💻 Question No: 02

⚙️ Setup

- Ensure the Python kernel has the necessary libraries: `pandas`, `matplotlib` and `lets-plot`, `os`, `numpy`, `statsmodels`
- Ensure the `Marketing Customer Value Analysis` file is in the `data` folder.

In [89]:
```
import matplotlib.pyplot as plt
import pandas as pd
import os
os.getcwd()
import numpy as np
import statsmodels.api as sm

from lets_plot import * # This imports all of ggplot2's functions
LetsPlot.setup_html()
```

i. Load the dataset and explore its structure using basic commands.

In [90]:
```
df = pd.read_csv('D:/Data Science for Marketing-I/data/WA_Fn-UseC_-Marketing-Custom
df
```

Out[90]:

	Customer	State	Customer Lifetime Value	Response	Coverage	Education	Effective To Date	Emp
0	BU79786	Washington	2763.519279	No	Basic	Bachelor	2/24/11	
1	QZ44356	Arizona	6979.535903	No	Extended	Bachelor	1/31/11	
2	AI49188	Nevada	12887.431650	No	Premium	Bachelor	2/19/11	
3	WW63253	California	7645.861827	No	Basic	Bachelor	1/20/11	
4	HB64268	Washington	2813.692575	No	Basic	Bachelor	2/3/11	
...	...	...	...	...	...	...	...	
9129	LA72316	California	23405.987980	No	Basic	Bachelor	2/10/11	
9130	PK87824	California	3096.511217	Yes	Extended	College	2/12/11	
9131	TD14365	California	8163.890428	No	Extended	Bachelor	2/6/11	
9132	UP19263	California	7524.442436	No	Extended	College	2/3/11	
9133	Y167826	California	2611.836866	No	Extended	College	2/14/11	

9134 rows × 24 columns

In [91]:
```
df.head()
```

Out[91]:

	Customer	State	Customer Lifetime Value	Response	Coverage	Education	Effective To Date	Employ
0	BU79786	Washington	2763.519279	No	Basic	Bachelor	2/24/11	
1	QZ44356	Arizona	6979.535903	No	Extended	Bachelor	1/31/11	
2	AI49188	Nevada	12887.431650	No	Premium	Bachelor	2/19/11	
3	WW63253	California	7645.861827	No	Basic	Bachelor	1/20/11	
4	HB64268	Washington	2813.692575	No	Basic	Bachelor	2/3/11	

5 rows × 24 columns

💡 The `head()` function is used to display the first five records of the dataset by default.

In [92]:
```
df.describe()
```

Out[92]:

	Customer Lifetime Value	Income	Monthly Premium Auto	Months Since Last Claim	Months Since Policy Inception	Number of Open Complaints	N
count	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	91
mean	8004.940475	37657.380009	93.219291	15.097000	48.064594	0.384388	
std	6870.967608	30379.904734	34.407967	10.073257	27.905991	0.910384	
min	1898.007675	0.000000	61.000000	0.000000	0.000000	0.000000	
25%	3994.251794	0.000000	68.000000	6.000000	24.000000	0.000000	
50%	5780.182197	33889.500000	83.000000	14.000000	48.000000	0.000000	
75%	8962.167041	62320.000000	109.000000	23.000000	71.000000	0.000000	
max	83325.381190	99981.000000	298.000000	35.000000	99.000000	5.000000	

💡 The `describe()` function provides a summary of statistical measures, such as count, mean, standard deviation, minimum, and maximum values, for numerical columns in the dataset.

ii. Create a new column named "Engaged" by transforming the categorical values in the "Response" variable into numerical values. Why is this transformation important?

In [93]:
```
df['engaged']=df['Response'].apply(lambda x:1 if x=='Yes' else 0)
df
```

Out[93]:

	Customer	State	Customer Lifetime Value	Response	Coverage	Education	Effective To Date	Emp
0	BU79786	Washington	2763.519279	No	Basic	Bachelor	2/24/11	
1	QZ44356	Arizona	6979.535903	No	Extended	Bachelor	1/31/11	
2	AI49188	Nevada	12887.431650	No	Premium	Bachelor	2/19/11	
3	WW63253	California	7645.861827	No	Basic	Bachelor	1/20/11	
4	HB64268	Washington	2813.692575	No	Basic	Bachelor	2/3/11	
...	...	...	...	...	...	...	...	
9129	LA72316	California	23405.987980	No	Basic	Bachelor	2/10/11	
9130	PK87824	California	3096.511217	Yes	Extended	College	2/12/11	
9131	TD14365	California	8163.890428	No	Extended	Bachelor	2/6/11	
9132	UP19263	California	7524.442436	No	Extended	College	2/3/11	
9133	Y167826	California	2611.836866	No	Extended	College	2/14/11	

9134 rows × 25 columns

💡 Creating the "Engaged" column changes the "Response" values ('Yes' to 1 and 'No' to 0) into numbers.

iii.Calculate and interpret the Engagement Rate. How is it computed, and what does it indicate about the customer responses?

In [94]:
```
df['engaged'].sum()/df['engaged'].count()*100
```

Out[94]:
```
np.float64(14.320122618786948)
```

💡 The Engagement Rate is calculated by finding the average of the "Engaged" column. This gives the percentage of customers who responded positively ("Yes"). It indicates how many customers are actively engaged.

iv. Analyze engagement rate by "Renew Offer Type" and "Sales Channel":

In [95]:
```
df.groupby('Renew Offer Type')['engaged'].sum()/df.groupby('Renew Offer Type')['eng
```

Out[95]:
```
Renew Offer Type
Offer1    15.831557
Offer2    23.376623
Offer3     2.094972
Offer4     0.000000
Name: engaged, dtype: float64
```

In [96]:
```
df.groupby('Sales Channel')['engaged'].sum()/df.groupby('Sales Channel')['engaged']
```

Out[96]:
```
Sales Channel
Agent          19.154443
Branch         11.453058
Call Center    10.878187
Web            11.773585
Name: engaged, dtype: float64
```

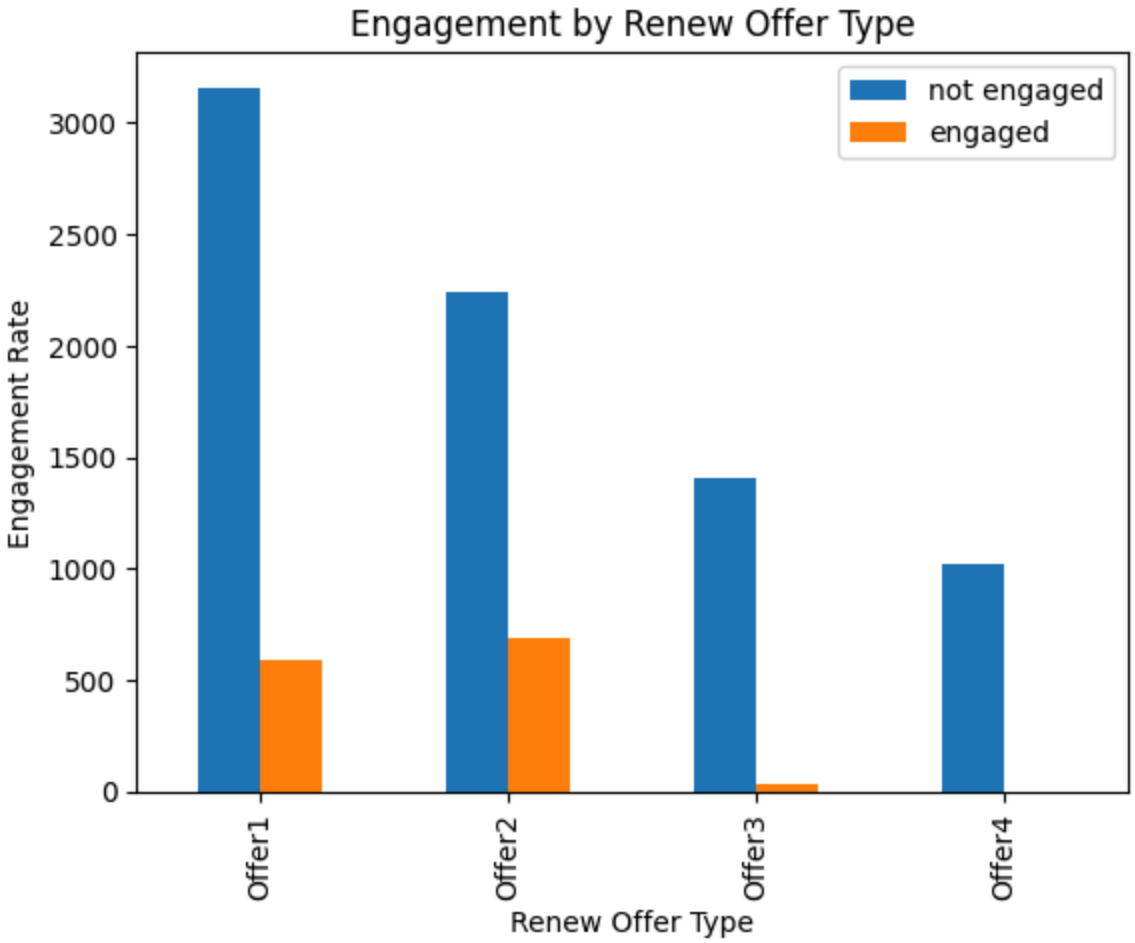
v. Use a pivot table to summarize engagement by "Renew Offer Type" and visualize the results using both bar and pie charts. Why are these visualizations helpful in understanding customer engagement patterns?

In [97]:
```
pivot_table=df.pivot_table(index='Renew Offer Type',values='Response',columns='enga
pivot_table.columns=['not engaged','engaged']
pivot_table
```

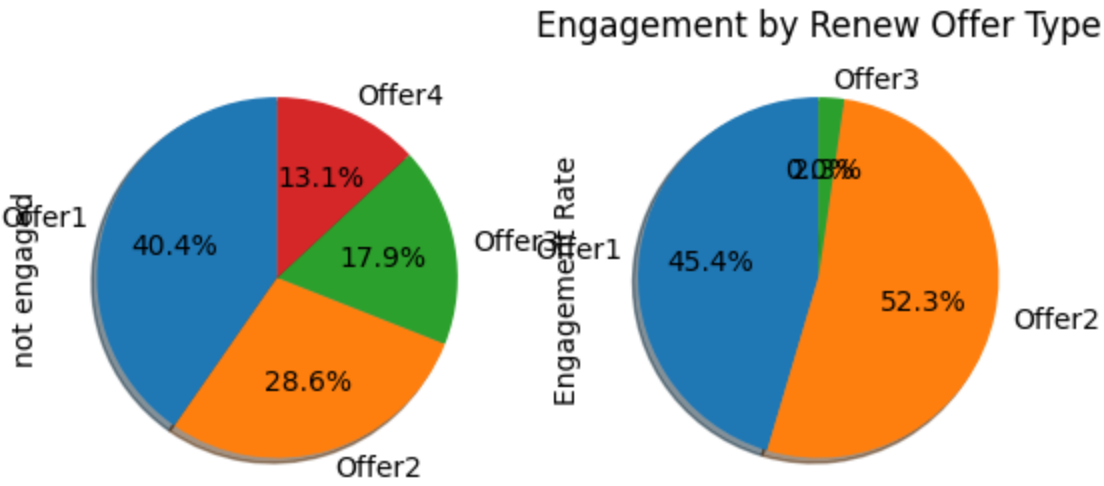
Out[97]:

Renew Offer Type	not engaged	engaged
Offer1	3158.0	594.0
Offer2	2242.0	684.0
Offer3	1402.0	30.0
Offer4	1024.0	0.0

In [98]:
```
pivot_table.plot(kind='bar')
plt.title("Engagement by Renew Offer Type")
plt.ylabel("Engagement Rate")
plt.show()
```



In [99]:
```
pivot_table.plot(kind='pie', subplots=True, autopct='%1.1f%%',startangle=90, shadow
plt.title("Engagement by Renew Offer Type")
plt.ylabel("Engagement Rate")
plt.show()
```



vi. Explain the purpose of regression analysis in this context. Describe how you would approach regression using

(i) continuous variables only

In [100…
```
df.describe()
```

Out[100…

	Customer Lifetime Value	Income	Monthly Premium Auto	Months Since Last Claim	Months Since Policy Inception	Number of Open Complaints	N
count	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000	91
mean	8004.940475	37657.380009	93.219291	15.097000	48.064594	0.384388	
std	6870.967608	30379.904734	34.407967	10.073257	27.905991	0.910384	
min	1898.007675	0.000000	61.000000	0.000000	0.000000	0.000000	
25%	3994.251794	0.000000	68.000000	6.000000	24.000000	0.000000	
50%	5780.182197	33889.500000	83.000000	14.000000	48.000000	0.000000	
75%	8962.167041	62320.000000	109.000000	23.000000	71.000000	0.000000	
max	83325.381190	99981.000000	298.000000	35.000000	99.000000	5.000000	

In [101…
```
continuous_vars=['Customer Lifetime Value', 'Income', 'Monthly Premium Auto',
                 'Months Since Last Claim', 'Months Since Policy Inception','Numb
```

In [102…
```
logit=sm.Logit(df['engaged'],df[continuous_vars])
logit_fit = logit.fit()
```

```
Optimization terminated successfully.
         Current function value: 0.421189
         Iterations 6
```

💡 The optimization process was successful, and the algorithm converged in 6 iterations.The final value of the objective function is 0.421189, which can be used as a benchmark for further improvements.

In [103…

```
print(logit_fit.summary())
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                engaged   No. Observations:                 9134
Model:                          Logit   Df Residuals:                     9126
Method:                           MLE   Df Model:                            7
Date:                Mon, 03 Feb 2025   Pseudo R-squ.:                -0.02546
Time:                        22:14:29   Log-Likelihood:                -3847.1
converged:                       True   LL-Null:                       -3751.6
Covariance Type:            nonrobust   LLR p-value:                    1.000
==============================================================================
                                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Customer Lifetime Value       -6.741e-06   5.04e-06     -1.337      0.181   -1.66e-05    3.14e-06
Income                        -2.857e-06   1.03e-06     -2.766      0.006   -4.88e-06   -8.33e-07
Monthly Premium Auto             -0.0084      0.001     -6.889      0.000      -0.011      -0.006
Months Since Last Claim          -0.0202      0.003     -7.238      0.000      -0.026      -0.015
Months Since Policy Inception    -0.0060      0.001     -6.148      0.000      -0.008      -0.004
Number of Open Complaints        -0.0829      0.034     -2.424      0.015      -0.150      -0.016
Number of Policies               -0.0810      0.013     -6.356      0.000      -0.106      -0.056
Total Claim Amount                0.0001      0.000      0.711      0.477      -0.000       0.000
==============================================================================
```

💡 Purpose: Regression analysis helps understand the relationships between the dependent variable (e.g., engagement) and independent variables (e.g., offer type, income)

The model does not explain the variance in the target variable (engaged) well, as indicated by the negative pseudo R-squared and high LLR p-value.(Income, Monthly Premium Auto, Months Since Last Claim, Months Since Policy Inception, Number of Open Complaints, and Number of Policies) are significant predictors of engagement,(Customer Lifetime Value and Total Claim Amount) not significantly impact engagement.

(ii) Categorical

In [104…
```
gender_values, gender_lables = df['Gender'].factorize()
df['genderfactorized'] = gender_values
```

In [105…
```
education_values, education_lables = df['Education'].factorize()
df['educationfactorized'] = education_values
```

In [108…
```
categorical_var = ['genderfactorized','educationfactorized']
```

In [109…
```
logit1 = sm.Logit(df['engaged'],df[categorical_var])
logit1_fit = logit1.fit()
```

```
Optimization terminated successfully.
         Current function value: 0.489140
         Iterations 6
```

In [110…
```
print(logit1_fit.summary())
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                engaged   No. Observations:                 9134
Model:                          Logit   Df Residuals:                     9132
Method:                           MLE   Df Model:                            1
Date:                Mon, 03 Feb 2025   Pseudo R-squ.:                -0.1909
Time:                        22:14:42   Log-Likelihood:                -4467.8
converged:                       True   LL-Null:                       -3751.6
Covariance Type:            nonrobust   LLR p-value:                    1.000
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
genderfactorized       -1.1269      0.046    -24.263      0.000      -1.218      -1.036
educationfactorized    -0.5536      0.018    -30.560      0.000      -0.589      -0.518
==============================================================================
```

💡 The Logit regression shows that both **gender** and **education** significantly negatively impact engagement.

- **Gender**: Coefficient of -1.1269, p-value 0.000.
- **Education**: Coefficient of -0.5536, p-value 0.000.

Both factors are highly significant with very low p-values, suggesting strong negative relationships with the dependent variable (engagement).

iii.Both Continous and Categorical

In [46]:
```
logit11 = sm.Logit(
    df['engaged'],
    df[[
```

```
        'genderfactorized',
        'educationfactorized',
        'Customer Lifetime Value',
        'Income',
        'Monthly Premium Auto',
        'Months Since Last Claim',
        'Months Since Policy Inception',
        'Number of Open Complaints',
        'Number of Policies',
        'Total Claim Amount'
    ]]
)
logit11_fit = logit11.fit()
```

```
Optimization terminated successfully.
         Current function value: 0.420108
         Iterations 6
```

💡 The optimization process was successful, and the algorithm converged in 6 iterations.The final value of the objective function is 0.420108

In [47]:
```
print(logit11_fit.summary())
```

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                engaged   No. Observations:                 9134
Model:                          Logit   Df Residuals:                     9124
Method:                           MLE   Df Model:                            9
Date:                Mon, 03 Feb 2025   Pseudo R-squ.:                -0.02283
Time:                        22:07:06   Log-Likelihood:                -3837.3
converged:                       True   LL-Null:                       -3751.6
Covariance Type:            nonrobust   LLR p-value:                    1.000
================================================================================
=============
                                   coef    std err          z      P>|z|      [0.02
5      0.975]
--------------------------------------------------------------------------------
-------------
genderfactorized                -0.1421      0.058     -2.458      0.014      -0.25
5      -0.029
educationfactorized             -0.0801      0.022     -3.570      0.000      -0.12
4      -0.036
Customer Lifetime Value       -6.625e-06   5.02e-06     -1.319      0.187   -1.65e-0
5    3.22e-06
Income                        -2.275e-06   1.04e-06     -2.188      0.029   -4.31e-0
6   -2.37e-07
Monthly Premium Auto            -0.0077      0.001     -6.343      0.000      -0.01
0      -0.005
Months Since Last Claim         -0.0186      0.003     -6.627      0.000      -0.02
4      -0.013
Months Since Policy Inception   -0.0054      0.001     -5.559      0.000      -0.00
7      -0.004
Number of Open Complaints       -0.0811      0.034     -2.375      0.018      -0.14
8      -0.014
Number of Policies              -0.0751      0.013     -5.888      0.000      -0.10
0      -0.050
Total Claim Amount               0.0002      0.000      1.173      0.241      -0.00
0       0.000
================================================================================
=============
```

💡

- **Significant variables** ($p < 0.05$):
  - **gender**, **education**, **Income**, **Monthly Premium Auto**, **Months Since Last Claim**, **Months Since Policy Inception**, **Number of Open Complaints**, and **Number of Policies**.
- **Not significant** ($p > 0.05$):
  - **Customer Lifetime Value** and **Total Claim Amount**.

The coefficients for significant variables suggest negative relationships with engagement, while others have no notable impact.