campaign-hypotest

April 23, 2024

[4]:
```
''' Importing the rquired libraries for analysis'''


import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns


''' Readig the csv file 'campaign - cmapaign' to df_cap '''


df_cap = pd.read_csv('campaign - campaign.csv')
df_cap.head()
```

[4]:

	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	…	NumCatalogPurchases	\
0	0	6/16/14	0	189	…	4	
1	0	6/15/14	0	464	…	3	
2	1	5/13/14	0	134	…	2	
3	1	5/11/14	0	10	…	0	
4	0	4/8/14	0	6	…	1	

	NumStorePurchases	NumWebVisitsMonth	AcceptedCmp3	AcceptedCmp4	\
0	6	1	0	0	
1	7	5	0	0	
2	5	2	0	0	
3	2	7	0	0	
4	2	7	1	0	

	AcceptedCmp5	AcceptedCmp1	AcceptedCmp2	Complain	Country

```
0              0              0              0         0        SP
1              0              0              1         0        CA
2              0              0              0         0        US
3              0              0              0         0       AUS
4              0              0              0         0        SP

[5 rows x 27 columns]
```

[4]:

[5]:
```
''' All columns in the datafile '''

df_cap.columns
```

[5]:
```
Index(['ID', 'Year_Birth', 'Education', 'Marital_Status', 'Income', 'Kidhome',
       'Teenhome', 'Dt_Customer', 'Recency', 'MntWines', 'MntFruits',
       'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',
       'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',
       'AcceptedCmp2', 'Complain', 'Country'],
      dtype='object')
```

Here, we are converting the different campaign columns to singke column of camoaignAcceptance and their paralle value to as 'value' by using pd.melt() function and returning the data to new dataframe pf 'df_cap1'.

[6]:
```
''' Checking for null values in dataframe '''

df_cap.isnull().sum()
```

[6]:
```
ID                   0
Year_Birth           0
Education            0
Marital_Status       0
Income               0
Kidhome              0
Teenhome             0
Dt_Customer          0
Recency              0
MntWines             0
MntFruits            0
MntMeatProducts      0
MntFishProducts      0
MntSweetProducts     0
MntGoldProds         0
NumDealsPurchases    0
```

```
NumWebPurchases        0
NumCatalogPurchases    0
NumStorePurchases      0
NumWebVisitsMonth      0
AcceptedCmp3           0
AcceptedCmp4           0
AcceptedCmp5           0
AcceptedCmp1           0
AcceptedCmp2           0
Complain               0
Country                0
dtype: int64
```

[7]:
```
''' dropping the null values for column income '''

df_cap['Income'].dropna(inplace = True,how = 'all')
```

0.0.1 Q: Is income of Customer dependent on their Education level?

[8]:
```
''' querying out the columns income and education '''

df_cap['Income'].isna().sum()
```

[8]:
```
0
```

[9]:
```
''' checking for the data types of the columns '''

df_cap.dtypes
```

[9]:
```
ID                    int64
Year_Birth            int64
Education            object
Marital_Status       object
Income               object
Kidhome               int64
Teenhome              int64
Dt_Customer          object
Recency               int64
MntWines              int64
MntFruits             int64
MntMeatProducts       int64
MntFishProducts       int64
MntSweetProducts      int64
MntGoldProds          int64
NumDealsPurchases     int64
NumWebPurchases       int64
NumCatalogPurchases   int64
```

```
NumStorePurchases       int64
NumWebVisitsMonth       int64
AcceptedCmp3            int64
AcceptedCmp4            int64
AcceptedCmp5            int64
AcceptedCmp1            int64
AcceptedCmp2            int64
Complain                int64
Country                object
dtype: object
```

[10]:
```
''' checking the number of values in column Education.'''

df_cap['Education'].value_counts()
```

[10]:
```
Education
Graduation    1126
PhD            486
Master         370
2n Cycle       203
Basic           54
Name: count, dtype: int64
```

[11]:
```
''' Removing the '$' symbol from income to convert it to float value '''

df_cap['Income'] = df_cap['Income'].str.slice(1)
```

[12]:
```
''' Removing the comma(,) from ncome column '''

df_cap["Income"] = df_cap['Income'].str.replace(',','')
```

[13]:
```
''' Changing the datatype of column income '''

df_cap['Income'] = df_cap["Income"].astype(float)
```

[14]:
```
''' Querying both columns education and income '''

df_cap['Income'],df_cap['Education']
```

[14]:
```
(0       84835.0
 1       57091.0
 2       67267.0
 3       32474.0
 4       21474.0
          ...
 2234    66476.0
 2235    31056.0
```

```
2236    46310.0
2237    65819.0
2238    94871.0
Name: Income, Length: 2239, dtype: float64,
0       Graduation
1       Graduation
2       Graduation
3       Graduation
4       Graduation
           …
2234           PhD
2235      2n Cycle
2236    Graduation
2237    Graduation
2238           PhD
Name: Education, Length: 2239, dtype: object)
```

[15]:
```
''' Taking sample of 20 incomes where educaton level is graduation '''

grd = df_cap[df_cap['Education'] == 'Graduation']['Income'].sample(20)
grd
```

[15]:
```
236     51287.0
1093    74190.0
1569    79593.0
1359    75922.0
949     28587.0
666     65486.0
1397    73803.0
985     42664.0
411     61014.0
1231    28647.0
1346    40321.0
922     46734.0
1845    68743.0
115     67225.0
1605    57959.0
1743    40800.0
1493    74293.0
1295    60714.0
1022    77298.0
1660    70503.0
Name: Income, dtype: float64
```

[16]:
```
''' Taking sample of 20 incomes where educaton level is basic '''
bsc = df_cap[df_cap['Education'] == 'Basic']["Income"].sample(20)
bsc
```

[16]:
```
688     25965.0
318     15253.0
1190    24882.0
2019    26997.0
984     13724.0
957     16014.0
1165    16581.0
1715    15056.0
146     20425.0
674     13084.0
1792    28249.0
1866    25443.0
165     24279.0
1060    22634.0
1485    23724.0
147     20425.0
1114    28389.0
1038    24480.0
1175    26868.0
590      8940.0
Name: Income, dtype: float64
```

[17]:
```
''' Taking sample of 20 incomes where educaton level is phd'''

phd = df_cap[df_cap['Education'] == 'PhD']['Income'].sample(20)
phd
```

[17]:
```
549     35946.0
295     80336.0
1130    36930.0
1653    74637.0
1610    46757.0
1217    82032.0
409     69389.0
1565    61467.0
277     41003.0
1839    64504.0
1215    52569.0
924     52869.0
1330    38443.0
1298    80427.0
628     87171.0
1808    68117.0
41      66465.0
1811    58086.0
1104    37929.0
1822    46854.0
```

```
Name: Income, dtype: float64
```

[18]:
```
''' Taking sample of 20 incomes where educaton level is 2n cycle '''

n2c = df_cap[df_cap['Education'] == ('2n Cycle')]['Income'].sample(20)
n2c
```

[18]:
```
2000    46772.0
1953    33812.0
903     62972.0
44      25959.0
878     82347.0
2221     7500.0
555     50334.0
1380    49514.0
1430    74859.0
695     82326.0
36      65370.0
1731    59060.0
1705    21282.0
2089    45204.0
1712    23718.0
342     74805.0
1832    21955.0
1677    54342.0
1297    83257.0
141     85710.0
Name: Income, dtype: float64
```

[19]:
```
''' Taking sample of 20 incomes where educaton level is masters '''

mas = df_cap[df_cap['Education']== 'Master']['Income'].sample(20)
mas
```

[19]:
```
546     41335.0
565     88097.0
1659    40101.0
1377    60432.0
1674    45143.0
117     65104.0
780     10979.0
118     81698.0
2093    47353.0
868     70053.0
601     82584.0
1450    63841.0
1076    57136.0
```

```
562     33444.0
1617    50353.0
375     54197.0
106     62845.0
1553    42394.0
224     39763.0
1996    50898.0
Name: Income, dtype: float64
```

here, we were asked whether the income of customer is dependent on their education level.

So we formulate a hypothesis test for the given data and check whether we can reject null or we can't reject null.

1. Null hypothesis - (H0): Income of visitor doesnot dependent on education
2. Alternative hypothesis - (Ha): Income of visitor dependent on eduction
3. significance level = 0.05

Below ,If the p_value(probablilty value) we get in the test result is lower than this significance level(0.05), w ecan reject null hyppthesis else we cannot reject null hypothesis and accpet the null.

[20]:
```
from scipy.stats import ttest_ind

#1.
ttest_ind(grd,bsc,alternative = 'two-sided')
```

[20]:
```
TtestResult(statistic=9.850828192362206, pvalue=5.175649675671145e-12, df=38.0)
```

[21]:
```
#2.
ttest_ind(phd,bsc,alternative = 'two-sided')
```

[21]:
```
TtestResult(statistic=9.572480293163347, pvalue=1.1335961859604426e-11, df=38.0)
```

[22]:
```
#3.
ttest_ind(n2c,bsc,alternative = 'two-sided')
```

[22]:
```
TtestResult(statistic=5.672343847957485, pvalue=1.6027176697902599e-06, df=38.0)
```

[23]:
```
#4.
ttest_ind(mas,bsc,alternative = 'two-sided')
```

[23]:
```
TtestResult(statistic=7.704844333084001, pvalue=2.7937587088386193e-09, df=38.0)
```

In Above, teste we compare dthe sample of 20 salries of education level of basic people to eduaction level of people who are higher like(PhD,masters,graduation,2nc)

We performed ttest_ind between two groups per each education level to check whther customer income is dependent on education level.

In all four tests(1,2,3,4) the p value came out to be lesser than the significance value of 0.05

So we can reject the null hypothesis and conclude that income of customers depends on Education level.

[23]:

0.0.2 Do higher income people spend more (take in account spending in all categories together)

[24]:
```
''' making a copy of dataframe '''

df_cap3 = df_cap.copy()
df_cap3
```

[24]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	\
0	1826	1970	Graduation	Divorced	84835.0	0	
1	1	1961	Graduation	Single	57091.0	0	
2	10476	1958	Graduation	Married	67267.0	0	
3	1386	1967	Graduation	Together	32474.0	1	
4	5371	1989	Graduation	Single	21474.0	1	
…	…	…	…	…	…	…	
2234	10142	1976	PhD	Divorced	66476.0	0	
2235	5263	1977	2n Cycle	Married	31056.0	1	
2236	22	1976	Graduation	Divorced	46310.0	1	
2237	528	1978	Graduation	Married	65819.0	0	
2238	4070	1969	PhD	Married	94871.0	0	

	Teenhome	Dt_Customer	Recency	MntWines	…	NumCatalogPurchases	\
0	0	6/16/14	0	189	…	4	
1	0	6/15/14	0	464	…	3	
2	1	5/13/14	0	134	…	2	
3	1	5/11/14	0	10	…	0	
4	0	4/8/14	0	6	…	1	
…	…	…	…	…	…	…	
2234	1	3/7/13	99	372	…	2	
2235	0	1/22/13	99	5	…	0	
2236	0	12/3/12	99	185	…	1	
2237	0	11/29/12	99	267	…	4	
2238	2	9/1/12	99	169	…	5	

	NumStorePurchases	NumWebVisitsMonth	AcceptedCmp3	AcceptedCmp4	\
0	6	1	0	0	
1	7	5	0	0	
2	5	2	0	0	
3	2	7	0	0	
4	2	7	1	0	
…	…	…	…	…	
2234	11	4	0	0	

```
2235                3                8            0             0
2236                5                8            0             0
2237               10                3            0             0
2238                4                7            0             1

      AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Country
0                0             0             0         0       SP
1                0             0             1         0       CA
2                0             0             0         0       US
3                0             0             0         0      AUS
4                0             0             0         0       SP
…              …             …             …         …       …
2234             0             0             0         0       US
2235             0             0             0         0       SP
2236             0             0             0         0       SP
2237             0             0             0         0      IND
2238             1             0             0         0       CA

[2239 rows x 27 columns]
```

[25]:
```
''' Finding median of income '''

med  = df_cap3['Income'].median()
med
```

[25]:
```
51373.0
```

[26]:
```
''' Making groups as lower and higher income groups '''

df_cap3['inc_grp'] = df_cap3['Income'].apply(lambda x : 'Higher' if x > med␣
 ↪else 'lower')
```

[27]:
```
''' creating df of high income gorup '''

hg = df_cap3[df_cap3['inc_grp'] == 'Higher']
```

[28]:
```
''' creating df of lower income group '''

lg = df_cap3[df_cap3['inc_grp'] == 'lower']
```

[29]:
```
lg.head()
```

[29]:
```
      ID  Year_Birth   Education Marital_Status   Income  Kidhome  Teenhome  \
3   1386        1967  Graduation       Together  32474.0        1         1
4   5371        1989  Graduation         Single  21474.0        1         0
7   1991        1967  Graduation       Together  44931.0        0         1
13  2964        1981  Graduation        Married  26872.0        0         0
```

```
14  10311        1969  Graduation        Married   4428.0        0         1

   Dt_Customer  Recency  MntWines  …  NumStorePurchases  NumWebVisitsMonth  \
3      5/11/14        0        10  …                  2                  7
4       4/8/14        0         6  …                  2                  7
7      1/18/14        0        78  …                  3                  5
13    10/16/13        0         3  …                  2                  6
14     10/5/13        0        16  …                  0                  1

    AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  \
3              0             0             0             0             0
4              1             0             0             0             0
7              0             0             0             0             0
13             0             0             0             0             0
14             0             0             0             0             0

    Complain  Country  inc_grp
3          0      AUS    lower
4          0       SP    lower
7          0       SP    lower
13         0       CA    lower
14         0       SP    lower

[5 rows x 28 columns]
```

[30]:
```
lg.columns
```

[30]:
```
Index(['ID', 'Year_Birth', 'Education', 'Marital_Status', 'Income', 'Kidhome',
       'Teenhome', 'Dt_Customer', 'Recency', 'MntWines', 'MntFruits',
       'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',
       'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',
       'AcceptedCmp2', 'Complain', 'Country', 'inc_grp'],
      dtype='object')
```

[31]:
```
''' pivoting the categorical spends to single column as catgory and value as␣
 ↪spent for lower income group'''


lg1 = pd.melt(lg,id_vars = ['ID', 'Year_Birth', 'Education', 'Marital_Status',␣
 ↪'Income', 'Kidhome',
       'Teenhome', 'Dt_Customer', 'Recency','NumDealsPurchases',␣
 ↪'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',
```

```
      'AcceptedCmp2', 'Complain', 'Country', 'inc_grp'],var_name =␣
↪'category',value_name = 'spent')
```

[32]:
```
lg1
```

[32]:
```
          ID  Year_Birth   Education Marital_Status   Income  Kidhome  \
0       1386        1967  Graduation       Together  32474.0        1
1       5371        1989  Graduation         Single  21474.0        1
2       1991        1967  Graduation       Together  44931.0        0
3       2964        1981  Graduation        Married  26872.0        0
4      10311        1969  Graduation        Married   4428.0        0
…        …          …          …              …       …       …
6787    8595        1973  Graduation          Widow  42429.0        0
6788    7232        1973  Graduation          Widow  42429.0        0
6789    7829        1900    2n Cycle       Divorced  36640.0        1
6790    5263        1977    2n Cycle        Married  31056.0        1
6791      22        1976  Graduation       Divorced  46310.0        1

      Teenhome Dt_Customer  Recency  NumDealsPurchases  …  AcceptedCmp3  \
0            1     5/11/14        0                  1  …             0
1            0      4/8/14        0                  2  …             1
2            1     1/18/14        0                  1  …             0
3            0    10/16/13        0                  1  …             0
4            1     10/5/13        0                  0  …             0
…          …          …      …                  …  …           …
6787         1     2/11/14       99                  2  …             0
6788         1     2/11/14       99                  2  …             0
6789         0     9/26/13       99                  1  …             0
6790         0     1/22/13       99                  1  …             0
6791         0     12/3/12       99                  2  …             0

      AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  \
0                0             0             0             0         0
1                0             0             0             0         0
2                0             0             0             0         0
3                0             0             0             0         0
4                0             0             0             0         0
…              …             …             …             …       …
6787             0             0             0             0         0
6788             0             0             0             0         0
6789             0             0             0             0         1
6790             0             0             0             0         0
6791             0             0             0             0         0

      Country  inc_grp      category  spent
0         AUS    lower      MntWines     10
1          SP    lower      MntWines      6
```

```
2          SP    lower      MntWines     78
3          CA    lower      MntWines      3
4          SP    lower      MntWines     16
…       …       …             …     …
6787      AUS    lower  MntGoldProds      4
6788       SP    lower  MntGoldProds      4
6789      IND    lower  MntGoldProds     25
6790       SP    lower  MntGoldProds     16
6791       SP    lower  MntGoldProds     14

[6792 rows x 24 columns]
```

[33]:
```
''' pivoting the categorical spends to single column as catgory and value as␣
 ↪spent for higher income group'''


hg1 = pd.melt(hg, id_vars = ['ID', 'Year_Birth', 'Education', 'Marital_Status',␣
 ↪'Income', 'Kidhome',
       'Teenhome', 'Dt_Customer', 'Recency','NumDealsPurchases',␣
 ↪'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',
       'AcceptedCmp2', 'Complain', 'Country', 'inc_grp'],var_name =␣
 ↪'category',value_name = 'spent')
```

[34]:
```
hg1
```

[34]:
```
          ID  Year_Birth   Education Marital_Status   Income  Kidhome  \
0       1826        1970  Graduation       Divorced  84835.0        0
1          1        1961  Graduation         Single  57091.0        0
2      10476        1958  Graduation        Married  67267.0        0
3       7348        1958         PhD         Single  71691.0        0
4       4073        1954    2n Cycle        Married  63564.0        0
…      …          …          …              …       …       …
6637    2415        1962  Graduation       Together  62568.0        0
6638    9977        1973  Graduation       Divorced  78901.0        0
6639   10142        1976         PhD       Divorced  66476.0        0
6640     528        1978  Graduation        Married  65819.0        0
6641    4070        1969         PhD        Married  94871.0        0

      Teenhome Dt_Customer  Recency  NumDealsPurchases  …  AcceptedCmp3  \
0            0     6/16/14        0                  1  …             0
1            0     6/15/14        0                  1  …             0
2            1     5/13/14        0                  1  …             0
3            0     3/17/14        0                  1  …             0
4            0     1/29/14        0                  1  …             1
…        …          …      …                …  …           …
```

```
6637         1      4/7/14       99                 3  …             0
6638         1     9/17/13       99                 3  …             0
6639         1      3/7/13       99                 2  …             0
6640         0    11/29/12       99                 1  …             0
6641         2      9/1/12       99                 1  …             0

      AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  \
0                0             0             0             0         0
1                0             0             0             1         0
2                0             0             0             0         0
3                0             0             0             0         0
4                0             0             0             0         0
…              …             …             …             …         …
6637             0             0             1             0         0
6638             0             0             0             0         0
6639             0             0             0             0         0
6640             0             0             0             0         0
6641             1             1             0             0         0

      Country  inc_grp      category  spent
0          SP   Higher      MntWines    189
1          CA   Higher      MntWines    464
2          US   Higher      MntWines    134
3          SP   Higher      MntWines    336
4         GER   Higher      MntWines    769
…       …       …           …     …
6637       SP   Higher  MntGoldProds     61
6638       US   Higher  MntGoldProds     34
6639       US   Higher  MntGoldProds     78
6640      IND   Higher  MntGoldProds     63
6641       CA   Higher  MntGoldProds    144

[6642 rows x 24 columns]
```

[35]:
```
''' taking sample of 20 values for spent values for higher income group '''

high20 = hg1['spent'].sample(20)
```

[36]:
```
high20
```

[36]:
```
198      28
4700    126
1734     12
6238     40
2033     61
5768    248
46       42
```

```
5118    103
2339    168
6545    155
3153    322
5061     46
4288     69
5378     67
3484    216
1702     91
1638     37
4368    120
3334      8
5406      6
Name: spent, dtype: int64
```

[37]:
```
''' taking sample of 20 values for spent values for lower income group '''

low20 = lg1['spent'].sample(20)
low20
```

[37]:
```
4668    44
3735     1
5180     1
3502    28
1134     0
5966     2
5957    14
2655     1
5250     2
6514    36
1273     0
5454     0
3428     3
1893     0
2669     2
3843     8
6064     4
4092    58
1161     0
1659    11
Name: spent, dtype: int64
```

1. Here, we separetd the income range by taking median of the income from data froame and separeted the income as two groups ‘lower’ and ;‘higher’
2. we have taken samples of 20 and conduct hypothesis testing of ttet individual and compare with low income group with high income groupto see whether higher income group spent more money.

3. H0: high income group has same amount spent along with low income group ha: high income group sents more money than low and medium income group.
4. let us consider the significance level = 0.05

[38]:
```
''' Conducting a hypothesis test for checking the hypothesis formed '''

ttest_ind(high20,low20,alternative = 'greater')
```

[38]:
```
TtestResult(statistic=4.455828080351984, pvalue=3.573320763312415e-05, df=38.0)
```

1. Here on both ttest results where p value is less than our considered significance level of high income group on test with low income group.
2. So, we can reject the null hypothesis and conclude that high income group visitors spends more money.

[38]:
```

```

0.0.3 do couples pent more or less on wines than people living alone.

[39]:
```
df_cap2 = df_cap.copy()
df_cap2
```

[39]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	\
0	1826	1970	Graduation	Divorced	84835.0	0	
1	1	1961	Graduation	Single	57091.0	0	
2	10476	1958	Graduation	Married	67267.0	0	
3	1386	1967	Graduation	Together	32474.0	1	
4	5371	1989	Graduation	Single	21474.0	1	
…	…	…	…	…	…	…	
2234	10142	1976	PhD	Divorced	66476.0	0	
2235	5263	1977	2n Cycle	Married	31056.0	1	
2236	22	1976	Graduation	Divorced	46310.0	1	
2237	528	1978	Graduation	Married	65819.0	0	
2238	4070	1969	PhD	Married	94871.0	0	

	Teenhome	Dt_Customer	Recency	MntWines	…	NumCatalogPurchases	\
0	0	6/16/14	0	189	…	4	
1	0	6/15/14	0	464	…	3	
2	1	5/13/14	0	134	…	2	
3	1	5/11/14	0	10	…	0	
4	0	4/8/14	0	6	…	1	
…	…	…	…	…	…	…	
2234	1	3/7/13	99	372	…	2	
2235	0	1/22/13	99	5	…	0	
2236	0	12/3/12	99	185	…	1	
2237	0	11/29/12	99	267	…	4	
2238	2	9/1/12	99	169	…	5	

```
      NumStorePurchases  NumWebVisitsMonth  AcceptedCmp3  AcceptedCmp4  \
0                     6                  1             0             0
1                     7                  5             0             0
2                     5                  2             0             0
3                     2                  7             0             0
4                     2                  7             1             0
…                   …                  …             …             …
2234                 11                  4             0             0
2235                  3                  8             0             0
2236                  5                  8             0             0
2237                 10                  3             0             0
2238                  4                  7             0             1

      AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Country
0                0             0             0         0       SP
1                0             0             1         0       CA
2                0             0             0         0       US
3                0             0             0         0      AUS
4                0             0             0         0       SP
…              …             …             …         …       …
2234             0             0             0         0       US
2235             0             0             0         0       SP
2236             0             0             0         0       SP
2237             0             0             0         0      IND
2238             1             0             0         0       CA

[2239 rows x 27 columns]
```

[40]:
```
''' checking number of customer for marital status '''

df_cap2['Marital_Status'].value_counts()
```

[40]:
```
Marital_Status
Married     864
Together    579
Single      480
Divorced    232
Widow        77
Alone         3
YOLO          2
Absurd        2
Name: count, dtype: int64
```

[41]:
```
''' function describing creating two groups of marital status

Here we considered mariied and toegther customers as incouple group
```

```
rest of all as alone group

'''

def fun(x):
  if x == 'Married' or x == 'Together':
    x = 'In_couple'
  else:
    x = 'Alone'
  return x
```

[42]:
```
''' Applying the function(fun) of groups created '''

df_cap2['Marital_Status'] = df_cap2['Marital_Status'].apply(fun)
```

[43]:
```
df_cap2
```

[43]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	\
0	1826	1970	Graduation	Alone	84835.0	0	
1	1	1961	Graduation	Alone	57091.0	0	
2	10476	1958	Graduation	In_couple	67267.0	0	
3	1386	1967	Graduation	In_couple	32474.0	1	
4	5371	1989	Graduation	Alone	21474.0	1	
…	…	…	…	…	…	…	
2234	10142	1976	PhD	Alone	66476.0	0	
2235	5263	1977	2n Cycle	In_couple	31056.0	1	
2236	22	1976	Graduation	Alone	46310.0	1	
2237	528	1978	Graduation	In_couple	65819.0	0	
2238	4070	1969	PhD	In_couple	94871.0	0	

	Teenhome	Dt_Customer	Recency	MntWines	…	NumCatalogPurchases	\
0	0	6/16/14	0	189	…	4	
1	0	6/15/14	0	464	…	3	
2	1	5/13/14	0	134	…	2	
3	1	5/11/14	0	10	…	0	
4	0	4/8/14	0	6	…	1	
…	…	…	…	…	…	…	
2234	1	3/7/13	99	372	…	2	
2235	0	1/22/13	99	5	…	0	
2236	0	12/3/12	99	185	…	1	
2237	0	11/29/12	99	267	…	4	
2238	2	9/1/12	99	169	…	5	

	NumStorePurchases	NumWebVisitsMonth	AcceptedCmp3	AcceptedCmp4	\
0	6	1	0	0	
1	7	5	0	0	
2	5	2	0	0	

```
3                    2                   7             0             0
4                    2                   7             1             0
…                  …                   …             …             …
2234                11                   4             0             0
2235                 3                   8             0             0
2236                 5                   8             0             0
2237                10                   3             0             0
2238                 4                   7             0             1

      AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Country
0                0             0             0         0       SP
1                0             0             1         0       CA
2                0             0             0         0       US
3                0             0             0         0      AUS
4                0             0             0         0       SP
…              …             …             …         …       …
2234             0             0             0         0       US
2235             0             0             0         0       SP
2236             0             0             0         0       SP
2237             0             0             0         0      IND
2238             1             0             0         0       CA

[2239 rows x 27 columns]
```

[44]:
```
''' fetching out columns of group in couple '''


df_cap2[df_cap2['Marital_Status'] == 'In_couple']
```

[44]:
```
         ID  Year_Birth   Education Marital_Status   Income  Kidhome  \
2     10476        1958  Graduation      In_couple  67267.0        0
3      1386        1967  Graduation      In_couple  32474.0        1
6      4073        1954    2n Cycle      In_couple  63564.0        0
7      1991        1967  Graduation      In_couple  44931.0        0
8      4047        1954         PhD      In_couple  65324.0        0
…     …          …          …             …       …      …
2228   2106        1974    2n Cycle      In_couple  20130.0        0
2229   3363        1974    2n Cycle      In_couple  20130.0        0
2235   5263        1977    2n Cycle      In_couple  31056.0        1
2237    528        1978  Graduation      In_couple  65819.0        0
2238   4070        1969         PhD      In_couple  94871.0        0

      Teenhome Dt_Customer  Recency  MntWines  …  NumCatalogPurchases  \
2            1     5/13/14        0       134  …                    2
3            1     5/11/14        0        10  …                    0
6            0     1/29/14        0       769  …                   10
7            1     1/18/14        0        78  …                    1
```

```
8            1     1/11/14        0       384  …                    2
…          …          …       …       …   …                    …
2228         0     3/17/14       99         0  …                    0
2229         0     3/17/14       99         0  …                    0
2235         0     1/22/13       99         5  …                    0
2237         0    11/29/12       99       267  …                    4
2238         2      9/1/12       99       169  …                    5

      NumStorePurchases  NumWebVisitsMonth  AcceptedCmp3  AcceptedCmp4  \
2                     5                  2             0             0
3                     2                  7             0             0
6                     7                  6             1             0
7                     3                  5             0             0
8                     9                  4             0             0
…                   …                  …             …             …
2228                  3                  8             0             0
2229                  3                  8             0             0
2235                  3                  8             0             0
2237                 10                  3             0             0
2238                  4                  7             0             1

      AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Country
2                0             0             0         0       US
3                0             0             0         0      AUS
6                0             0             0         0      GER
7                0             0             0         0       SP
8                0             0             0         0       US
…              …             …             …         …       …
2228             0             0             0         0       SP
2229             0             0             0         0       SP
2235             0             0             0         0       SP
2237             0             0             0         0      IND
2238             1             0             0         0       CA

[1443 rows x 27 columns]
```

[45]:
```
''' Taking samples of 20 customers for spending on wine from alone group '''


alone20 = df_cap2[df_cap2['Marital_Status'] == 'Alone']['MntWines'].sample(20)
alone20
```

[45]:
```
814      24
1222      2
1842    825
40       12
1092     66
```

```
2118    292
1702     22
1629     44
355      16
1920    557
1069     95
1494    375
503      29
887       2
415     163
583     265
291     912
1526    185
1077    189
1420     35
Name: MntWines, dtype: int64
```

[46]:
```
''' Taking samples of 20 customers for spending on wine from in_couple group '''


couple20 = df_cap2[df_cap2['Marital_Status'] == 'In_couple']['MntWines'].
 ↪sample(20)
couple20
```

[46]:
```
116      738
1047     303
1969     197
1353    1000
1280      25
212        3
236      117
432      658
554      290
904      313
1752      12
2204     199
967        6
184        1
2        134
298      641
1365     593
1750      26
1229       9
679      243
Name: MntWines, dtype: int64
```

1. Here we have taken sample of 20 for each of both groups created for 'In couple' and 'alone'.

2. we will conduct a hypothesis test of ttest individula for both groups created to know if couples spend moere or less on wines.
3. H0: incouple and alone group spents equal amount on wines

   Ha: incouple spents more or less amount on wines than alone group
4. Significance level = 0.05

[47]:
```
ttest_ind(couple20,alone20,alternative = 'less')
```

[47]:
```
TtestResult(statistic=0.7792938314608708, pvalue=0.779683408949251, df=38.0)
```

[48]:
```
ttest_ind(couple20,alone20,alternative = 'greater')
```

[48]:
```
TtestResult(statistic=0.7792938314608708, pvalue=0.2203165910507489, df=38.0)
```

Here we can obsereve that p value for both tetst is greater than our significance value.

Hence we can't reject the null hypothesis and there is no confident evidence that couple spends more than Alone group people.

.

[48]:
```

```

0.0.4 Are people with lower income are more attracted towards campaign or simply put accept more campaigns

[49]:
```
''' creating a copy of df '''

df_cap1 = df_cap.copy()
```

[50]:
```
df_cap1 = pd.
 ↪melt(df_cap,id_vars=['ID','Year_Birth','Education','Marital_Status','Income','Kidhome','Tee
                       'MntMeatProducts', 'MntFishProducts',␣
 ↪'MntSweetProducts','MntFruits',
      'MntGoldProds', 'NumDealsPurchases',␣
 ↪'NumWebPurchases','NumCatalogPurchases','NumStorePurchases','NumWebVisitsMonth','Complain',
 ↪= 'Acceptancecmp',value_name = 'yes/No')
```

[51]:
```
''' fetching out columns of income and  accpetance '''

df_cap1[['Income','Acceptancecmp']]
```

[51]:

	Income	Acceptancecmp
0	84835.0	AcceptedCmp3
1	57091.0	AcceptedCmp3
2	67267.0	AcceptedCmp3
3	32474.0	AcceptedCmp3

```
4       21474.0  AcceptedCmp3
…           …           …
11190   66476.0  AcceptedCmp2
11191   31056.0  AcceptedCmp2
11192   46310.0  AcceptedCmp2
11193   65819.0  AcceptedCmp2
11194   94871.0  AcceptedCmp2

[11195 rows x 2 columns]
```

[52]:
```
''' creating two groups of income as - lower and higher by taking base value as␣
 ↪median of the income column '''

med  = df_cap1['Income'].median()
med
df_cap1['lh_inc'] = df_cap1['Income'].apply(lambda x : 'Higher' if x > med else␣
 ↪'lower')
```

[53]:
```
''' Taking sample of 20 for lower income group '''

a = df_cap1[df_cap1['lh_inc']=='lower']['yes/No'].sample(20)
a
```

[53]:
```
2063     0
9060     0
496      1
4035     0
10475    0
5207     0
4171     0
7941     0
908      0
10582    0
5701     0
9525     1
4410     0
9314     0
10895    0
10147    0
4187     0
7289     0
5152     0
7538     0
Name: yes/No, dtype: int64
```

[54]:
```
''' Taking sample of 20 for higher income group '''
```

```
b = df_cap1[df_cap1['lh_inc']=='Higher']['yes/No'].sample(20)
b
```

[54]:
```
7352    0
558     0
8129    0
330     0
8394    0
2678    0
307     0
6383    0
7435    0
4585    0
9046    0
2773    0
3498    0
2114    0
6242    0
437     0
9607    0
1647    0
9446    0
5073    0
Name: yes/No, dtype: int64
```

here, we from a null hypothesis for lower and higher income groups by taking sample of 20 for both groups with campaingnacceptance (yes/no) columns

Null hypothesis H0: lower income group doesn't accept more campaign programs

Alternatuve hypothesis Ha: lower income group simply accept more campaign programs.

significance level : 0.05

[55]:
```
''' conducting a hypothesis test on the data for lower and higher income groups␣
 ↪'''

ttest_ind(a,b,alternative = 'greater')
```

[55]:
```
TtestResult(statistic=1.4529663145135578, pvalue=0.07722086265939598, df=38.0)
```

here, the p_value obtained is larger than our significance level .

so, we cannot reject null value hence lower income group doesn't accept more campaigns.

[55]:
```

```

0.1 conclusion

here, we worked with campaign dataset which has information about the customers details and their cmaigh=n results, purchases and amount spent.

We did hypothesis testing and data manipulation of this data and accumulated the important insights from oour analysis.

[55]: