marketing_analysis

April 14, 2024

1 Goals

Optimasi Campaign Results terkait penawaran produk baru, sehingga mendapatkan profit yang maksimal di bulan depan dan memahami karakteristik customers yang menerima penawaran produk baru tersebut.

1.1 Data Quality Check

- Q1: Apakah ada features yang data type-nya tidak sesuai?
- Q2: Adakah duplicate, missing values atau outliers? Bagaimana mengatasi hal-hal tersebut?
- Q3: Apakah terdapat distribusi yang tidak masuk akal?

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

pd.set_option('display.max_columns',None)
```

[2]:
```
df = pd.read_csv("marketing_data.csv")
df.head()
```

[2]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	\
0	1826	1970	Graduation	Divorced	$84,835.00	0	
1	1	1961	Graduation	Single	$57,091.00	0	
2	10476	1958	Graduation	Married	$67,267.00	0	
3	1386	1967	Graduation	Together	$32,474.00	1	
4	5371	1989	Graduation	Single	$21,474.00	1	

	Teenhome	Dt_Customer	Recency	MntWines	MntFruits	MntMeatProducts	\
0	0	6/16/14	0	189	104	379	
1	0	6/15/14	0	464	5	64	
2	1	5/13/14	0	134	11	59	
3	1	5/11/14	0	10	0	1	
4	0	4/8/14	0	6	16	24	

	MntFishProducts	MntSweetProducts	MntGoldProds	NumDealsPurchases	\
0	111	189	218	1	

```
1                7                 0            37                 1
2               15                 2            30                 1
3                0                 0             0                 1
4               11                 0            34                 2

   NumWebPurchases  NumCatalogPurchases  NumStorePurchases  NumWebVisitsMonth  \
0                4                    4                  6                  1
1                7                    3                  7                  5
2                3                    2                  5                  2
3                1                    0                  2                  7
4                3                    1                  2                  7

   AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  \
0             0             0             0             0             0
1             0             0             0             0             1
2             0             0             0             0             0
3             0             0             0             0             0
4             1             0             0             0             0

   Response  Complain Country
0         1         0      SP
1         1         0      CA
2         0         0      US
3         0         0     AUS
4         1         0      SP
```

[3]:
```
print(df.shape)
df.describe()
```

```
(2240, 28)
```

[3]:
```
                ID   Year_Birth      Kidhome     Teenhome      Recency  \
count   2240.000000  2240.000000  2240.000000  2240.000000  2240.000000
mean    5592.159821  1968.805804     0.444196     0.506250    49.109375
std     3246.662198    11.984069     0.538398     0.544538    28.962453
min        0.000000  1893.000000     0.000000     0.000000     0.000000
25%     2828.250000  1959.000000     0.000000     0.000000    24.000000
50%     5458.500000  1970.000000     0.000000     0.000000    49.000000
75%     8427.750000  1977.000000     1.000000     1.000000    74.000000
max    11191.000000  1996.000000     2.000000     2.000000    99.000000

          MntWines    MntFruits  MntMeatProducts  MntFishProducts  \
count  2240.000000  2240.000000      2240.000000      2240.000000
mean    303.935714    26.302232       166.950000        37.525446
std     336.597393    39.773434       225.715373        54.628979
min       0.000000     0.000000         0.000000         0.000000
25%      23.750000     1.000000        16.000000         3.000000
50%     173.500000     8.000000        67.000000        12.000000
```

```
75%     504.250000    33.000000       232.000000        50.000000
max    1493.000000   199.000000      1725.000000       259.000000

       MntSweetProducts  MntGoldProds  NumDealsPurchases  NumWebPurchases  \
count       2240.000000   2240.000000        2240.000000      2240.000000
mean          27.062946     44.021875           2.325000         4.084821
std           41.280498     52.167439           1.932238         2.778714
min            0.000000      0.000000           0.000000         0.000000
25%            1.000000      9.000000           1.000000         2.000000
50%            8.000000     24.000000           2.000000         4.000000
75%           33.000000     56.000000           3.000000         6.000000
max          263.000000    362.000000          15.000000        27.000000

       NumCatalogPurchases  NumStorePurchases  NumWebVisitsMonth  \
count          2240.000000        2240.000000        2240.000000
mean              2.662054           5.790179           5.316518
std               2.923101           3.250958           2.426645
min               0.000000           0.000000           0.000000
25%               0.000000           3.000000           3.000000
50%               2.000000           5.000000           6.000000
75%               4.000000           8.000000           7.000000
max              28.000000          13.000000          20.000000

       AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  \
count   2240.000000   2240.000000   2240.000000   2240.000000   2240.000000
mean       0.072768      0.074554      0.072768      0.064286      0.013393
std        0.259813      0.262728      0.259813      0.245316      0.114976
min        0.000000      0.000000      0.000000      0.000000      0.000000
25%        0.000000      0.000000      0.000000      0.000000      0.000000
50%        0.000000      0.000000      0.000000      0.000000      0.000000
75%        0.000000      0.000000      0.000000      0.000000      0.000000
max        1.000000      1.000000      1.000000      1.000000      1.000000

          Response     Complain
count  2240.000000  2240.000000
mean      0.149107     0.009375
std       0.356274     0.096391
min       0.000000     0.000000
25%       0.000000     0.000000
50%       0.000000     0.000000
75%       0.000000     0.000000
max       1.000000     1.000000
```

1.1.1 Q1: Data Type Check

Apakah ada features yang data type-nya tidak sesuai?

[4]:
```
df.info()
# df.dtypes
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 28 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   ID                   2240 non-null   int64
 1   Year_Birth           2240 non-null   int64
 2   Education            2240 non-null   object
 3   Marital_Status       2240 non-null   object
 4    Income              2216 non-null   object
 5   Kidhome              2240 non-null   int64
 6   Teenhome             2240 non-null   int64
 7   Dt_Customer          2240 non-null   object
 8   Recency              2240 non-null   int64
 9   MntWines             2240 non-null   int64
 10  MntFruits            2240 non-null   int64
 11  MntMeatProducts      2240 non-null   int64
 12  MntFishProducts      2240 non-null   int64
 13  MntSweetProducts     2240 non-null   int64
 14  MntGoldProds         2240 non-null   int64
 15  NumDealsPurchases    2240 non-null   int64
 16  NumWebPurchases      2240 non-null   int64
 17  NumCatalogPurchases  2240 non-null   int64
 18  NumStorePurchases    2240 non-null   int64
 19  NumWebVisitsMonth    2240 non-null   int64
 20  AcceptedCmp3         2240 non-null   int64
 21  AcceptedCmp4         2240 non-null   int64
 22  AcceptedCmp5         2240 non-null   int64
 23  AcceptedCmp1         2240 non-null   int64
 24  AcceptedCmp2         2240 non-null   int64
 25  Response             2240 non-null   int64
 26  Complain             2240 non-null   int64
 27  Country              2240 non-null   object
dtypes: int64(23), object(5)
memory usage: 490.1+ KB
```

[5]:
```
df.columns = df.columns.str.replace(' ','')

df['Income'] = df['Income'].str.replace('$','')
df['Income'] = df['Income'].str.replace(',','')
df['Income'] = df['Income'].astype('float')

df['Dt_Customer'] = pd.to_datetime(df['Dt_Customer'])
```

```
C:\Users\M S I\AppData\Local\Temp\ipykernel_19704\1509624653.py:7: UserWarning:
```

```
Could not infer format, so each element will be parsed individually, falling
back to `dateutil`. To ensure parsing is consistent and as-expected, please
specify a format.
  df['Dt_Customer'] = pd.to_datetime(df['Dt_Customer'])
```

[13]:
```
for col, dtype in zip(df.columns,df.dtypes):
    print(f'{col} [{dtype}]: {df[col].isna().sum()} null')
    if dtype =='int64' or dtype=='float64':
        print(df[col].min(),'-',df[col].max())
    else:
        print(df[col].unique())
    print()
```

```
ID [int64]: 0 null
0 - 11191

Year_Birth [int64]: 0 null
1893 - 1996

Education [object]: 0 null
['Graduation' 'PhD' '2n Cycle' 'Master' 'Basic']

Marital_Status [object]: 0 null
['Divorced' 'Single' 'Married' 'Together' 'Widow' 'YOLO' 'Alone' 'Absurd']

Income [float64]: 24 null
1730.0 - 666666.0

Kidhome [int64]: 0 null
0 - 2

Teenhome [int64]: 0 null
0 - 2

Dt_Customer [datetime64[ns]]: 0 null
<DatetimeArray>
['2014-06-16 00:00:00', '2014-06-15 00:00:00', '2014-05-13 00:00:00',
 '2014-05-11 00:00:00', '2014-04-08 00:00:00', '2014-03-17 00:00:00',
 '2014-01-29 00:00:00', '2014-01-18 00:00:00', '2014-01-11 00:00:00',
 '2013-12-27 00:00:00',
 …
 '2013-06-26 00:00:00', '2013-05-24 00:00:00', '2012-08-15 00:00:00',
 '2013-12-10 00:00:00', '2012-09-19 00:00:00', '2012-08-10 00:00:00',
 '2014-06-06 00:00:00', '2013-05-25 00:00:00', '2013-04-09 00:00:00',
 '2012-09-01 00:00:00']
Length: 663, dtype: datetime64[ns]

Recency [int64]: 0 null
```

0 - 99

MntWines [int64]: 0 null
0 - 1493

MntFruits [int64]: 0 null
0 - 199

MntMeatProducts [int64]: 0 null
0 - 1725

MntFishProducts [int64]: 0 null
0 - 259

MntSweetProducts [int64]: 0 null
0 - 263

MntGoldProds [int64]: 0 null
0 - 362

NumDealsPurchases [int64]: 0 null
0 - 15

NumWebPurchases [int64]: 0 null
0 - 27

NumCatalogPurchases [int64]: 0 null
0 - 28

NumStorePurchases [int64]: 0 null
0 - 13

NumWebVisitsMonth [int64]: 0 null
0 - 20

AcceptedCmp3 [int64]: 0 null
0 - 1

AcceptedCmp4 [int64]: 0 null
0 - 1

AcceptedCmp5 [int64]: 0 null
0 - 1

AcceptedCmp1 [int64]: 0 null
0 - 1

AcceptedCmp2 [int64]: 0 null

```
0 - 1

Response [int64]: 0 null
0 - 1

Complain [int64]: 0 null
0 - 1

Country [object]: 0 null
['SP' 'CA' 'US' 'AUS' 'GER' 'IND' 'SA' 'ME']
```

1.1.2 Q2: Dirty Data Check

duplicate, missing values atau outliers?

[ ]: