WGU D208 PA 1

Andrew Shrestha

Part I: Research Question

A1. Question: How much on average will a customer be charged per month? Is it possible to predict this using multiple regression analysis?

A2. Objectives and Goals: Predicting with confidence the amount of revenue per customer will be a significant question for companies, corporations, and stakeholders as it is part of sales forecasting. This figure/knowledge allows stakeholders to make efficient decisions on how company invests, grows, and creates overall valuation of the company (**Kripa Mahalingam 2020)**.

Part II: Method Justification

B1. Assumptions of Multiple Regression Model

i. Linear Relationship: Independent and Dependent variables will have an approximate linear relationship
ii. Homoscedasticity: residual variance will be equal for each of the explanatory variables
iii. Independent Errors: residuals will be uncorrelated
iv. Variance in Predictors: explanatory variables results must vary within a range
v. Multicollinearity: there may be at least two or likely more variables that are highly correlated

B2. Benefits of Chosen Tools

The tools leveraged in this analysis will be Python along with various packages such as scikit-learn. I choose to use this programming language since Python has an abundance of packages for ease of processing and interpreting data as well as visualizing capabilities. Python is also has the ability to integrate into other programs such as Java or C++ which can be significant in joint projects. Finally, Python can deliver and run code with greater speed than other programs such as R which will be time efficient when regards to the codes in this analysis.

I will also be leveraging Jupyter notebook to display and present the coding as well as outputs. Jupyter notebook has both the ease of understanding the codes and outputs as well as being sharable. Independent cell by cell running capabilities presented in the program provides additional understanding of what each cell does. Finally, it is also very secure as it does not store the data on local machines and is protective of data sensitive information.

B3. Appropriate Technique Justification

We are utilizing the multiple regression technique in our analysis since there are a variety of independent variables in our churn data collection that have possible relationships with our dependent variable of Monthly Charge. This is where a multiple regression analysis shines, since it allows us to not only be flexible in which independent variables to take into consideration, but also to describe how the changes in each of these has to the changes in the dependent Monthly Charge variable.

We can also control variables by predicting the effects of changing one independent variable on the dependent, while simultaneously holding all other independent variables constant. Thus, giving us the insights into separating complex research questions from our data source.

The coefficients that we obtain from our results of each independent variables will be immensely helpful to the corporation as they are able to see clearly which independent variables positively and negatively impact the Monthly Charge, and therefore make proper resource allocation decisions based on these findings.

Part III: Data Preparation

C1. Data Preparation Goals and Manipulation

i. Obtaining the churn data set into jupyter notebooks via read_csv code
ii. Address any repetitions, irrelevant, duplicated, inconsistent, or unnecessary data
iii. Impute missing data values with statistically significant calculations to keep data accuracy intact
iv. Identify outliers and remove them if they are more than 3 standard deviations above or below the mean
v. Test and Train our data set to give us a prediction on customer monthly charges with respect to our explanatory variables

The goal for the preparation stage of the analysis is data cleaning to ensure that we are dealing with accurate and correct data, free from any errors that may influence our prediction outcome via ii.

The goal for the manipulation stage of the analysis is to first ensure that we have a complete and appropriate data set, by means of imputation and disregard for missing values and outliers respectively as shown in step iii.

These preparation and manipulation stages will set us up for success when we then test and train our data to gain a prediction on our desired variable of Monthly Charge

C2. Summary Statistics

Out of the 10,000 observations, we decided to remove a couple variables that were deemed no relation to our target variable “MontlyCharge”. Thus, CaseOrder, Customer_id, Interaction, UID, City, State, Country,Zip,Lat,Lng,Population, Area, TimeZone, Marital, and PaymentMethod were all removed, leaving only 18 variables left.

In order to ensure that our regression analysis could be completed, we also changed the categorical binary outputs on select variables from yes/no to numeric 1/0.

Finally, when running the describe function for our new cleaned data set, we see that average children = 2, Age = 53, Income = $39806, Outage_Sec_perweek = 10, Email = 12, Contacts = 0.99, Yearly_equip_failure = 0.39, and tenure =34.52.

C3. Steps used to Prepare Data

i. Upload given Churn_Clean dataset CSV file via python programming language and create a churn data frame

ii. Observe the dataset columns and rows and understand the relationships between the independent variables and the dependent variable "MonthlyCharge"

iii. Rename survey questions into more descriptive labels, thus making it easier to keep track of and understand

iv. Get initial statistics of the dataframe via describe and dtypes method to further understand the data before getting into cleaning and manipulation

v. Reduce the dataset by removing independent variables that have little/no relation to our dependent variable of interest. We would like to leave in only significant and meaningful data in our analysis as to get an accurate outcome

vi. Ensure that missing variables are imputed with significant values via descriptive statistics such as mean. This keeps the integrity of the dataset and will be much more meaningful than leaving it blank/removing them

vii. Ensure that outliers are removed that are greater than 3 standard deviations away from the mean

viii. Utilize dummy variables for our binary categorical values from "yes/no" to "0/1". This will change the data type from string to numerical which is necessary for our multiple regression analysis

ix. Observe both univariate and bivariate graphical visualizations

x. The final prepared data will be saved as Final_Churn_Clean

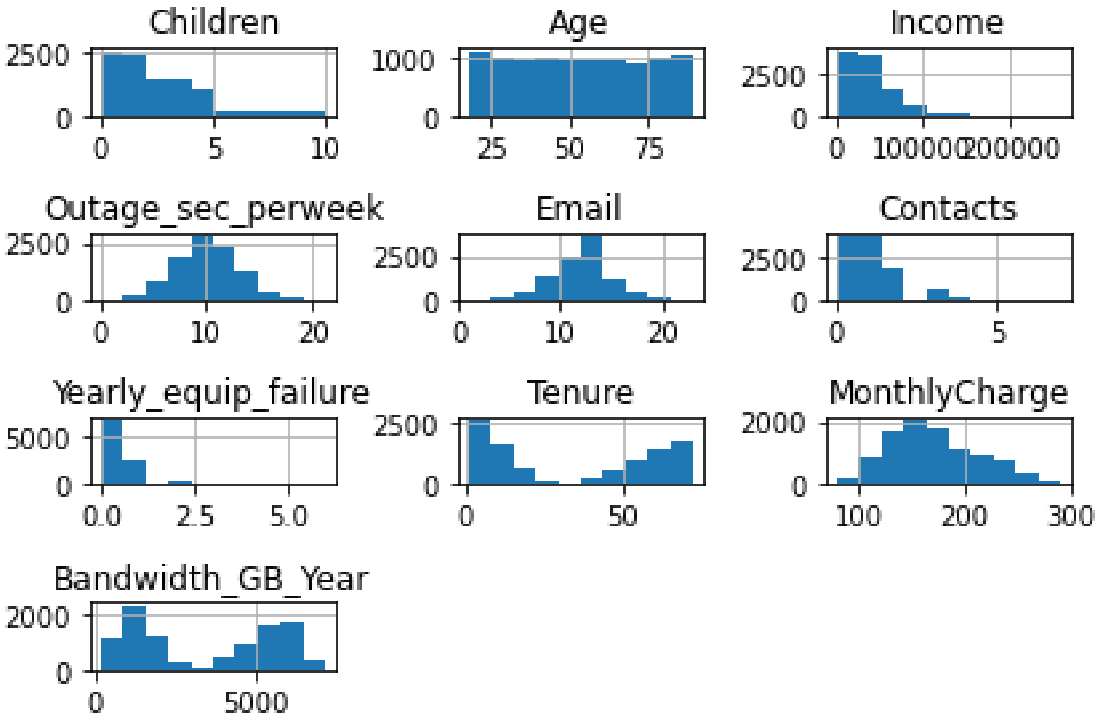
```
#importing the necessary libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

import pylab
from pylab import rcParams
import statsmodels.api as sm
import statistics
from scipy import stats

import sklearn
from sklearn import preprocessing
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.metrics import classification_report

from scipy.stats import chisquare
from scipy.stats import chi2_contingency


#uploading our initial churn dataset into the pandas dataframe
churn_clean_df =
pd.read_csv(r"C:\Users\andre\OneDrive\Desktop\churn_clean.csv")


#re-labeling the Survey Questions as to make it more meaningful instead of the
generic 1-7 labling originally done
churn_clean_df.rename (columns = {'Item1': 'Timely_Response',
'Item2':'Timely_Fixes', 'Item3':'Timely_Replacements',

'Item4':'Reliability','Item5':'Options','Item6':'Respectful_Response','Item7':'
Couteous_Exchange',
                        'Item8':'Evidence_Of_Active_Listening'}, inplace =
True)


#observe the dataset and get some descriptive statistics before implementing
cleaning and manipulation
churn_clean_df.shape
churn_clean_df.describe()


#based off of observations between relationships between the independent
variables and our dependent variable "MonthlyCharge",
#remove less significant columns in order to reduce dataset and make it easier
for analysis

churn_clean_df = churn_clean_df.drop (columns = ['CaseOrder',
'Customer_id','Interaction','UID','City','State','County','Zip','Lat','Lng','Po
pulation','Area','TimeZone','Marital','PaymentMethod'])


#addressing any missing data
missing_data_churn_df = churn_clean_df.isnull().sum()
missing_data_churn_df


#we will now transform our binary categorical variables into dummy variables
taking on either a value of 0 or 1
churn_clean_df ['DummyGender'] = [1 if v =='Male' else 0 for v in
churn_clean_df['Gender']]
churn_clean_df ['DummyChurn'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Churn']]
churn_clean_df ['DummyTechie'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Techie']]
churn_clean_df ['DummyContract'] = [1 if v =='Two Year' else 0 for v in
churn_clean_df['Contract']]
churn_clean_df ['DummyPort_modem'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Port_modem']]
```

```
churn_clean_df ['DummyTablet'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Tablet']]
churn_clean_df ['DummyInternetService'] = [1 if v =='Fiber Optic' else 0 for v
in churn_clean_df['InternetService']]
churn_clean_df ['DummyPhone'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Phone']]
churn_clean_df ['DummyMultiple'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['Multiple']]
churn_clean_df ['DummyOnlineSecurity'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['OnlineSecurity']]
churn_clean_df ['DummyOnlineBackup'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['OnlineBackup']]
churn_clean_df ['DummyDeviceProtection'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['DeviceProtection']]
churn_clean_df ['DummyTechSupport'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['TechSupport']]
churn_clean_df ['DummyStreamingTV'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['StreamingTV']]
churn_clean_df ['DummyStreamingMovies'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['StreamingMovies']]
churn_clean_df ['DummyPaperlessBilling'] = [1 if v =='Yes' else 0 for v in
churn_clean_df['PaperlessBilling']]


#now we will drop the original (Yes/No) categorical variables as we essentially
already created a duplicate of them with binary 0 or 1 values
churn_clean_df = churn_clean_df.drop (columns = ['Gender',
'Churn','Techie','Contract','Port_modem','Tablet','InternetService','Phone','Mu
ltiple','OnlineSecurity','OnlineBackup','DeviceProtection','TechSupport','Strea
mingTV','StreamingMovies','PaperlessBilling'])


#now we will double check the columns to see if we have just the dummy
variables to avoid duplications
churn_clean_df.columns
```

C4. Univariate and Bivariate Visualizations

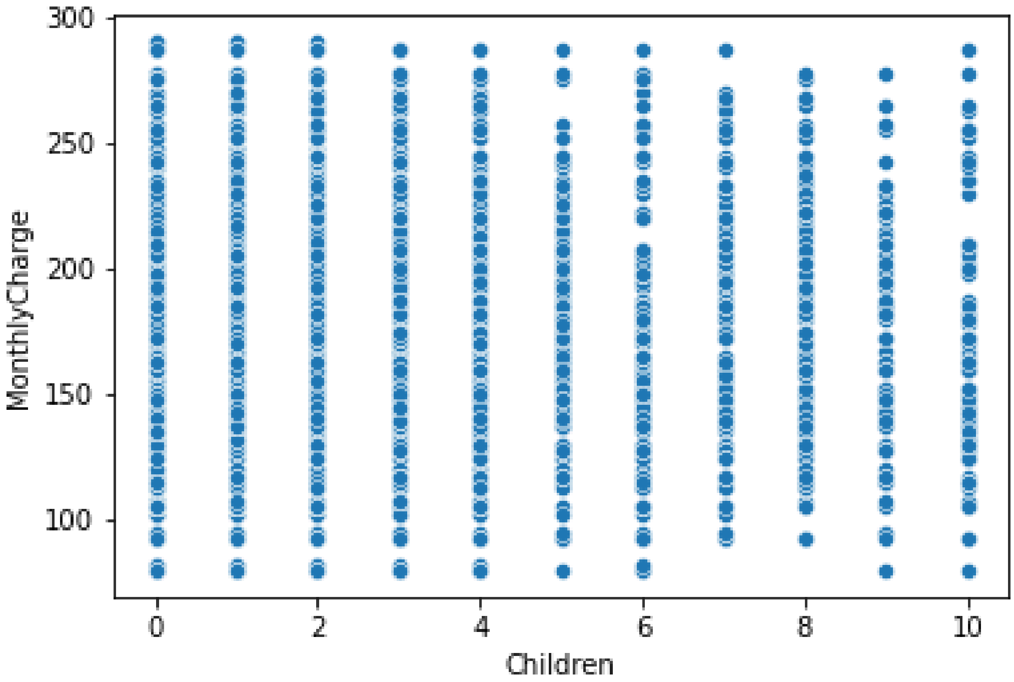
i. Univariate Visualizations

```
#we will use histograms for our Univariate continuous variable analysis
visualizations as they are extremely insightful into understanding the
distribution of the data

churn_clean_df[['Job','Children','Age','Income','Outage_sec_perweek','Ema
il','Contacts','Yearly_equip_failure','Tenure','MonthlyCharge','Bandwidth
_GB_Year']].hist()
plt.tight_layout()
```

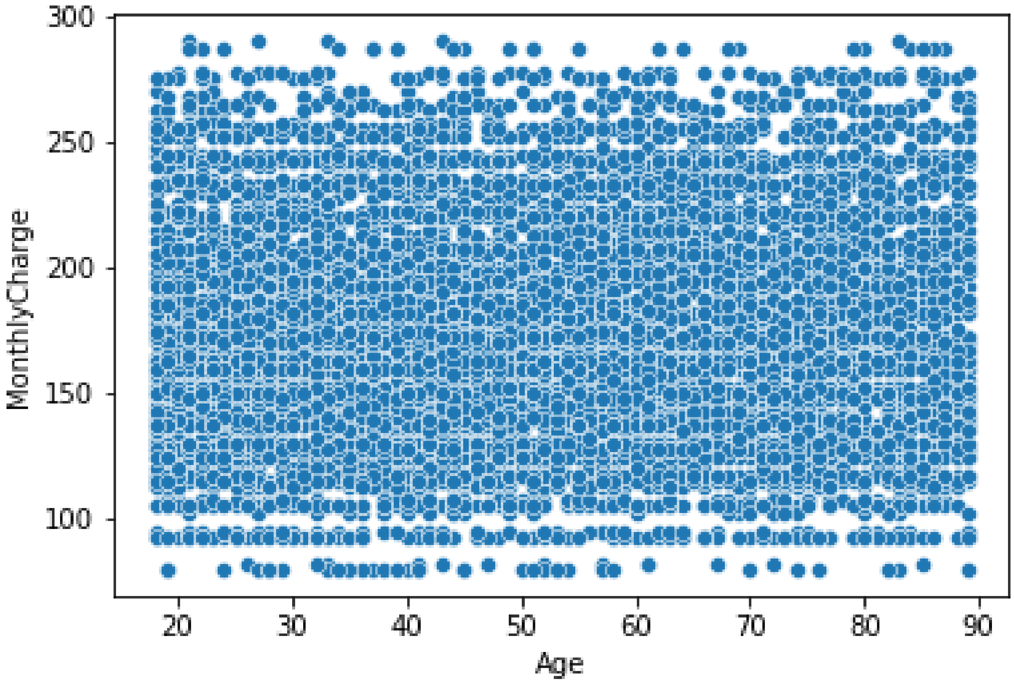


ii. Bivariate Visualizations

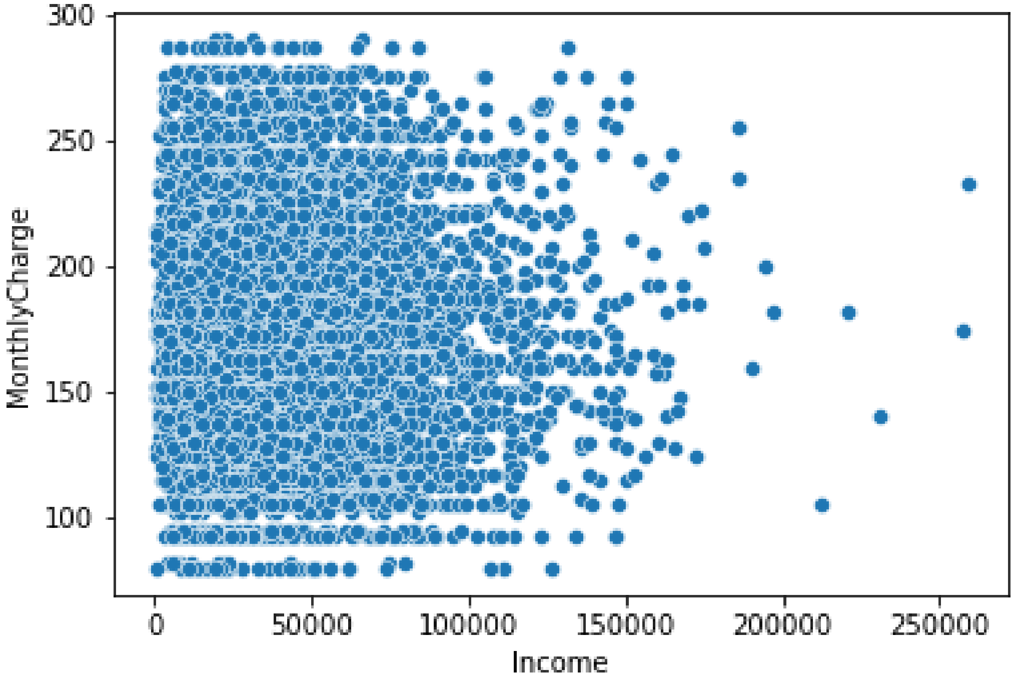
```
#Bivariate Scatter plots to observe relationship between the independent
variables and the dependent variable "MonthlyCharge"
sns.scatterplot(x=churn_clean_df['Children'],
y=churn_clean_df['MonthlyCharge'])
plt.show()
```



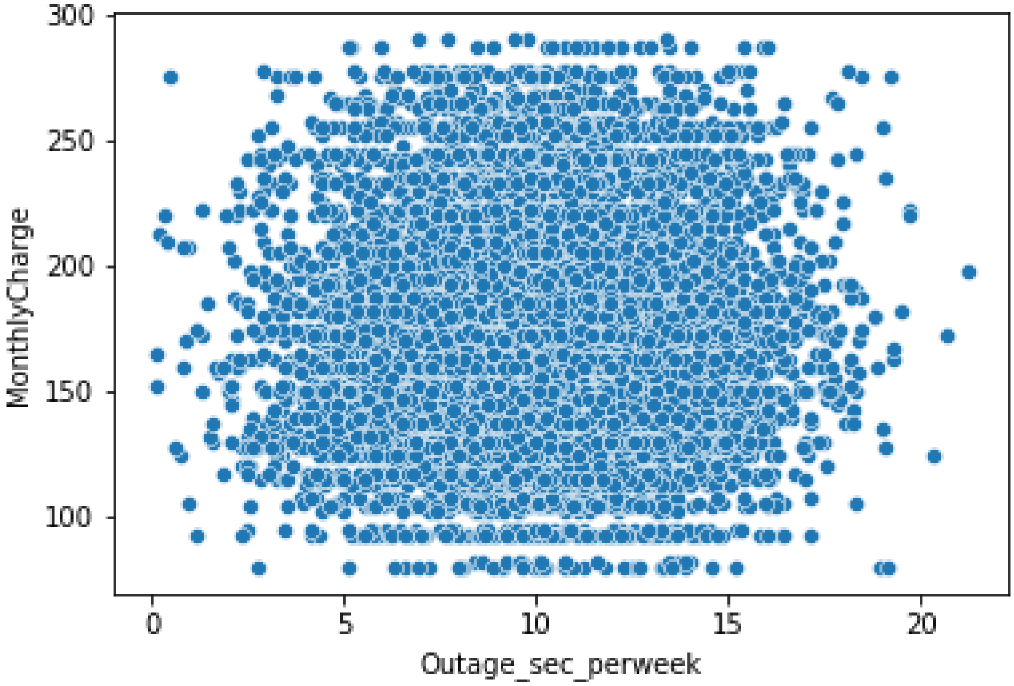
```
sns.scatterplot(x=churn_clean_df['Age'], y=churn_clean_df['MonthlyCharge'])
plt.show()
```



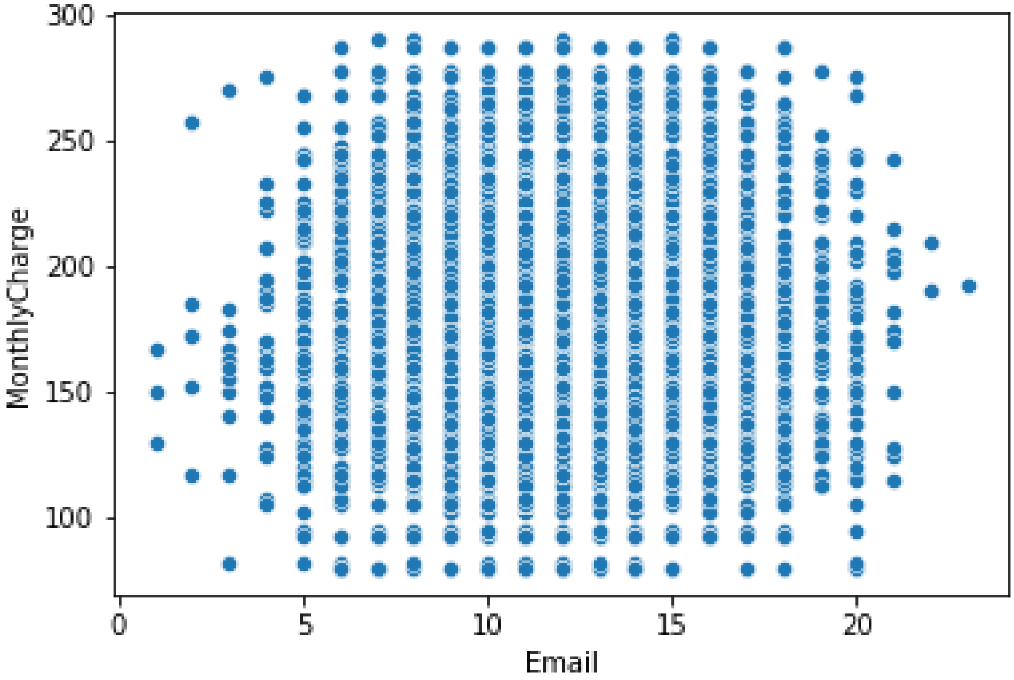
```
sns.scatterplot(x=churn_clean_df['Income'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



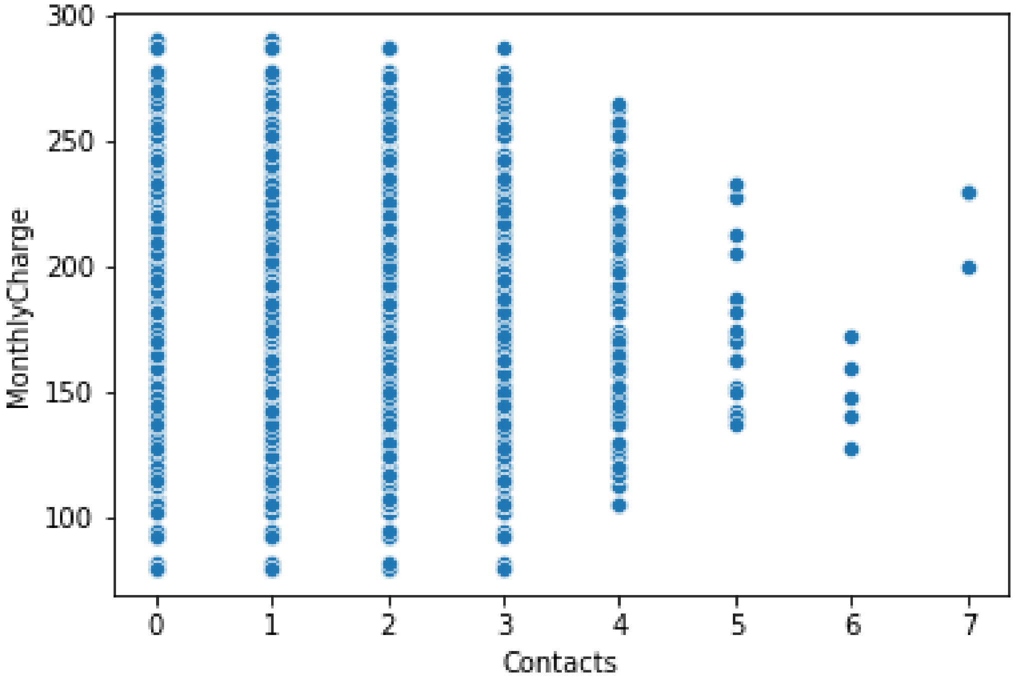
```
sns.scatterplot(x=churn_clean_df['Outage_sec_perweek'],y=churn_clean_df['Monthl
yCharge'])
plt.show()
```



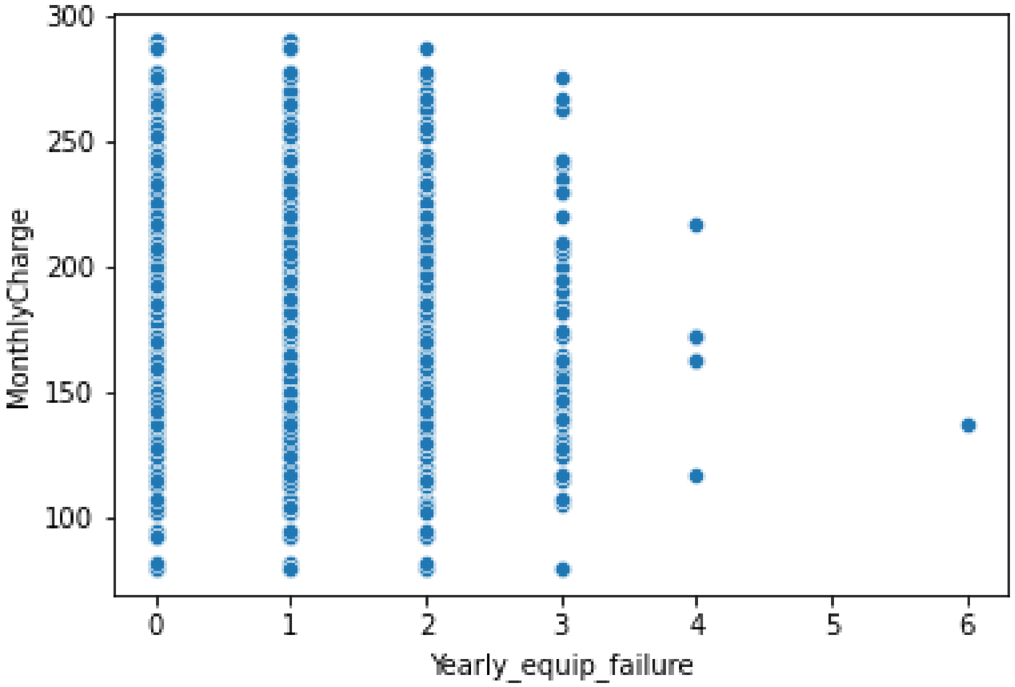
```
sns.scatterplot(x=churn_clean_df['Email'], y=churn_clean_df['MonthlyCharge'])
plt.show()
```



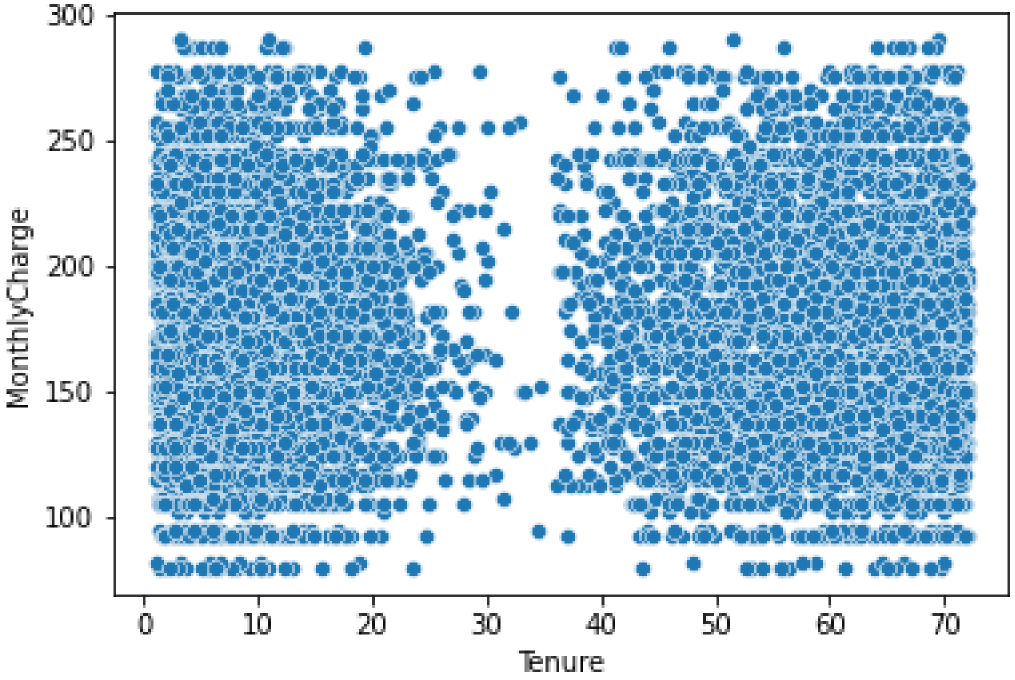
```
sns.scatterplot(x=churn_clean_df['Contacts'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



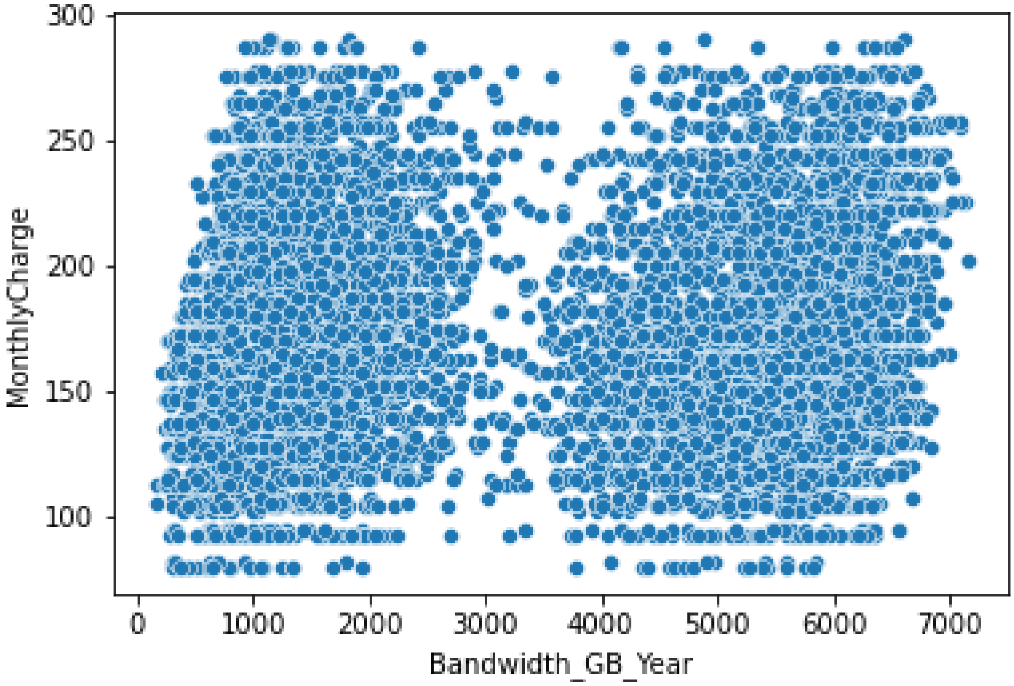
```
sns.scatterplot(x=churn_clean_df['Yearly_equip_failure'],y=churn_clean_df['Mont
hlyCharge'])
plt.show()
```



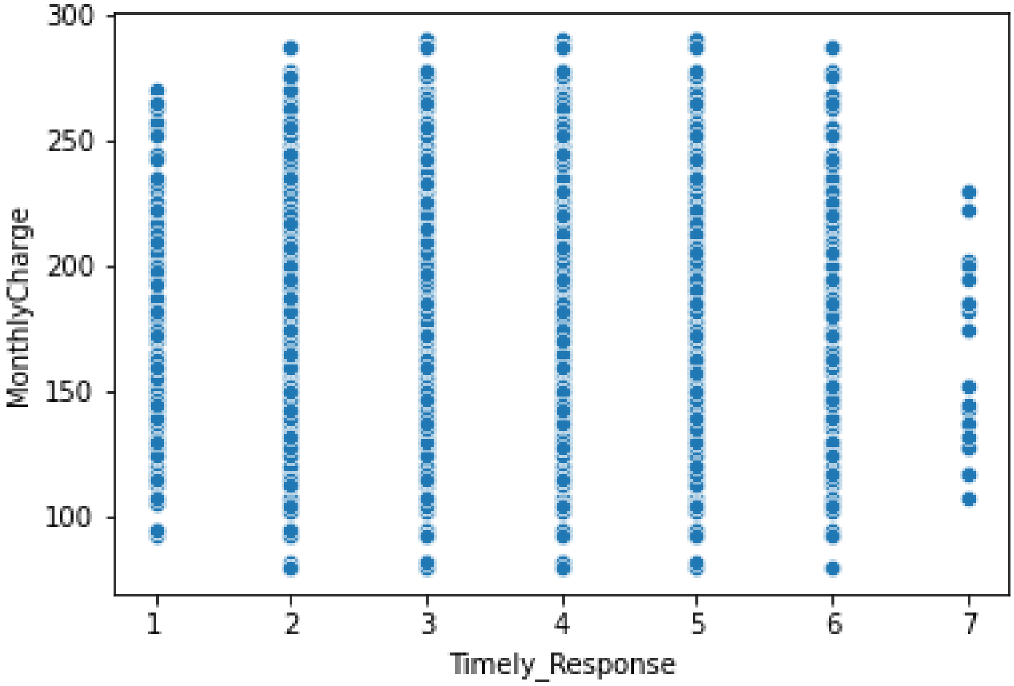
```
sns.scatterplot(x=churn_clean_df['Tenure'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



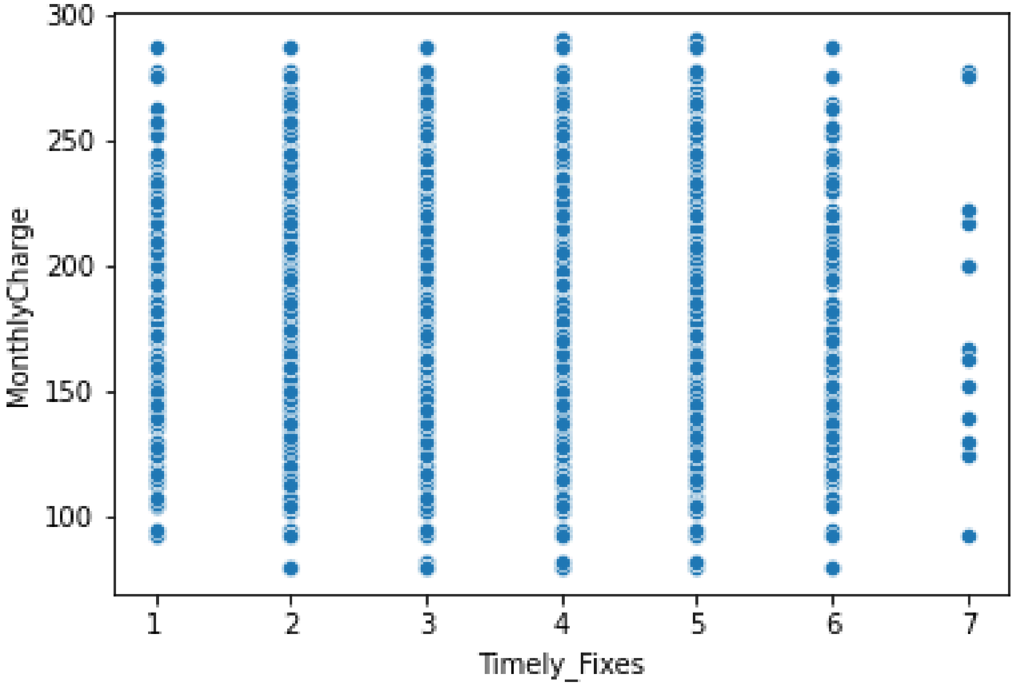
```
sns.scatterplot(x=churn_clean_df['Bandwidth_GB_Year'],y=churn_clean_df['Monthly
Charge'])
plt.show()
```



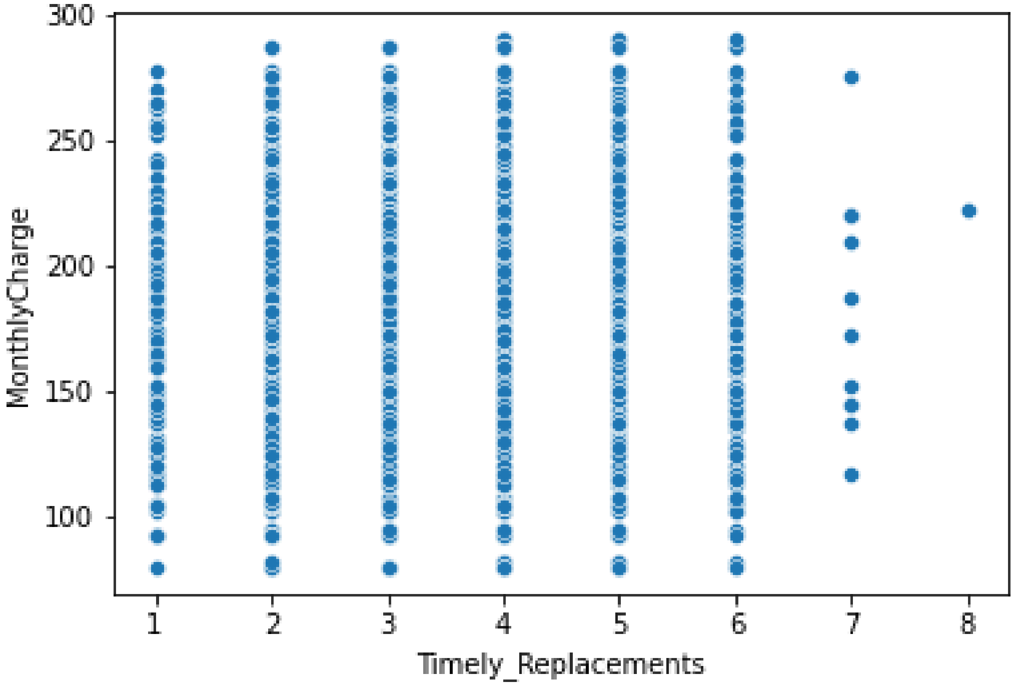
```
sns.scatterplot(x=churn_clean_df['Timely_Response'],y=churn_clean_df['MonthlyCh
arge'])
plt.show()
```



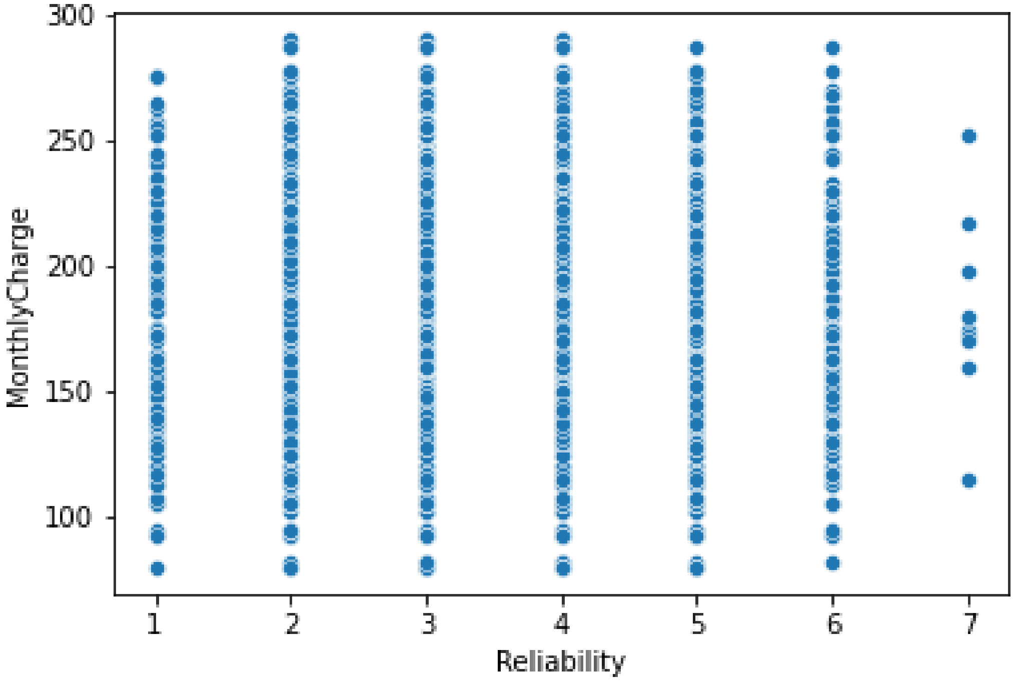
```
sns.scatterplot(x=churn_clean_df['Timely_Fixes'],y=churn_clean_df['MonthlyCharg
e'])
plt.show()
```



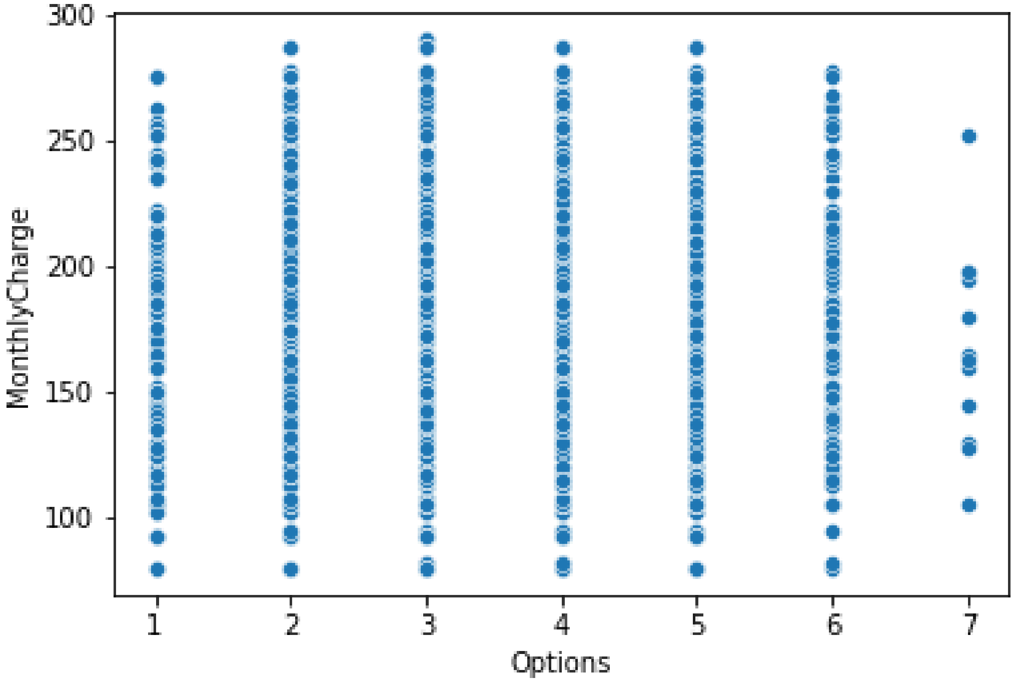
```
sns.scatterplot(x=churn_clean_df['Timely_Replacements'],y=churn_clean_df['Month
lyCharge'])
plt.show()
```



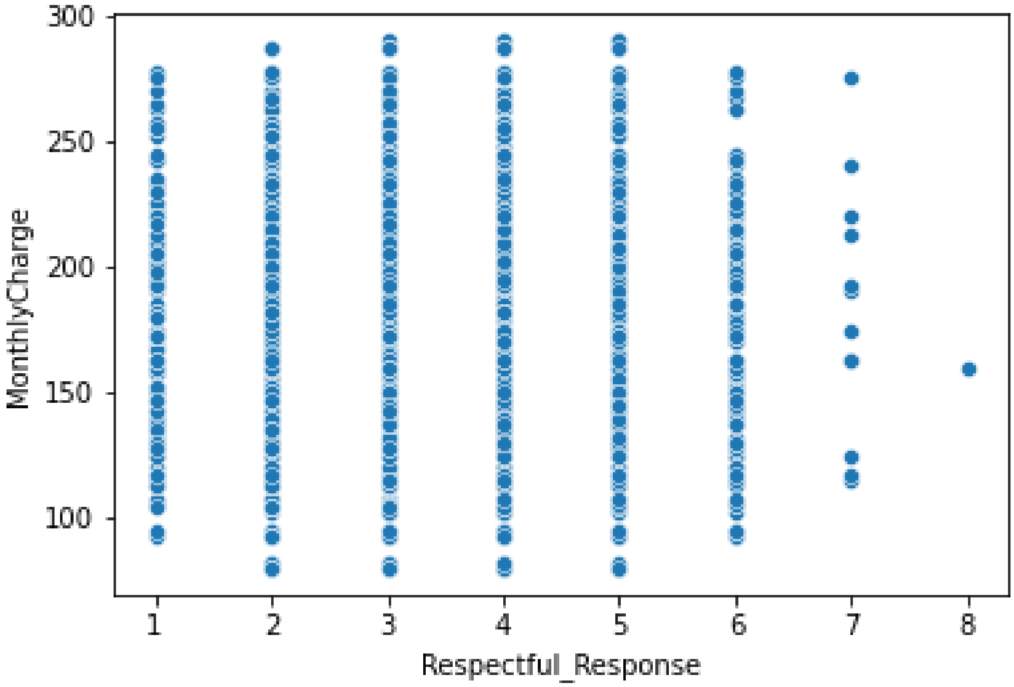
```
sns.scatterplot(x=churn_clean_df['Reliability'],y=churn_clean_df['MonthlyCharge
'])
plt.show()
```



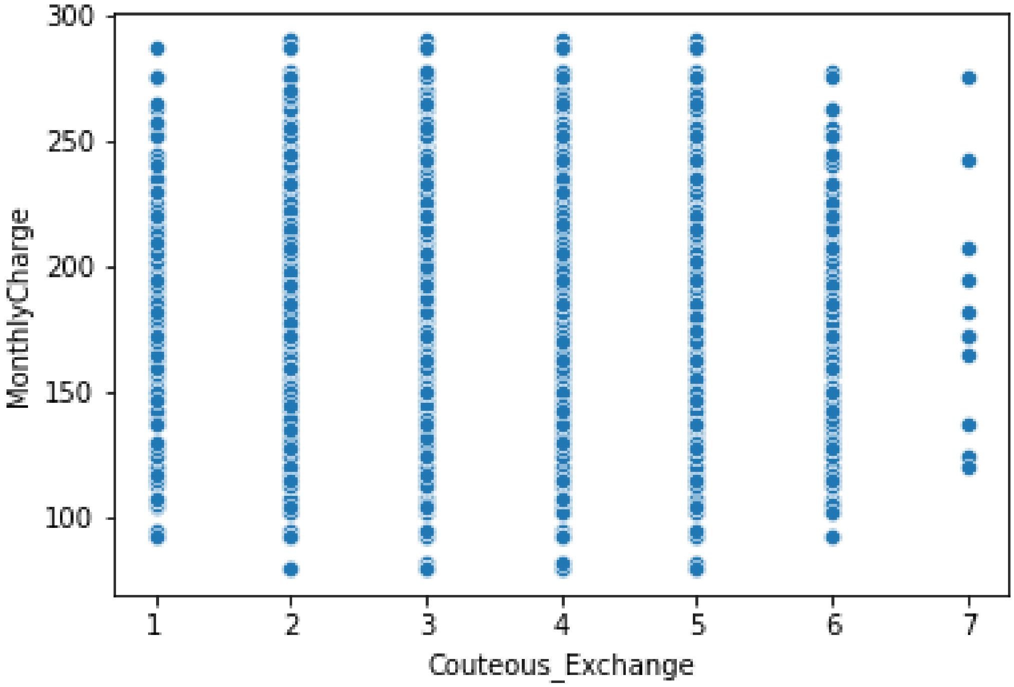
```
sns.scatterplot(x=churn_clean_df['Options'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



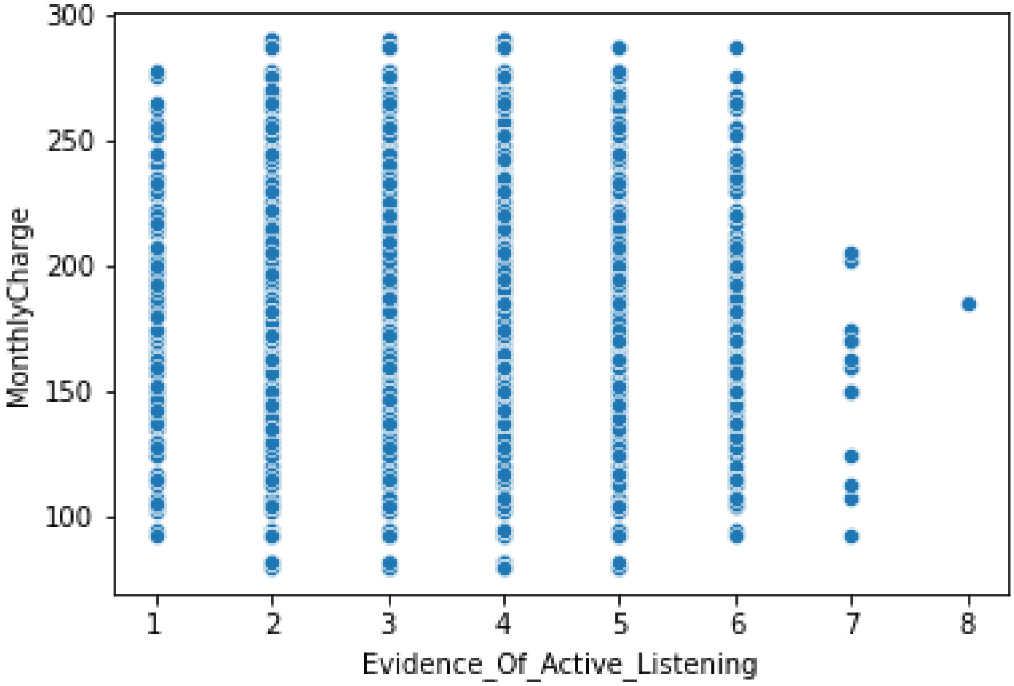
```
sns.scatterplot(x=churn_clean_df['Respectful_Response'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



```
sns.scatterplot(x=churn_clean_df['Couteous_Exchange'],y=churn_clean_df['MonthlyCharge'])
plt.show()
```



```
sns.scatterplot(x=churn_clean_df['Evidence_Of_Active_Listening'],y=churn_clean_
df['MonthlyCharge'])
plt.show()
```



C5. Copy of Prepared Data Set

(attached)

```
#we will now prepare the cleaned data set for multiple regression analysis
churn_clean_df.to_csv('final_churn_clean.csv')
```

Part IV: Model Comparison & Analysis

D1. Initial Multiple Regression from all Predictors

```
#we will now create an initial multiple regresssion with all variables stated
in part C2 (without dummy variables)
final_churn_clean_df['intercept'] = 1
lm_MonthlyCharge =
sm.OLS(final_churn_clean_df['MonthlyCharge'],final_churn_clean_df[['Children','
Age','Income','Outage_sec_perweek','Email','Contacts','Yearly_equip_failure','T
enure','Bandwidth_GB_Year','Timely_Response','Timely_Fixes',
'Timely_Replacements',

'Reliability','Options','Respectful_Response','Couteous_Exchange',
                      'Evidence_Of_Active_Listening','intercept']]).fit()
print(lm_MonthlyCharge.summary())
```

```
OLS Regression Results
==============================================================================
====
Dep. Variable:          MonthlyCharge   R-squared:                       0.276
Model:                            OLS   Adj. R-squared:                  0.275
Method:                 Least Squares   F-statistic:                     224.1
Date:                Thu, 22 Jul 2021   Prob (F-statistic):               0.00
Time:                        16:23:41   Log-Likelihood:
-50171.
No. Observations:               10000   AIC:
1.004e+05
Df Residuals:                    9982   BIC:
1.005e+05
Df Model:                          17
Covariance Type:            nonrobust
==============================================================================
======================
                                   coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
----------------------
Children                        -2.7385      0.175    -15.611      0.000
-3.082      -2.395
Age                              0.2952      0.018     16.189      0.000
0.259       0.331
Income                       -1.096e-05    1.3e-05     -0.844      0.398
-3.64e-05    1.45e-05
Outage_sec_perweek               0.2431      0.123      1.976      0.048
0.002       0.484
Email                            0.0433      0.121      0.358      0.720
-0.194       0.280
Contacts                        -0.1428      0.370     -0.386      0.700
-0.869       0.583
Yearly_equip_failure            -0.4497      0.575     -0.781      0.435
-1.578       0.678
Tenure                          -6.8947      0.113    -61.163      0.000
-7.116      -6.674
Bandwidth_GB_Year                0.0840      0.001     61.585      0.000
0.081       0.087
Timely_Response                  1.5024      0.524      2.869      0.004
0.476       2.529
Timely_Fixes                    -0.2776      0.491     -0.566      0.572
-1.240       0.684
Timely_Replacements             -0.5958      0.450     -1.323      0.186
-1.478       0.287
Reliability                     -0.0642      0.403     -0.159      0.873
-0.853       0.725
Options                         -0.5080      0.418     -1.215      0.224
-1.328       0.312
```

```
Respectful_Response             -0.0982      0.431     -0.228      0.820
-0.942       0.746
Couteous_Exchange               -0.2433      0.407     -0.598      0.550
-1.042       0.555
Evidence_Of_Active_Listening    -0.4908      0.388     -1.266      0.206
-1.251       0.269
intercept                      116.2018      4.025     28.870      0.000
108.312     124.092
==========================================================================
====
Omnibus:                      241.815   Durbin-Watson:                1.972
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           146.097
Skew:                           0.147   Prob(JB):                  1.89e-32
Kurtosis:                       2.486   Cond. No.                  5.40e+05
==========================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 5.4e+05. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

```
#now we will create a multiple regression with all the variables including
dummy variables
final_churn_clean_df['intercept'] = 1
lm_MonthlyCharge_all =
sm.OLS(final_churn_clean_df['MonthlyCharge'],final_churn_clean_df[['Children','
Age','Income','Outage_sec_perweek','Email','Contacts','Yearly_equip_failure','T
enure','Bandwidth_GB_Year','DummyGender',
      'DummyChurn', 'DummyTechie', 'DummyContract', 'DummyPort_modem',
      'DummyTablet', 'DummyInternetService', 'DummyPhone', 'DummyMultiple',
      'DummyOnlineSecurity', 'DummyOnlineBackup', 'DummyDeviceProtection',
      'DummyTechSupport', 'DummyStreamingTV', 'DummyStreamingMovies',
      'DummyPaperlessBilling','Timely_Response','Timely_Fixes',
'Timely_Replacements',

'Reliability','Options','Respectful_Response','Couteous_Exchange',
                      'Evidence_Of_Active_Listening','intercept']]).fit()
print(lm_MonthlyCharge_all.summary())
```

```
                            OLS Regression Results
==========================================================================
====
Dep. Variable:          MonthlyCharge   R-squared:                    0.967
Model:                            OLS   Adj. R-squared:               0.966
Method:                 Least Squares   F-statistic:
8740.
Date:                Thu, 22 Jul 2021   Prob (F-statistic):            0.00
Time:                        17:03:35   Log-Likelihood:
-34791.
```

```
No. Observations:               10000   AIC:                     6.965e+04
Df Residuals:                    9966   BIC:                     6.990e+04
Df Model:                          33
Covariance Type:            nonrobust
==========================================================================
======================
                                   coef    std err          t      P>|t|
[0.025      0.975]
--------------------------------------------------------------------------
----------------------
Children                        -1.1885      0.040    -29.774      0.000
-1.267      -1.110
Age                              0.1307      0.004     31.294      0.000
0.122       0.139
Income                        2.421e-06   2.79e-06      0.867      0.386
-3.05e-06     7.9e-06
Outage_sec_perweek              -0.0036      0.026     -0.135      0.892
-0.055       0.048
Email                           -0.0030      0.026     -0.116      0.908
-0.054       0.048
Contacts                        -0.0670      0.080     -0.841      0.400
-0.223       0.089
Yearly_equip_failure            -0.0973      0.124     -0.786      0.432
-0.340       0.145
Tenure                          -3.1806      0.043    -74.060      0.000
-3.265      -3.096
Bandwidth_GB_Year                0.0390      0.001     75.206      0.000
0.038       0.040
DummyGender                     -2.8079      0.162    -17.375      0.000
-3.125      -2.491
DummyChurn                       2.4574      0.238     10.320      0.000
1.991       2.924
DummyTechie                      0.1853      0.211      0.877      0.380
-0.229       0.599
DummyContract                    0.4834      0.188      2.574      0.010
0.115       0.852
DummyPort_modem                 -0.2077      0.157     -1.319      0.187
-0.516       0.101
DummyTablet                     -0.1277      0.172     -0.742      0.458
-0.465       0.210
DummyInternetService            34.8859      0.206    168.947      0.000
34.481      35.291
DummyPhone                      -0.3759      0.271     -1.387      0.165
-0.907       0.155
DummyMultiple                   29.4633      0.165    179.037      0.000
29.141      29.786
DummyOnlineSecurity             -0.2806      0.170     -1.655      0.098
-0.613       0.052
DummyOnlineBackup               18.7465      0.166    113.217      0.000
18.422      19.071
DummyDeviceProtection            9.1251      0.164     55.481      0.000
8.803       9.448
DummyTechSupport                12.2992      0.163     75.545      0.000
11.980      12.618
```

```
DummyStreamingTV                32.7847      0.199    164.396      0.000
32.394      33.176
DummyStreamingMovies            43.4914      0.197    220.667      0.000
43.105      43.878
DummyPaperlessBilling            0.1274      0.160      0.796      0.426
-0.186       0.441
Timely_Response                 -0.0860      0.113     -0.762      0.446
-0.307       0.135
Timely_Fixes                     0.2169      0.106      2.053      0.040
0.010       0.424
Timely_Replacements              0.0093      0.097      0.096      0.923
-0.181       0.199
Reliability                      0.0592      0.087      0.683      0.494
-0.111       0.229
Options                          0.0339      0.090      0.376      0.707
-0.143       0.210
Respectful_Response             -0.0733      0.093     -0.791      0.429
-0.255       0.108
Couteous_Exchange               -0.0080      0.088     -0.091      0.928
-0.180       0.164
Evidence_Of_Active_Listening    -0.0708      0.083     -0.849      0.396
-0.234       0.093
intercept                       62.9387      0.937     67.153      0.000
61.102      64.776
==========================================================================
====
Omnibus:                    39029.228   Durbin-Watson:                1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1370.373
Skew:                           0.024   Prob(JB):                2.67e-298
Kurtosis:                       1.187   Cond. No.                 5.87e+05
==========================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 5.87e+05. This might indicate that
there are
strong multicollinearity or other numerical problems.
```

The R^2 of this model is 0.967. This means that the model explains about 97% of the variance which is high but not shocking since we have most of the original variables considered. At the bottom of the statistics, it indicates a condition number of 5.87e+05 which is very large and suggests strong multicollinearity in this model.

D2. Justify Selection Procedure for Reducing Initial Model

As stated above, the R^2 value of the initial multiple regression model is high at 97%. However, this does not necessarily mean that all of the variables taken into account are significant. Infact, in this next step we can take out several of the variables and still have a high R^2 value, leaving only significant variables in place while removing the insignificant ones.

We can achieve this by looking at the P-values of each variable in the initial multiple regression model. If the P-value is higher than the chosen alpha level of 0.05, we can safely assume that they are insignificant and can remove them to reduce the model and thus, making the model easier to work with (**Minitab 2019)**.

D3. Reduced Multiple Regression Model

```
#we will now justify a variable selection for reducing the model so that it
aligns better with the research question. We will do this by removing the
highest p values greater than alpha of 0.05 so that only significant variables
remain
#we see that even after removing the variables above p value 0.05, the R -
square value of 0.967 or 97% of data the model is still able to explain.
final_churn_clean_df['intercept'] = 1
lm_MonthlyCharge_Reduced =
sm.OLS(final_churn_clean_df['MonthlyCharge'],final_churn_clean_df[['Children','
Age','Tenure','Bandwidth_GB_Year','DummyGender',
       'DummyChurn', 'DummyContract', 'DummyInternetService', 'DummyMultiple',
        'DummyOnlineBackup', 'DummyDeviceProtection',
       'DummyTechSupport', 'DummyStreamingTV', 'DummyStreamingMovies',
    'Timely_Fixes','intercept']]).fit()
print(lm_MonthlyCharge_Reduced.summary())
```

```
OLS Regression Results
==============================================================================
====
Dep. Variable:          MonthlyCharge   R-squared:                       0.967
Model:                            OLS   Adj. R-squared:                  0.967
Method:                 Least Squares   F-statistic:                 1.923e+04
Date:                Fri, 23 Jul 2021   Prob (F-statistic):
0.00
Time:                        11:42:47   Log-Likelihood:
-34799.
No. Observations:               10000   AIC:                         6.963e+04
Df Residuals:                    9984   BIC:                         6.974e+04
Df Model:                          15
Covariance Type:            nonrobust
```

```
===========================================================================
===============
                            coef    std err          t      P>|t|      
[0.025      0.975]
---------------------------------------------------------------------------
---------------
Children                 -1.1823      0.040    -29.775      0.000    -1.260
-1.104
Age                       0.1298      0.004     31.257      0.000     0.122
0.138
Tenure                   -3.1624      0.042    -76.057      0.000    -3.244
-3.081
Bandwidth_GB_Year         0.0388      0.001     77.252      0.000     0.038
0.040
DummyGender              -2.8097      0.161    -17.431      0.000    -3.126
-2.494
DummyChurn                2.5053      0.237     10.580      0.000     2.041
2.970
DummyContract             0.4919      0.188      2.622      0.009     0.124
0.860
DummyInternetService     34.8258      0.203    171.234      0.000    34.427
35.225
DummyMultiple            29.4748      0.164    179.686      0.000    29.153
29.796
DummyOnlineBackup        18.7675      0.165    113.617      0.000    18.444
19.091
DummyDeviceProtection     9.1530      0.164     55.803      0.000     8.831
9.474
DummyTechSupport         12.2984      0.163     75.636      0.000    11.980
12.617
DummyStreamingTV         32.8196      0.197    166.259      0.000    32.433
33.207
DummyStreamingMovies     43.5178      0.196    222.563      0.000    43.135
43.901
Timely_Fixes              0.1164      0.076      1.532      0.126    -0.033
0.265
intercept                62.3580      0.468    133.179      0.000    61.440
63.276
===========================================================================
====
Omnibus:                    38804.477   Durbin-Watson:                1.999
Prob(Omnibus):                  0.000   Jarque-Bera (JB):          1381.110
Skew:                           0.023   Prob(JB):                1.25e-300
Kurtosis:                       1.180   Cond. No.                  2.48e+04
===========================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is 
correctly specified.
[2] The condition number is large, 2.48e+04. This might indicate that 
there are
strong multicollinearity or other numerical problems.
```

E1. Comparing the Initial and Reduced Multiple Regression Models

Here we find that even after removing 18 variables, with only 15 remaining the R^2 value of the model is exactly the same at 0.967 or 97% capable of explaining the variability. It further emphasizes that the variables with P-values above the 0.05 cut-off were indeed insignificant.

The most statistically significant variables were those with a P-value of 0, which in this case were quite a few with 13 in total:

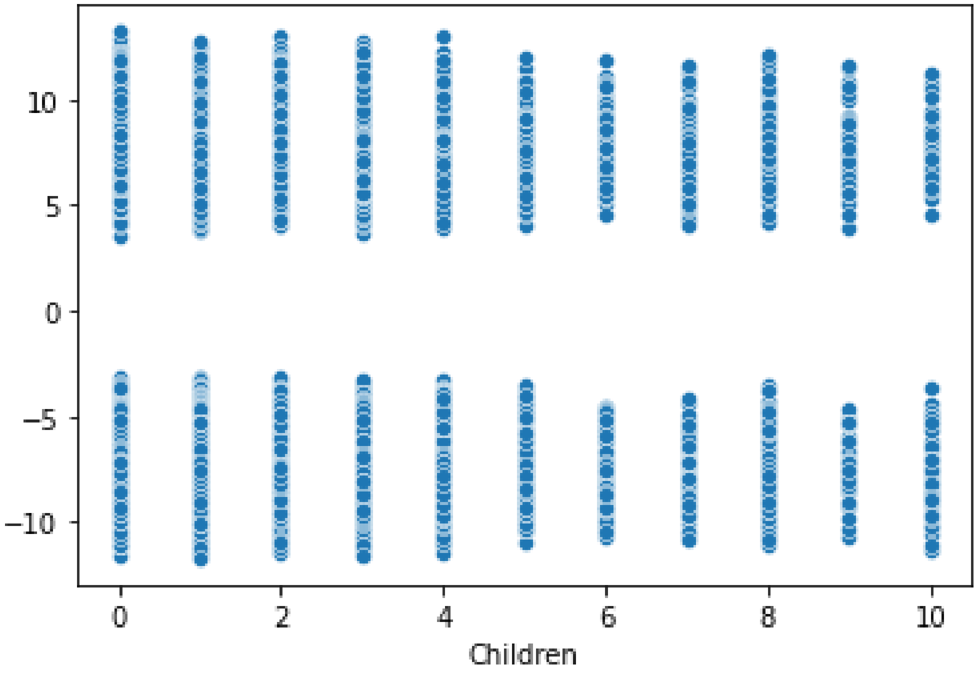
Children, Age, Tenure, Bandwidth_GB_Year, DummyGender,DummyChurn, DummyInternetService, DummyMultiple, DummyOnlineBackup, DummyDeviceProtection, DummyTechSupport, DummyStreamingTV, and DummyStreamingMovies

Our New Reduced Multiple Regression Model Equation is as follows:

Y = 62.3580 - 1.1823*Children +0.1298*Age - 3.1624*Tenure + 0.0388*Bandwidth_GB_Year - 2.8097*DummyGender + 2.5053*DummyChurn + 0.4919*DummyContract + 34.8258*DummyInternetService + 29.4748*DummyMultiple + 18.7675*DummyOnlineBackup + 9.1530*DummyDeviceProtection + 12.2984*DummyTechSupport + 32.8196*DummyStreamingTV + 43.5178*StreamingMovies + 0.1164*Timely_Fixes

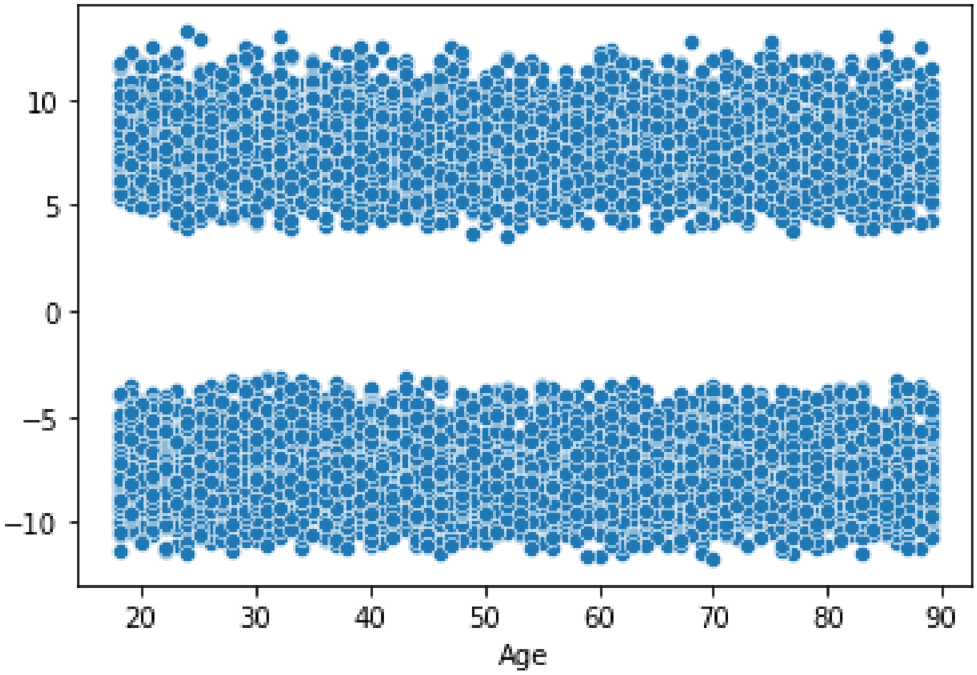
Also as shown below, the residuals express that the model is ideal due to the residuals distributed are trending towards the middle of the plot (**Qualtrics 2021)**

```
#now we will create a residual plot for the comparison of initial and reduced
regression models for each significant variable
final_churn_clean_df ['intercept']=1
residuals = final_churn_clean_df['MonthlyCharge'] -
lm_MonthlyCharge_Reduced.predict(final_churn_clean_df[['Children','Age','Tenure
','Bandwidth_GB_Year','DummyGender',
       'DummyChurn', 'DummyContract', 'DummyInternetService', 'DummyMultiple',
        'DummyOnlineBackup', 'DummyDeviceProtection',
       'DummyTechSupport', 'DummyStreamingTV', 'DummyStreamingMovies',
    'Timely_Fixes','intercept']])
sns.scatterplot(x=final_churn_clean_df['Children'], y=residuals)
plt.show()
```



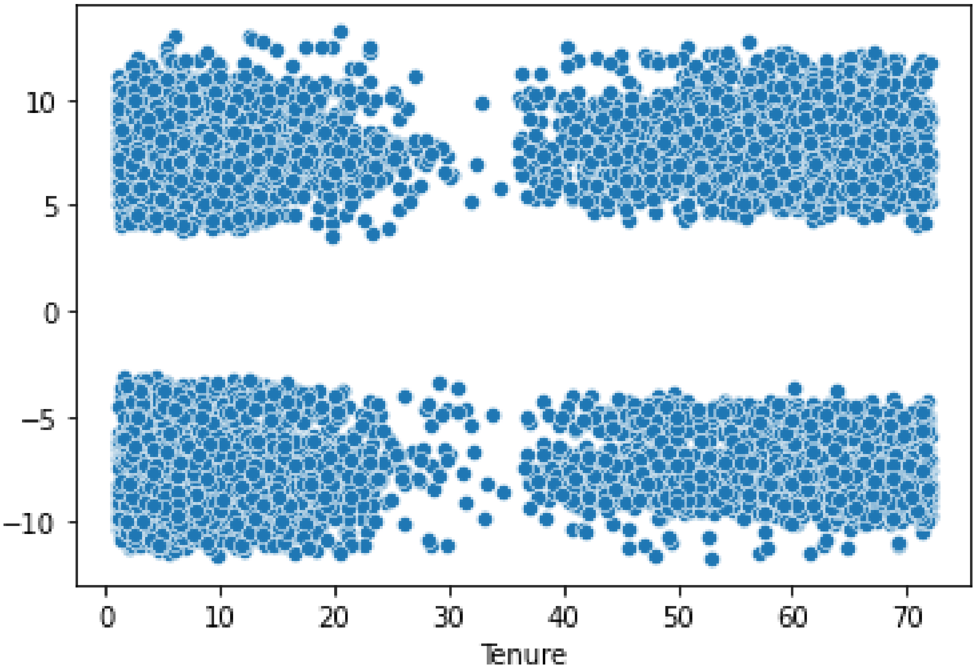
sns.scatterplot(x=final_churn_clean_df['Age'], y=residuals)

plt.show()

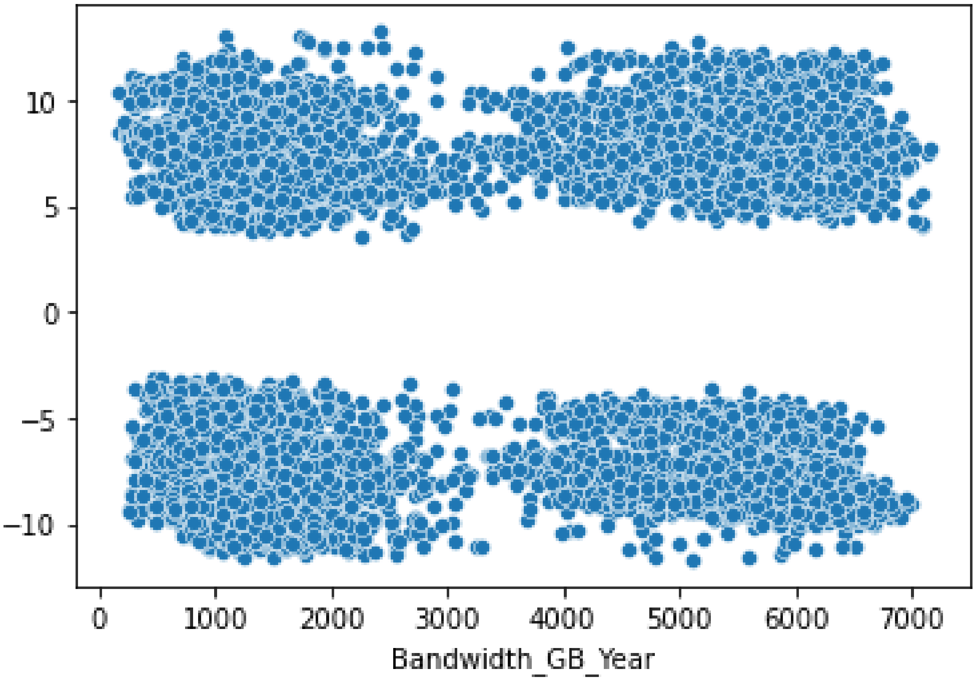


sns.scatterplot(x=final_churn_clean_df['Tenure'], y=residuals)

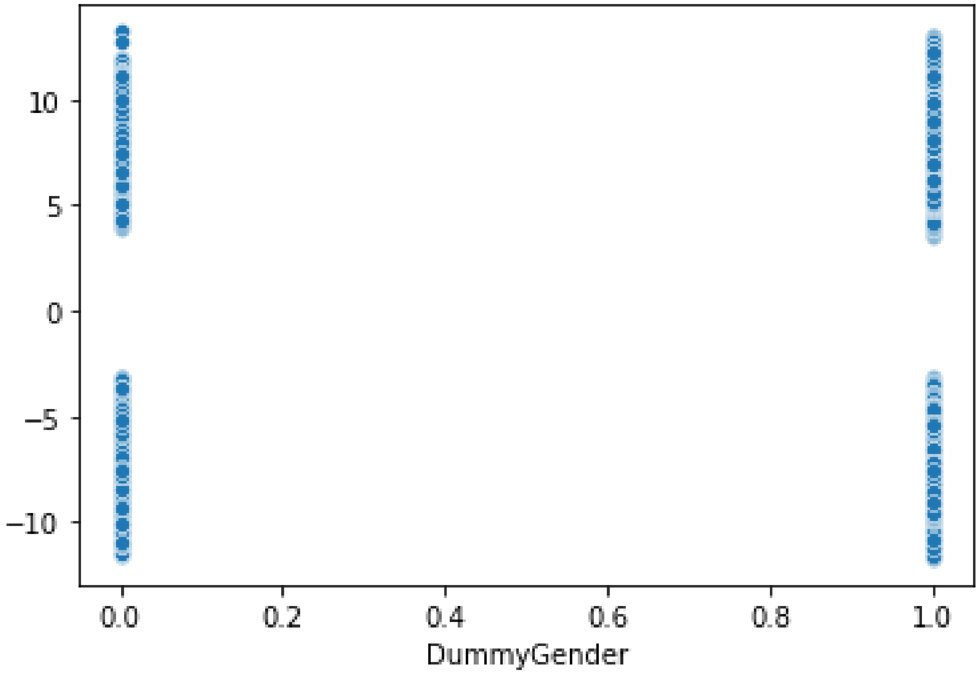
plt.show()



```
sns.scatterplot(x=final_churn_clean_df['Bandwidth_GB_Year'], y=residuals)

plt.show()
```

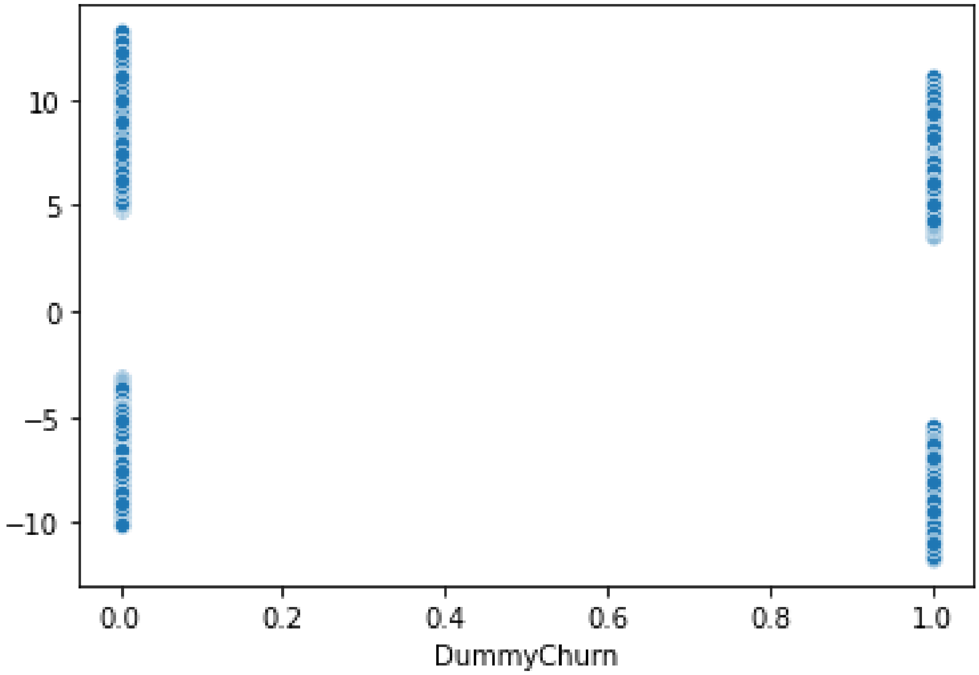


```
sns.scatterplot(x=final_churn_clean_df['DummyGender'], y=residuals)

plt.show()
```



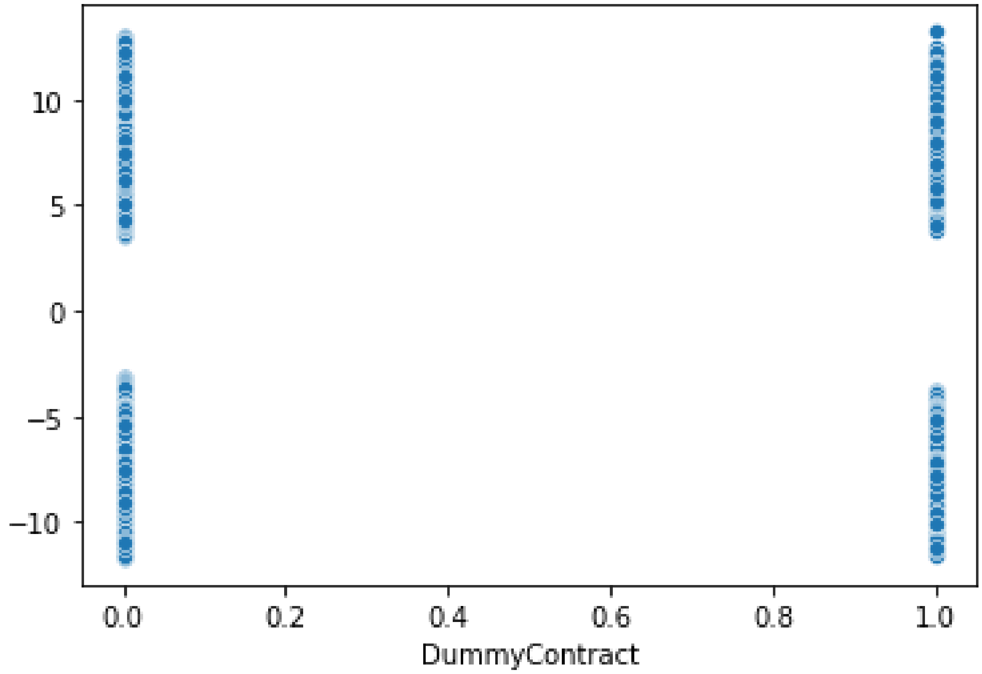
sns.scatterplot(x=final_churn_clean_df['DummyChurn'], y=residuals)

plt.show()



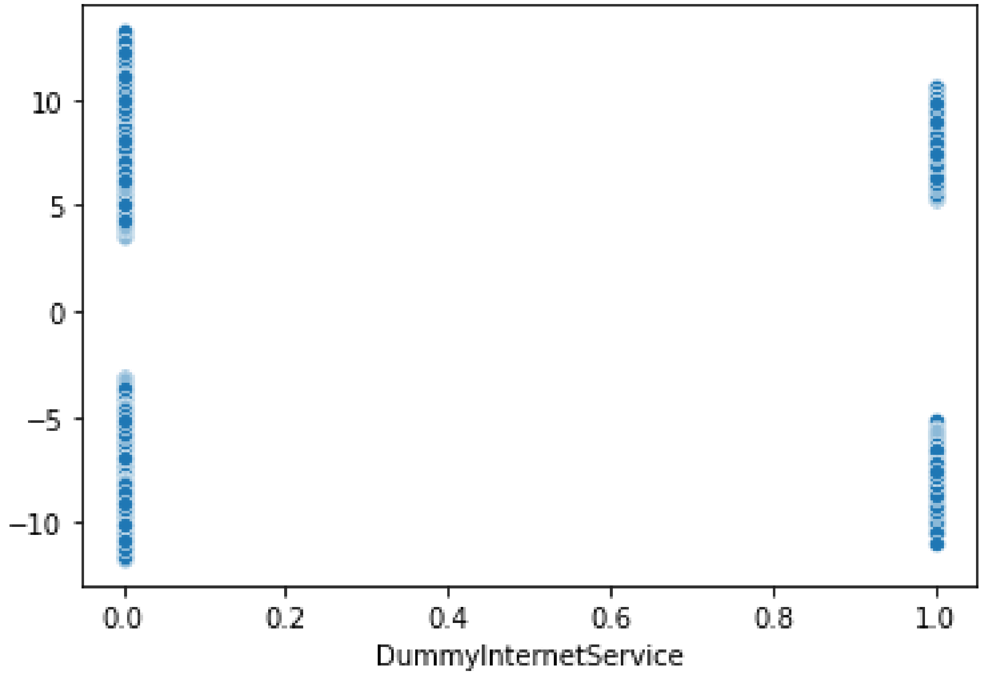
sns.scatterplot(x=final_churn_clean_df['DummyContract'], y=residuals)

plt.show()



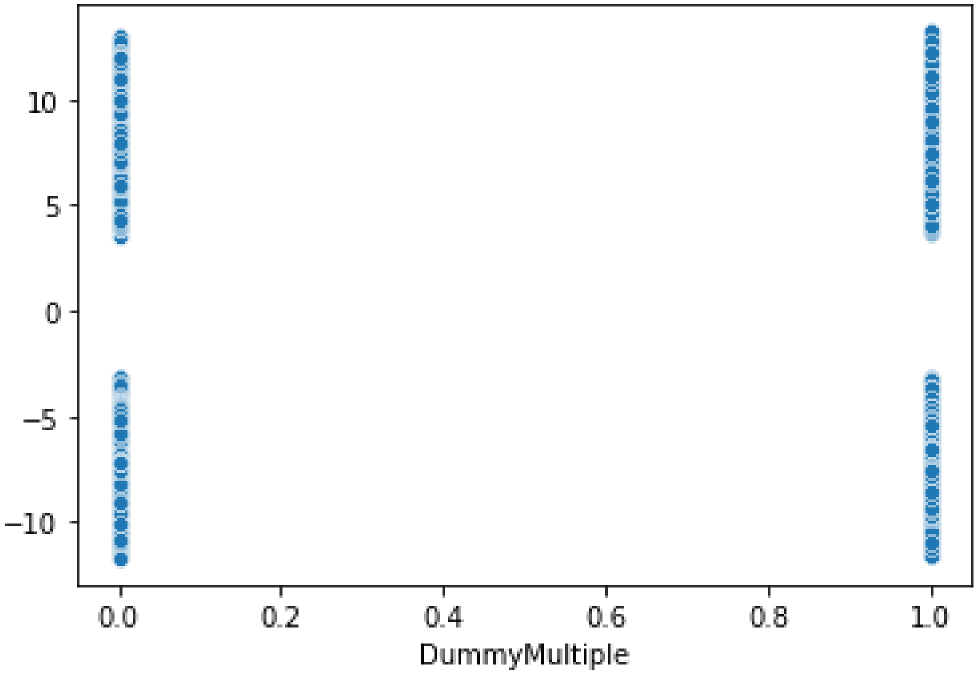
sns.scatterplot(x=final_churn_clean_df['DummyInternetService'], y=residuals)

plt.show()



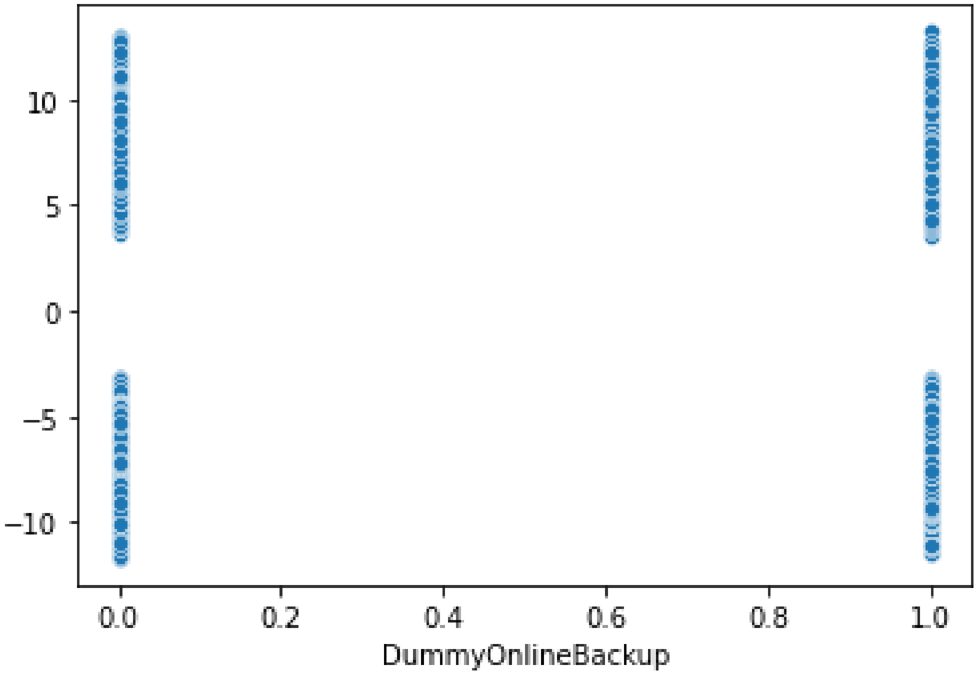
sns.scatterplot(x=final_churn_clean_df['DummyMultiple'], y=residuals)

plt.show()



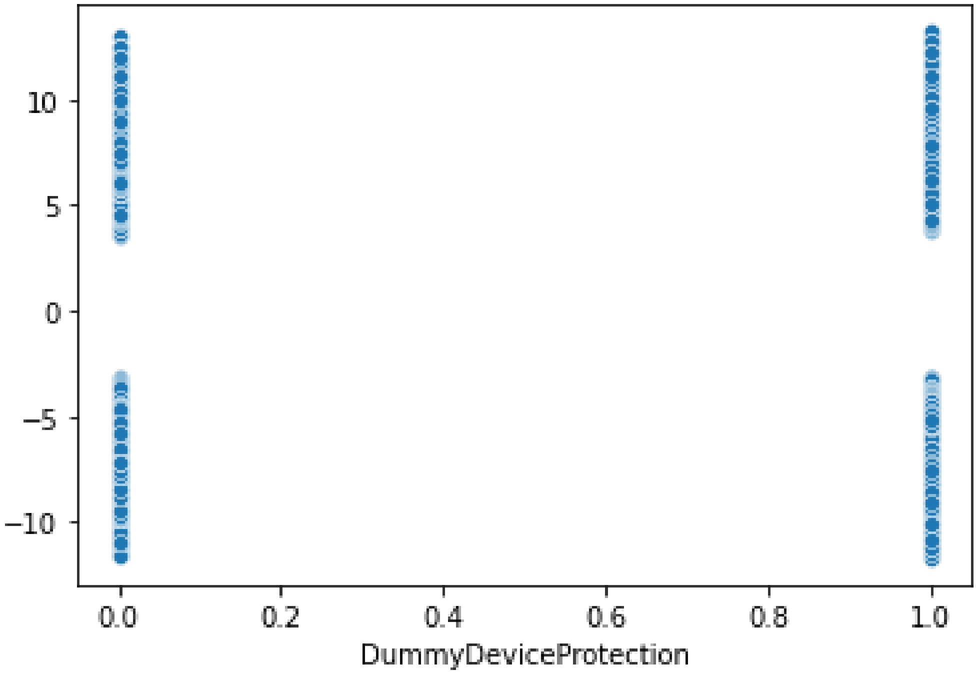
sns.scatterplot(x=final_churn_clean_df['DummyOnlineBackup'], y=residuals)

plt.show()



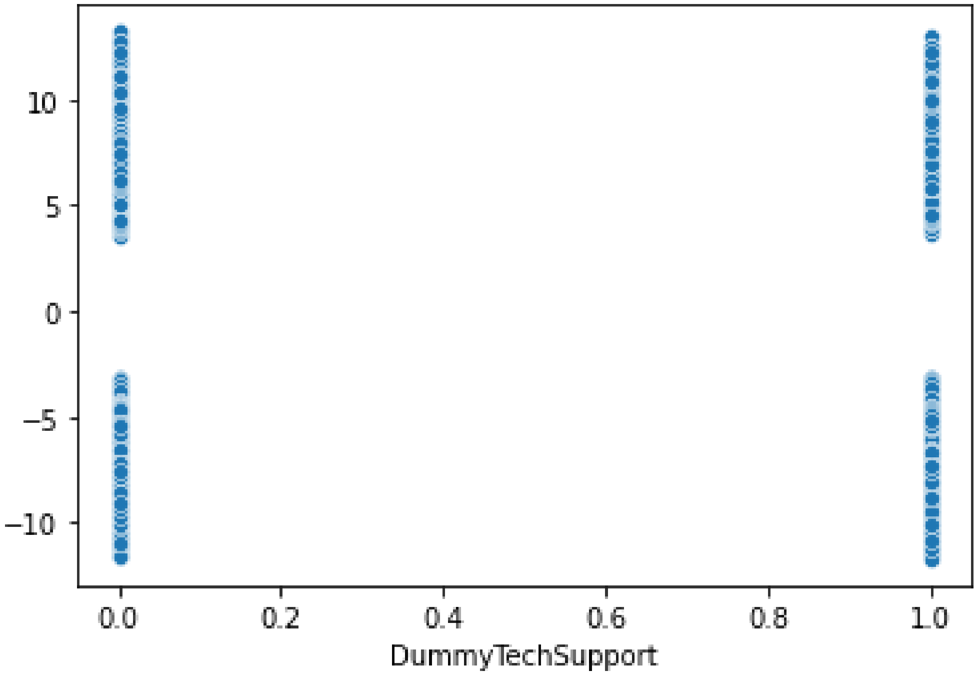
sns.scatterplot(x=final_churn_clean_df['DummyDeviceProtection'], y=residuals)

plt.show()



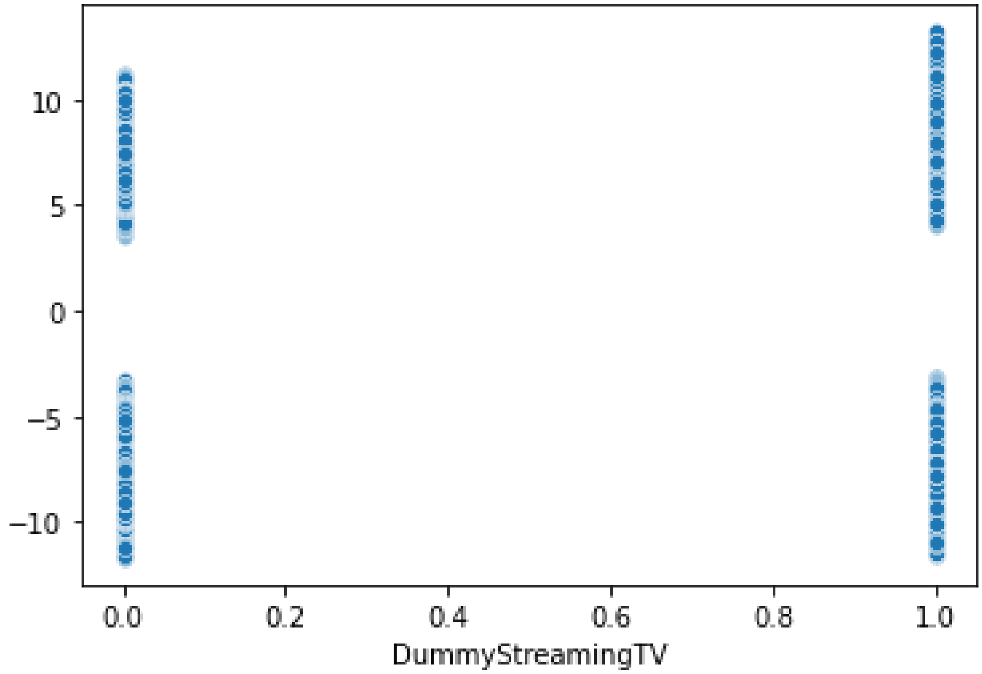
sns.scatterplot(x=final_churn_clean_df['DummyTechSupport'], y=residuals)

plt.show()



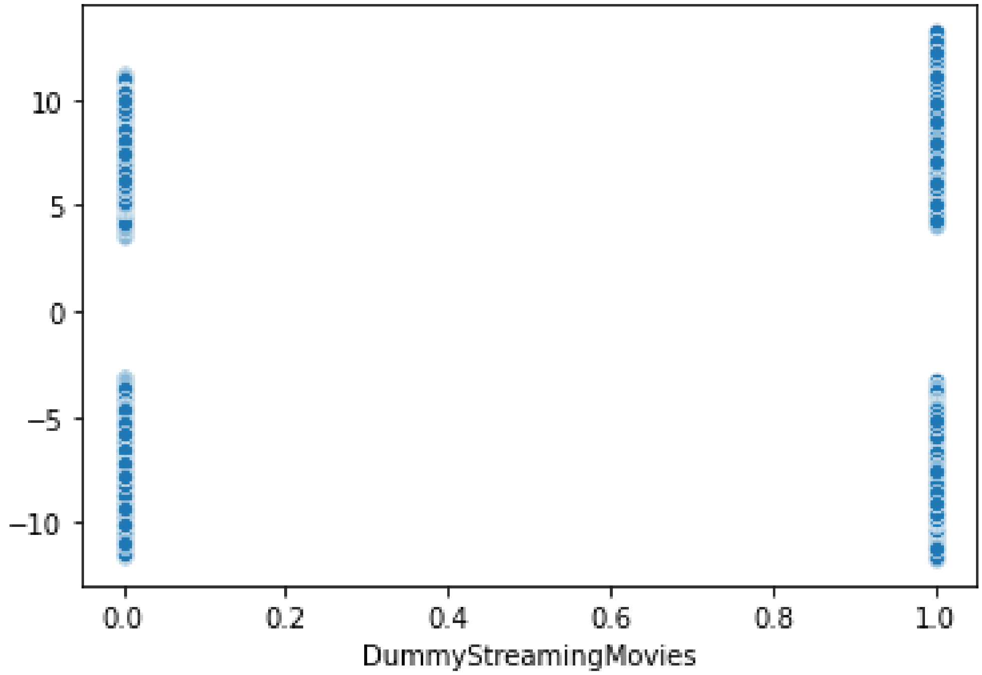
sns.scatterplot(x=final_churn_clean_df['DummyStreamingTV'], y=residuals)

plt.show()



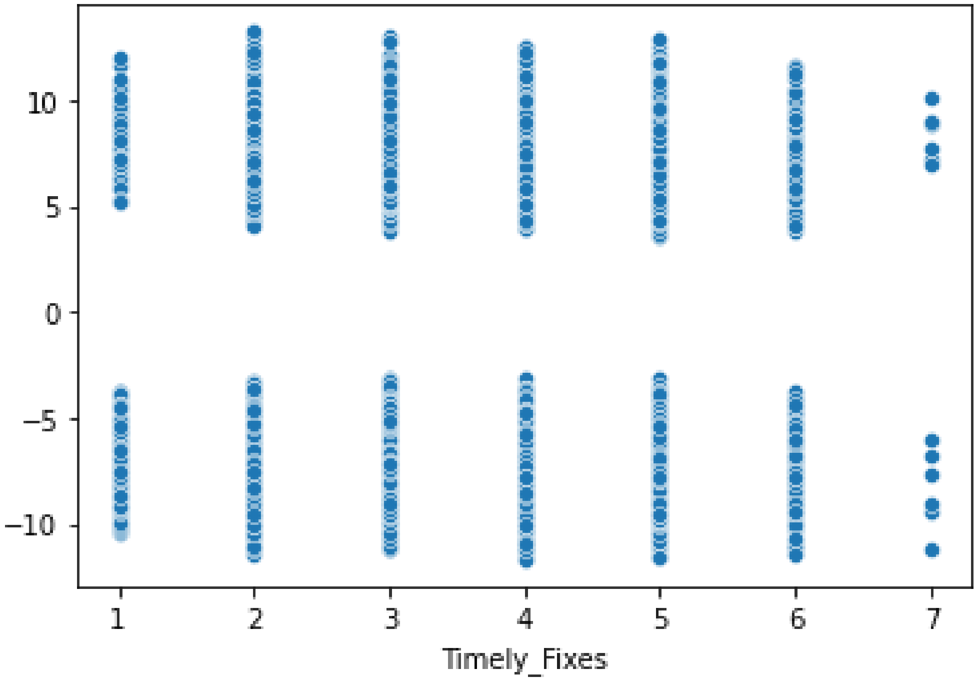
sns.scatterplot(x=final_churn_clean_df['DummyStreamingMovies'], y=residuals)

plt.show()



sns.scatterplot(x=final_churn_clean_df['Timely_Fixes'], y=residuals)

plt.show()



E2. Output and Calculations of the Analysis

Outputs and Calculations are provided above

E3. Code of Multiple Regression Analysis

Code provided above

Part V: Data Summary and Implications

F1. Results of Data Analysis

The regression equation for the reduced model is as follows

Y = 62.3580 - 1.1823*Children +0.1298*Age - 3.1624*Tenure + 0.0388*Bandwidth_GB_Year - 2.8097*DummyGender + 2.5053*DummyChurn + 0.4919*DummyContract + 34.8258*DummyInternetService + 29.4748*DummyMultiple + 18.7675*DummyOnlineBackup + 9.1530*DummyDeviceProtection + 12.2984*DummyTechSupport + 32.8196*DummyStreamingTV + 43.5178*StreamingMovies + 0.1164*Timely_Fixes

According to the Regression Analysis, it seems that the top 3 coefficients and there for contributors to the change in MonthlyCharge were: StreamingMovies, StreamingTV, Multiple, and InternetService

The Coefficients of each Continuous variable results are as follows:

Children = MonthlyCharge will decrease by 1.1823 units

Age = MonthlyCharge will increase by 0.1298 units

Tenure = MonthlyCharge will decrease by 3.1624 units

Bandwidth_GB_Year = MonthlyCharge will increase by 0.0388 units

Gender = MonthlyCharge will decrease by 2.8097 units

Churn = MonthlyCharge will increase by 2.5053 units

Contract = MonthlyCharge will increase by 0.4919 units

InternetService = MonthlyCharge will increase by 34.8258 units

Multiple = MonthlyCharge will increase by 29.4748 units

OnlineBackup = MonthlyCharge will increase by 18.7675 units

DeviceProtection = MonthlyCharge will increase by 9.1530 units

TechSupport = MonthlyCharge will increase by 12.2984 units

StreamingTV = MonthlyCharge will increase by 32.8196 unit

StreamingMovies = MonthlyCharge will increase by 43.5178 units

Timely_Fixes = MonthlyCharge will increase by 0.1164 units

While all of the above variable's P-values are below alpha level of 0.05, all the variables above except for Timely_Fixes had a P-Value of 0.00, making them statistically significant.

As far as limitations, it seems that the data set that we have of 10,000 observations is abit small, and so perhaps if we focus more on collection of data to increase the observations, we will have a more accurate prediction of the relationships.

F2. Recommended Course of Action

When looking at the results of our Regression Analysis, it is shown that the highest coefficient variables in relation to MonthlyCharge were StreamingTV and StreamingMovies. This shows us that a big portion of the service provider is being used for movies and TV shows. To capitalize on this, the company should focus on allocating more resources on making their platform easier, faster and more compatible for entertainment streaming.

If this is pursued, we can be sure that the company will increase the sales numbers per customer with the factor of about 43 units and 32 units respectively for movie and TV streaming.

G. Panapto Video

https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=7ac404c1-59ea-4695-814a-ad710002c6c7

H. Third Party Code Sources

Tim McAleer (2020). Interpreting Linear Regression Through Statsmodels.Summary(). https://medium.com/swlh/interpreting-linear-regression-through-statsmodels-summary-4796d359035a

Mirko Stojiljkovic (2021). Linear Regression in Python. https://realpython.com/linear-regression-in-python/#implementing-linear-regression-in-python

W3Schools (2021). Machine Learning – Multiple Regression. https://www.w3schools.com/python/python_ml_multiple_regression.asp

I. Acknowledged Sources

Minitab (2019). Model Reduction. https://support.minitab.com/en-us/minitab-express/1/help-and-how-to/modeling-statistics/regression/supporting-topics/regression-models/model-reduction/

Qualtrics (2021). Interpreting Residual Plots to Improve your Regression. https://www.qualtrics.com/support/stats-iq/analyses/regression-guides/interpreting-residual-plots-improve-regression/

Kripa Mahalingam (2020). The Importance of Sales Forecasting. https://www.chargebee.com/blog/importance-of-sales-forecasting/