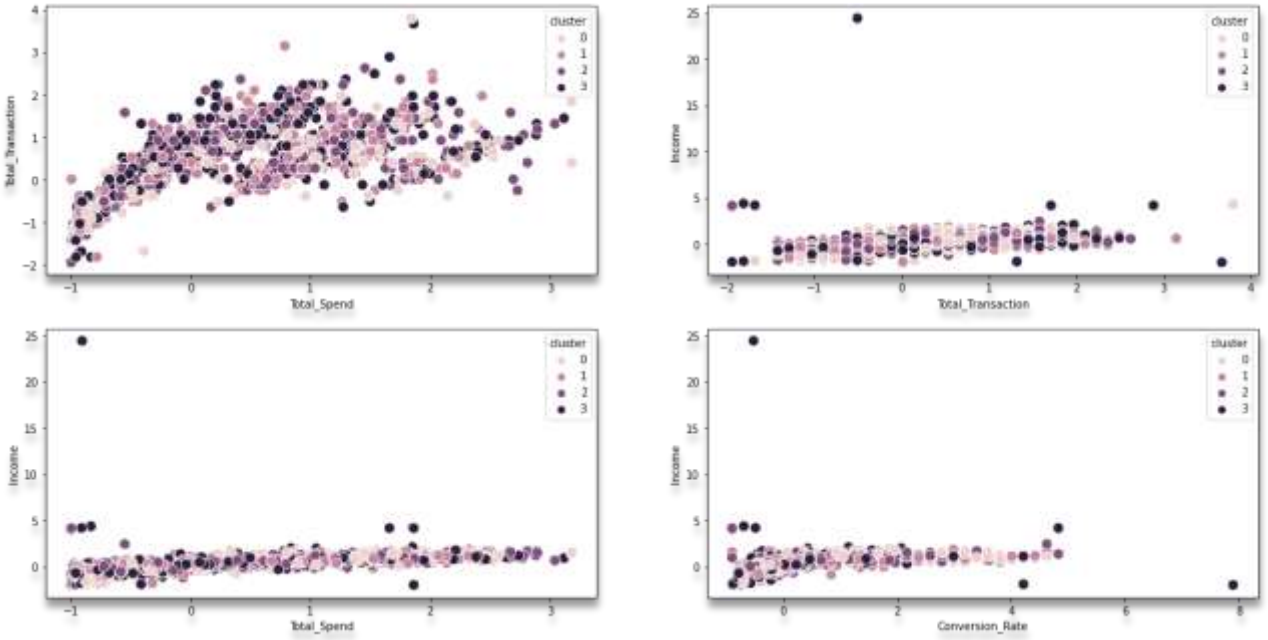


2

Karena terlalu sulit melihat hasil clustering pada visualisasi 3d, disini saya juga membuat visualisasi 2d menggunakan PCA



Jika hanya dilihat dari visualisasi sebelumnya dapat dilihat bahwa tiap-tiap cluster tidak terlalu berbeda antar satu sama lain. Maka dari itu disini saya juga melihat pembagian cluster per dua faktor. Namun seperti yang terlihat pada scatter plot di bawah ini, tetap terlihat bahwa pembagian cluster masih tercampur satu sama lain. Sehingga disini kita perlu mengevaluasi performa clustering model.



CLUSTERING EVALUATION : SILHOUETTE SCORE

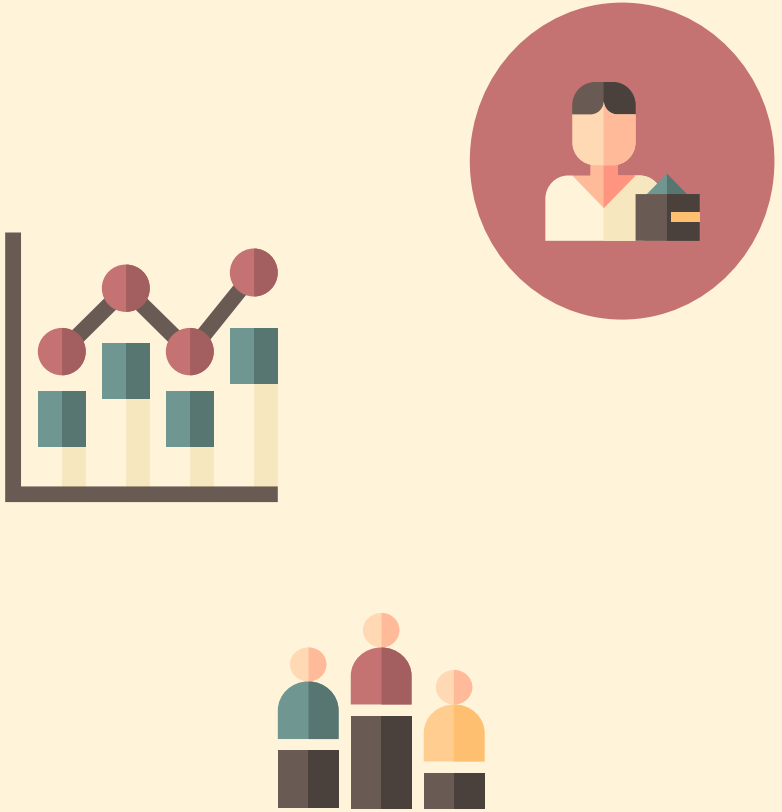
Untuk evaluasi clustering disini saya menggunakan metode Silhouette Score. Pada metode ini nilai yang dihasilkan berkisar diantara -1 s/d 1. Semakin mendekati angka 1 berarti cluster yang dihasilkan semakin baik dan karakteristik suatu cluster sangat berbeda dengan cluster lainnya. Silhouette score yang didapat pada modeling dengan 4 cluster ini adalah sebesar **0.33.** Dari nilai evaluasi ini, clustering yang dihasilkan cukup baik dan berbeda satu sama lain.

```
# Calculate Silhoutte Score
X = df_standard.drop(columns=['cluster'])

from sklearn.metrics import silhouette_score
score_list = []
for i in range(2,11):
    kmeans = KMeans(n_clusters=i, random_state=0).fit(df_standard)
    score = silhouette_score(X, kmeans.labels_, metric='euclidean')
    score_list.append(score)

clustering_eval = pd.DataFrame({'clusters' : [2,3,4,5,6,7,8,9,10],
                  'silhouette_score' : score_list})
clustering_eval
```

clusters	silhouette_score
2	0.488740
3	0.397460
4	0.330578
5	0.270582
6	0.270596
7	0.232660
8	0.236571
9	0.232021
10	0.226689

04

Customer Personality Analysis for Marketing Retargeting



Setelah melakukan clustering dan menghaslkan 4 clustering yang berbeda disini saya melakukan analisis terhadap hasil clustering tersebut. Dilihat dari statistik yang ada, 4 cluster tersebut terbagi menjadi cluster **High Spender, Mid Spender, Low Spender,** dan **Risk of Churn**. Pada bagian ini akan dijelaskan karakteristik masing-masing cluster.

VISUALISASI & ANALISIS

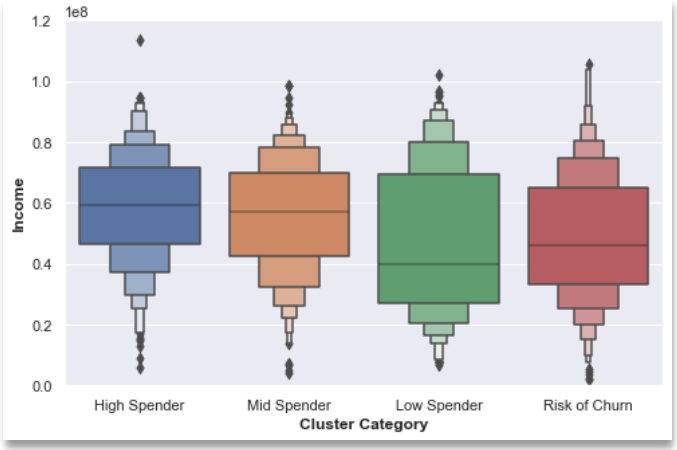
Dalam melakukan analisis clustering customer ini saya membuat beberapa visualisasi dari hasil statistik yang ada. Analisis tersebut meliputi jumlah customer, rata-rata usia, income, conversion rate, dll.

Total User per Cluster

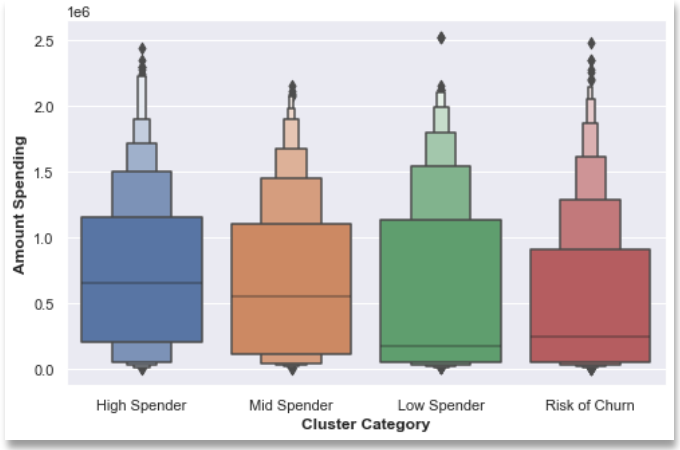
Average Age per Cluster

cluster_category	Average Age
High Spender	70.730310
Mid Spender	59.699443
Risk of Churn	48.436234
Low Spender	36.739857

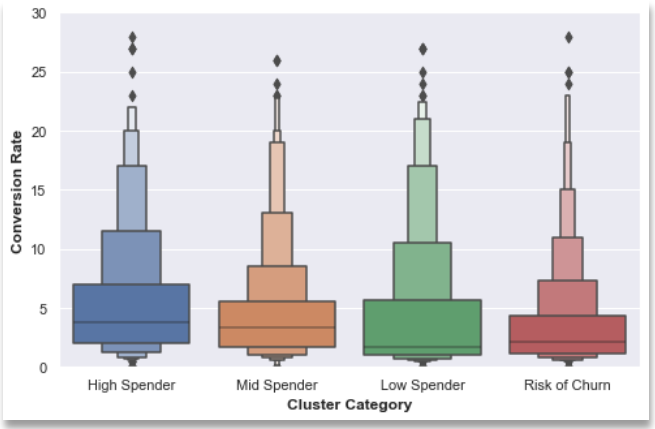
Income per Cluster



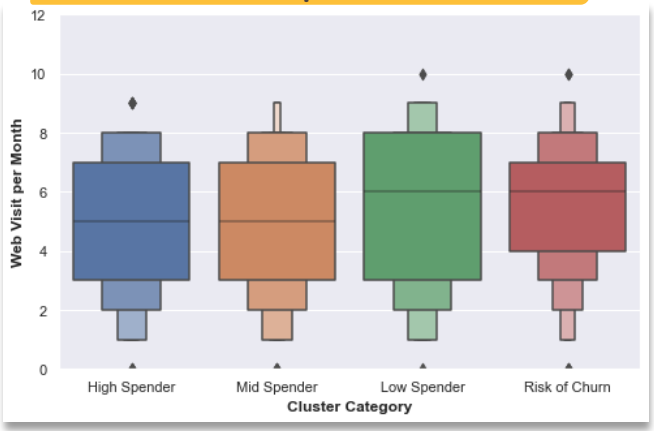
Amount Spending per Cluster



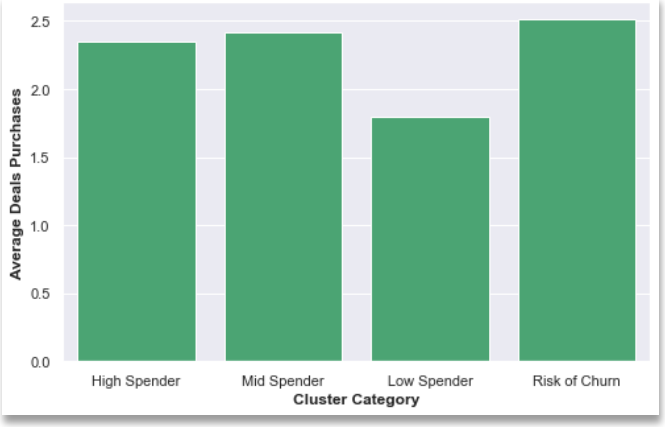
Conversion Rate per Cluster



Web Visit per Cluster



Average Deals Purchase per Cluster



INTERPRETATION SUMMARY

1 **High Spender :** Kelompok ini memiliki jumlah customer terkecil dengan rata-rata usia tertinggi. Kelompok customer ini memiliki Income dan total spending yang paling besar dibandingkan dengan kelompok cluster yang lainnya. Meskipun cluster ini cukup jarang melakukan visit web, namun cluster ini merupakan kelompok dengan conversion rate terbesar. Sangat tepat bagi perusahaan untuk melakukan retarget marketing ke dalam cluster ini.

2 **Mid Spender :** Kelompok ini memiliki jumlah customer terbesar kedua dengan rata-rata usia customer 59th. Karakteristik pada cluster ini cukup mirip dengan cluster high spender. Hanya saja jika dilihat dari income dan total spending, cluster ini memiliki nilai yang lebih rendah. Meskipun begitu walaupun kelompok ini lebih jarang melakukan visit web, cluster ini cukup sering merespon campaign dilihat dari conversion rate yang cukup tinggi.

3 **Low Spender :** Kelompok ini memiliki jumlah customer yang cukup kecil dengan rata-rata usia paling rendah yaitu 36 tahun. Kelompok ini memiliki income terkecil dan total spending terkecil kedua dibandingkan dengan kelompok lainnya. Customer ini memiliki karakteristik cukup sering mengunjungi web namun tidak banyak berbelanja. Customer di cluster ini juga tidak banyak merespon campaign dilihat dari conversion rate yang cukup kecil dibandingkan dengan cluster lainnya.

4 **Risk of Churn :** Kelompok ini memiliki jumlah customer yang paling banyak dibandingkan dengan cluster lainnya. Tetapi total spending dari kelompok ini adalah yang paling kecil dibandingkan kelompok lainnya. Dalam melakukan pembelian, cluster ini paling sering melakukan pembelian menggunakan promo. Customer di kelompok ini adalah yang paling sering mengunjungi web namun juga paling sedikit merespon campaign dilihat dari rata-rata conversion rate yang paling kecil dibandingkan dengan cluster lainnya.

RECOMMENDATION & POTENTIAL IMPACT

1. Untuk kelompok **Low Spender** dan **Risk of Churn** perlu dianalisis lebih lanjut mengapa pada dua kelompok ini customernya sering mengunjungi web namun tidak melakukan pembelian. Akan disayangkan apabila nantinya customer dari kelompok ini benar-benar churn karena jumlah customer pada kedua cluster ini cukup tinggi. Jika sebanyak 5% saja pelanggan dari kalangan Risk of Churn dan 5% pelanggan dari cluster Low Spender memilih untuk churn, revenue perusahaan berpotensi berkurang sebesar 33 juta. Apabila hal tersebut terjadi maka dapat merugikan perusahaan.

2. Untuk kalangan **Mid Spender** juga perlu analisis lebih lanjut mengenai pemberian promosi yang cocok dengan karakteristik customer. Bagaimana agar customer pada kelompok ini dapat meningkatkan transaksi, serta bagaimana customer dari kelompok ini tetap berbelanja di platform perusahaan dan tidak churn. Dengan mempertahankan kelompok ini, perusahaan dapat tetap menghasilkan revenue hingga 360 juta.

3. Perusahaan harus fokus untuk meningkatkan service bisa dengan memberikan free gift atau dengan cara lain untuk customer dari kalangan **High Spender** supaya customer di kelompok ini tetap memilih berbelanja di perusahaan dan tidak churn. Untuk marketing selanjutnya, menargetkan kelompok high spender merupakan pilihan yang tepat. Karena sesuai dengan interpretasi sebelumnya, customer dari kalangan ini cenderung lebih sering merespon campaign. Sehingga akan memberikan revenue lebih besar ke perusahaan. Dengan mempertahankan kelompok ini, perusahaan dapat menghasilkan revenue hingga 308 juta

Thank You!

This presentation is created by:
Dewi Ayu Rahmawati
ayudewi.ar@gmail.com
https://linkedin.com/in/dewiayurahmawati

Hello! I'm a Civil Engineering graduate from ITS. I'm a person who love to learn and start learning Data Science by joining Rakamin Bootcamp in early 2022. Now I want to shifting my career and looking for opportunities into the data field.