Predictive Modeling
Task 1

Briana Churchill

Student ID: 011009463

Western Governors University, Data Analytics M.S. D208: Predictive Modeling

Dr. Eric Straw

August 22, 2023

Table of Contents

Part I: Research Question

A1. Question and Method Used to Analyze

A research question applicable to the dataset is "What factors are primarily contributed to long term customer tenure?" Answering this question involves analyzing one target variable, and various explanatory variables utilizing multiple linear regression. Such an analysis could allow the business to determine if there is a relationship between specific factors and length of tenure. It is costly to obtain new customers, and maximizing the probability of keeping customers for longer could benefit the company financially.

A2. Goals of the Data Analysis

The objective for this analysis is to determine which factors (independent/explanatory variables) contribute most significantly to the length of customer tenure (dependent/target variable). Achieving this goal will be done by using a multiple regression model, assessing the regression statistics and viewing visualizations of the data. Determining which factors have the most significant relationship to tenure will allow the business to build on its strengths and address any weaknesses. Ensuring that practices related to lengthy tenure stay consistent, or improve if needed, allows the business to avoid potential costs of churn.

Part II: Method Justification

B1. Four Assumptions of a Multiple Linear Regression Model

Four of the five assumptions of Multiple Linear Regression as described by the Statology article (Zach, 2021) and the potential effects of violations to these assumptions are:

- Between each and every predictor variable and the response variable a linear relationship is present.
    - If a linear relationship does not exist, the data is not useful to the model.
- High correlation between predictor variables is not present (no multicollinearity).
    - Multicollinearity leads to unreliable coefficient estimates.
- There is independence in each observation within the dataset.
    - Without independence of each observation, there is the possibility of autocorrelation. Autocorrelation may be utilized in other time series methods but would not be beneficial for this model.
- The residuals are not heteroscedastic. Instead, there is constant variance at each point in the linear model, or homoscedasticity.
    - Similar to the effects of multicollinearity, heteroscedasticity causes unreliable coefficient estimates.

B2. Benefits of Using Python

Initially I was interested in switching over to R for this assessment, as I wanted to get more familiar with that programming language. Switching languages during this course was not recommended per Dr. Straw's "Tips for Success" however. As such, one of the great benefits of using Python again was that I already had reliable scripts from D206 and D207 that I could utilize again for this assessment. Using my previous code allowed me to clean the data quicker and implement the univariate and bivariate explorations easier as well.

In addition to the ease of having previously used scripts available to me, overall Python is simple and easy to understand. From the packages to the visualizations, Python has been a great resource for me to analyze as much data as I have thus far. Not only are there multitudes of packages to use, but any error messages I encountered explained the issue well. Adjusting the code to be correct was not a difficult task, which made the overall analysis better to digest.

Each package used for this assessment, and its purpose:

- Numpy
    - Allows mathematical equations needed to transform the data
- Pandas
    - Provides a logical structure/data frame
- Matplotlib & Seaborn
    - Creates the visualizations (ie: pie chart, scatter plots, box plots, etc.)
- Scipy statsmodels
    - Used many times for various scripts related to:
        - Statistical models, including multiple regression models
        - Creating the linear regression plots
        - Completing the variance inflation factor (VIF)
- Sklearn
    - Used to normalize the data to better understand the data in the regression models (minmax scaler)

B3. Why Multiple Linear Regression is an Appropriate Technique

For this analysis I will be using multiple categorical and continuous independent variables (described below in part III) and one continuous dependent variable. The appropriate statistical tool to model the various relationships between each independent variable and the dependent variable would be Multiple Linear Regression. (Middleton, 2023) Using this regression method will allow for correct and appropriate predictions of the data, which may assist the business in answering the question at hand and implementing any necessary changes.

Part III: Data Preparation

C1. Data cleaning goals and the steps used

For the data to be prepared for analysis, it was essential to review and visually inspect the data frame to observe any outliers or necessary changes. Having known that the CSV file would not change between D207-D209, I was familiar with the dataset already. Having seen and worked with the data before, my goals of detecting and treating necessary variables were easy to achieve. This included ensuring appropriate spacing within the names of variables, scanning for and correcting outliers as well as null and duplicate values. To ensure smooth processing during latter steps of the analysis I also limited decimal points in certain columns.

The functions and scripts I used can be seen here:

```
# Clean up and prepare the data
# Rename the Case order column for proper spacing df = df.rename(columns={'Caseorder':
'Case_order'})

# Rename the Outage column for proper spacing

df = df.rename(columns={'Outage_sec_perweek': 'Outage_sec_per_week'})

# Rename the Onlinesecurity column for proper spacing

df = df.rename(columns={'OnlineSecurity': 'Online_Security'})

# Rename the Internetservice column for proper spacing

df = df.rename(columns={'InternetService': 'Internet_Service'})

# Rename the Onlinebackup column for proper spacing

df = df.rename(columns={'OnlineBackup': 'Online_Backup'})

# Rename the Deviceprotection column for proper spacing

df = df.rename(columns={'DeviceProtection': 'Device_Protection'})

# Rename the Techsupport column for proper spacing

df = df.rename(columns={'TechSupport': 'Tech_Support'})

# Rename the Streamingtv column for proper spacing

df = df.rename(columns={'StreamingTV': 'Streaming_TV'})

# Rename the Streamingmovies column for proper spacing

df = df.rename(columns={'StreamingMovies': 'Streaming_Movies'})

# Rename the Paperlessbilling column for proper spacing

df = df.rename(columns={'PaperlessBilling': 'Paperless_Billing'})

# Rename the Paymentmethod column for proper spacing
```

```
df = df.rename(columns={'PaymentMethod': 'Payment_Method'})
# Rename the Monthlycharge column for proper spacing
df = df.rename(columns={'MonthlyCharge': 'Monthly_Charge'})
# Minimalize decimal places in applicable columns
df["Age"] = df.Age.round(2)
df["Outage_sec_per_week"] = df.Outage_sec_per_week.round(2)df["Yearly_equip_failure"] =
df.Yearly_equip_failure.round(2)df["Monthly_Charge"] = df.Monthly_Charge.round(2)
df["Bandwidth_GB_Year"] = df.Bandwidth_GB_Year.round(2)
```

C2. Describe and show the summary statistics for all variables used

Using the .value_counts() and .describe() functions I was able to assess the various data points within each column. For categorical variables the .value_counts() function shows the applicable categories and the amount of customers within them. For the continuous variables, the .describe() function shows the summary statistics for each applicable variable. All variables and their applicable values assessed are described and can be seen in the images below.

Having a research question centered around the length of time a customer has stayed with the company, the dependent variable is Tenure. This variable is measured in months, with the average amount of tenure being 34.53 months. The minimum amount is about 1 month, and maximum nearing 72 months.

```
In [6]: # View data counts to ensure appropriate values for Tenure
        df.Tenure.describe()

Out[6]: count    10000.000000
        mean        34.526188
        std         26.443063
        min          1.000259
        25%          7.917694
        50%         35.430507
        75%         61.479795
        max         71.999280
        Name: Tenure, dtype: float64
```

Age is the first independent variable needed for the model. The summary statistics indicate that customer ages range from 18 years old to 89 years old.

```
In [7]: # View values for Age
        df.Age.describe()

Out[7]: count    10000.000000
        mean        53.078400
        std         20.698882
        min         18.000000
        25%         35.000000
        50%         53.000000
        75%         71.000000
        max         89.000000
        Name: Age, dtype: float64
```

Customer Income per year in USD has an average of $39,806.93, with the lowest value being $348.67 and the highest being $258,900. Initially the large variation between the minimum and maximum values was alarming. However, they are not out of the realm of possibilities, and allow for a diversified assessment.

```
In [8]: # View data counts to ensure appropriate values for Income
        df.Income.describe()

Out[8]: count     10000.000000
        mean      39806.926771
        std       28199.916702
        min         348.670000
        25%       19224.717500
        50%       33170.605000
        75%       53246.170000
        max      258900.700000
        Name: Income, dtype: float64
```

Various marital categories can be seen in the image along with the counts of customers in each particular group. Separating customers into their respective groups is most accurate for this analysis, rather than grouping all non-married people in one group and all married people in another.

```
In [9]: # View data counts to ensure appropriate values for Marital
        df.Marital.value_counts()

Out[9]: Divorced         2092
        Widowed          2027
        Separated        2014
        Never Married    1956
        Married          1911
        Name: Marital, dtype: int64
```

Gender is another independent variable that could potentially be attributed to the length of tenure. Not only are those that identify as female or male accounted for, but nonbinary people as well. Allowing people to choose their preferred gender identity ensures that everyone is accounted for correctly.

```
In [10]: # View data counts to ensure appropriate values for Gender
         df.Gender.value_counts()

Out[10]: Female       5025
         Male         4744
         Nonbinary     231
         Name: Gender, dtype: int64
```

All independent variables so far are related to customer demographics. There are many potential factors associated with lengthy tenure, however. Including the outages variable was necessary to determine if service outages small or large play a role in tenure. The average amount of outages in seconds per week is about 10 seconds, with the maximum being 21.21 seconds.

```
In [11]: # View data counts to ensure appropriate values for Outages
         df.Outage_sec_perweek.describe()

Out[11]: count    10000.000000
         mean        10.001848
         std          2.976019
         min          0.099747
         25%          8.018214
         50%         10.018560
         75%         11.969485
         max         21.207230
         Name: Outage_sec_perweek, dtype: float64
```

Yearly equipment failure or rather the lack thereof may contribute to extended customer tenure as well. Including this variable allows the business to see any potential effects that yearly equipment failures have on customer retention rates. As the statistics state, most customers that experienced equipment failures encountered the issue only once.

```
In [12]: # View values for Yearly equipment failure
         df.Yearly_equip_failure.describe()

Out[12]: count    10000.000000
         mean         0.398000
         std          0.635953
         min          0.000000
         25%          0.000000
         50%          0.000000
         75%          1.000000
         max          6.000000
         Name: Yearly_equip_failure, dtype: float64
```

Surprisingly more customers have month-to-month contracts as compared to one- or two-year contracts. Having the ability to change carriers at any given time may result in less time as a customer, and as such this variable is a valuable choice to include.

```
In [13]: # View data counts to ensure appropriate values for Contract
         df.Contract.value_counts()

Out[13]: Month-to-month    5456
         Two Year          2442
         One year          2102
         Name: Contract, dtype: int64
```

Most customers utilize fiber optic for their internet service type. DSL being the next most used option. Then there are 2,129 customers that use none at all. Analyzing the various categories of customers compared to tenure is important for this analysis as well.

```
In [14]: # View data counts to ensure appropriate values for Internet Service
         df.InternetService.value_counts()

Out[14]: Fiber Optic    4408
         DSL            3463
         None           2129
         Name: InternetService, dtype: int64
```

There is not an equal representation of whether customers have multiple phone lines or not, but the values are somewhat close, nonetheless. This variable was included because I imagine that having multiple phone lines would be more difficult to move from company to company. Determining if there is such a relationship of multiple lines on tenure will inform the research question.

```
In [15]: # View data counts to ensure appropriate values for Multiple
         df.Multiple.value_counts()

Out[15]: No     5392
         Yes    4608
         Name: Multiple, dtype: int64
```

More than half of the customers in the dataset do not utilize tech support. With technology advancing each and every day it can be overwhelming for consumers to enjoy the products they use. Weighing this variable in the analysis could also inform the research question.

```
In [16]: # View data counts to ensure appropriate values for Tech Support
         df.TechSupport.value_counts()

Out[16]: No     6250
         Yes    3750
         Name: TechSupport, dtype: int64
```

Monthly charge values range from 79.98 to 290.16. Every customer may have a different phone and internet plan compared to others. There are additional packages that customers can purchase also. Instead of comparing the monthly charge from customer to customer, it is most appropriate to review this variable in comparison to tenure for the purpose of this analysis.

```
In [17]: # View data counts to ensure appropriate values for Monthly Charge
         df.MonthlyCharge.describe()

Out[17]: count    10000.000000
         mean       172.624816
         std         42.943094
         min         79.978860
         25%        139.979239
         50%        167.484700
         75%        200.734725
         max        290.160419
         Name: MonthlyCharge, dtype: float64
```

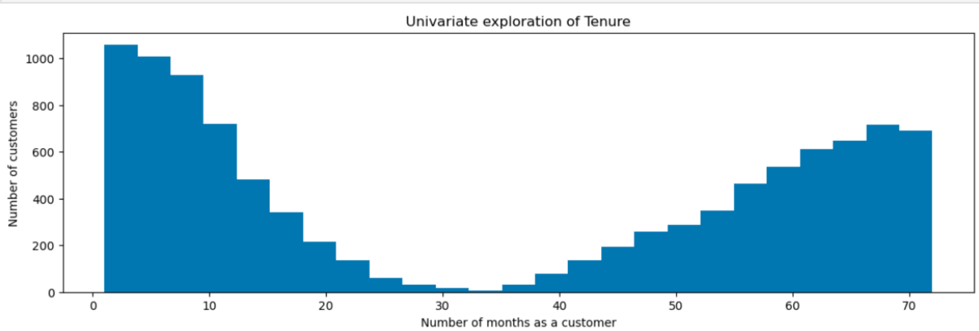
Bandwidth is measured in GB per year. Some customers use a lot, with the maximum being 7,158.98GB. Others use very little as the minimum amount of 155.51GB indicates.

```
In [18]: # View data counts to ensure appropriate values for Bandwidth
         df.Bandwidth_GB_Year.describe()

Out[18]: count    10000.000000
         mean      3392.341550
         std       2185.294852
         min        155.506715
         25%       1236.470827
         50%       3279.536903
         75%       5586.141370
         max       7158.981530
         Name: Bandwidth_GB_Year, dtype: float64
```

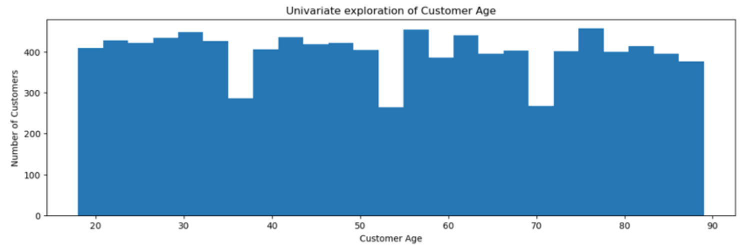
C3. Generate univariate and bivariate visualizations

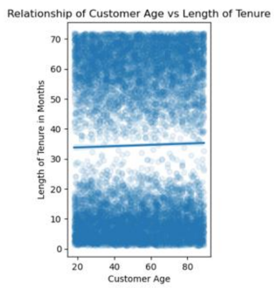
For the first plot, I completed a univariate exploration of the dependent variable Tenure. The remaining plots are of various types as can be seen (pie chart, scatter, etc.) and are of univariate explorations of the independent variables. As well as bivariate explorations of each independent variable with the dependent variable.

```
In [22]: # Determine size of visualization
         plt.figure(figsize = [14,4])
         #Univariate exploration of Tenure
         plt.title("Univariate exploration of Tenure")
         bins = np.arange(0, df["Tenure"].max() + 2, 1)
         plt.hist(model_df["Tenure"], bins=25)
         plt.xlabel("Number of months as a customer")
         plt.ylabel("Number of customers")
         plt.show()
```

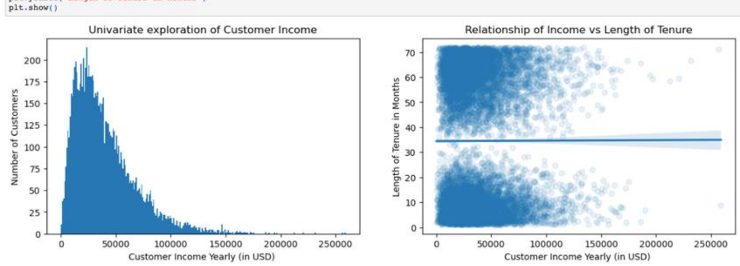


In [23]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Age
plt.title("Univariate exploration of Customer Age")
bins = np.arange(0, df["Age"].max() + 2, 1)
plt.hist(model_df["Age"], bins=25)
plt.xlabel("Customer Age")
plt.ylabel("Number of Customers")
plt.show()

# Second plot: Bivariate exploration of Age vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Customer Age vs Length of Tenure")
sns.regplot(data=model_df, x="Age", y="Tenure", scatter_kws={'alpha' : 1/10})
plt.xlabel("Customer Age")
plt.ylabel("Length of Tenure in Months")
plt.show()
```

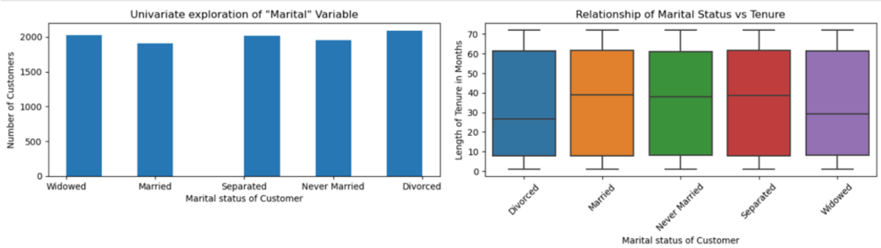




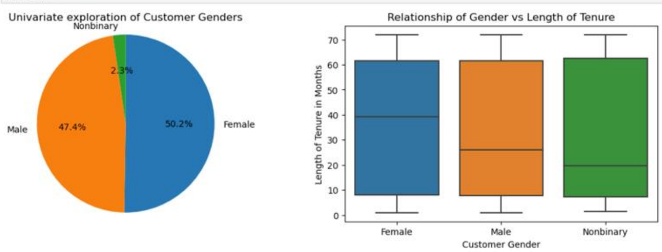
In [24]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of income
plt.subplot(1, 2, 1)
plt.title('Univariate exploration of Customer Income')
bins = np.arange(0, df.Income.max() + 500, 1000)
plt.hist(data=model_df, x="Income", bins=bins)
plt.xlabel('Customer Income Yearly (in USD) ')
plt.ylabel("Number of Customers")

# Second plot: Bivariate exploration of income vs tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Income vs Length of Tenure")
sns.regplot(data=model_df, x="Income", y="Tenure", scatter_kws={'alpha' : 1/10})
plt.xlabel("Customer Income Yearly (in USD) ")
plt.ylabel("Length of Tenure in Months")
plt.show()
```



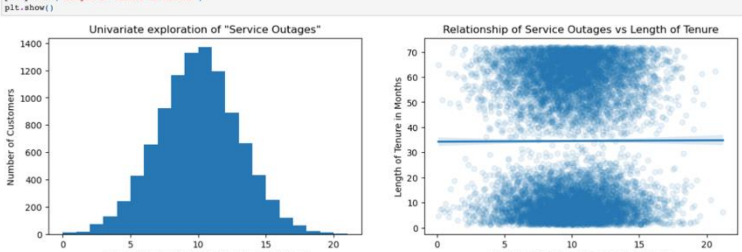
In [25]:
```python
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Marital Variable
plt.subplot(1, 2, 1)
plt.title('Univariate exploration of "Marital" Variable')
Marital_counts = df["Marital"].value_counts()
Marital_labels = ["Divorced", "Widowed", "Separated", "Never Married", "Married"]
plt.hist(x=df["Marital"], bins=10)
plt.xlabel('Marital status of Customer')
plt.ylabel("Number of Customers")

# Second plot: Bivariate exploration of Marital vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Marital Status vs Tenure")
sns.boxplot(data=df, x="Marital", y="Tenure")
plt.xlabel("Marital status of Customer")
plt.ylabel("Length of Tenure in Months")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



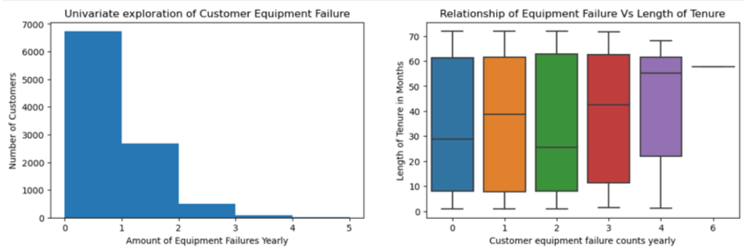
In [26]:
```python
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Gender
plt.subplot(1, 2, 1)
plt.title("Univariate exploration of Customer Genders")
Gender_counts = df["Gender"].value_counts()
plt.pie(Gender_counts, labels=Gender_counts.index, autopct='%1.1f%%', startangle=90, counterclock = False)
plt.axis('square')

# Second plot: Bivariate exploration of Gender vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Gender vs Length of Tenure")
sns.boxplot(data=df, x="Gender", y='Tenure')
plt.xlabel("Customer Gender")
plt.ylabel("Length of Tenure in Months")
plt.show()
```



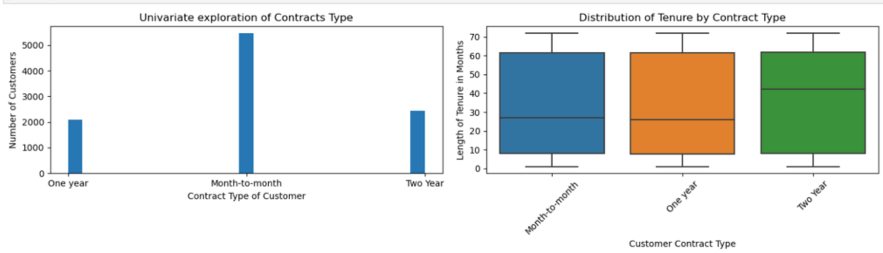
In [27]:
```python
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Outage Variable
plt.subplot(1, 2, 1)
plt.title('Univariate exploration of "Service Outages"')
bins = np.arange(0, df.Outage_sec_per_week.max())
plt.hist(data=model_df, x="Outage_sec_per_week", bins=bins)
plt.xlabel('Weekly Service Outages in Seconds per Week)')
plt.ylabel("Number of Customers")

# Second plot: Bivariate exploration of Outages vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Service Outages vs Length of Tenure")
sns.regplot(data=model_df, x="Outage_sec_per_week", y="Tenure", scatter_kws={'alpha' : 1/10})
plt.xlabel("Customer Outages in Seconds per Week")
plt.ylabel("Length of Tenure in Months")
plt.show()
```



In [28]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Customer Equipment Failure
plt.subplot(1, 2, 1)
plt.title("Univariate exploration of Customer Equipment Failure")
bins = np.arange(0, model_df.Yearly_equip_failure.max())
plt.hist(data=model_df, x="Yearly_equip_failure", bins=bins)
plt.xlabel('Amount of Equipment Failures Yearly')
plt.ylabel("Number of Customers")

# Second plot: Bivariate exploration of Equipment Failure vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Equipment Failure Vs Length of Tenure")
sns.boxplot(data=model_df, x="Yearly_equip_failure", y='Tenure')
plt.xlabel("Customer equipment failure counts yearly")
plt.ylabel("Length of Tenure in Months")
plt.show()
```

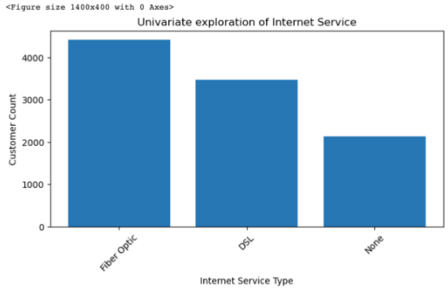


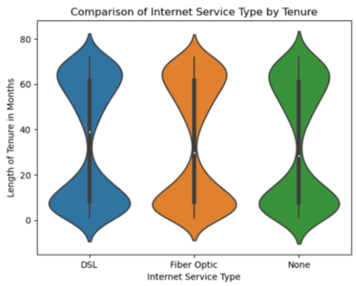
In [29]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Contracts
plt.subplot(1, 2, 1)
plt.title('Univariate exploration of Contracts Type')
Contract_counts = df["Contract"].value_counts()
Contract_labels = ["Month-to-month", "Two Year", "One year"]
plt.hist(x=df["Contract"], bins=25)
plt.xlabel('Contract Type of Customer')
plt.ylabel("Number of Customers")

# Second plot: Bivariate exploration of Contracts vs Tenure
plt.subplot(1, 2, 2)
plt.title("Distribution of Tenure by Contract Type")
sns.boxplot(data=df, x="Contract", y="Tenure")
plt.xlabel("Customer Contract Type")
plt.ylabel("Length of Tenure in Months")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



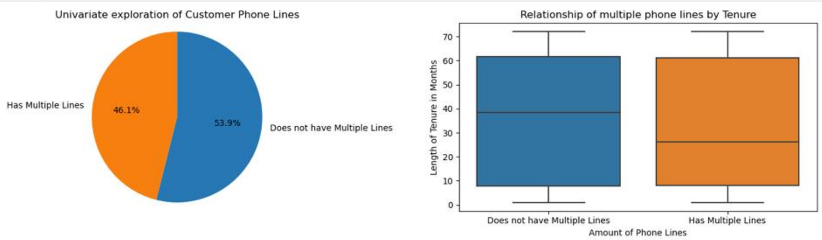
In [30]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Internet Service
Internet_Service_counts = df["Internet_Service"].value_counts()
plt.figure(figsize=(8, 4))
plt.bar(Internet_Service_counts.index, Internet_Service_counts.values)
plt.title("Univariate exploration of Internet Service")
plt.xlabel("Internet Service Type")
plt.ylabel("Customer Count")
plt.xticks(rotation=45)
plt.show()

# Second plot: Bivariate exploration of Service Type vs. Tenure
sns.violinplot(x='Internet_Service', y='Tenure', data=df)
plt.xlabel('Internet Service Type')
plt.ylabel('Length of Tenure in Months')
plt.title('Comparison of Internet Service Type by Tenure')
plt.show()
```

```
<Figure size 1400x400 with 0 Axes>
```

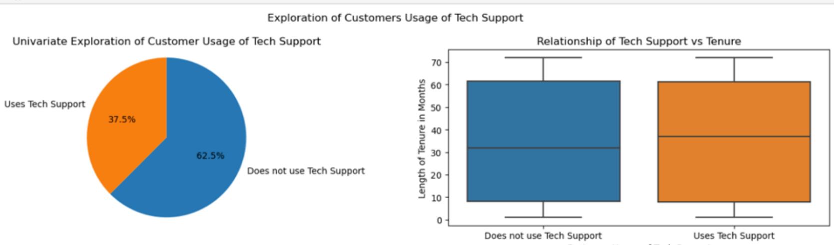




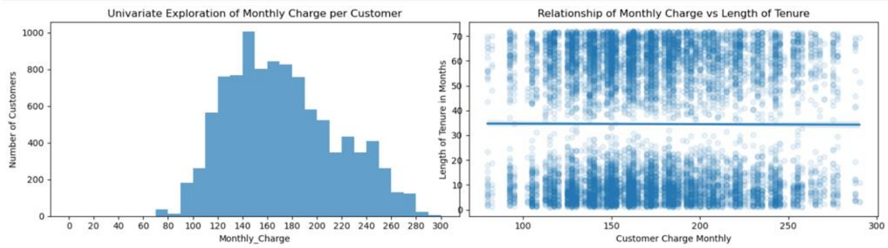
In [31]:
```
# Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of variable "Multiple"
plt.subplot(1, 2, 1)
plt.title("Univariate exploration of Customer Phone Lines")
Multiple_counts = model_df['Multiple'].value_counts()
Multiple_labels = ["Does not have Multiple Lines", "Has Multiple Lines"]
plt.pie(Multiple_counts, labels=Multiple_labels, autopct='%1.1f%%', startangle=90, counterclock=False)
plt.axis('equal')

# Second plot: Bivariate exploration of Multiple Lines vs. Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of multiple phone lines by Tenure")
sns.boxplot(data=model_df, x="Multiple", y='Tenure')
plt.xticks(ticks=[0, 1], labels=Multiple_labels)
plt.xlabel("Amount of Phone Lines")
plt.ylabel("Length of Tenure in Months")
plt.tight_layout()
plt.show()
```



In [32]:
```
# Determine size and title of visualization
plt.figure(figsize = [14,4])
plt.suptitle("Exploration of Customers Usage of Tech Support")

# First plot: Univariate exploration of variable Tech_Support
plt.subplot(1, 2, 1)
plt.title("Univariate Exploration of Customer Usage of Tech Support")
Tech_Support_counts = model_df['Tech_Support'].value_counts()
Tech_Support_labels = ["Does not use Tech Support", "Uses Tech Support"]
plt.pie(Tech_Support_counts, labels=Tech_Support_labels, autopct='%1.1f%%', startangle=90, counterclock=False)
plt.axis('equal')

# Second plot: Bivariate exploration of Tech Support vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Tech Support vs Tenure")
sns.boxplot(data=model_df, x="Tech_Support", y='Tenure')
plt.xticks(ticks=[0, 1], labels=Tech_Support_labels)
plt.xlabel("Customer Usage of Tech Support")
plt.ylabel("Length of Tenure in Months")
plt.tight_layout()
plt.show()
```

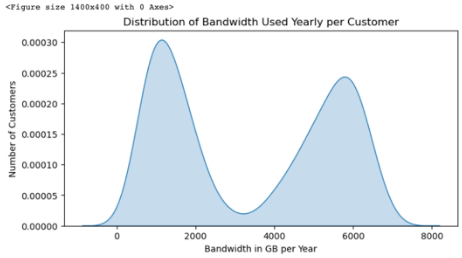


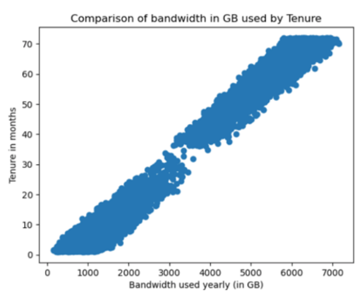
```
In [33]: # Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Monthly Charge variable
plt.subplot(1, 2, 1)
plt.title('Univariate Exploration of Monthly Charge per Customer')
bins = np.arange(0, df.Monthly_Charge.max() + 10, 10)
plt.hist(data=model_df, x="Monthly_Charge", bins=bins, alpha=0.7)
plt.xlabel('Monthly_Charge')
plt.ylabel("Number of Customers")
plt.xticks(np.arange(0, df['Monthly_Charge'].max() + 10, 20))
plt.tight_layout()

# Second plot: A plot for the Bivariate exploration of Monthly Charge vs Tenure
plt.subplot(1, 2, 2)
plt.title("Relationship of Monthly Charge vs Length of Tenure")
sns.regplot(data=model_df, x="Monthly_Charge", y="Tenure", scatter_kws={'alpha' : 1/10})
plt.xlabel("Customer Charge Monthly")
plt.ylabel("Length of Tenure in Months")
plt.show()
```




```
In [34]: # Determine size of visualization
plt.figure(figsize = [14,4])

# First plot: Univariate exploration of Bandwidth Variable
plt.figure(figsize=(8, 4))
sns.kdeplot(model_df['Bandwidth_GB_Year'], fill=True)
plt.title("Distribution of Bandwidth Used Yearly per Customer")
plt.xlabel("Bandwidth in GB per Year")
plt.ylabel("Number of Customers")
plt.show()

# Second plot: Bivariate exploration of Bandwidth vs Tenure
plt.scatter(model_df['Bandwidth_GB_Year'], model_df['Tenure'])
plt.xlabel('Bandwidth used yearly (in GB)')
plt.ylabel('Tenure in months')
plt.title('Comparison of bandwidth in GB used by Tenure')
plt.show()
```

```
<Figure size 1400x400 with 0 Axes>
```





C4. Data transformation goals

While cleaning the data I noted the data types for the variables that needed to be transformed in order to complete the analysis. Some of the data types were not the optimal choice for particular columns to begin with. Adding the fact that this analysis requires a Multiple Regression Model, I knew all variables that would be used needed to be numeric in nature. Not all data variables were numeric initially, so certain variables required re-expression. Starting with "yes" or "no" data points within the "Multiple" and "TechSupport" variables, I applied ordinal encoding to make them numeric; 1 for "yes" and 0 for "no." With the information obtained from Unifying Data Science, I created dummy columns for the remaining categorical variables and included these values in a new data frame with all the other variables I needed for the model.

(Eubank, 2022) Creating these dummy columns, otherwise known as one hot encoding, allows the applicable data points to be accounted for in a numeric sense by also using 1 or 0 to identify the specific categorical value applicable to each customer. One value from each column is omitted, and the remaining variables will have a value of 1 if they are applicable to the customer. If all remaining variables have a value of 0, then the omitted value is applicable. All code used can be seen attached to this assessment, and here:

```
# Convert Marital column to category from object
df["Marital"] = df["Marital"].astype("category")
# Convert Gender column to category from object
df["Gender"] = df["Gender"].astype("category")
# Convert Internet Service column to category from object
df["Internet_Service"] = df["Internet_Service"].astype("category")
# Convert Payment Method column to category from object
df["Payment_Method"] = df["Payment_Method"].astype("category")
# Convert Contract column to category from object
df["Contract"] = df["Contract"].astype("category")
# Change all yes/no values to 1 or 0 by mapping
mapping = {'Yes': 1, 'No': 0, 'unknown': np.nan}
# Apply the mapping to applicable columns
convert = ["Multiple" , "Tech_Support"] df[convert] = df[convert].replace(mapping)
# Create dummy variables for applicable columns and new dataframe
Dummy_Variables = ["Gender", "Marital", "Internet_Service", "Contract"]dummy_dfs = []
for column in Dummy_Variables:
dummy_df = pd.get_dummies(data=df[column], prefix=column, drop_first=True)
dummy_dfs.append(dummy_df)
model_df = df[["Age", "Income" , "Outage_sec_per_week", "Yearly_equip_failure",
"Multiple", "Tech_Support", "Tenure" , "Monthly_Charge", "Bandwidth_GB_Year"]]
# Concatenate the dummy variables/df with the original df
model_df = pd.concat([model_df] + dummy_dfs, axis=1)
# Visually inspect the new dataframe
pd.set_option("display.max_columns", None) print(model_df.head(5))
```

C5. Provide the prepared data set as a CSV file.

Attached to the submission of this assessment the CSV file model_df.csv can be found.

Part IV: Model Comparison and Analysis

D1. Construct an initial multiple linear regression model

The initial multiple linear regression model was created along with the residual standard error so that the results could be compared to the residual model in the ladder steps. For now, a screengrab of the initial model results and the residual standard error, can be seen here:

```
In [37]: # Create model for Multiple Linear Regression
         # Set dependent variable
         y = model_df.Tenure
         # Set independent variables
         X = model_df[["Age", "Income" , "Outage_sec_per_week", "Yearly_equip_failure",
                          "Multiple", "Tech_Support" , "Monthly_Charge", "Bandwidth_GB_Year" ,
                              "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                             "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" , "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)
         model = sm.OLS(y, X)
         reg_results = model.fit()
         print(reg_results.summary())

         # Retrieve residual standard error
         reg_results.resid.std(ddof=X.shape[1])
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.999
Model:                            OLS   Adj. R-squared:                  0.999
Method:                 Least Squares   F-statistic:                 4.121e+05
Date:                Thu, 17 Aug 2023   Prob (F-statistic):               0.00
Time:                        16:24:37   Log-Likelihood:                -13878.
No. Observations:               10000   AIC:                         2.779e+04
Df Residuals:                    9981   BIC:                         2.793e+04
Df Model:                          18
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Age                              0.0414      0.000     88.147      0.000       0.040       0.042
Income                       -1.562e-07   3.44e-07     -0.454      0.650   -8.31e-07    5.19e-07
Outage_sec_per_week             -0.0010      0.003     -0.310      0.756      -0.007       0.005
Yearly_equip_failure            -0.0031      0.015     -0.201      0.841      -0.033       0.027
Multiple                         0.9629      0.021     45.167      0.000       0.921       1.005
Tech_Support                     0.6247      0.020     30.786      0.000       0.585       0.664
Monthly_Charge                  -0.0553      0.000   -210.126      0.000      -0.056      -0.055
Bandwidth_GB_Year                0.0122   4.48e-06   2721.418      0.000       0.012       0.012
Gender_Male                     -0.8285      0.020    -42.149      0.000      -0.867      -0.790
Gender_Nonbinary                 0.2409      0.065      3.687      0.000       0.113       0.369
Marital_Married                  0.0095      0.031      0.308      0.758      -0.051       0.070
Marital_Never Married           -0.0117      0.031     -0.382      0.703      -0.072       0.048
Marital_Separated               -0.0086      0.030     -0.285      0.776      -0.068       0.051
Marital_Widowed                 -0.0084      0.030     -0.279      0.780      -0.068       0.051
Internet_Service_Fiber Optic     6.1391      0.023    269.340      0.000       6.094       6.184
Internet_Service_None            4.3292      0.027    160.172      0.000       4.276       4.382
Contract_One year               -0.0541      0.025     -2.170      0.030      -0.103      -0.005
Contract_Two Year               -0.0597      0.024     -2.524      0.012      -0.106      -0.013
const                           -3.3456      0.067    -50.039      0.000      -3.477      -3.215
==============================================================================
Omnibus:                     1316.576   Durbin-Watson:                   2.005
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2023.011
Skew:                          -0.943   Prob(JB):                         0.00
Kurtosis:                       4.141   Cond. No.                     3.54e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 3.54e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
Out[37]: 0.9702197452097101
```

D2. Justify a statistically based feature selection procedure

Dr. Middleton's presentations regarding feature selection methods in part 1 of her webinar for this course included different wrapper methods applicable to this step in the process. Given the number of variables I was working with, and the complexity presented in this assessment, I felt that the backward stepwise elimination method would be best to reduce the model. Starting with all variables and eliminating variables one by one based on the statistical significance of the p-value improved the model with each step. P-values of less than or equal to 0.05 having no statistical significance meant that they were to be excluded; until only statistically significant variables were present. The success of this method was verified by comparing the standard error for the initial and reduced models. Also discussed by Dr. Middleton, a lower standard error indicates less variance between the model and actual data points, resulting in a better model.

Backward stepwise elimination will remove statistically insignificant variables but does not assess or correct multicollinearity. Before removing variables for there p-value it was vital that I checked for multicollinearity independent of the feature selection method. To check and correct any multicollinearity present the variance inflation factor (VIF) for all variables was measured. (Zach, 2020) Any variables with a VIF value of 5 or greater were removed, starting with the highest value. This process would be repeated until no variables were present with a VIF value of > 5.

In an attempt to further understand the process of the elimination method and regression modeling in general, I did seek outside resources. In my search I found two videos that were a tremendous help in understanding normalization of the data. Both referenced below, one from a Professor Ryan Ahmed, and another by the channel titled "Your Data Teacher." Obtaining the added information regarding normalization of data, and applicable code that allows such normalization allowed me to correct errors I had been seeing as well as have a better understanding of the data overall. Considering the various differences within the data points between the variables (ie: age range: 18-89, tenure range:1-72 months etc.). Normalizing the data to have the minimum value represented by 0 and the maximum value represented by 1 allowed all of the variables to be comparable to one another in a more understandable manner.

D3. Reduced linear regression model and the output for each feature selection procedure

As described above, the initial step to reducing the model was to address high multicollinearity. Variables monthly charge, age and outages per year were removed. All other variables had a VIF value of < 5. This process can be seen here:

```
[49]: # Check for high multicollinearity (VIF > 5) among variables
X = model_df[["Age", "Income" , "Outage_sec_per_week", "Yearly_equip_failure",
              "Multiple", "Tech_Support" , "Monthly_Charge", "Bandwidth_GB_Year" ,
                 "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]]
vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns

vif_df["VIF"] = [variance_inflation_factor(X.values, i)
for i in range(len(X.columns))]

print(vif_df)
```

```
                        Variable        VIF
0                            Age   6.524852
1                         Income   2.857040
2            Outage_sec_per_week   9.503381
3           Yearly_equip_failure   1.383426
4                       Multiple   2.201175
5                   Tech_Support   1.638538
6                 Monthly_Charge  16.390106
7              Bandwidth_GB_Year   3.331884
8                    Gender_Male   1.909368
9               Gender_Nonbinary   1.045101
10               Marital_Married   1.821227
11         Marital_Never Married   1.834162
12            Marital_Separated   1.867104
13               Marital_Widowed   1.865264
14  Internet_Service_Fiber Optic   2.420401
15         Internet_Service_None   1.563068
16             Contract_One year   1.374498
17             Contract_Two Year   1.435734
```

```
[50]: # Remove Variable "Monthly_Charge" due to VIF value, re-check for high multicollinearity
X = model_df[["Age", "Income" , "Outage_sec_per_week", "Yearly_equip_failure",
              "Multiple", "Tech_Support" , "Bandwidth_GB_Year" ,
                 "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]]
vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns

vif_df["VIF"] = [variance_inflation_factor(X.values, i)
for i in range(len(X.columns))]

print(vif_df)
```

```
                        Variable       VIF
0                            Age  6.013390
1                         Income  2.797734
2            Outage_sec_per_week  8.123456
3           Yearly_equip_failure  1.380671
4                       Multiple  1.813058
5                   Tech_Support  1.571616
6              Bandwidth_GB_Year  3.158560
7                    Gender_Male  1.886602
8               Gender_Nonbinary  1.043724
9                Marital_Married  1.785993
10         Marital_Never Married  1.797085
11             Marital_Separated  1.822895
12               Marital_Widowed  1.824696
13  Internet_Service_Fiber Optic  2.158097
14         Internet_Service_None  1.562979
15             Contract_One year  1.366177
16             Contract_Two Year  1.430341
```

```
[53]: # Remove Variable "Outage_sec_per_week" due to VIF value, re-check for high multicollinearity
X = model_df[["Age", "Income" , "Yearly_equip_failure",
                 "Multiple", "Tech_Support" , "Bandwidth_GB_Year" ,
                      "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                    "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" ,
               "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]]
vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns

vif_df["VIF"] = [variance_inflation_factor(X.values, i)
for i in range(len(X.columns))]

print(vif_df)
```

```
                        Variable       VIF
0                            Age  5.247361
1                         Income  2.707416
2           Yearly_equip_failure  1.373020
3                       Multiple  1.783396
4                   Tech_Support  1.560069
5              Bandwidth_GB_Year  2.999005
6                    Gender_Male  1.849183
7               Gender_Nonbinary  1.041896
8                Marital_Married  1.715132
9          Marital_Never Married  1.741388
10             Marital_Separated  1.751765
11               Marital_Widowed  1.762666
12  Internet_Service_Fiber Optic  2.076508
13         Internet_Service_None  1.521913
14             Contract_One year  1.355906
15             Contract_Two Year  1.415629
```

```
[54]: # Remove Variable "Age" due to VIF value, re-check for high multicollinearity
X = model_df[["Income" , "Yearly_equip_failure",
                 "Multiple", "Tech_Support" , "Bandwidth_GB_Year" ,
                      "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                    "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" ,
               "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]]
vif_df = pd.DataFrame()
vif_df["Variable"] = X.columns

vif_df["VIF"] = [variance_inflation_factor(X.values, i)
for i in range(len(X.columns))]

print(vif_df)
```

```
                        Variable       VIF
0                         Income  2.584467
1           Yearly_equip_failure  1.361086
2                       Multiple  1.752980
3                   Tech_Support  1.533119
4              Bandwidth_GB_Year  2.825511
5                    Gender_Male  1.803762
6               Gender_Nonbinary  1.041229
7                Marital_Married  1.636399
8          Marital_Never Married  1.657109
9              Marital_Separated  1.667846
10               Marital_Widowed  1.680086
11  Internet_Service_Fiber Optic  1.975361
12         Internet_Service_None  1.471880
13             Contract_One year  1.344559
14             Contract_Two Year  1.403528
```

Now that there were no variables with high multicollinearity present, the data was ready to be transformed. The following script normalized the data prior to the creation of the regression models.

```
In [40]: # Normalize the data to allow better interpretation of the results
scaler = MinMaxScaler()
norm_df = scaler.fit_transform(model_df)
norm_df = pd.DataFrame(norm_df, columns=model_df.columns)
print (norm_df.head(5))
```

```
        Age    Income  Outage_sec_per_week  Yearly_equip_failure  Multiple  \
0  0.704225  0.109120             0.373283              0.166667       0.0
1  0.126761  0.082599             0.549503              0.166667       1.0
2  0.450704  0.035818             0.504500              0.166667       1.0
3  0.422535  0.071848             0.701563              0.000000       0.0
4  0.915493  0.153646             0.381336              0.166667       0.0

   Tech_Support    Tenure  Monthly_Charge  Bandwidth_GB_Year  Gender_Male  \
0           0.0  0.081624        0.440004           0.106951          1.0
1           0.0  0.002203        0.773861           0.092164          0.0
2           0.0  0.207804        0.380483           0.271180          0.0
3           0.0  0.226580        0.190218           0.286868          1.0
4           1.0  0.009447        0.332905           0.016560          1.0

   Gender_Nonbinary  Marital_Married  Marital_Never Married  \
0               0.0              0.0                    0.0
1               0.0              1.0                    0.0
2               0.0              0.0                    0.0
3               0.0              1.0                    0.0
4               0.0              0.0                    0.0

   Marital_Separated  Marital_Widowed  Internet_Service_Fiber Optic  \
0                0.0              1.0                           1.0
1                0.0              0.0                           1.0
2                0.0              1.0                           0.0
3                0.0              0.0                           0.0
4                1.0              0.0                           1.0

   Internet_Service_None  Contract_One year  Contract_Two Year
0                    0.0                1.0                0.0
1                    0.0                0.0                0.0
2                    0.0                0.0                1.0
3                    0.0                0.0                1.0
4                    0.0                0.0                0.0
```

With the data normalized, the backward stepwise elimination was started. Each variable with a p-value > or equal to 0.05 was removed one by one starting with the highest value. With each elimination a new model was created. There were 11 models created, eliminating the following variables:

- Marital Widowed, p-value: 0.924
- Tech Support, p-value: 0.825
- Marital Married, p-value: 0.755
- Income, p-value: 0.750
- Nonbinary Gender, p-value: 0.688
- Marital Never Married, p-value: 0.551
- Equipment Failures Yearly, p-value: 0.275
- Two-year Contract p-value: 0.255
- Marital Separated, p-value: 0.154
- One-year Contract p-value: 0.045

Each individual model/step can be seen in the screengrabs below.

```
[55…
# Begin backward elimination by checking OLS results for P-values
y = norm_df.Tenure
X = norm_df[["Income" , "Yearly_equip_failure",
              "Multiple", "Tech_Support" , "Bandwidth_GB_Year" ,
                   "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                  "Marital_Separated" , "Marital_Widowed" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model01 = sm.OLS(y, X)
reg_results = model01.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 7.981e+04
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:23:14   Log-Likelihood:                 19663.
No. Observations:               10000   AIC:                        -3.929e+04
Df Residuals:                    9984   BIC:                        -3.918e+04
Df Model:                          15
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Income                          -0.0010      0.003     -0.323      0.747      -0.007       0.005
Yearly_equip_failure             0.0035      0.003      1.083      0.279      -0.003       0.010
Multiple                        -0.0122      0.001    -17.907      0.000      -0.014      -0.011
Tech_Support                     0.0002      0.001      0.220      0.826      -0.001       0.002
Bandwidth_GB_Year                1.1948      0.001   1093.431      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.145      0.000      -0.014      -0.011
Gender_Nonbinary                 0.0009      0.002      0.402      0.688      -0.004       0.005
Marital_Married                  0.0003      0.001      0.317      0.751      -0.002       0.002
Marital_Never Married            0.0007      0.001      0.627      0.531      -0.001       0.003
Marital_Separated               -0.0009      0.001     -0.859      0.390      -0.003       0.001
Marital_Widowed                  0.0001      0.001      0.096      0.924      -0.002       0.002
Internet_Service_Fiber Optic     0.0706      0.001     91.218      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.449      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.262      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.139      0.255      -0.003       0.001
const                           -0.1142      0.001    -87.117      0.000      -0.117      -0.112
==============================================================================
Omnibus:                      220.523   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              221.580
Skew:                          -0.341   Prob(JB):                     7.66e-49
Kurtosis:                       2.740   Cond. No.                         14.7
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[56…
```python
# Check OLS results for additional Backward Elimination after removing "Marital_Widowed" for P-value: 0.924
y = norm_df.Tenure
X = norm_df[["Income" , "Yearly_equip_failure",
                 "Multiple", "Tech_Support" , "Bandwidth_GB_Year" ,
                        "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                      "Marital_Separated" , "Internet_Service_Fiber Optic" ,
              "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model02 = sm.OLS(y, X)
reg_results = model02.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 8.551e+04
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:25:06   Log-Likelihood:                 19663.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9985   BIC:                        -3.919e+04
Df Model:                          14
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Income                          -0.0010      0.003     -0.323      0.747      -0.007       0.005
Yearly_equip_failure             0.0035      0.003      1.084      0.278      -0.003       0.010
Multiple                        -0.0122      0.001    -17.909      0.000      -0.014      -0.011
Tech_Support                     0.0002      0.001      0.221      0.825      -0.001       0.002
Bandwidth_GB_Year                1.1948      0.001   1093.498      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.149      0.000      -0.014      -0.011
Gender_Nonbinary                 0.0009      0.002      0.402      0.688      -0.004       0.005
Marital_Married                  0.0003      0.001      0.309      0.757      -0.002       0.002
Marital_Never Married            0.0006      0.001      0.665      0.506      -0.001       0.002
Marital_Separated               -0.0010      0.001     -1.040      0.298      -0.003       0.001
Internet_Service_Fiber Optic     0.0706      0.001     91.233      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.471      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.261      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.139      0.255      -0.003       0.001
const                           -0.1142      0.001    -94.471      0.000      -0.117      -0.112
==============================================================================
Omnibus:                      220.549   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              221.589
Skew:                          -0.341   Prob(JB):                     7.63e-49
Kurtosis:                       2.740   Cond. No.                         14.6
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[58…
```python
# Check OLS results for additional Backward Elimination after removing Tech_Support for P-value: 0.825
y = norm_df.Tenure
X = norm_df[["Income" , "Yearly_equip_failure",
                 "Multiple" , "Bandwidth_GB_Year" ,
                        "Gender_Male" , "Gender_Nonbinary" , "Marital_Married" , "Marital_Never Married" ,
                      "Marital_Separated" , "Internet_Service_Fiber Optic" ,
              "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model03 = sm.OLS(y, X)
reg_results = model03.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 9.210e+04
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:28:01   Log-Likelihood:                 19663.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9986   BIC:                        -3.920e+04
Df Model:                          13
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Income                          -0.0010      0.003     -0.321      0.748      -0.007       0.005
Yearly_equip_failure             0.0035      0.003      1.085      0.278      -0.003       0.010
Multiple                        -0.0122      0.001    -17.913      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.552      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.154      0.000      -0.014      -0.011
Gender_Nonbinary                 0.0009      0.002      0.403      0.687      -0.004       0.005
Marital_Married                  0.0003      0.001      0.312      0.755      -0.002       0.002
Marital_Never Married            0.0006      0.001      0.665      0.506      -0.001       0.002
Marital_Separated               -0.0010      0.001     -1.041      0.298      -0.003       0.001
Internet_Service_Fiber Optic     0.0706      0.001     91.264      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.476      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.261      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.139      0.255      -0.003       0.001
const                           -0.1141      0.001    -97.011      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      220.501   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              221.529
Skew:                          -0.341   Prob(JB):                     7.86e-49
Kurtosis:                       2.740   Cond. No.                         14.1
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[59…
```python
# Check OLS results for additional Backward Elimination after removing "Marital_Married" for P-value: 0.755
y = norm_df.Tenure
X = norm_df[["Income" , "Yearly_equip_failure",
                 "Multiple" , "Bandwidth_GB_Year" ,
                        "Gender_Male" , "Gender_Nonbinary" , "Marital_Never Married" ,
                      "Marital_Separated" , "Internet_Service_Fiber Optic" ,
              "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model04 = sm.OLS(y, X)
reg_results = model04.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 9.978e+04
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:29:06   Log-Likelihood:                 19663.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9987   BIC:                        -3.921e+04
Df Model:                          12
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Income                          -0.0010      0.003     -0.319      0.750      -0.007       0.005
Yearly_equip_failure             0.0035      0.003      1.083      0.279      -0.003       0.010
Multiple                        -0.0122      0.001    -17.916      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.629      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.155      0.000      -0.014      -0.011
Gender_Nonbinary                 0.0009      0.002      0.402      0.688      -0.004       0.005
Marital_Never Married            0.0005      0.001      0.597      0.551      -0.001       0.002
Marital_Separated               -0.0011      0.001     -1.206      0.228      -0.003       0.001
Internet_Service_Fiber Optic     0.0706      0.001     91.269      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.479      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.262      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.142      0.253      -0.003       0.001
const                           -0.1140      0.001   -100.192      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      220.560   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              221.560
Skew:                          -0.341   Prob(JB):                     7.74e-49
Kurtosis:                       2.739   Cond. No.                         14.0
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[60…
```
# Check OLS results for additional Backward Elimination after removing "Income" for P-value: 0.750
y = norm_df.Tenure
X = norm_df[[ "Yearly_equip_failure",
              "Multiple" , "Bandwidth_GB_Year" ,
                    "Gender_Male" , "Gender_Nonbinary" , "Marital_Never Married" ,
                  "Marital_Separated" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model05 = sm.OLS(y, X)
reg_results = model05.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.089e+05
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:29:57   Log-Likelihood:                 19663.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9988   BIC:                        -3.921e+04
Df Model:                          11
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Yearly_equip_failure             0.0035      0.003      1.081      0.280      -0.003       0.010
Multiple                        -0.0122      0.001    -17.917      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.680      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.153      0.000      -0.014      -0.011
Gender_Nonbinary                 0.0009      0.002      0.401      0.688      -0.004       0.005
Marital_Never Married            0.0005      0.001      0.596      0.551      -0.001       0.002
Marital_Separated               -0.0010      0.001     -1.203      0.229      -0.003       0.001
Internet_Service_Fiber Optic     0.0706      0.001     91.297      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.489      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.263      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.141      0.254      -0.003       0.001
const                           -0.1142      0.001   -111.213      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      220.694   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              221.671
Skew:                          -0.341   Prob(JB):                     7.32e-49
Kurtosis:                       2.739   Cond. No.                         14.0
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[62…
```
# Check OLS results for additional Backward Elimination after removing "Gender_Nonbinary" for P-value: 0.688
y = norm_df.Tenure
X = norm_df[["Yearly_equip_failure", "Multiple" , "Bandwidth_GB_Year" ,
                   "Gender_Male" , "Marital_Never Married" ,
                  "Marital_Separated" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model06 = sm.OLS(y, X)
reg_results = model06.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.198e+05
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:31:34   Log-Likelihood:                 19662.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9989   BIC:                        -3.922e+04
Df Model:                          10
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Yearly_equip_failure             0.0035      0.003      1.092      0.275      -0.003       0.010
Multiple                        -0.0122      0.001    -17.917      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.818      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.410      0.000      -0.014      -0.011
Marital_Never Married            0.0005      0.001      0.596      0.551      -0.001       0.002
Marital_Separated               -0.0011      0.001     -1.204      0.229      -0.003       0.001
Internet_Service_Fiber Optic     0.0706      0.001     91.300      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.493      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.262      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.141      0.254      -0.003       0.001
const                           -0.1141      0.001   -111.729      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      221.090   Durbin-Watson:                   1.967
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              222.144
Skew:                          -0.341   Prob(JB):                     5.78e-49
Kurtosis:                       2.739   Cond. No.                         13.9
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

[63…
```
# Check OLS results for additional Backward Elimination after removing "Marital_Never Married" for P-value: 0.551
y = norm_df.Tenure
X = norm_df[["Yearly_equip_failure", "Multiple" , "Bandwidth_GB_Year" , "Gender_Male" ,
                  "Marital_Separated" , "Internet_Service_Fiber Optic" ,
            "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model07 = sm.OLS(y, X)
reg_results = model07.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.331e+05
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:32:40   Log-Likelihood:                 19662.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9990   BIC:                        -3.923e+04
Df Model:                           9
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Yearly_equip_failure             0.0035      0.003      1.092      0.275      -0.003       0.010
Multiple                        -0.0122      0.001    -17.911      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.858      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.419      0.000      -0.014      -0.011
Marital_Separated               -0.0012      0.001     -1.395      0.163      -0.003       0.000
Internet_Service_Fiber Optic     0.0706      0.001     91.314      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.493      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.258      0.024      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.133      0.257      -0.003       0.001
const                           -0.1140      0.001   -114.004      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      221.171   Durbin-Watson:                   1.968
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              222.314
Skew:                          -0.341   Prob(JB):                     5.31e-49
Kurtosis:                       2.740   Cond. No.                         13.8
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
# Check OLS results for additional Backward Elimination after removing "Yearly_equip_failure" for P-value: 0.275
y = norm_df.Tenure
X = norm_df[[ "Multiple" , "Bandwidth_GB_Year" , "Gender_Male" ,
                 "Marital_Separated" , "Internet_Service_Fiber Optic" ,
             "Internet_Service_None" , "Contract_One year" , "Contract_Two Year"]].assign(const=1)

model08 = sm.OLS(y, X)
reg_results = model08.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.497e+05
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:34:20   Log-Likelihood:                 19662.
No. Observations:               10000   AIC:                        -3.931e+04
Df Residuals:                    9991   BIC:                        -3.924e+04
Df Model:                           8
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Multiple                        -0.0122      0.001    -17.909      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1949      0.001   1093.934      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.418      0.000      -0.014      -0.011
Marital_Separated               -0.0012      0.001     -1.415      0.157      -0.003       0.000
Internet_Service_Fiber Optic     0.0706      0.001     91.309      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.489      0.000       0.069       0.073
Contract_One year               -0.0020      0.001     -2.247      0.025      -0.004      -0.000
Contract_Two Year               -0.0009      0.001     -1.138      0.255      -0.003       0.001
const                           -0.1138      0.001   -116.358      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      221.291   Durbin-Watson:                   1.968
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              222.330
Skew:                          -0.341   Prob(JB):                     5.27e-49
Kurtosis:                       2.739   Cond. No.                         6.07
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
# Check OLS results for additional Backward Elimination after removing "Contract_Two Year" for P-value: 0.255
y = norm_df.Tenure
X = norm_df[["Multiple" , "Bandwidth_GB_Year" , "Gender_Male" ,
                 "Marital_Separated" , "Internet_Service_Fiber Optic" ,
             "Internet_Service_None" , "Contract_One year"]].assign(const=1)

model09 = sm.OLS(y, X)
reg_results = model09.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.711e+05
Date:                Sun, 20 Aug 2023   Prob (F-statistic):               0.00
Time:                        22:35:36   Log-Likelihood:                 19661.
No. Observations:               10000   AIC:                        -3.931e+04
Df Residuals:                    9992   BIC:                        -3.925e+04
Df Model:                           7
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Multiple                        -0.0122      0.001    -17.926      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1094.110      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.431      0.000      -0.014      -0.011
Marital_Separated               -0.0012      0.001     -1.425      0.154      -0.003       0.000
Internet_Service_Fiber Optic     0.0706      0.001     91.306      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.495      0.000       0.069       0.073
Contract_One year               -0.0017      0.001     -2.002      0.045      -0.003   -3.46e-05
const                           -0.1140      0.001   -120.196      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      221.891   Durbin-Watson:                   1.968
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              223.100
Skew:                          -0.342   Prob(JB):                     3.59e-49
Kurtosis:                       2.740   Cond. No.                         5.92
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
# Check OLS results for additional Backward Elimination after removing "Marital_Separated" for P-value: 0.154
y = norm_df.Tenure
X = norm_df[["Multiple" , "Bandwidth_GB_Year" , "Gender_Male" , "Internet_Service_Fiber Optic" ,
             "Internet_Service_None" , "Contract_One year"]].assign(const=1)

model10 = sm.OLS(y, X)
reg_results = model10.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 1.996e+05
Date:                Mon, 21 Aug 2023   Prob (F-statistic):               0.00
Time:                        20:23:56   Log-Likelihood:                 19660.
No. Observations:               10000   AIC:                        -3.931e+04
Df Residuals:                    9993   BIC:                        -3.926e+04
Df Model:                           6
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Multiple                        -0.0122      0.001    -17.924      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1094.054      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.434      0.000      -0.014      -0.011
Internet_Service_Fiber Optic     0.0706      0.001     91.292      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.486      0.000       0.069       0.073
Contract_One year               -0.0017      0.001     -2.001      0.045      -0.003   -3.37e-05
const                           -0.1143      0.001   -122.288      0.000      -0.116      -0.112
==============================================================================
Omnibus:                      221.952   Durbin-Watson:                   1.968
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              223.314
Skew:                          -0.342   Prob(JB):                     3.22e-49
Kurtosis:                       2.741   Cond. No.                         5.84
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```
[77… # Check OLS results for additional Backward Elimination after removing "Contract_One year" for P-value: 0.045
y = norm_df.Tenure
X = norm_df[["Multiple" , "Bandwidth_GB_Year" , "Gender_Male" , "Internet_Service_Fiber Optic" ,
             "Internet_Service_None"]].assign(const=1)

model11 = sm.OLS(y, X)
reg_results = model11.fit()
print(reg_results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 2.394e+05
Date:                Mon, 21 Aug 2023   Prob (F-statistic):               0.00
Time:                        20:24:28   Log-Likelihood:                 19658.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9994   BIC:                        -3.926e+04
Df Model:                           5
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Multiple                        -0.0122      0.001    -17.955      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.908      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.408      0.000      -0.014      -0.011
Internet_Service_Fiber Optic     0.0706      0.001     91.301      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.461      0.000       0.069       0.073
const                           -0.1146      0.001   -124.926      0.000      -0.116      -0.113
==============================================================================
Omnibus:                      222.161   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              223.760
Skew:                          -0.343   Prob(JB):                     2.58e-49
Kurtosis:                       2.742   Cond. No.                         5.74
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

E1. Explain your data analysis process

By utilizing 18 columns for the initial model, there is certainly potential to have high multicollinearity and/or statistical insignificance among the variables. This is evident in the initial model when looking at the variation inflation factors and p-values. Once the variables with high multicollinearity and insignificant statistical p-values were removed, the reduced model resulted in 5 columns. Using the OLS regression results from both the initial and reduced model many values were compared for the analysis.

The coefficient value informs the analyst as to how much the dependent variable tenure would be affected with a 1-unit change for each independent variable. The higher the R^2 value, the better regression. The probability of the f-statistic also identifies statistical significance. Given this information, these values and the p-values for each individual variable were used to compare the models. (Zach, 2019) In addition, scripts to obtain the standard residual error for both models was used to review the values to determine which model was best. (Malekpour, 2021) Finally, regression plots were created to verify the assumptions of multiple linear regression were present.

E2. Provide the output and *all* calculations of the analysis you performed

Following the feature selection methods discussed above, the reduced model was created. While analyzing the initial and reduced models, the residual standard error was compared. All outputs related to the regression model and the residual standard error can be found in the screengrab below. Within the image, one can see the 5 columns that remained, which are:

- Multiple
- Bandwidth per year
- Gender_Male
- Internet Service: Fiber Optics
- Internet Service: None

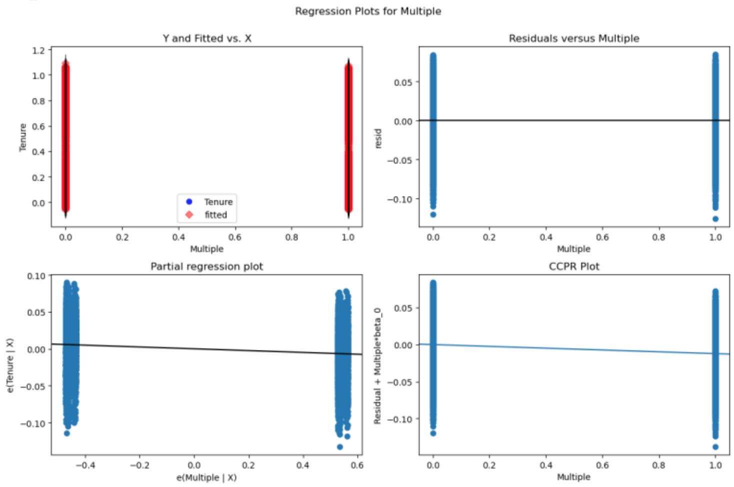
```
[78… # Create the Reduced model
y = norm_df.Tenure
X = norm_df[["Multiple" , "Bandwidth_GB_Year" , "Gender_Male" , "Internet_Service_Fiber Optic" ,
             "Internet_Service_None"]].assign(const=1)

red_model = sm.OLS(y, X)
reg_results = red_model.fit()
print(reg_results.summary())

# Retrieve residual standard error for reduced model
reg_results.resid.std(ddof=X.shape[1])
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.992
Method:                 Least Squares   F-statistic:                 2.394e+05
Date:                Mon, 21 Aug 2023   Prob (F-statistic):               0.00
Time:                        20:24:39   Log-Likelihood:                 19658.
No. Observations:               10000   AIC:                        -3.930e+04
Df Residuals:                    9994   BIC:                        -3.926e+04
Df Model:                           5
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Multiple                        -0.0122      0.001    -17.955      0.000      -0.014      -0.011
Bandwidth_GB_Year                1.1948      0.001   1093.908      0.000       1.193       1.197
Gender_Male                     -0.0125      0.001    -18.408      0.000      -0.014      -0.011
Internet_Service_Fiber Optic     0.0706      0.001     91.301      0.000       0.069       0.072
Internet_Service_None            0.0707      0.001     75.461      0.000       0.069       0.073
const                           -0.1146      0.001   -124.926      0.000      -0.116      -0.113
==============================================================================
Omnibus:                      222.161   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              223.760
Skew:                          -0.343   Prob(JB):                     2.58e-49
Kurtosis:                       2.742   Cond. No.                         5.74
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[78… 0.033896724786738164
```

Statology had an abundance of helpful information for many of the steps involved in this assessment. This extends to creating and evaluating a residual plot. Using the code obtained from Statology regarding residual plots, I created the following visuals for each independent variable. (Zach, 2020)

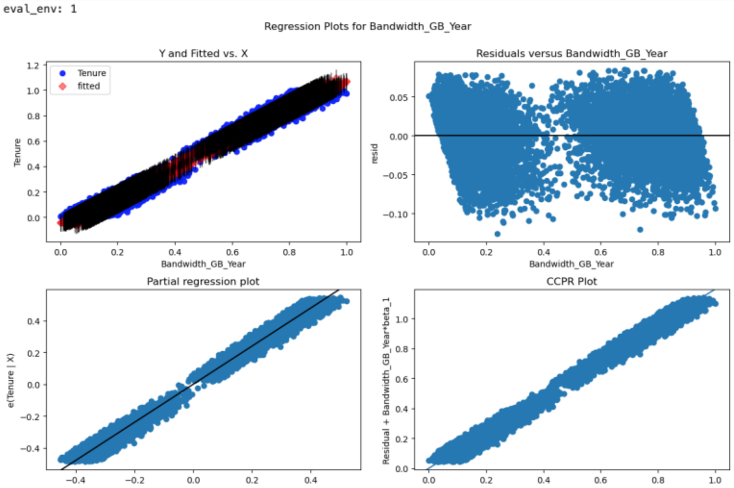
[84…
```
# Define figure size
fig = plt.figure(figsize = [12,8])

# Create Regression Plot for Multiple
sm.graphics.plot_regress_exog(reg_results, 'Multiple', fig=fig);
```

eval_env: 1



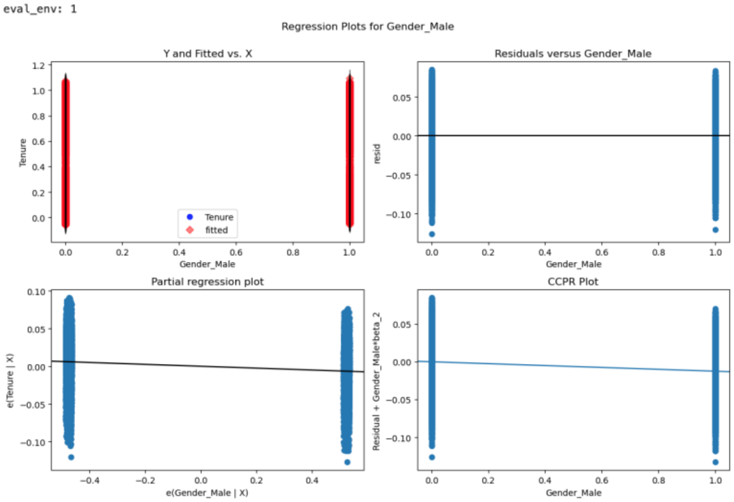
[85…
```
# Define figure size
fig = plt.figure(figsize = [12,8])

# Create Regression Plot for Bandwidth_GB_Year
sm.graphics.plot_regress_exog(reg_results, 'Bandwidth_GB_Year', fig=fig);
```

eval_env: 1



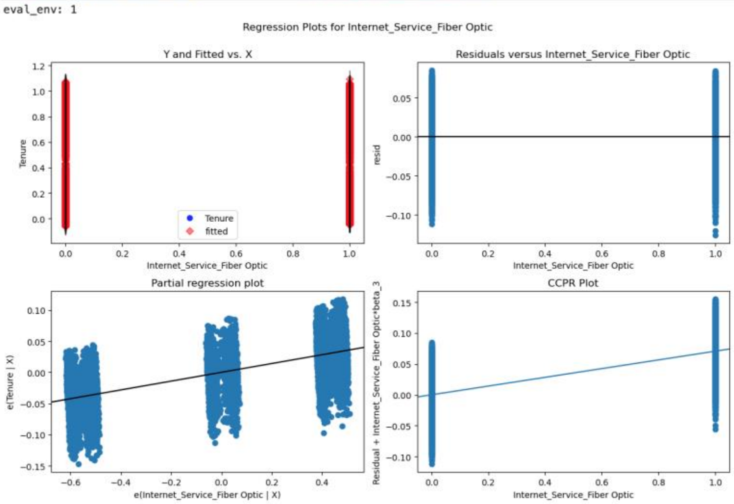
[86…
```python
# Define figure size
fig = plt.figure(figsize = [12,8])

# Create Regression Plot for Gender_Male
sm.graphics.plot_regress_exog(reg_results, 'Gender_Male', fig=fig);
```

eval_env: 1

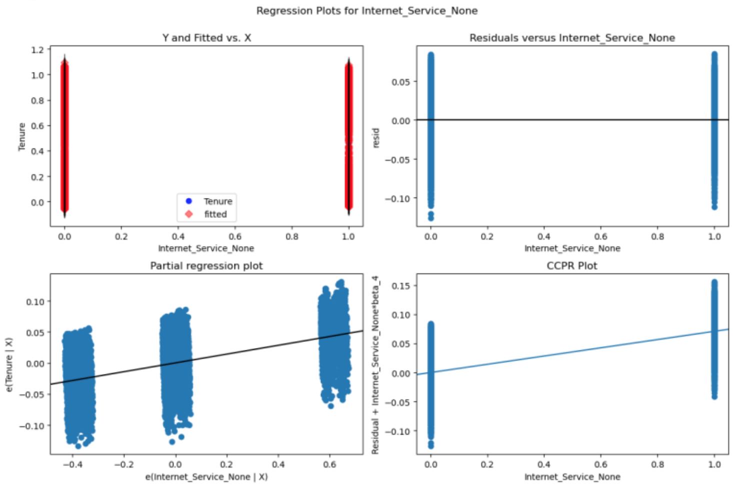


[87…
```python
# Define figure size
fig = plt.figure(figsize = [12,8])

# Create Regression Plot for Internet_Service_Fiber Optic
sm.graphics.plot_regress_exog(reg_results, 'Internet_Service_Fiber Optic', fig=fig);
```

eval_env: 1



```
[88… # Define figure size
fig = plt.figure(figsize = [12,8])

# Create Regression Plot for Internet_Service_None
sm.graphics.plot_regress_exog(reg_results, 'Internet_Service_None', fig=fig);
eval_env: 1
```



E3. Provide an executable error-free copy of the code used

The executable script file associated with this analysis is attached to the submission of this assessment.

Part V: Data Summary and Implications

F1. Discuss the results

The regression equation for the reduced model:

Y^ = -0.1146 - 0.0122 (Multiple) + 1.1948 (Bandwidth) - 0.0125 (Male) + 0.0706 (Optic internet service) + 0.0707 (No internet service)

For the reduced dataset the coefficients for the remaining columns and their significance is as follows:

- All other factors held constant, customers with multiple lines stay with the business 0.0122 months less than customers not having multiple lines
- All other factors held constant; a one unit increase in a customer's Bandwidth in GB per year is associated with a 0.0121 increase in months of tenure
    - Bandwidth coefficient in the reduced model has a value of 1.1948 due to normalization. Scaling this value back results in a coefficient value of 0.0121
- All other factors held constant, customers that identify as male stay with the business 0.0125 months less than customers that do not identify as male
- All other factors held constant; customers that utilize the optic internet service type stay with the business 0.0706 months more than customers that do not have optic

- All other factors held constant; customers that have no internet service stay with the business 0.0707 months more than customers that have internet service

The p-values for all columns were equal to 0.00. The probability of the f-statistic is also 0.00. These values indicate that each variable and the model overall have statistical significance and that it is not by chance. The R^2 surprisingly decreased in the residual model, from 0.999 to 0.992. A higher R^2 value results in a better regression, so this was a bit concerning. Comparing the residual standard error values brought back confidence in my residual model, however.

The initial model has a value of 0.97 for the residual standard error, and the residual model has a value of 0.03. As discussed previously, Dr. Middleton in webinar 1 mentioned that a smaller residual standard error indicates a better model because there is less variance between the model and the true data points. Given the analysis of the results from each regression model, it was evident that I had statistically significant variables remaining, and that there was a relationship present between each independent variable and the dependent variable.

To further analyze these relationships, the regression plots were interpreted using knowledge I gained from Statology. To verify the accuracy of the regression plots in comparison to the assumptions for multiple linear regression, I took a look at the residuals. Unfortunately, the majority of the variables and their residuals plots did not show homoscedasticity. For multiple, gender_male, Optic Internet Service and no internet service the residuals are clearly plotted along the y axis at various points. The residuals are not centered around the line of best fit, which is 0 for all plots. The only plot with a different distribution is that of the bandwidth variable. The residuals bandwidth plot has data points that seem distributed fairly symmetrically and most of the data points are closer to the line of best fit in comparison to the other plots. However, there does seem to be a pattern, and given the information for all the residual plots in this model, I cannot be sure that the assumptions for multiple linear regression are satisfied. (Zach, 2020)

Given the model outputs, I believe that statistical significance is present in the residual model. There does not seem to be practical significance though. Given the residuals plots and the unsatisfied assumptions for multiple linear regression, it does not make much sense to make business decisions based off this model. In addition, multiple lines, bandwidth usage, and internet service type may be relevant variables to inform the question, but male gender is not. In comparison to other gender expressions and their rates of tenure, there is relevancy for data reporting purposes. That is not the case in this model however, instead male gender is used among variables related to service. The business could market changes in relation to deals or promotions for all of the residual variables, except for gender_male. A customer's gender expression is out of the control of the business, and even if they did market special deals for male identifying individuals, that would be discrimination and a large issue for the company and its brand.

Another problem with the model is that it has two variables pertaining to internet service that have similar statistical outputs. The p-value for optic internet service is 0.0706, and no internet service has a p-value of 0.707. Marketing for one of these would contradict the relevance of marketing for the other. Having more information regarding phone service in comparison to no internet service would be more appropriate prior to making changes related to the statistical significance of no internet service in this case.

Needing more information before decision making leads me to the limitations of the analysis. Overall, there is not enough data to move forward with utilizing this reduced model to inform the question. Only 10,000 customers (about the seating capacity of Cameron basketball stadium at Duke University) were accounted for within a time period of one month to just under 6 years. Having more customer data over a longer period of time could be much more beneficial to the question, especially considering that it pertains to time spent as a customer of the company.

F2. Recommended course of action

Given the absense of reliability within the residual model, it would be recommended that the business does not move forward with any business decisions based off these statistics. Instead, I would recommend reassessing the question with a larger dataset. In addition, I would hone in on data related to customer tenure of 1 year and beyond, preferably up to 10 years. Instead of analyzing customer demographics such as gender or marital status, I would assess only variables in which the company does have some control over. Transforming and reviewing data that is more specific to the research question and various service aspects would be more beneficial for the company to apply changes where need be in regards to business practices.

If a larger dataset is not available for analysis, then I would recommend reassessing the scope of the analysis. Potentially changing the research question and/or independent variables could allow for a more reliable model. Sometimes the exact question at hand may not be a viable inquiry to assess. Making decisions based off an unreliable model could be catestrophic for the business. Rephraming the question and viewing the data from another point of view could provide an accurate predictive model that would inevitably be of more benefit to the company as compared to the residual model available in this assessment.

Part VI: Demonstration

G. Panopto Video

The Panopto video recorded for this assessment can be found in the corresponding folder for this course.

H. List the web sources used to acquire data or segments of third-party code

Churchill, Briana. (2023, July 14). *Performance Assessment: Exploratory Data Analysis (OEM2).* Assignment for MS Data Analytics Course D207. Western Governors University.

Ahmed, R. (2022, April 24). *Normalization Vs. Standardization (Feature Scaling in Machine Learning) [Video].* Youtube. <https://youtu.be/bqhQ2LWBheQ>

Eubank, N. (2022). *Using and Interpreting Indicator (Dummy) Variables.* Unifying Data Science. <https://www.unifyingdatascience.org/html/interpreting_indicator_vars.html>

Malekpour, M. (2021, Sept. 30). *Residual standard error of a regression in python.* Stackoverflow. <https://stackoverflow.com/questions/63333999/residual-standard-error-of-a-regression-in-python>

YourDataTeacher. (2020, Sep. 16). *Normalization and Standardization in Python*. [Video]. Youtube. <https://youtu.be/RyUQT7SqmyI>

Zach. (2022, April 1). *How to Get Regression Model Summary from Scikit-Learn.* Statology.

<https://www.statology.org/sklearn-linear-regression-summary/>

Zach. (2020, July 20). *How to Calculate VIF in Python*. Statology. <https://www.statology.org/how-to-calculate-vif-in-python/>

Zach. (2020, July 21). *How to Create a Residual Plot in Python.* Statology. <https://www.statology.org/residual-plot-python/>

I. Acknowledge sources

Middleton, K. (2023, July 13). "*Getting Started With D208*" Part I. Western Governors University. Pages 22, 35-38.

Straw, E. (2023). *Tips for Success* [Unpublished document]. Western Governors University.

Zach. (2021, Nov. 16). *The Five Assumptions of Multiple Linear Regression.* Statology. <https://www.statology.org/multiple-linear-regression-assumptions/>

Zach. (2020, Dec. 7). *What Are Residuals in Statistics.* Statology. <https://www.statology.org/residuals/>

Zach. (2019, Mar. 20). *How to Read and Interpret a Regression Table*. Statology. <https://www.statology.org/read-interpret-regression-table/