PREDICTIVE MODELING — D208
Task 1
Eric D. Otten
Student ID: 011399183

## A1. Research Question

My research question is: “Which customer factors contribute most to a customer’s tenure with the service provider?” This question is relevant to an organization because decision-makers use customer behavioral patterns to increase customer retention.

## A1. Goals

To understand which customer characteristics influence how long they stay with a service provider, it's crucial to start with cleaning the data. This step ensures that the data is accurate and reliable by removing any errors or inconsistencies, such as missing values or different formats. This is necessary to make sure the analysis is based on good data.

After the data has been cleaned, the next step, building a linear regression model, will help to identify and measure how different factors like customer age, how often they use the service, their level of satisfaction, and their payment methods affect their tenure with the provider. This model will show the relationships between these variables and customer tenure, thus allowing companies to see which aspects are most important with regards to keeping customers as long as possible. This information is vital for making informed business decisions that aim to improve customer retention and increase their lifetime value (LTV.)

Once the linear regression model is set up, it will provide coefficients of each factor. These coefficients indicate the strength and direction of the impact each factor has on customer tenure. For instance, a positive coefficient for service usage might indicate that customers who use the service more frequently are likely to stay longer. This type of insight can help the company focus on encouraging more frequent use through targeted promotions or improved service offerings.

Additionally, examining the statistical significance of each coefficient will allow the organization to distinguish between genuinely impactful factors and those that might not make a substantial difference. By focusing on the most significant predictors of customer tenure, the company can allocate resources more effectively and implement strategic changes that directly address the factors that encourage customers to remain with the service longer.

Finally, the model's overall fit and predictive accuracy will be evaluated to ensure it can reliably inform business decisions. This involves assessing metrics such as R-squared and p-values, which help determine how well the model explains customer behavior and whether the results are statistically significant. With a well-constructed model, the company can make informed decisions to enhance customer satisfaction and retention, ultimately leading to increased profitability.

## B1. Assumptions of a Linear Regression Model

The four assumptions of a linear regression model are as follows:

1. **Linear relationship**: Linear regression models assume a linear relationship between the independent and dependent variables.This means that the change in the dependent variable is proportional to the change in the independent variables.
2. **Independence**: Linear regression models assume there is no correlation between residuals (errors) in time series data. This means that the value of the error term for one observation is not correlated with the value of the error term for any other observation
3. **Homoscedasticity:** Linear regression models assume that the variance of the residuals (errors) should be constant across all levels of the independent variables.
4. **Normality of Residuals:** Linear regression models assume that residuals (errors) are normally distributed, especially for hypothesis testing and constructing confidence intervals. Normality can be assessed using various diagnostic plots or statistical tests.

## B2. Benefits of Using Python

I will be using Python for this analysis due to its flexibility and simplicity. According to Dekkati (2021), "Python is a high-level programming language that is simple and easy to learn, free to use and open source, platform-independent, portable, dynamically typed, procedure-oriented and object-oriented, interpreted, extendable, embedded, and has an extensive library."

Additionally, I will import the following Python libraries:

- pandas, which allows for handling large datasets and importing .csv files.
- numpy which allows for mathematical operations on the dataset.
- scikit-learn/sklearn for machine learning, linear regression, and model evaluation.
- matplotlib for graphing functionality.
- statsmodels which allows for statistical modeling, including regression analysis.
- seaborn which allows for informative statistical graphics.

## B3. Rationale for Multiple Linear Regression

Multiple linear regression will be used to answer my research question for several reasons.

Multiple linear regression allows one to analyze the relationship between multiple independent variables (customer factors) and a single dependent variable (customer tenure.) This helps in identifying which factors have the most significant impact on customer tenure. "We analyze residuals to see if there are any discernible patterns in those residuals when they are arranged in order according to the corresponding values of any of the independent variables," Wheeler (2013) clarifies.

Additionally, multiple linear regression allows one to control for confounding variables by including multiple independent variables. This means an analyst can isolate the effect of each factor on customer tenure, providing a clearer understanding of what truly influences customer retention.

The appropriateness of using multiple linear regression also heavily depends on the nature of the dependent variable, in this case, customer tenure. For multiple linear regression to be suitable, the dependent variable should ideally be continuous. This means it can take on any value within a range, such as the number of months or years a customer has been with the service provider. This continuity allows for more precise measurements and interpretations of changes in the dependent variable due to shifts in the independent variables.

If the dependent variable were categorical, such as whether a customer is still with the provider (yes or no), other types of models, like logistic regression, would be more appropriate. However, since customer tenure involves continuous data, multiple linear regression is applicable and can effectively handle this type of analysis.

Moreover, the linear relationship assumption in multiple regression requires that changes in the independent variables lead to proportional changes in the dependent variable. This is essential for the model to provide accurate predictions and meaningful insights. Before proceeding with the regression analysis, it's crucial to verify this assumption by plotting each of the independent variables against the dependent variable to check for linearity. If the relationships appear linear, it further confirms the suitability of using multiple linear regression for analyzing how customer factors influence tenure. This step helps ensure that the model will be both robust and relevant to making informed business decisions based on the data analysis.

### C1. Data Cleaning Goals and Steps

My goal with regards to cleaning the sample data is to create a uniform DataFrame to which multiple linear regression can be applied and from which useful business insights can be drawn.

The provided dataset is cleaned yet contains several data anomalies which should be corrected:

- Zip codes are provided in int28 format instead of as a string and as a result have lost their leading zeroes. These rows will be converted to strings.
- Time zone categories are redundant. For example, separate time zones exist for EST: New York, Detroit, and others. These categories will be reduced to the standard US time zones. The following mappings will be used:
    - America/New_York will be mapped to EST.
    - America/Detroit will be mapped to EST.
    - America/Indiana/Indianapolis will be mapped to EST.
    - America/Kentucky/Louisville will be mapped to EST.
    - America/Indiana/Vincennes will be mapped to EST.
    - America/Indiana/Tell_City will be mapped to EST.
    - America/Indiana/Petersburg will be mapped to EST.
    - America/Indiana/Knox will be mapped to EST.
    - America/Indiana/Winamac will be mapped to EST.
    - America/Indiana/Marengo will be mapped to EST.
    - America/Toronto will be mapped to EST.
    - America/Chicago will be mapped to CST.

- America/Menominee will be mapped to CST.
- America/North_Dakota/New_Salem will be mapped to CST.
- America/Denver will be mapped to MST.
- America/Phoenix will be mapped to MST.
- America/Boise will be mapped to MST.
- America/Los_Angeles will be mapped to PST.
- America/Anchorage will be mapped to AKST.
- America/Nome will be mapped to AKST.
- America/Sitka will be mapped to AKST.
- America/Juneau will be mapped to AKST.
- Pacific/Honolulu will be mapped to HAST.
- America/Puerto_Rico will be mapped to AST.
- America/Ojinaga will be mapped to MST.

For nominal categorical data, one hot encoding will be used as it is the most widespread approach. This approach involves creating a new column for each category, which contains a binary encoding of 0 or 1 to denote whether a particular row belongs to this category. This can be achieved using the `get_dummies()` method within the `pandas` library.

### C2:  Dependent and All Independent Variables Summary Statistics

My research question is “Which customer factors contribute most to a customer’s tenure with the service provider?” Therefore, the dependent variable in my analysis is customer tenure. The remaining customer factors are independent variables. To generate summary statistics, the `value_counts()` method will be run on all categorical data and the `describe()` method will be run on all numerical data. An overview of the variables used to answer my research question is as follows:

- **Population**: Population within a mile radius of the customer, based on census data.
- **Area**: Classification of the customer's area (rural, urban, suburban).
- **TimeZone**: Time zone of the customer's residence.
- **Children**: Number of children in the customer’s household as reported during sign-up.
- **Age**: Age of the customer as reported during sign-up.
- **Income**: Annual income of the customer as reported at the time of sign-up.
- **Marital**: Marital status of the customer.
- **Gender**: Gender of the customer as they self-identify.
- **Outage_sec_perweek**: Average number of seconds per week of system outages experienced by the customer.
- **Email**: Number of emails sent to the customer in the last year.
- **Contacts**: Number of times the customer contacted technical support.
- **Yearly_equip_failure**: Number of times the customer’s equipment failed and needed replacement or reset in the past year.
- **Techie**: Indicates whether the customer considers themselves technically inclined.
- **Contract**: The type of service contract the customer has (month-to-month, one year, two years).
- **Port_modem**: Whether the customer uses a portable modem.
- **Tablet**: Whether the customer owns a tablet device.
- **InternetService**: Type of internet service the customer has (DSL, fiber optic, none).
- **Phone**: Whether the customer has a phone service.
- **Multiple**: Whether the customer has multiple lines.
- **OnlineSecurity**: Whether the customer has an online security service.
- **OnlineBackup**: Whether the customer uses an online backup service.
- **DeviceProtection**: Whether the customer uses a device protection service.
- **TechSupport**: Whether the customer has technical support service.
- **StreamingTV**: Whether the customer uses streaming TV service.
- **StreamingMovies**: Whether the customer uses streaming movies service.
- **PaperlessBilling**: Whether the customer has opted for paperless billing.
- **PaymentMethod**: Method by which the customer makes payments (e.g., electronic check, mailed check, bank transfer, credit card).
- **Tenure (dependent variable)**: Number of months the customer has been with the service provider.
- **MonthlyCharge**: Average monthly charge billed to the customer.
- **Bandwidth_GB_Year**: Average amount of data in GB used by the customer per year.

The following code generates summary statistics for each column in the dataframe:

| **Code** |
|---|
| (code below) |

```
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import missingno as msno
import numpy as np
import pandas as pd
import seaborn as sns
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor

# Read the CSV file
filename = "churn_clean.csv"
df = pd.read_csv(filename, keep_default_na = False, na_values=["NA"])

# Temporarily include 'CaseOrder' and 'Zip' as categorical variables
categorical_cols = df.select_dtypes(include=["object", "bool"]).copy()
categorical_cols['CaseOrder'] = df['CaseOrder'].astype('category')
categorical_cols['Zip'] = df['Zip'].astype('category')

# Update numerical columns to exclude 'CaseOrder' and 'Zip' if they were included
numerical_cols = df.select_dtypes(include=["int64", "float64"]).drop(columns=['CaseOrder',
'Zip'], errors='ignore')

# Display the results
print("Categorical Data Summary:")
for key, value in categorical_summary.items():
    print(f"\nColumn: {key}\n{value}")

print("\nNumerical Data Summary:")
for key, value in numerical_summary.items():
    print(f"\nColumn: {key}\n{value}")
```

**Result**

```
Categorical Data Summary:

Column: Customer_id
Customer_id
K409198    1
X300173    1
M155745    1
G126132    1
O148559    1
          ..
F454437    1
W845098    1
P854487    1
K983374    1
T38070     1
Name: count, Length: 10000, dtype: int64

Column: Interaction
Interaction
aa90260b-4141-4a24-8e36-b04ce1f4f77b    1
26769b47-8eda-4e14-9baf-7348b64b7da3    1
6d65ca83-1001-4d01-a3f9-c3ae5ac33a83    1
448944cf-10f6-4a04-a8e0-4079b6791e26    1
a9890702-06c6-4337-9d5b-65f7d1e30466    1
                                       ..
c650b63b-2d68-48f2-911d-6e8c838c8185    1
3006986f-69e9-4c80-8dcb-1f8d917f2071    1
0e3b8690-177a-4bce-a4e9-823682ce8aec    1
25400298-b615-407d-9e79-25fb89b38429    1
9de5fb6e-bd33-4995-aec8-f01d0172a499    1
Name: count, Length: 10000, dtype: int64

Column: UID
UID
e885b299883d4f9fb18e39c75155d990    1
7df0305ba8ef7f90baf1c5ab300cb5b3    1
4e112d99a62f69c893048b4ffd9af8f3    1
d4c79435b769c307ec763e59af26271b    1
c2ed5dd33d623ad8490e7f819b400c98    1
                                   ..
c409ba0987500ec59628290e98d38437    1
97a1d34a85577c5c60fd040bbe34a3a7    1
899ed4f66698422742afd91512db2e89    1
369e8e3c53e8275a8e668739ddd84572    1
0ea683a03a3cd544aefe8388aab16176    1
Name: count, Length: 10000, dtype: int64

Column: City
City
Houston         34
New York        24
Springfield     23
Buffalo         23
San Antonio     22
                ..
Cottontown       1
San Dimas        1
Fort Hill        1
Webster          1
Clarkesville     1
```

```
Name: count, Length: 6058, dtype: int64

Column: State
State
TX    603
NY    558
PA    550
CA    526
IL    413
OH    359
FL    324
MO    310
VA    285
NC    280
IA    279
MI    279
MN    264
WV    247
IN    241
GA    238
KY    238
WI    228
OK    203
KS    195
NJ    190
TN    185
AL    181
NE    181
AR    176
WA    175
MA    172
CO    155
LA    141
MS    126
SC    124
MD    123
ND    118
NM    114
OR    114
AZ    112
ME    112
SD    101
MT     96
NH     85
VT     84
ID     81
AK     77
CT     71
UT     66
NV     48
WY     43
PR     40
HI     35
DE     21
RI     19
DC     14
Name: count, dtype: int64

Column: County
County
```

```
Washington     111
Jefferson      100
Montgomery      99
Franklin        92
Los Angeles     91
              ...
Rooks            1
Cochise          1
Yauco            1
Hoke             1
Briscoe          1
Name: count, Length: 1620, dtype: int64

Column: Area
Area
Suburban    3346
Urban       3327
Rural       3327
Name: count, dtype: int64

Column: TimeZone
TimeZone
America/New_York                  4072
America/Chicago                   3672
America/Los_Angeles                887
America/Denver                     552
America/Detroit                    265
America/Indiana/Indianapolis       186
America/Phoenix                    104
America/Boise                       57
America/Anchorage                   55
America/Puerto_Rico                 40
Pacific/Honolulu                    35
America/Menominee                   16
America/Nome                        12
America/Kentucky/Louisville         10
America/Sitka                        8
America/Indiana/Vincennes            6
America/Indiana/Tell_City            6
America/Toronto                      5
America/Indiana/Petersburg           4
America/Juneau                       2
America/North_Dakota/New_Salem       2
America/Indiana/Knox                 1
America/Indiana/Winamac              1
America/Indiana/Marengo              1
America/Ojinaga                      1
Name: count, dtype: int64

Column: Job
Job
Occupational psychologist               30
Comptroller                             28
Hospital pharmacist                     28
Horticultural therapist                 28
Ranger/warden                           27
                                        ..
Control and instrumentation engineer     6
Travel agency manager                    6
Accountant, chartered certified          6
```

```
Arboriculturist                           6
Toxicologist                              6
Name: count, Length: 639, dtype: int64

Column: Marital
Marital
Divorced         2092
Widowed          2027
Separated        2014
Never Married    1956
Married          1911
Name: count, dtype: int64

Column: Gender
Gender
Female       5025
Male         4744
Nonbinary     231
Name: count, dtype: int64

Column: Churn
Churn
No     7350
Yes    2650
Name: count, dtype: int64

Column: Techie
Techie
No     8321
Yes    1679
Name: count, dtype: int64

Column: Contract
Contract
Month-to-month    5456
Two Year          2442
One year          2102
Name: count, dtype: int64

Column: Port_modem
Port_modem
No     5166
Yes    4834
Name: count, dtype: int64

Column: Tablet
Tablet
No     7009
Yes    2991
Name: count, dtype: int64

Column: InternetService
InternetService
Fiber Optic    4408
DSL            3463
None           2129
Name: count, dtype: int64

Column: Phone
Phone
```

```
Yes    9067
No      933
Name: count, dtype: int64

Column: Multiple
Multiple
No     5392
Yes    4608
Name: count, dtype: int64

Column: OnlineSecurity
OnlineSecurity
No     6424
Yes    3576
Name: count, dtype: int64

Column: OnlineBackup
OnlineBackup
No     5494
Yes    4506
Name: count, dtype: int64

Column: DeviceProtection
DeviceProtection
No     5614
Yes    4386
Name: count, dtype: int64

Column: TechSupport
TechSupport
No     6250
Yes    3750
Name: count, dtype: int64

Column: StreamingTV
StreamingTV
No     5071
Yes    4929
Name: count, dtype: int64

Column: StreamingMovies
StreamingMovies
No     5110
Yes    4890
Name: count, dtype: int64

Column: PaperlessBilling
PaperlessBilling
Yes    5882
No     4118
Name: count, dtype: int64

Column: PaymentMethod
PaymentMethod
Electronic Check            3398
Mailed Check                2290
Bank Transfer(automatic)    2229
Credit Card (automatic)     2083
Name: count, dtype: int64
```

```
Column: CaseOrder
CaseOrder
1        1
6671     1
6664     1
6665     1
6666     1
        ..
3334     1
3335     1
3336     1
3337     1
10000    1
Name: count, Length: 10000, dtype: int64

Column: Zip
Zip
32340    4
75077    4
44310    4
61764    4
16115    4
        ..
43788    1
58579    1
53526    1
79104    1
30523    1
Name: count, Length: 8583, dtype: int64

Numerical Data Summary:

Column: Lat
count    10000.000000
mean        38.757567
std          5.437389
min         17.966120
25%         35.341828
50%         39.395800
75%         42.106908
max         70.640660
Name: Lat, dtype: float64

Column: Lng
count    10000.000000
mean       -90.782536
std         15.156142
min       -171.688150
25%        -97.082812
50%        -87.918800
75%        -80.088745
max        -65.667850
Name: Lng, dtype: float64

Column: Population
count     10000.000000
mean       9756.562400
std       14432.698671
min           0.000000
25%         738.000000
```

```
50%        2910.500000
75%       13168.000000
max      111850.000000
Name: Population, dtype: float64

Column: Children
count    10000.0000
mean         2.0877
std          2.1472
min          0.0000
25%          0.0000
50%          1.0000
75%          3.0000
max         10.0000
Name: Children, dtype: float64

Column: Age
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

Column: Income
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

Column: Outage_sec_perweek
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

Column: Email
count    10000.000000
mean        12.016000
std          3.025898
min          1.000000
25%         10.000000
50%         12.000000
75%         14.000000
max         23.000000
Name: Email, dtype: float64
```

```
Column: Contacts
count    10000.000000
mean         0.994200
std          0.988466
min          0.000000
25%          0.000000
50%          1.000000
75%          2.000000
max          7.000000
Name: Contacts, dtype: float64

Column: Yearly_equip_failure
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

Column: Tenure
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

Column: MonthlyCharge
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

Column: Bandwidth_GB_Year
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

Column: Item1
count    10000.000000
mean         3.490800
std          1.037797
min          1.000000
```

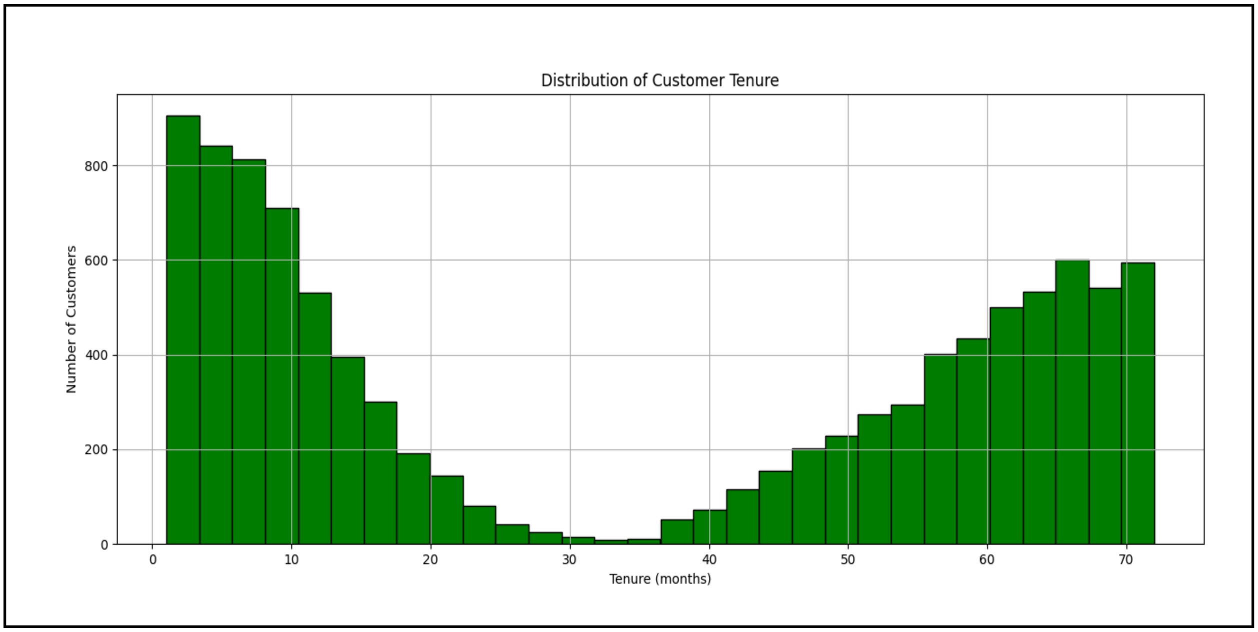
```
25%          3.000000
50%          3.000000
75%          4.000000
max          7.000000
Name: Item1, dtype: float64

Column: Item2
count    10000.000000
mean         3.505100
std          1.034641
min          1.000000
25%          3.000000
50%          4.000000
75%          4.000000
max          7.000000
Name: Item2, dtype: float64

Column: Item3
count    10000.000000
mean         3.487000
std          1.027977
min          1.000000
25%          3.000000
50%          3.000000
75%          4.000000
max          8.000000
Name: Item3, dtype: float64

Column: Item4
count    10000.000000
mean         3.497500
std          1.025816
min          1.000000
25%          3.000000
50%          3.000000
75%          4.000000
max          7.000000
Name: Item4, dtype: float64

Column: Item5
count    10000.000000
mean         3.492900
std          1.024819
min          1.000000
25%          3.000000
50%          3.000000
75%          4.000000
max          7.000000
Name: Item5, dtype: float64

Column: Item6
count    10000.000000
mean         3.497300
std          1.033586
min          1.000000
25%          3.000000
50%          3.000000
75%          4.000000
max          8.000000
Name: Item6, dtype: float64
```

```
Column: Item7
count    10000.000000
mean         3.509500
std          1.028502
min          1.000000
25%          3.000000
50%          4.000000
75%          4.000000
max          7.000000
Name: Item7, dtype: float64

Column: Item8
count    10000.000000
mean         3.495600
std          1.028633
min          1.000000
25%          3.000000
50%          3.000000
75%          4.000000
max          8.000000
Name: Item8, dtype: float64
```

### C3. Univariate and Bivariate Visualizations

Shown below is a histogram of the dependent variable, customer tenure, generated using the `hist()` method within the `matplotlib` library.

| **Code** |
|---|
| `plt.figure(figsize=(12, 6))`<br>`plt.hist(df["Tenure"], bins=30, color="green", edgecolor="black")`<br>`plt.title("Distribution of Customer Tenure")`<br>`plt.xlabel("Tenure (months)")`<br>`plt.ylabel("Number of Customers")`<br>`plt.grid(True)`<br>`plt.show()` |
| **Result** |



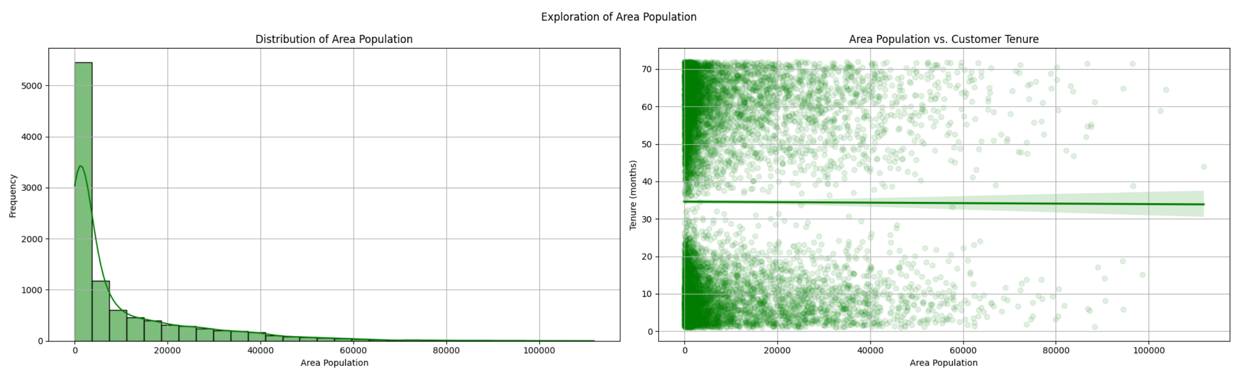

Shown below is a univariate analysis of each numerical variable and a bivariate analysis of the aforementioned variable and tenure in the form of a histogram and linear regression plot respectively.

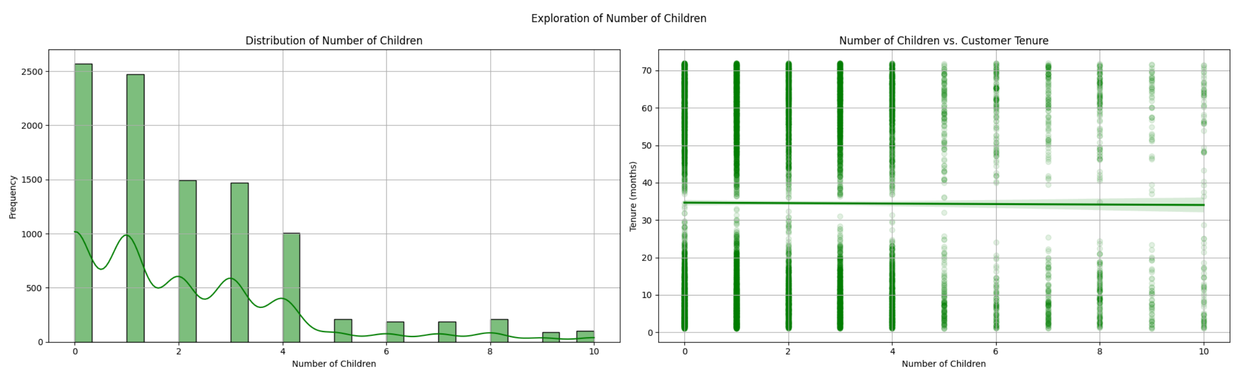
**Code**

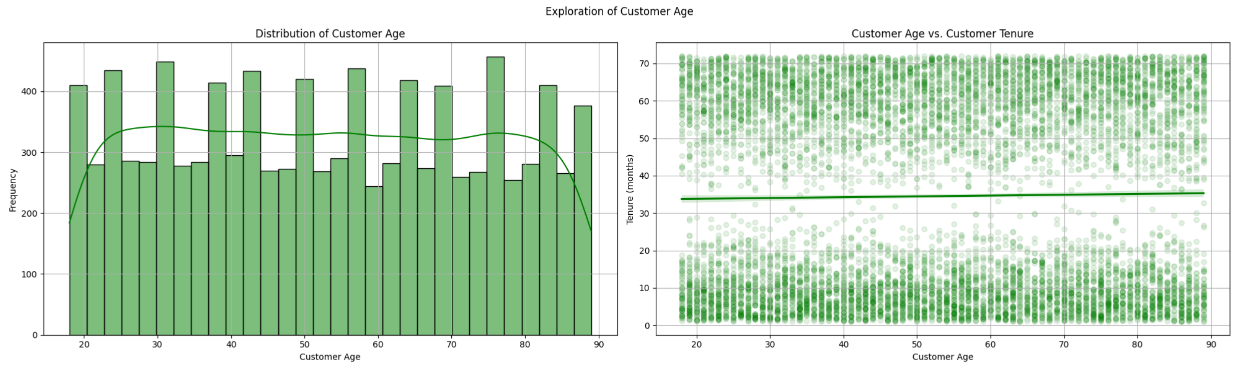
```
numerical_column_titles = {
    # "CaseOrder":            "Case Order",
    "Lat": "Latitude",
    "Lng": "Longitude",
    "Population": "Area Population",
    "Children": "Number of Children",
    "Age": "Customer Age",
    "Income": "Customer Income",
    "Outage_sec_perweek": "Outage Seconds Per Week",
    "Email": "Number of Emails Sent",
    "Contacts": "Number of Support Contacts",
    "Yearly_equip_failure": "Annual Equipment Failures",
    # "Tenure":               "Customer Tenure",
    "MonthlyCharge": "Average Monthly Charge",
    "Bandwidth_GB_Year": "Annual Bandwidth Usage",
}

for column, variable_name in numerical_column_titles.items():
    # Setting up the figure and axes for side-by-side plots
    fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(20, 6))
    plt.suptitle(f"Exploration of {variable_name}")

    # Plot 1: Histogram of Age on ax1
    sns.histplot(df[column], bins=30, kde=True, color="green", ax=ax1)
    ax1.set_title(f"Distribution of {variable_name}")
    ax1.set_xlabel(variable_name)
    ax1.set_ylabel("Frequency")
    ax1.grid(True)
```

```
    # Plot 2: Scatter Plot of Age vs. Tenure on ax2 using regplot for a potential regression
line
    sns.regplot(
        x=column,
        y="Tenure",
        data=df,
        color="green",
        ax=ax2,
        scatter_kws={"alpha": 1 / 10},
    )
    ax2.set_title(f"{variable_name} vs. Customer Tenure")
    ax2.set_xlabel(variable_name)
    ax2.set_ylabel("Tenure (months)")
    ax2.grid(True)

    # Show the plots
    plt.tight_layout()  # Adjusts plot parameters to give some padding and prevent overlap
    plt.show()
```

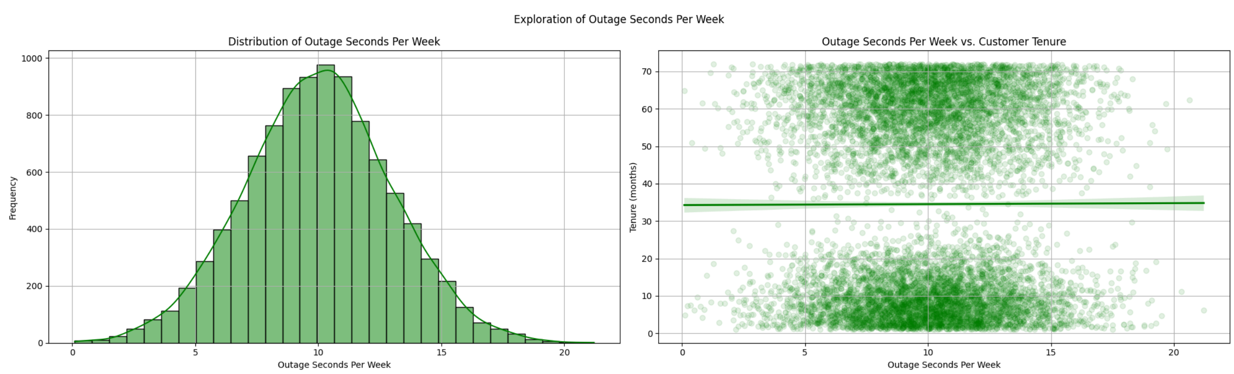
**Result:**

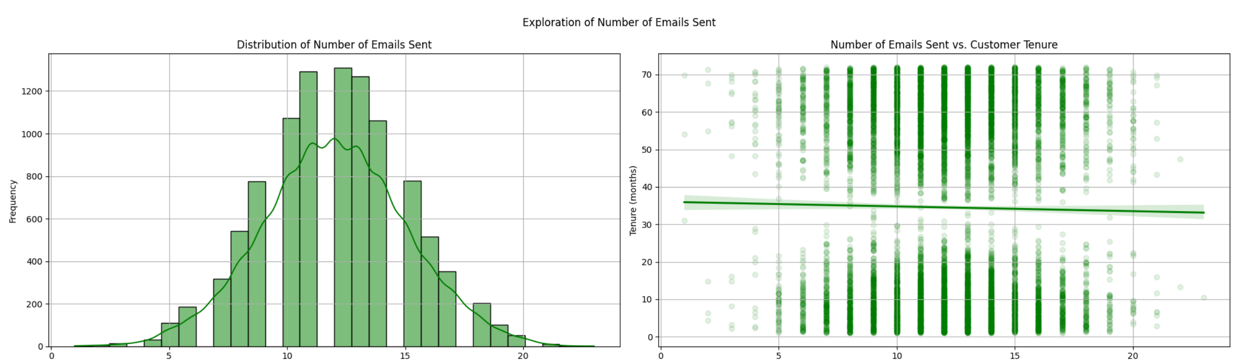


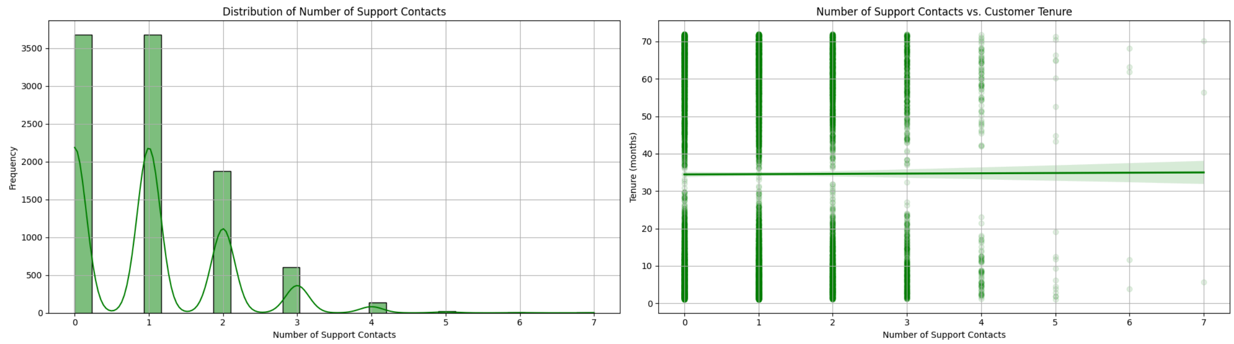

Exploration of Customer Age
Distribution of Customer Age
Frequency
Customer Age
Customer Age vs. Customer Tenure
Tenure (months)
Customer Age


Exploration of Customer Income
Distribution of Customer Income
Frequency
Customer Income
Customer Income vs. Customer Tenure
Tenure (months)
Customer Income

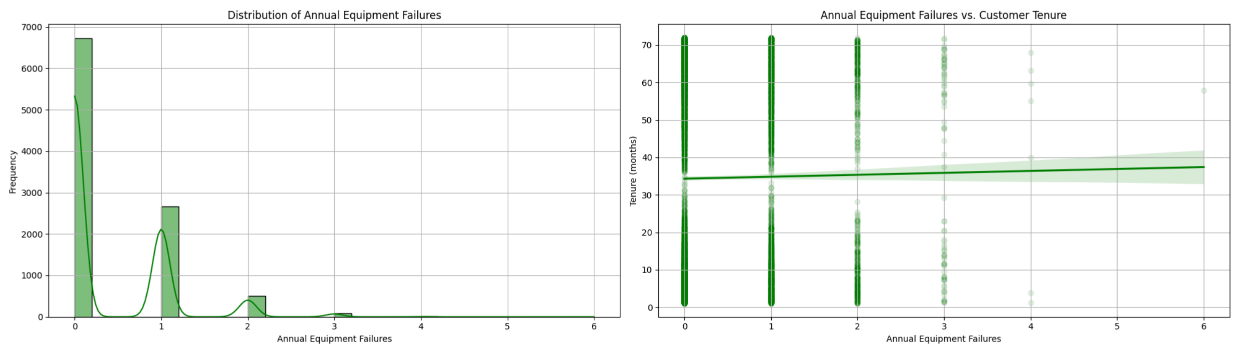

Exploration of Outage Seconds Per Week
Distribution of Outage Seconds Per Week
Frequency
Outage Seconds Per Week
Outage Seconds Per Week vs. Customer Tenure
Tenure (months)
Outage Seconds Per Week


Exploration of Number of Emails Sent
Distribution of Number of Emails Sent
Frequency
Number of Emails Sent
Number of Emails Sent vs. Customer Tenure
Tenure (months)
Number of Emails Sent

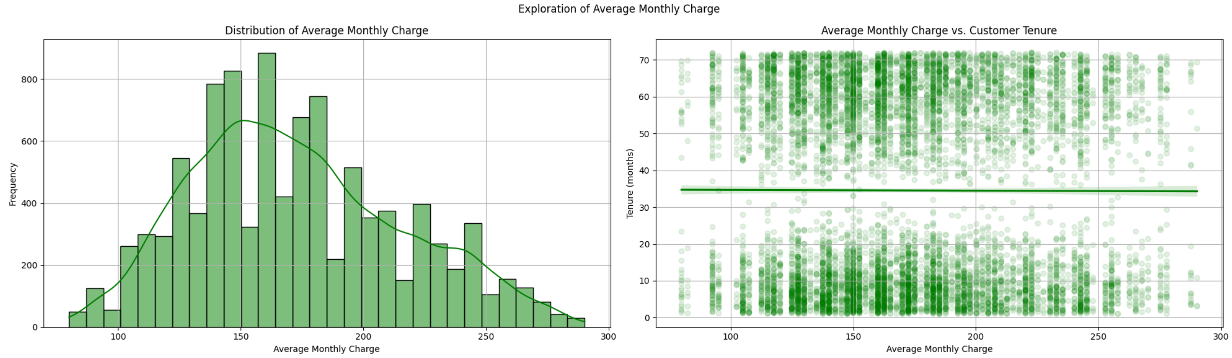
Exploration of Number of Support Contacts



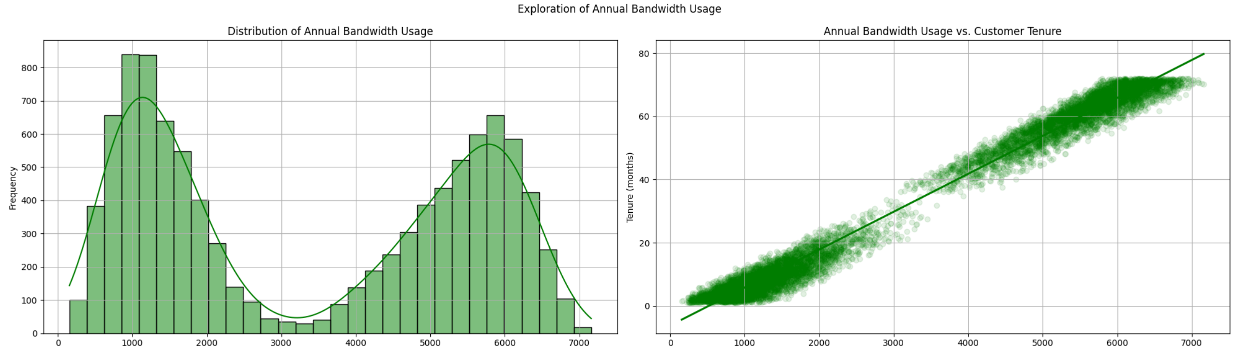

Exploration of Annual Equipment Failures




Exploration of Average Monthly Charge




Exploration of Annual Bandwidth Usage

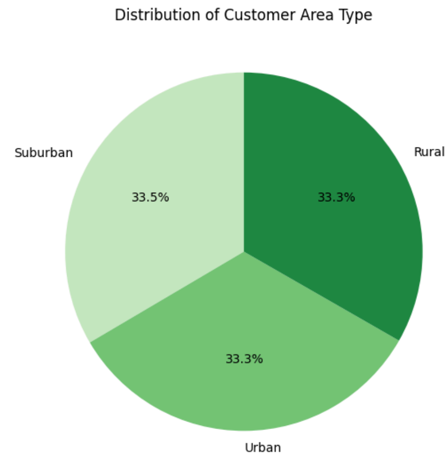


Shown below is a univariate analysis of each categorical variable and a bivariate analysis of the aforementioned variable and tenure in the form of a pie chart and box plot respectively.

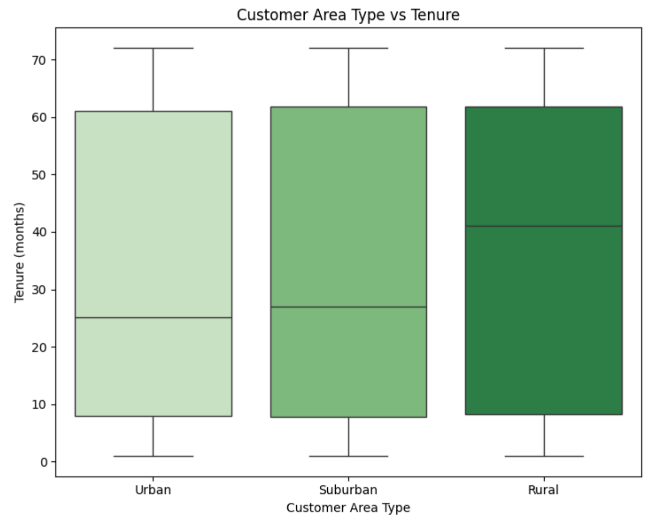
| **Code** |
|---|

```
categorical_column_titles = {
    "State": "Customer State of Residence",
    "Area": "Customer Area Type",
    "TimeZone": "Time Zone of Customer Residence",
    "Marital": "Marital Status of Customer",
    "Gender": "Gender of Customer",
    "Churn": "Customer Churn Status Last Month",
    "Techie": "Whether Customer is Tech-Savvy",
    "Contract": "Type of Customer Contract",
    "Port_modem": "Whether Customer Uses a Portable Modem",
    "Tablet": "Whether Customer Owns a Tablet",
    "InternetService": "Type of Internet Service Customer Uses",
    "Phone": "Whether Customer Has Phone Service",
    "Multiple": "Whether Customer Has Multiple Lines",
    "OnlineSecurity": "Whether Customer Uses Online Security Service",
    "OnlineBackup": "Whether Customer Uses Online Backup Service",
    "DeviceProtection": "Whether Customer Uses Device Protection Service",
    "TechSupport": "Whether Customer Has Technical Support Service",
    "StreamingTV": "Whether Customer Uses Streaming TV Service",
    "StreamingMovies": "Whether Customer Uses Streaming Movies Service",
    "PaperlessBilling": "Whether Customer Uses Paperless Billing",
    "PaymentMethod": "Customer's Payment Method",
}


for column, variable_name in categorical_column_titles.items():
    fig, ax = plt.subplots(1, 2, figsize=(14, 6))

    # Pie chart
    area_counts = df[column].value_counts()
    colors = sns.color_palette(
        "Greens", n_colors=area_counts.size
    )  # Using green color palette
    ax[0].pie(
        area_counts,
        labels=area_counts.index,
        autopct="%1.1f%%",
        startangle=90,
        colors=colors,
    )
    ax[0].set_title(f"Distribution of {variable_name}")

    # Box plot
    sns.boxplot(x=column, y="Tenure", data=df, ax=ax[1], palette="Greens")
    ax[1].set_title(f"{variable_name} vs Tenure")
    ax[1].set_xlabel(variable_name)
    ax[1].set_ylabel("Tenure (months)")

    # Show the plot
    plt.tight_layout()
    plt.show()
```

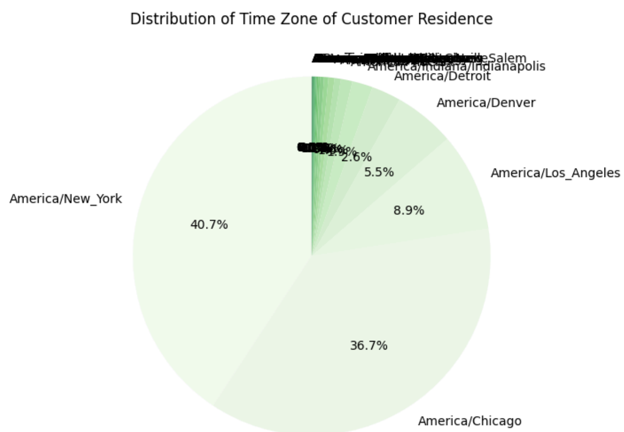
| **Result:** |
|---|



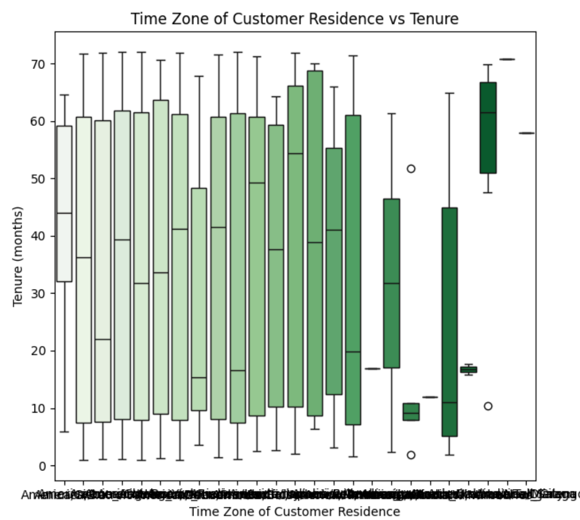
Distribution of Customer Area Type
Suburban
33.5%
Rural
33.3%
33.3%
Urban

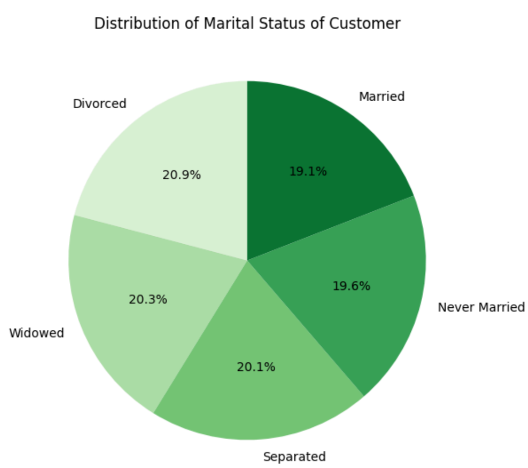
Customer Area Type vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Urban
Suburban
Rural
Customer Area Type

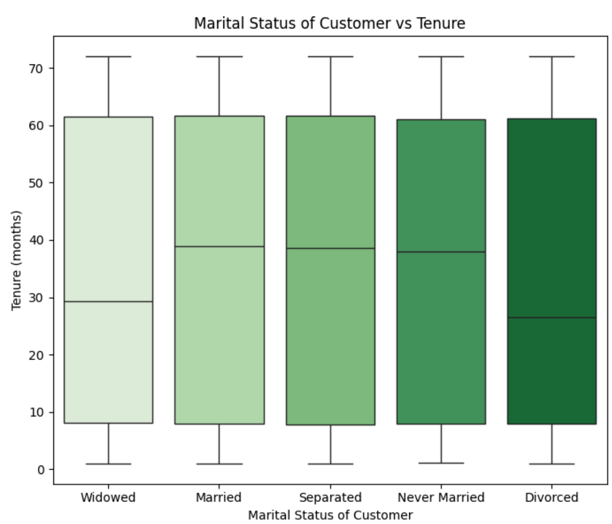
Distribution of Time Zone of Customer Residence
America/Indiana/Indianapolis
America/Detroit
America/Denver
America/Los_Angeles
America/New_York
2.6%
5.5%
8.9%
40.7%
36.7%
America/Chicago

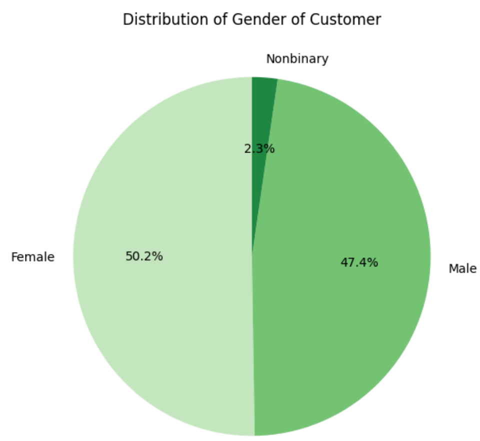
Time Zone of Customer Residence vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Time Zone of Customer Residence

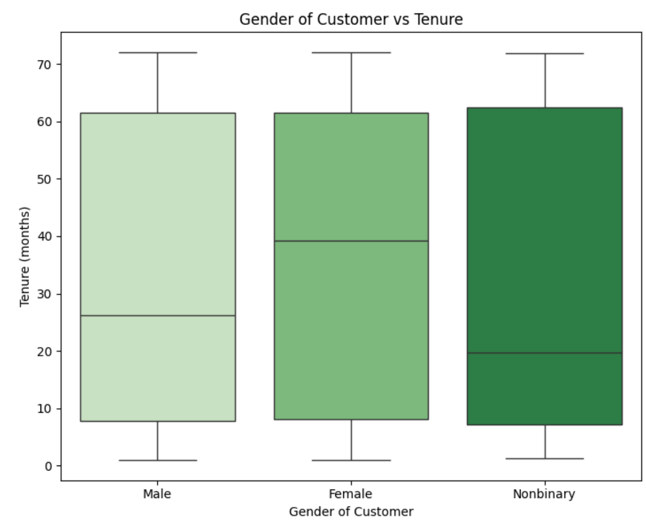



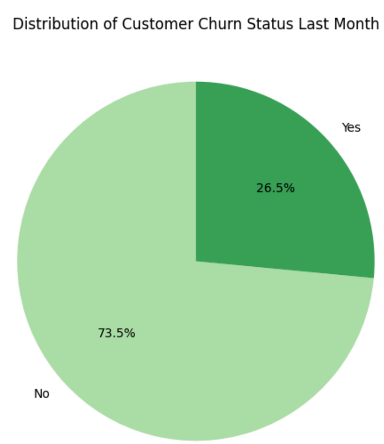
Distribution of Marital Status of Customer
Divorced
20.9%
Married
19.1%
19.6%
Never Married
Widowed
20.3%
20.1%
Separated

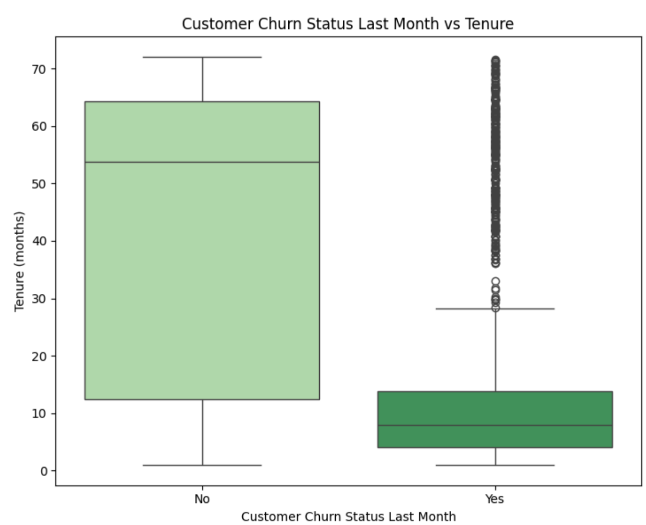



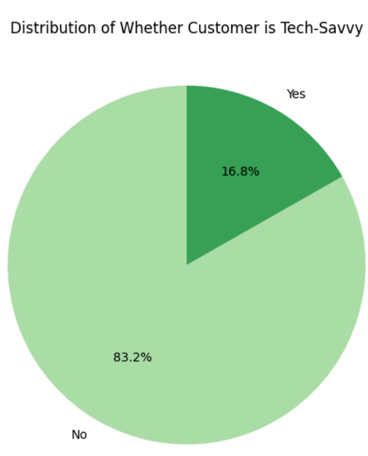
Marital Status of Customer vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Widowed
Married
Separated
Never Married
Divorced
Marital Status of Customer

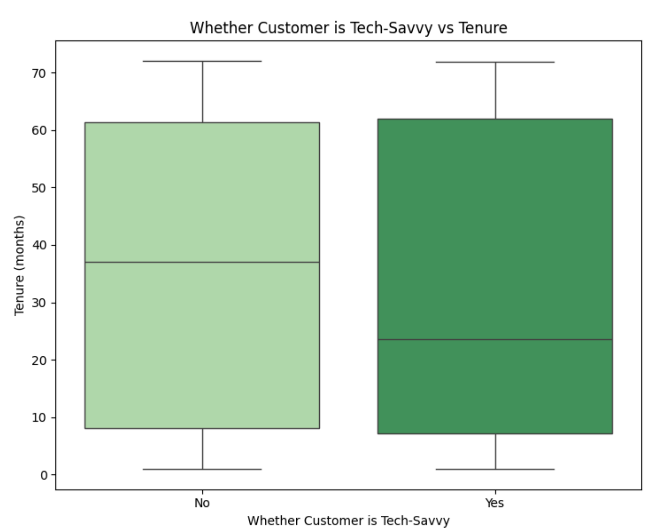

Distribution of Gender of Customer
Nonbinary
2.3%
Female
50.2%
47.4%
Male

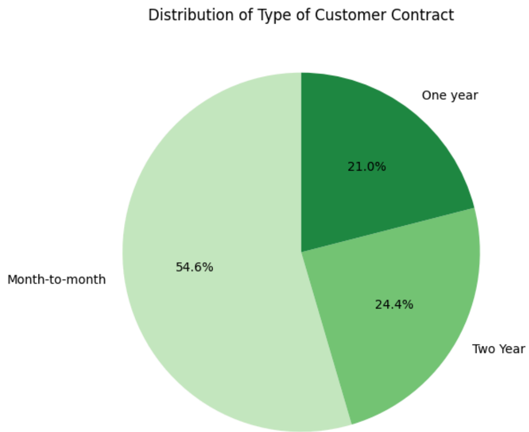

Gender of Customer vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Male
Female
Nonbinary
Gender of Customer

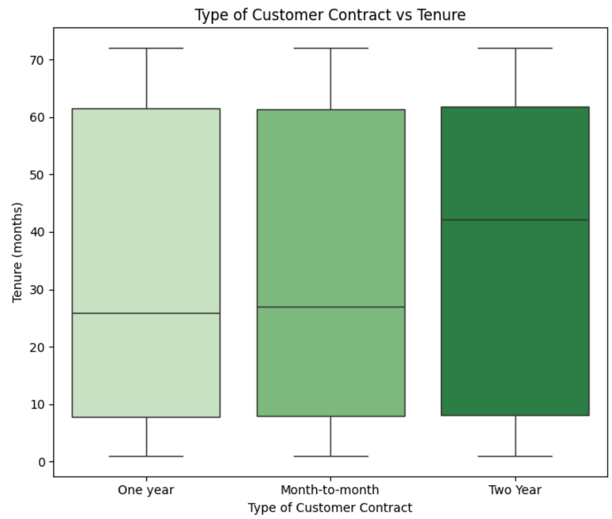

Distribution of Customer Churn Status Last Month
Yes
26.5%
73.5%
No

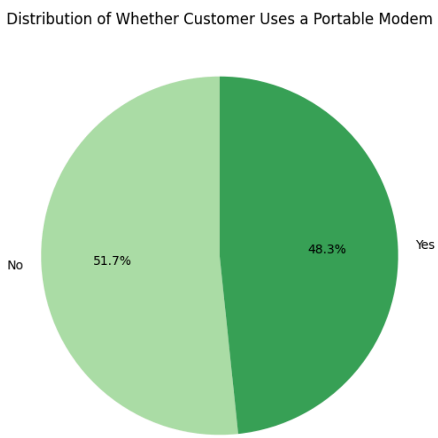

Customer Churn Status Last Month vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Customer Churn Status Last Month

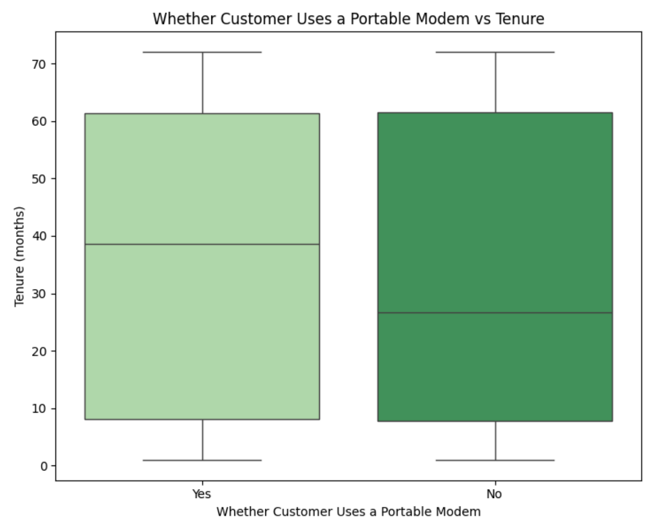

Distribution of Whether Customer is Tech-Savvy
Yes
16.8%
83.2%
No

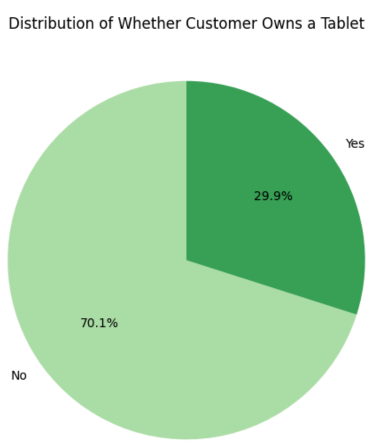

Whether Customer is Tech-Savvy vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Whether Customer is Tech-Savvy

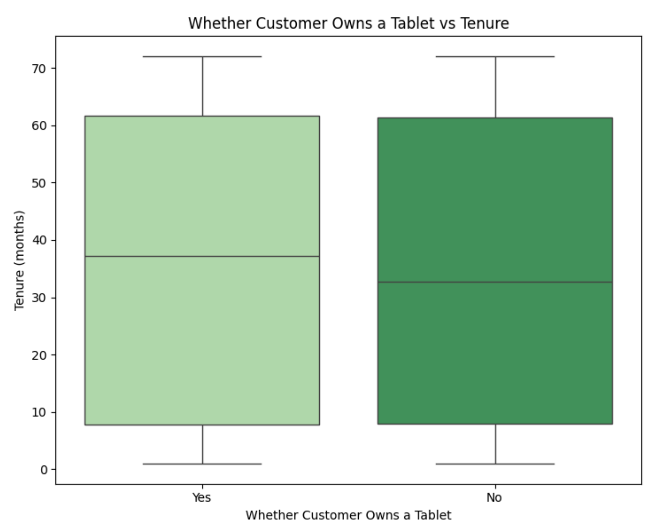

Distribution of Type of Customer Contract
One year
21.0%
Month-to-month
54.6%
24.4%
Two Year

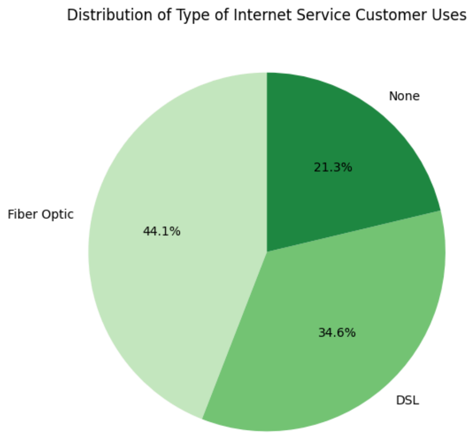

Type of Customer Contract vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
One year
Month-to-month
Two Year
Type of Customer Contract

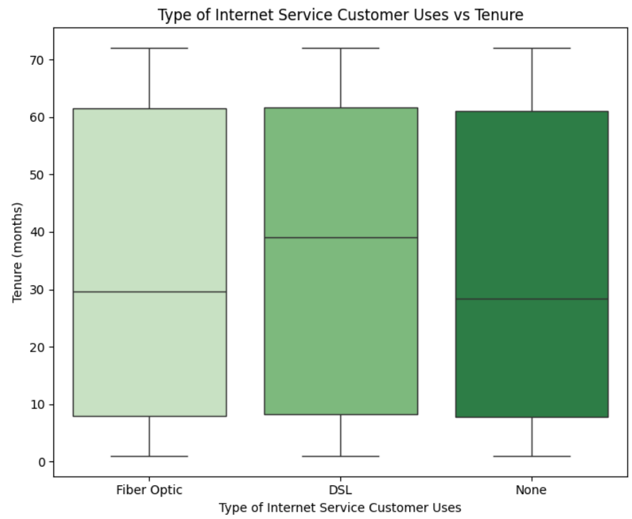

Distribution of Whether Customer Uses a Portable Modem
No
51.7%
48.3%
Yes

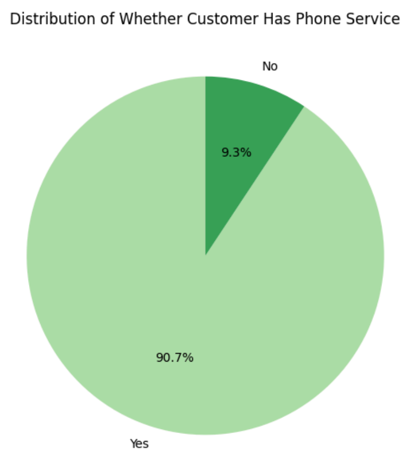

Whether Customer Uses a Portable Modem vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Uses a Portable Modem


Distribution of Whether Customer Owns a Tablet
Yes
29.9%
70.1%
No

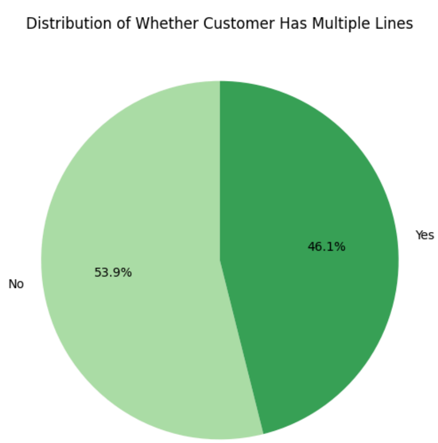

Whether Customer Owns a Tablet vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Owns a Tablet

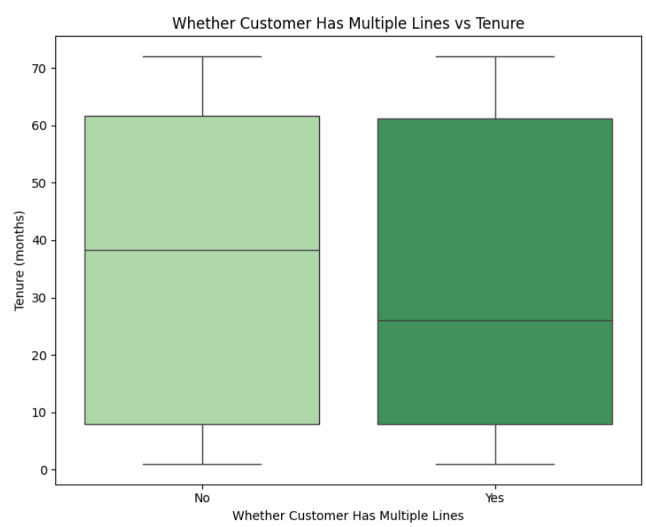

Distribution of Type of Internet Service Customer Uses
None
21.3%
Fiber Optic
44.1%
34.6%
DSL


Type of Internet Service Customer Uses vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Fiber Optic
DSL
None
Type of Internet Service Customer Uses


Distribution of Whether Customer Has Phone Service
No
9.3%
90.7%
Yes

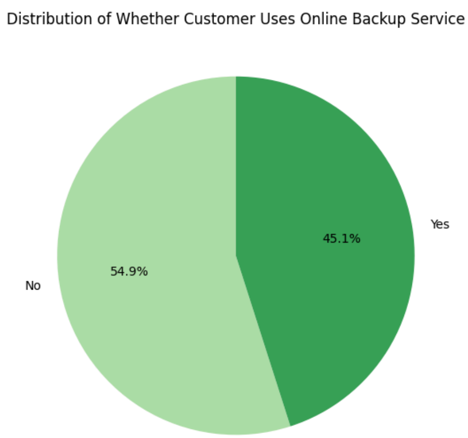

Whether Customer Has Phone Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Has Phone Service

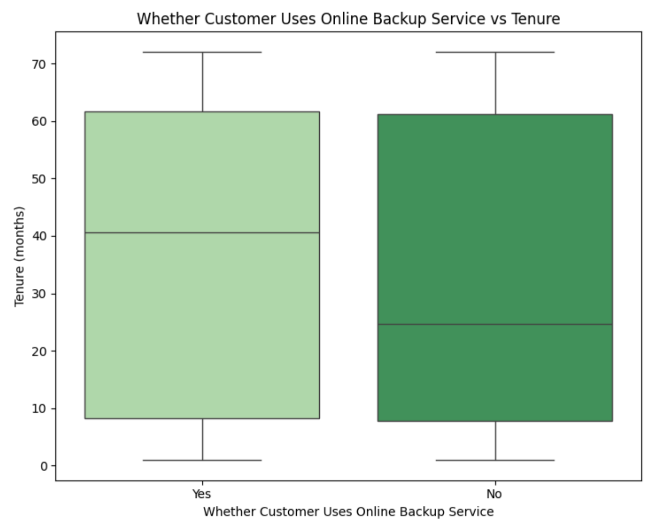

Distribution of Whether Customer Has Multiple Lines
Yes
46.1%
No
53.9%

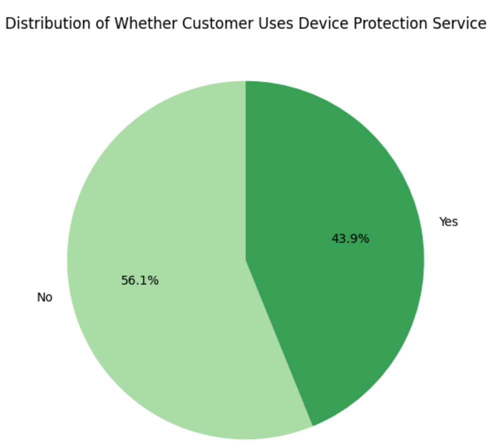

Whether Customer Has Multiple Lines vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Whether Customer Has Multiple Lines

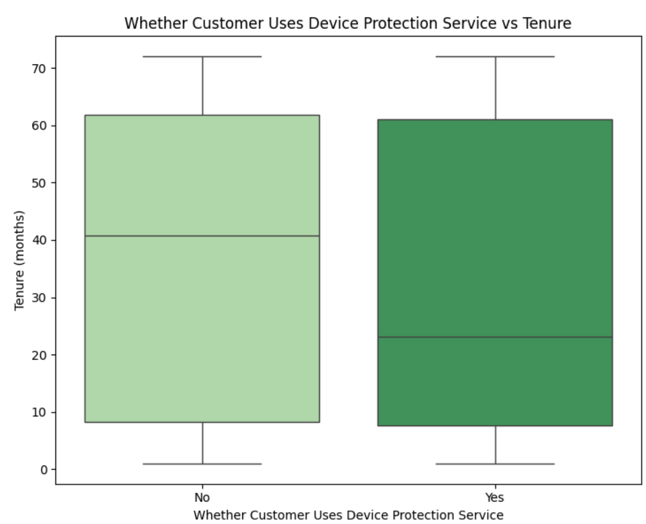

Distribution of Whether Customer Uses Online Security Service
Yes
35.8%
64.2%
No

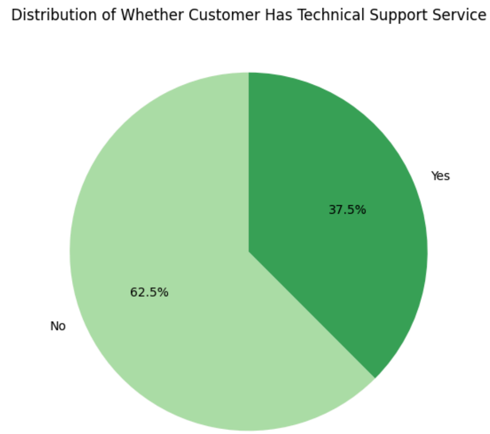

Whether Customer Uses Online Security Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Uses Online Security Service

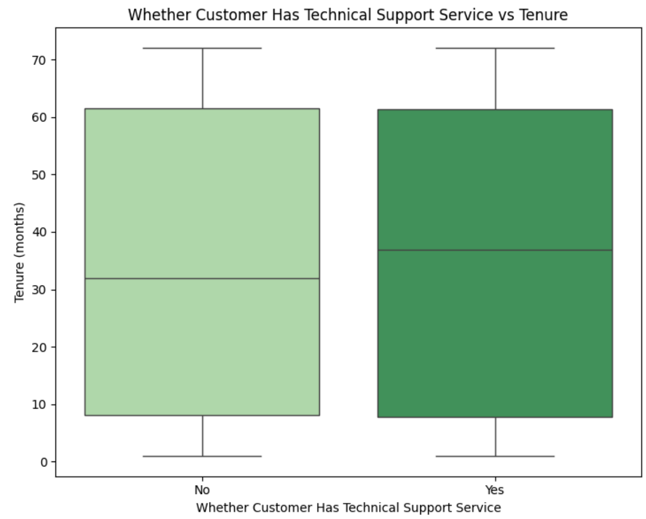

Distribution of Whether Customer Uses Online Backup Service
Yes
45.1%
54.9%
No

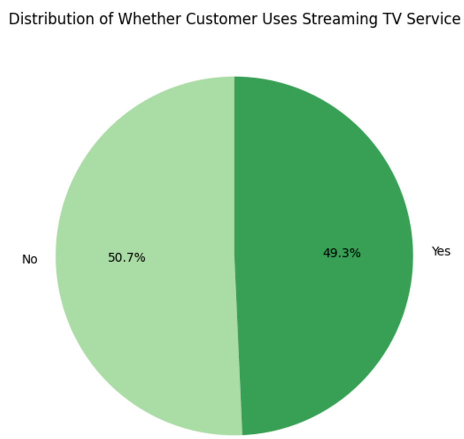

Whether Customer Uses Online Backup Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Uses Online Backup Service

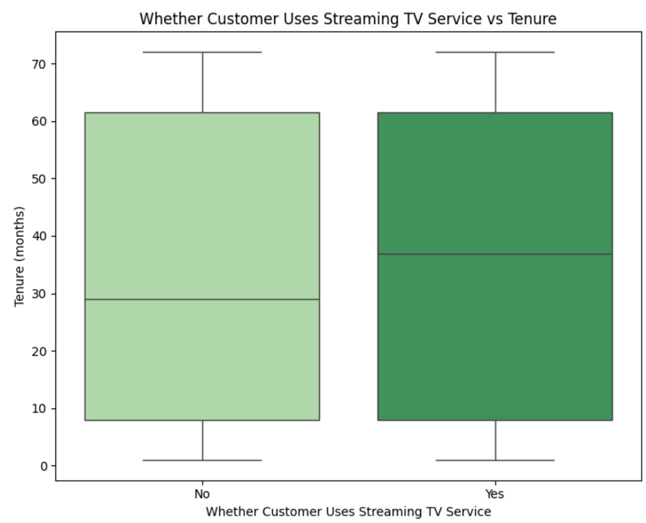

Distribution of Whether Customer Uses Device Protection Service
Yes
43.9%
56.1%
No

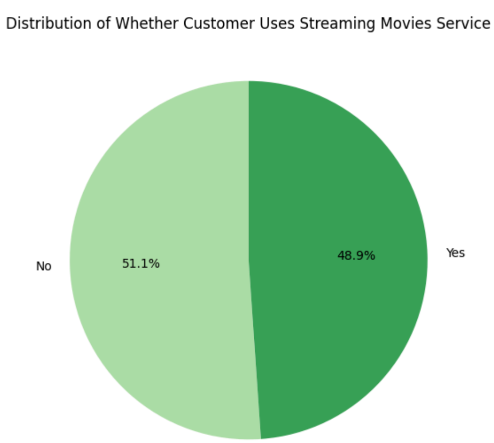

Whether Customer Uses Device Protection Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Whether Customer Uses Device Protection Service


Distribution of Whether Customer Has Technical Support Service
Yes
37.5%
62.5%
No

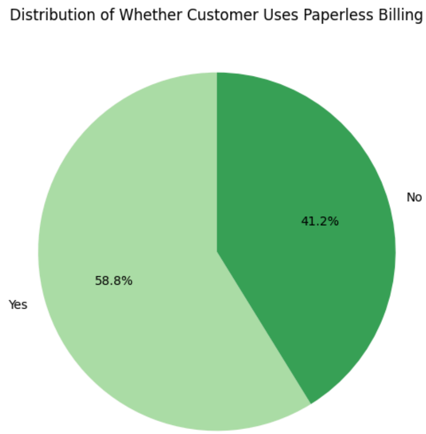

Whether Customer Has Technical Support Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Whether Customer Has Technical Support Service


Distribution of Whether Customer Uses Streaming TV Service
No
50.7%
49.3%
Yes

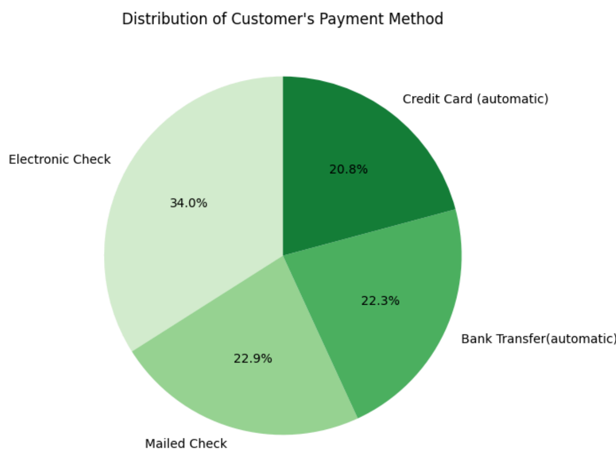

Whether Customer Uses Streaming TV Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
No
Yes
Whether Customer Uses Streaming TV Service

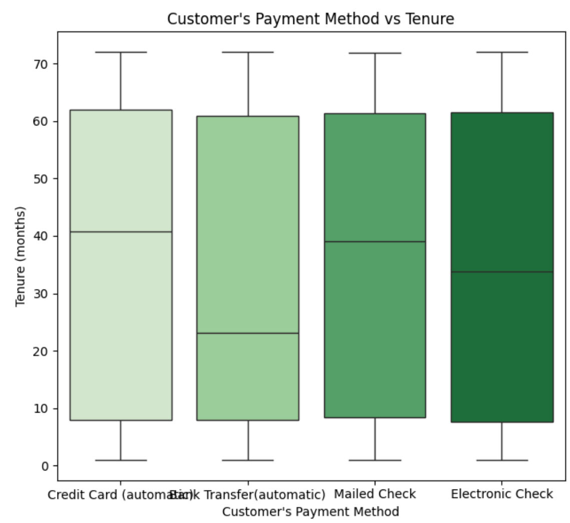

Distribution of Whether Customer Uses Streaming Movies Service
No
51.1%
48.9%
Yes


Whether Customer Uses Streaming Movies Service vs Tenure
Tenure (months)
0
10
20
30
40
50
60
70
Yes
No
Whether Customer Uses Streaming Movies Service








## C4. Data Transformation

My goal with regards to cleaning the sample data is to create a uniform DataFrame to which multiple linear regression can be applied and from which useful business insights can be drawn. To achieve this, the following data transformations will be performed:

- Zip codes will be converted to strings with leading zeroes
- Time zones will be standardized using a mapping function
- Columns intended to represent boolean values will be converted to booleans with “Yes” being mapped to True and “No” being mapped to False.
- Nominal variables that include repeating values will be converted to categories.
- Dummy columns will be generated to allow for multiple linear regression

The annotated code below achieves these objectives.

**Code**

```
# C4: Data Transformation

# Convert zip codes to string to preserve leading zeros
df["Zip"] = (
    df["Zip"].astype(str).str.zfill(5)
)  # Assuming "zip" is the name of the column for zip codes

# Mapping of locations to time zones
time_zone_map = {
    "America/New_York": "EST",
    "America/Detroit": "EST",
    "America/Indiana/Indianapolis": "EST",
    "America/Kentucky/Louisville": "EST",
    "America/Indiana/Vincennes": "EST",
    "America/Indiana/Tell_City": "EST",
    "America/Indiana/Petersburg": "EST",
    "America/Indiana/Knox": "EST",
    "America/Indiana/Winamac": "EST",
    "America/Indiana/Marengo": "EST",
    "America/Toronto": "EST",
    "America/Chicago": "CST",
    "America/Menominee": "CST",
    "America/North_Dakota/New_Salem": "CST",
    "America/Denver": "MST",
    "America/Phoenix": "MST",
    "America/Boise": "MST",
    "America/Los_Angeles": "PST",
    "America/Anchorage": "AKST",
    "America/Nome": "AKST",
    "America/Sitka": "AKST",
    "America/Juneau": "AKST",
    "Pacific/Honolulu": "HAST",
    "America/Puerto_Rico": "AST",
    "America/Ojinaga": "MST",
}

# Replace the TimeZone column with the mapped values
df["TimeZone"] = df["TimeZone"].map(time_zone_map)

# Convert boolean columns to actual boolean types
boolean_columns = [
    "Techie",
    "Port_modem",
    "Tablet",
    "Phone",
    "Multiple",
    "OnlineSecurity",
    "OnlineBackup",
    "DeviceProtection",
    "TechSupport",
    "StreamingTV",
    "StreamingMovies",
    "PaperlessBilling",
]
for column in boolean_columns:
    df[column] = df[column].map({"Yes": True, "No": False})
```

```python
# Convert remaining categorical data to category dtype
nominal_columns = [
    "Area",
    "TimeZone",
    "Job",
    "Marital",
    "Gender",
    "Contract",
    "InternetService",
    "PaymentMethod",
]
for column in nominal_columns:
    df[column] = df[column].astype("category")

# Create a new dataframe with only relevant variables
df_encoded = df[
    [
        "Population",
        "Area",
        "TimeZone",
        "Children",
        "Age",
        "Income",
        "Marital",
        "Gender",
        "Outage_sec_perweek",
        "Email",
        "Contacts",
        "Yearly_equip_failure",
        "Techie",
        "Contract",
        "Port_modem",
        "Tablet",
        "InternetService",
        "Phone",
        "Multiple",
        "OnlineSecurity",
        "OnlineBackup",
        "DeviceProtection",
        "TechSupport",
        "StreamingTV",
        "StreamingMovies",
        "PaperlessBilling",
        "PaymentMethod",
        "Tenure",
        "MonthlyCharge",
        "Bandwidth_GB_Year",
    ]
].copy()
df_encoded = pd.get_dummies(
    df_encoded,
    columns=[
        "Area",
        "Gender",
        "Contract",
        "Marital",
        "TimeZone",
        "InternetService",
        "PaymentMethod",
```

```
    ],
    drop_first = True,
)

print(df_encoded)

df_encoded.to_csv("churn_encoded.csv")
```

**Result:**

```
Press enter to begin Part C4
      Population  Children  Age    Income  Outage_sec_perweek  Email  Contacts  Yearly_equip_failure  ...  TimeZone_HAST  TimeZone_MST  TimeZone_PST  InternetService_Fiber Optic  InternetService_None  PaymentMethod_Credit Card (automatic)  PaymentMethod_Electronic Check  PaymentMethod_Mailed Check
0             38         0   68  28561.99            7.978323     10         0                     1  ...          False         False         False                         True                 False                                   True                           False                       False
1          10446         1   27  21704.77           11.699080     12         0                     1  ...          False         False         False                         True                 False                                  False                           False                       False
2           3735         4   50   9609.57           10.752800      9         0                     1  ...          False         False          True                        False                 False                                   True                           False                       False
3          13863         1   48  18925.23           14.913540     15         2                     0  ...          False         False          True                        False                 False                                  False                           False                        True
4          11352         0   83  40074.19            8.147417     16         2                     1  ...          False         False         False                         True                 False                                  False                           False                        True
...          ...       ...  ...       ...                 ...    ...       ...                   ...  ...            ...           ...           ...                          ...                   ...                                    ...                             ...                         ...
9995         640         3   23  55723.74            9.415935     12         2                     0  ...          False         False         False                        False                 False                                  False                            True                       False
9996       77168         4   48  34129.34            6.740547     15         2                     0  ...          False         False         False                         True                 False                                  False                            True                       False
9997         406         1   48  45983.43            6.590911     10         0                     0  ...          False         False         False                         True                 False                                  False                           False                       False
9998       35575         1   39  16667.58           12.071910     14         1                     0  ...          False         False         False                         True                 False                                   True                           False                       False
9999       12230         1   28   9020.92           11.754720     17         1                     0  ...          False         False         False                         True                 False                                  False                            True                       False

[10000 rows x 44 columns]
```

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

See churn_encoded.csv.

### D1.  Initial Multiple Linear Regression Model

I constructed an initial multiple linear regression model using the following code which includes `.assign(const=1)` as a baseline.

**Code**

```
# D1: Initial Linear Regression Model

Y = df_encoded["Tenure"]
X = df_encoded.drop(columns=["Tenure"])
X = sm.add_constant(X)
model = sm.OLS(Y, X.astype(float))
results = model.fit()
print(results.summary())
```

**Result:**

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       1.000
Model:                            OLS   Adj. R-squared:                  1.000
Method:                 Least Squares   F-statistic:                 1.408e+07
Date:                Wed, 29 May 2024   Prob (F-statistic):               0.00
Time:                        18:36:00   Log-Likelihood:                 8139.9
No. Observations:               10000   AIC:                        -1.619e+04
Df Residuals:                    9956   BIC:                        -1.587e+04
Df Model:                          43
Covariance Type:            nonrobust
==============================================================================================
                                     coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------------
const                             -3.8535      0.018   -210.202      0.000      -3.889      -3.818
Population                      -6.25e-08   7.64e-08     -0.818      0.413   -2.12e-07    8.72e-08
Children                          -0.3755      0.001   -748.564      0.000      -0.377      -0.375
```

| | | | | | | |
|---|---|---|---|---|---|---|
| Age | 0.0400 | 5.21e-05 | 768.016 | 0.000 | 0.040 | 0.040 |
| Income | 1.316e-08 | 3.82e-08 | 0.345 | 0.730 | -6.16e-08 | 8.8e-08 |
| Outage_sec_perweek | 0.0003 | 0.000 | 0.770 | 0.442 | -0.000 | 0.001 |
| Email | -6.667e-05 | 0.000 | -0.187 | 0.851 | -0.001 | 0.001 |
| Contacts | -0.0006 | 0.001 | -0.594 | 0.553 | -0.003 | 0.001 |
| Yearly_equip_failure | 0.0001 | 0.002 | 0.081 | 0.935 | -0.003 | 0.003 |
| Techie | 0.0002 | 0.003 | 0.085 | 0.932 | -0.005 | 0.006 |
| Port_modem | 0.0026 | 0.002 | 1.199 | 0.230 | -0.002 | 0.007 |
| Tablet | 0.0007 | 0.002 | 0.293 | 0.770 | -0.004 | 0.005 |
| Phone | 0.0015 | 0.004 | 0.418 | 0.676 | -0.006 | 0.009 |
| Multiple | 0.2685 | 0.005 | 59.177 | 0.000 | 0.260 | 0.277 |
| OnlineSecurity | -0.8312 | 0.002 | -365.966 | 0.000 | -0.836 | -0.827 |
| OnlineBackup | -0.3553 | 0.004 | -101.255 | 0.000 | -0.362 | -0.348 |
| DeviceProtection | -0.5970 | 0.003 | -224.761 | 0.000 | -0.602 | -0.592 |
| TechSupport | 0.3850 | 0.003 | 142.418 | 0.000 | 0.380 | 0.390 |
| StreamingTV | -1.2983 | 0.006 | -232.041 | 0.000 | -1.309 | -1.287 |
| StreamingMovies | -0.7227 | 0.007 | -106.958 | 0.000 | -0.736 | -0.709 |
| PaperlessBilling | -0.0036 | 0.002 | -1.665 | 0.096 | -0.008 | 0.001 |
| MonthlyCharge | -0.0352 | 0.000 | -287.578 | 0.000 | -0.035 | -0.035 |
| Bandwidth_GB_Year | 0.0122 | 4.97e-07 | 2.46e+04 | 0.000 | 0.012 | 0.012 |
| Area_Suburban | -0.0069 | 0.003 | -2.622 | 0.009 | -0.012 | -0.002 |
| Area_Urban | -0.0033 | 0.003 | -1.268 | 0.205 | -0.009 | 0.002 |
| Gender_Male | -0.7925 | 0.002 | -363.483 | 0.000 | -0.797 | -0.788 |
| Gender_Nonbinary | 0.2616 | 0.007 | 36.118 | 0.000 | 0.247 | 0.276 |
| Contract_One year | 0.0006 | 0.003 | 0.225 | 0.822 | -0.005 | 0.006 |
| Contract_Two Year | 0.0019 | 0.003 | 0.726 | 0.468 | -0.003 | 0.007 |
| Marital_Married | -0.0004 | 0.003 | -0.124 | 0.901 | -0.007 | 0.006 |
| Marital_Never Married | -0.0010 | 0.003 | -0.293 | 0.769 | -0.008 | 0.006 |
| Marital_Separated | 0.0028 | 0.003 | 0.845 | 0.398 | -0.004 | 0.009 |
| Marital_Widowed | 0.0001 | 0.003 | 0.039 | 0.969 | -0.006 | 0.007 |
| TimeZone_AST | 0.0103 | 0.021 | 0.488 | 0.626 | -0.031 | 0.052 |
| TimeZone_CST | 0.0135 | 0.012 | 1.092 | 0.275 | -0.011 | 0.038 |
| TimeZone_EST | 0.0098 | 0.012 | 0.790 | 0.429 | -0.014 | 0.034 |
| TimeZone_HAST | -0.0009 | 0.022 | -0.042 | 0.967 | -0.044 | 0.042 |
| TimeZone_MST | 0.0146 | 0.013 | 1.131 | 0.258 | -0.011 | 0.040 |
| TimeZone_PST | 0.0099 | 0.013 | 0.774 | 0.439 | -0.015 | 0.035 |
| InternetService_Fiber Optic | 5.7538 | 0.003 | 1665.144 | 0.000 | 5.747 | 5.761 |
| InternetService_None | 4.6005 | 0.003 | 1367.056 | 0.000 | 4.594 | 4.607 |
| PaymentMethod_Credit Card (automatic) | 0.0019 | 0.003 | 0.586 | 0.558 | -0.005 | 0.008 |
| PaymentMethod_Electronic Check | 0.0029 | 0.003 | 0.992 | 0.321 | -0.003 | 0.009 |
| PaymentMethod_Mailed Check | 0.0071 | 0.003 | 2.225 | 0.026 | 0.001 | 0.013 |

==============================================================================

| | | | |
|---|---|---|---|
| Omnibus: | 34814.468 | Durbin-Watson: | 2.003 |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 1633.922 |
| Skew: | -0.034 | Prob(JB): | 0.00 |
| Kurtosis: | 1.021 | Cond. No. | 1.59e+06 |

==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.59e+06. This might indicate that there are
strong multicollinearity or other numerical problems.

**D2. Justification for Reduced Model**

The previous code output includes a warning that there are strong multicollinearity problems. One of the assumptions of a multiple linear regression model is independence; variables should not display multicollinearity.

To identify which factors have high multicollinearity, we can use Variance Inflation Factor (VIF) which measures how much the behavior of one explanatory variable in a statistical model is influenced by its relationships with other variables. The `statsmodels.stats.outliers_influence` library contains a method called `variance_inflation_factor` which will be used to identify the multicollinearity of each factor.

The process of identifying factors with high multicollinearity is essential for answering the research question, “Which customer factors contribute most to a customer’s tenure with the service provider?” Without this process, it is difficult to identify the individual effect of each factor and thus the predictive power of the model is undermined.

**Code**

```
# D2: Reduced Feature Set
X = df_encoded.drop(columns=["Tenure"])
vif_data = pd.DataFrame()
vif_data["feature"] = X.columns

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

print(vif_data)
```

**Result:**

```
                                  feature        VIF
0                              Population   1.533004
1                                Children   1.947016
2                                     Age   7.489890
3                                  Income   2.981887
4                       Outage_sec_perweek  11.843031
5                                   Email  15.968287
6                                Contacts   2.013882
7                     Yearly_equip_failure   1.396949
8                                  Techie   1.205564
9                              Port_modem   1.936711
10                                 Tablet   1.433701
11                                  Phone  10.354601
12                               Multiple   6.438211
13                         OnlineSecurity   1.595813
14                           OnlineBackup   3.986759
```

```
15                       DeviceProtection    2.460137
16                            TechSupport    2.183592
17                            StreamingTV   10.099297
18                        StreamingMovies   14.180044
19                       PaperlessBilling    2.426246
20                          MonthlyCharge  281.498686
21                      Bandwidth_GB_Year    3.458410
22                          Area_Suburban    1.999114
23                             Area_Urban    1.992967
24                            Gender_Male    1.946952
25                       Gender_Nonbinary    1.049067
26                      Contract_One year    1.389313
27                      Contract_Two Year    1.452644
28                        Marital_Married    1.906090
29                  Marital_Never Married    1.925616
30                      Marital_Separated    1.955907
31                        Marital_Widowed    1.960063
32                           TimeZone_AST    1.320836
33                           TimeZone_CST   28.147184
34                           TimeZone_EST   34.598560
35                          TimeZone_HAST    1.266848
36                           TimeZone_MST    6.272298
37                           TimeZone_PST    7.594158
38             InternetService_Fiber Optic    4.051438
39                   InternetService_None    1.864629
40  PaymentMethod_Credit Card (automatic)    1.929718
41         PaymentMethod_Electronic Check    2.509253
42             PaymentMethod_Mailed Check    2.020339
```

The code output shows that MonthlyCharge has a VIF of approximately 23.93, indicating a very high level of multicollinearity. This could be because this variable is highly related to several other services and features, which impairs the model's ability to distinguish its individual effect.

The following steps were implemented to reduce the multicollinearity of the feature selection:

- Remove MonthlyCharge, which has a high variance inflation factor.
- Apply backward elimination to remove insignificant features with the highest p-values. To accomplish this, I used a custom function based on a wrapper function [found here](found here).

The function `backward_elimination_for_vif`, is designed to refine a set of predictors for a regression model by eliminating those that either don't significantly contribute to the model or cause multicollinearity. The steps below should give you an understanding if its internal mechanisms:

1. **Initialization**: The function takes in predictors (`X`), the outcome variable (`Y`), and two thresholds: one for the Variance Inflation Factor (VIF) and another for the p-value of the predictors. VIF measures how much the variance of a regression coefficient is increased due to multicollinearity, and the p-value assesses if the relationship between the predictor and the outcome is statistically significant.

2. **Adding a Constant**: `sm.add_constant(X)` adds a column of ones to `X`, which represents the intercept in the regression model.
3. **Loop Until Conditions Are Met**: The function uses a loop that continues until all predictors have a VIF less than the given threshold (default is 5.0) and all have p-values less than the specified threshold (default is 0.05). These conditions ensure that the remaining predictors are not only significant but also do not excessively inflate each other's variance.
4. **Fit the Model**: In each iteration, the function fits an Ordinary Least Squares (OLS) regression model using the predictors against the outcome. See [statsmodels.regression.linear_model.OLS]().
5. **Check High p-values and VIFs**: After fitting the model, it checks for predictors with high p-values (indicating they are not statistically significant) and high VIFs (indicating multicollinearity). The p-values are contained within the dictionary [statsmodels.regression.linear_model.OLSResults.pvalues]() which includes the two-tailed p values for the t-stats of the params.
6. **Remove Problematic Predictors**: If any predictors have a high VIF, the one with the highest VIF is removed first since it's the most problematic in terms of multicollinearity. If there are no high VIFs but there are predictors with high p-values, the one with the highest p-value is removed, as it is the least statistically significant.
7. **Repeat**: This process of checking and removing continues until all predictors meet the desired thresholds for both VIF and p-values.
8. **Return Refined Predictors and VIF Data**: Once the loop finishes, the function returns the refined set of predictors and the VIF data for these predictors.

**Code**

```
def backward_elimination_for_vif(X, Y, vif_threshold=5.0, p_value_threshold=0.05):
    def calculate_vif(X):
        """ Helper function to calculate VIFs for features in a given dataset. """
        vif_data = pd.DataFrame()
        vif_data["feature"] = X.columns
        vif_data["VIF"] = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
        return vif_data

    X = sm.add_constant(X)

    vif_data = pd.DataFrame()
    while True:
        # Fit the model
        model = sm.OLS(Y, X).fit()

        # Check for high p-values
        high_p_value = model.pvalues[model.pvalues > p_value_threshold]

        # Calculate VIFs
        vif_data = calculate_vif(X)
        high_vif = vif_data[vif_data["VIF"] > vif_threshold]
```

```
        # Check conditions to remove: high VIF and, if applicable, high p-value
        if high_vif.empty and high_p_value.empty:
            break

        # Prefer to remove high VIF variables first
        if not high_vif.empty:
            # Find the variable with the highest VIF
            feature_to_remove = high_vif.sort_values("VIF", ascending=False).iloc[0]['feature']
        elif not high_p_value.empty:
            # Or remove the least significant variable (highest p-value)
            feature_to_remove = high_p_value.idxmax()

        # Drop the feature with the highest VIF or p-value
        X = X.drop(columns=[feature_to_remove])

    return X, vif_data

# Analysis with MonthlyCharge removed
X = df_encoded.drop(columns=["Tenure", "MonthlyCharge"])
vif_data = pd.DataFrame()
vif_data["feature"] = X.columns

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

print(vif_data)

# Drop one dummy category per categorical variable
columns_to_drop = [
    "Area_Urban",
    "Gender_Nonbinary",
    "Contract_Two Year",
    "Marital_Widowed",
    "TimeZone_PST",
    "InternetService_None",
    "PaymentMethod_Mailed Check",
]

X = df_encoded.drop(columns=["Tenure", "MonthlyCharge"] + columns_to_drop)

vif_data = pd.DataFrame()
vif_data["feature"] = X.columns

vif_data["VIF"] = [
    variance_inflation_factor(X.values.astype(float), i) for i in range(X.shape[1])
]

# Backward Elimination
Y = df_encoded["Tenure"]
columns_to_drop = [
    "Area_Urban",
    "Gender_Nonbinary",
    "Contract_Two Year",
    "Marital_Widowed",
    "TimeZone_PST",
    "InternetService_None",
    "PaymentMethod_Mailed Check",
]
X = df_encoded.drop(columns=["Tenure", "MonthlyCharge"] + columns_to_drop)
```

```
X_optimal, vif_data = backward_elimination_for_vif(
    X=X.astype(float), Y=df_encoded["Tenure"]
)

print(vif_data)
```

**Result:**

```
                                  feature       VIF
0                              Population  1.452544
1                                Children  1.898591
2                                  Income  2.787244
3                                Contacts  1.951215
4                                  Techie  1.195238
5                              Port_modem  1.882926
6                                Multiple  1.800834
7                          OnlineSecurity  1.542545
8                            OnlineBackup  1.787449
9                        DeviceProtection  1.747709
10                            TechSupport  1.565799
11                            StreamingTV  1.916186
12                        StreamingMovies  1.921011
13                       PaperlessBilling  2.320493
14                       Bandwidth_GB_Year  3.242733
15                          Area_Suburban  1.477781
16                            Gender_Male  1.860279
17                        Marital_Married  1.427135
18                  Marital_Never Married  1.439136
19                      Marital_Separated  1.447942
20                           TimeZone_AST  1.037323
21                           TimeZone_CST  3.542379
22                           TimeZone_EST  4.103851
23                          TimeZone_HAST  1.024899
24                           TimeZone_MST  1.494840
25            InternetService_Fiber Optic  1.757166
26  PaymentMethod_Credit Card (automatic)  1.434189
27         PaymentMethod_Electronic Check  1.700449
```

The result of this code is a feature set which includes 28 features. The features in this set have a variance inflation factor that is less than 5.0.

### D3. Reduced Linear Regression Model

Using the reduced feature selection, we can generate a reduced linear regression model.

**Code**

```
# D3: Create a reduced model
model_reduced = sm.OLS(Y, sm.add_constant(X_optimal).astype(float))
results_reduced = model_reduced.fit()
print(results_reduced.summary())
```

**Result:**

```
                            OLS Regression Results
==============================================================================
```

```
Dep. Variable:                 Tenure   R-squared:                       0.994
Model:                            OLS   Adj. R-squared:                  0.994
Method:                 Least Squares   F-statistic:                 5.986e+04
Date:                Wed, 29 May 2024   Prob (F-statistic):               0.00
Time:                        18:39:06   Log-Likelihood:                -21287.
No. Observations:               10000   AIC:                         4.263e+04
Df Residuals:                    9971   BIC:                         4.284e+04
Df Model:                          28
Covariance Type:            nonrobust
==========================================================================================================
                                           coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------------------------
const                                   -2.3704      0.112    -21.182      0.000      -2.590      -2.151
Population                           -5.606e-07   1.44e-06     -0.389      0.697   -3.39e-06    2.27e-06
Children                                -0.3803      0.009    -40.035      0.000      -0.399      -0.362
Income                               -8.779e-07   7.23e-07     -1.214      0.225    -2.3e-06    5.39e-07
Contacts                                -0.0171      0.021     -0.829      0.407      -0.058       0.023
Techie                                  -0.0696      0.055     -1.276      0.202      -0.177       0.037
Port_modem                               0.0287      0.041      0.705      0.481      -0.051       0.109
Multiple                                -0.9630      0.041    -23.555      0.000      -1.043      -0.883
OnlineSecurity                          -0.9924      0.043    -23.319      0.000      -1.076      -0.909
OnlineBackup                            -1.1314      0.041    -27.596      0.000      -1.212      -1.051
DeviceProtection                        -0.9897      0.041    -24.078      0.000      -1.070      -0.909
TechSupport                             -0.0579      0.042     -1.375      0.169      -0.140       0.025
StreamingTV                             -2.7639      0.041    -67.696      0.000      -2.844      -2.684
StreamingMovies                         -2.5412      0.041    -62.259      0.000      -2.621      -2.461
PaperlessBilling                         0.0226      0.041      0.545      0.586      -0.059       0.104
Bandwidth_GB_Year                        0.0121   9.38e-06   1293.041      0.000       0.012       0.012
Area_Suburban                           -0.0097      0.043     -0.226      0.822      -0.094       0.075
Gender_Male                             -0.8119      0.041    -19.875      0.000      -0.892      -0.732
Marital_Married                         -0.0466      0.056     -0.825      0.409      -0.157       0.064
Marital_Never Married                   -0.0358      0.056     -0.639      0.523      -0.146       0.074
Marital_Separated                       -0.0155      0.055     -0.279      0.780      -0.124       0.093
TimeZone_AST                            -0.1999      0.330     -0.606      0.544      -0.846       0.446
TimeZone_CST                            -0.0799      0.075     -1.066      0.286      -0.227       0.067
TimeZone_EST                            -0.0165      0.073     -0.226      0.821      -0.159       0.126
TimeZone_HAST                           -0.4896      0.351     -1.395      0.163      -1.177       0.198
TimeZone_MST                             0.0577      0.101      0.569      0.569      -0.141       0.256
InternetService_Fiber Optic              3.1128      0.041     75.606      0.000       3.032       3.194
PaymentMethod_Credit Card (automatic)    0.0540      0.054      1.000      0.317      -0.052       0.160
PaymentMethod_Electronic Check           0.0525      0.046      1.134      0.257      -0.038       0.143
==============================================================================
Omnibus:                      570.059   Durbin-Watson:                   1.971
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              505.877
Skew:                           0.485   Prob(JB):                    1.41e-110
Kurtosis:                       2.477   Cond. No.                     8.62e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.62e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

When generating this model, a warning of multicollinearity is included, however, I have done my due diligence to check for multicollinearity and can proceed with this model.

### E1: Comparison of Initial and Reduced Models

Shown below are the initial and reduced model outputs.

| **Initial Model Output** |
|---|
| OLS Regression Results |

```
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       1.000
Model:                            OLS   Adj. R-squared:                  1.000
Method:                 Least Squares   F-statistic:                 1.408e+07
Date:                Wed, 29 May 2024   Prob (F-statistic):               0.00
Time:                        18:36:00   Log-Likelihood:                 8139.9
No. Observations:               10000   AIC:                        -1.619e+04
Df Residuals:                    9956   BIC:                        -1.587e+04
Df Model:                          43
Covariance Type:            nonrobust
==========================================================================================================
                                           coef    std err          t      P>|t|      [0.025      0.975]
----------------------------------------------------------------------------------------------------------
const                                   -3.8535      0.018   -210.202      0.000      -3.889      -3.818
Population                            -6.25e-08   7.64e-08     -0.818      0.413   -2.12e-07    8.72e-08
Children                                -0.3755      0.001   -748.564      0.000      -0.377      -0.375
Age                                      0.0400   5.21e-05    768.016      0.000       0.040       0.040
Income                                1.316e-08   3.82e-08      0.345      0.730   -6.16e-08     8.8e-08
Outage_sec_perweek                       0.0003      0.000      0.770      0.442      -0.000       0.001
Email                                -6.667e-05      0.000     -0.187      0.851      -0.001       0.001
Contacts                                -0.0006      0.001     -0.594      0.553      -0.003       0.001
Yearly_equip_failure                     0.0001      0.002      0.081      0.935      -0.003       0.003
Techie                                   0.0002      0.003      0.085      0.932      -0.005       0.006
Port_modem                               0.0026      0.002      1.199      0.230      -0.002       0.007
Tablet                                   0.0007      0.002      0.293      0.770      -0.004       0.005
Phone                                    0.0015      0.004      0.418      0.676      -0.006       0.009
Multiple                                 0.2685      0.005     59.177      0.000       0.260       0.277
OnlineSecurity                          -0.8312      0.002   -365.966      0.000      -0.836      -0.827
OnlineBackup                            -0.3553      0.004   -101.255      0.000      -0.362      -0.348
DeviceProtection                        -0.5970      0.003   -224.761      0.000      -0.602      -0.592
TechSupport                              0.3850      0.003    142.418      0.000       0.380       0.390
StreamingTV                             -1.2983      0.006   -232.041      0.000      -1.309      -1.287
StreamingMovies                         -0.7227      0.007   -106.958      0.000      -0.736      -0.709
PaperlessBilling                        -0.0036      0.002     -1.665      0.096      -0.008       0.001
MonthlyCharge                           -0.0352      0.000   -287.578      0.000      -0.035      -0.035
Bandwidth_GB_Year                        0.0122   4.97e-07   2.46e+04      0.000       0.012       0.012
Area_Suburban                           -0.0069      0.003     -2.622      0.009      -0.012      -0.002
Area_Urban                              -0.0033      0.003     -1.268      0.205      -0.009       0.002
Gender_Male                             -0.7925      0.002   -363.483      0.000      -0.797      -0.788
Gender_Nonbinary                         0.2616      0.007     36.118      0.000       0.247       0.276
Contract_One year                        0.0006      0.003      0.225      0.822      -0.005       0.006
Contract_Two Year                        0.0019      0.003      0.726      0.468      -0.003       0.007
Marital_Married                         -0.0004      0.003     -0.124      0.901      -0.007       0.006
Marital_Never Married                   -0.0010      0.003     -0.293      0.769      -0.008       0.006
Marital_Separated                        0.0028      0.003      0.845      0.398      -0.004       0.009
Marital_Widowed                          0.0001      0.003      0.039      0.969      -0.006       0.007
TimeZone_AST                             0.0103      0.021      0.488      0.626      -0.031       0.052
TimeZone_CST                             0.0135      0.012      1.092      0.275      -0.011       0.038
TimeZone_EST                             0.0098      0.012      0.790      0.429      -0.014       0.034
TimeZone_HAST                           -0.0009      0.022     -0.042      0.967      -0.044       0.042
TimeZone_MST                             0.0146      0.013      1.131      0.258      -0.011       0.040
TimeZone_PST                             0.0099      0.013      0.774      0.439      -0.015       0.035
InternetService_Fiber Optic              5.7538      0.003   1665.144      0.000       5.747       5.761
InternetService_None                     4.6005      0.003   1367.056      0.000       4.594       4.607
PaymentMethod_Credit Card (automatic)    0.0019      0.003      0.586      0.558      -0.005       0.008
PaymentMethod_Electronic Check           0.0029      0.003      0.992      0.321      -0.003       0.009
PaymentMethod_Mailed Check               0.0071      0.003      2.225      0.026       0.001       0.013
==============================================================================
Omnibus:                    34814.468   Durbin-Watson:                   2.003
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1633.922
Skew:                          -0.034   Prob(JB):                         0.00
Kurtosis:                       1.021   Cond. No.                     1.59e+06
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.59e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

**Reduced Model Output**

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                 Tenure   R-squared:                       0.994
Model:                            OLS   Adj. R-squared:                  0.994
Method:                 Least Squares   F-statistic:                 5.986e+04
Date:                Wed, 29 May 2024   Prob (F-statistic):               0.00
Time:                        18:39:06   Log-Likelihood:                -21287.
No. Observations:               10000   AIC:                         4.263e+04
Df Residuals:                    9971   BIC:                         4.284e+04
Df Model:                          28                                         
Covariance Type:            nonrobust                                         
=========================================================================================================
                                            coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------------------------
const                                    -2.3704      0.112    -21.182      0.000      -2.590      -2.151
Population                            -5.606e-07   1.44e-06     -0.389      0.697   -3.39e-06    2.27e-06
Children                                 -0.3803      0.009    -40.035      0.000      -0.399      -0.362
Income                                -8.779e-07   7.23e-07     -1.214      0.225    -2.3e-06    5.39e-07
Contacts                                 -0.0171      0.021     -0.829      0.407      -0.058       0.023
Techie                                   -0.0696      0.055     -1.276      0.202      -0.177       0.037
Port_modem                                0.0287      0.041      0.705      0.481      -0.051       0.109
Multiple                                 -0.9630      0.041    -23.555      0.000      -1.043      -0.883
OnlineSecurity                           -0.9924      0.043    -23.319      0.000      -1.076      -0.909
OnlineBackup                             -1.1314      0.041    -27.596      0.000      -1.212      -1.051
DeviceProtection                         -0.9897      0.041    -24.078      0.000      -1.070      -0.909
TechSupport                              -0.0579      0.042     -1.375      0.169      -0.140       0.025
StreamingTV                              -2.7639      0.041    -67.696      0.000      -2.844      -2.684
StreamingMovies                          -2.5412      0.041    -62.259      0.000      -2.621      -2.461
PaperlessBilling                          0.0226      0.041      0.545      0.586      -0.059       0.104
Bandwidth_GB_Year                         0.0121   9.38e-06   1293.041      0.000       0.012       0.012
Area_Suburban                            -0.0097      0.043     -0.226      0.822      -0.094       0.075
Gender_Male                              -0.8119      0.041    -19.875      0.000      -0.892      -0.732
Marital_Married                          -0.0466      0.056     -0.825      0.409      -0.157       0.064
Marital_Never Married                    -0.0358      0.056     -0.639      0.523      -0.146       0.074
Marital_Separated                        -0.0155      0.055     -0.279      0.780      -0.124       0.093
TimeZone_AST                             -0.1999      0.330     -0.606      0.544      -0.846       0.446
TimeZone_CST                             -0.0799      0.075     -1.066      0.286      -0.227       0.067
TimeZone_EST                             -0.0165      0.073     -0.226      0.821      -0.159       0.126
TimeZone_HAST                            -0.4896      0.351     -1.395      0.163      -1.177       0.198
TimeZone_MST                              0.0577      0.101      0.569      0.569      -0.141       0.256
InternetService_Fiber Optic               3.1128      0.041     75.606      0.000       3.032       3.194
PaymentMethod_Credit Card (automatic)     0.0540      0.054      1.000      0.317      -0.052       0.160
PaymentMethod_Electronic Check            0.0525      0.046      1.134      0.257      -0.038       0.143
==============================================================================
Omnibus:                      570.059   Durbin-Watson:                   1.971
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              505.877
Skew:                           0.485   Prob(JB):                    1.41e-110
Kurtosis:                       2.477   Cond. No.                     8.62e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.62e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

The initial model and reduced model vary in multiple model evaluation metrics:

- **Complexity**: The initial model includes 43 predictors whereas the reduced model includes 28. More complex models with a larger number of predictors require more computational resources. This simplification allows predictions to be made while using less system resources.

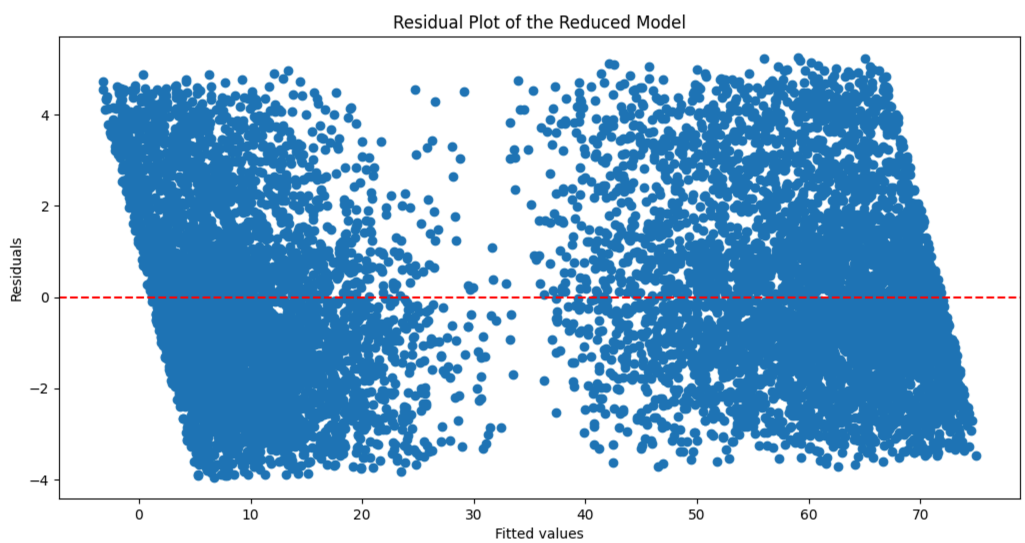
- **R-squared**: The initial model has an R-squared value of 1.000 whereas the reduced model maintains an R-squared value of 0.994. This indicates that even after simplification, the model retains most of its accuracy. Tatachar (2021) clarifies, "R-Squared (R2) – R2 is called the coefficient of Determination. R-Squared determines the proportion of variance in the dependent variable that can be explained by the independent variables."

### E2. Residual Plot and RSE

The following code generates a residual plot of the reduced model. "Linearity is examined through scatter plots, residual plot and remedy way is the transforming variables," Thu (2019) outlines.

| Code |
|---|

```
# Residual Plot for the Reduced Model
plt.figure(figsize=(12, 6))
plt.scatter(results_reduced.fittedvalues, results_reduced.resid)
plt.axhline(y=0, color="red", linestyle="--")
plt.xlabel("Fitted values")
plt.ylabel("Residuals")
plt.title("Residual Plot of the Reduced Model")
plt.show()
```




The following code calculates the Residual Standard Error (RSE) of the reduced model.

| Code |
|---|

```
# Calculate and print the Residual Standard Error
RSE = np.sqrt(results_reduced.scale)
print(f"Residual Standard Error (RSE) of the Reduced Model: {RSE:.3f}")
```

| Result: |
| --- |
| `Residual Standard Error (RSE) of the Reduced Model: 2.036` |

The Residual Standard Error (RSE) of the reduced model is 2.036. This indicates that, on average, the actual customer tenure deviates from the predicted customer tenure by 2.036 months. The histogram created in C3 shows that customer tenure ranges between 0-70 months, and therefore, for business use cases this deviation is somewhat significant.

### E3. Code

See main.py

### F1. Results of Data Analysis

The result of my data analysis process is the creation of a reduced model which provides a clearer insight as to which variables influence customer tenure most heavily. Shown below is the regression equation for the reduced model:

$$\text{Tenure} = -2.3704 - 5.606\times10^{-7}\cdot\text{Population} - 0.3803\cdot\text{Children} - 8.779\times10^{-7}\cdot\text{Income} - 0.0171\cdot\text{Contacts} - 0.0696\cdot\text{Techie} + 0.0287\cdot\text{Port\_modem} - 0.9630\cdot\text{Multiple} - 0.9924\cdot\text{OnlineSecurity} - 1.1314\cdot\text{OnlineBackup} - 0.9897\cdot\text{DeviceProtection} - 0.0579\cdot\text{TechSupport} - 2.7639\cdot\text{StreamingTV} - 2.5412\cdot\text{StreamingMovies} + 0.0226\cdot\text{PaperlessBilling} + 0.0121\cdot\text{Bandwidth\_GB\_Year} - 0.0097\cdot\text{Area\_Suburban} - 0.8119\cdot\text{Gender\_Male} - 0.0466\cdot\text{Marital\_Married} - 0.0358\cdot\text{Marital\_Never Married} - 0.0155\cdot\text{Marital\_Separated} - 0.1999\cdot\text{TimeZone\_AST} - 0.0799\cdot\text{TimeZone\_CST} - 0.0165\cdot\text{TimeZone\_EST} - 0.4896\cdot\text{TimeZone\_HAST} + 0.0577\cdot\text{TimeZone\_MST} + 3.1128\cdot\text{InternetService\_Fiber Optic} + 0.0540\cdot\text{PaymentMethod\_Credit Card (automatic)} + 0.0525\cdot\text{PaymentMethod\_Electronic Check}$$

This equations tells us that:

- Constant: When all other variables are zero, the Tenure is -2.3704 months.
- Population: Keeping all other variables constant, an increase of 1 in Population will decrease the Tenure by approximately 5.606×10^-7 months.

- Children: Keeping all other variables constant, an increase of 1 in the number of Children will decrease the Tenure by 0.3803 months.

- Income: Keeping all other variables constant, an increase of 1 in Income will decrease the Tenure by approximately $8.779 \times 10^{-7}$ months.

- Contacts: Keeping all other variables constant, an increase of 1 in Contacts will decrease the Tenure by 0.0171 months.

- Techie: Keeping all other variables constant, an increase of 1 in Techie presence will decrease the Tenure by 0.0696 months.

- Port Modem: Keeping all other variables constant, an increase of 1 in Port Modem usage will increase the Tenure by 0.0287 months.

- Multiple: Keeping all other variables constant, an increase of 1 in Multiple line usage will decrease the Tenure by 0.9630 months.

- Online Security: Keeping all other variables constant, an increase of 1 in Online Security will decrease the Tenure by 0.9924 months.

- Online Backup: Keeping all other variables constant, an increase of 1 in Online Backup will decrease the Tenure by 1.1314 months.

- Device Protection: Keeping all other variables constant, an increase of 1 in Device Protection will decrease the Tenure by 0.9897 months.

- Tech Support: Keeping all other variables constant, an increase of 1 in Tech Support will decrease the Tenure by 0.0579 months.

- Streaming TV: Keeping all other variables constant, an increase of 1 in Streaming TV will decrease the Tenure by 2.7639 months.

- Streaming Movies: Keeping all other variables constant, an increase of 1 in Streaming Movies will decrease the Tenure by 2.5412 months.

- Paperless Billing: Keeping all other variables constant, an increase of 1 in Paperless Billing will increase the Tenure by 0.0226 months.

- Bandwidth GB Year: Keeping all other variables constant, an increase of 1 in Bandwidth GB Year will increase the Tenure by 0.0121 months.

- Area Suburban: Keeping all other variables constant, being in a Suburban area will decrease the Tenure by 0.0097 months.

- Gender Male: Keeping all other variables constant, being Male will decrease the Tenure by 0.8119 months.

- Marital Married: Keeping all other variables constant, being Married will decrease the Tenure by 0.0466 months.

- Marital Never Married: Keeping all other variables constant, being Never Married will decrease the Tenure by 0.0358 months.

- Marital Separated: Keeping all other variables constant, being Separated will decrease the Tenure by 0.0155 months.

- TimeZone AST: Keeping all other variables constant, being in the AST time zone will decrease the Tenure by 0.1999 months.

- TimeZone CST: Keeping all other variables constant, being in the CST time zone will decrease the Tenure by 0.0799 months.

- TimeZone EST: Keeping all other variables constant, being in the EST time zone will decrease the Tenure by 0.0165 months.

- TimeZone HAST: Keeping all other variables constant, being in the HAST time zone will decrease the Tenure by 0.4896 months.

- TimeZone MST: Keeping all other variables constant, being in the MST time zone will increase the Tenure by 0.0577 months.

- Internet Service Fiber Optic: Keeping all other variables constant, having Fiber Optic as the Internet Service will increase the Tenure by 3.1128 months.

- Payment Method Credit Card (automatic): Keeping all other variables constant, using Credit Card (automatic) as a Payment Method will increase the Tenure by 0.0540 months.

- Payment Method Electronic Check: Keeping all other variables constant, using Electronic Check as a Payment Method will increase the Tenure by 0.0525 months.

This model is significant because it identifies which customer factors have the greatest impact on customer tenure and uses a simplified model to determine this value. Wheeler (2013) explains, "In the regression problem you are looking for some function, or combination of functions, of the independent variables that will explain a substantial proportion of the variation in the dependent variable."

One limitation of this data analysis is that, even after reducing the feature set, multicollinearity issues still persist. This can cause the reduced model to be inaccurate as each factor is not isolated, but rather can influence one another, which can lead to inaccurate results.

Additionally, both the initial and reduced models highlight the dependent variable tenure. However, in business use cases, other metrics are important to consider. By maximizing customer tenure, an organization may inadvertently reduce their revenue by focusing on keeping customers retained for as long as possible regardless of how much revenue they generate. For maximizing revenue and, consequently, profit, and organization should take into account the customer’s monthly billing amount.

### F2. Recommended Course of Action

My research question is: “Which customer factors contribute most to a customer’s tenure with the service provider?” By using this reduced model, decision makers can identify which factors contribute most to a customer’s tenure and work to increase those.

As organizations have limited resources, it makes sense for them to focus their efforts on customer factors which have the greatest impact on customer tenure. In the reduced model, the factors StreamingTV, StreamingMovies, OnlineBackup, and DeviceProtection have the greatest absolute coefficients, meaning these factors have the greatest impact on customer tenure.

As these are all boolean variables representing a customer’s choice to use a specific service and their coefficients are negative, an organization could aim to reduce the percentage of customers who use these services.

Assuming the goal of the organization is to maximize customer retention (limitations of this assumption are addressed in the previous section,) the marketing team of the organization should avoid creating marketing campaigns that highlight these service offerings: TV Streaming, Movie Streaming, Online Backup, and Device Protection.

### H. Web Sources

The following web sources were referenced for code documentation and to help me further understand statistical concepts:

Statsmodels docs - https://www.statsmodels.org/stable/generated/statsmodels.regression.linear_model.OLS.html

WGU Course Material - https://westerngovernorsuniversity-my.sharepoint.com/:p:/g/personal/william_sewell_wgu_edu/ER_vJMbYtxJGpxImpZ0DUQcBoVcORYKanFVKNKFcEXkRow?rtime=nf4c7rZ43Eg

Backwards elimination wrapper method – https://www.analyticsvidhya.com/blog/2020/10/a-comprehensive-guide-to-feature-selection-using-wrapper-methods-in-python/

Backward elimination in Python – https://www.javatpoint.com/backward-elimination-in-machine-learning

regress_exog documentation – https://www.statsmodels.org/v0.14.0/_modules/statsmodels/graphics/regressionplots.html

Seaborn color palettes – https://seaborn.pydata.org/tutorial/color_palettes.html

Regression equation – https://sixsigmadsi.com/glossary/regression-equation/

statsmodels.regression.linear_model.OLSResults.pvalues Documentation – https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLSResults.pvalues.html

### I. Works Consulted

Dekkati, S. (2021). Python Programming Language for Data-Driven Web Applications. International Journal of Reciprocal Symmetry and Theoretical Physics, 8, 1-10.

Tatachar, A. V. (2021). Comparative assessment of regression models based on model evaluation metrics. International Research Journal of Engineering and Technology (IRJET), 8(09), 2395-0056.

Thu, M. (2019). The Violation for assumptions of multiple regression model (Doctoral dissertation, Yangon Univarsity of Economics).

Wheeler, D. J. (2013). Should the Residuals be Normal?.