D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Exploratory Data Analysis Report (EDA)
On
Telecom Customer Churn Prediction

Subject: Big Data Analysis

Name: Sandip Verma

PRN: 20240804044

(Under the guidance of)

Dr. Maheshwari Biradar
Course Faculty

CONTENT TABLE

1. Introduction and Overview

Customer churn is a major concern in the **telecom industry**, as retaining customers is more cost - effective than acquiring new ones. The **Telecom Customer Churn dataset** provides valuable insights into customer behaviour, helping telecom providers develop better retention strategies.

This **Exploratory Data Analysis (EDA) Report** aims to uncover patterns and key factors influencing customer churn. By analyzing customer demographics, account details, service usage, and billing information, we can identify trends that impact customer retention.

The **Telco Customer Churn dataset** contains **7,043 entries** with **21 attributes** related to customer behaviour and subscription details. The dataset includes:

- **Demographic Data:** Gender, Senior Citizen, Partner, and Dependents
- **Account Information:** Tenure, Contract Type, Payment Method
- **Subscription Details:** Internet Service, Streaming Services, Phone Service
- **Billing Information:** Monthly Charges, Total Charges
- **Target Variable: Churn (Yes/No)** indicating whether a customer left the service

2. Data Collection and Description

2.1 Data Collection

The dataset used in this analysis is the **Telco Customer Churn dataset**, sourced from **Kaggle**. It contains **customer subscription details** and their churn status, which helps identify factors contributing to customer retention and churn.

- **Source:** Kaggle - Telco Customer Churn Dataset
- **Dataset Format:** CSV (Comma-Separated Values)
- **Number of Records:** 7,043 customers
- **Number of Features:** 21 columns

2.2 Data Description

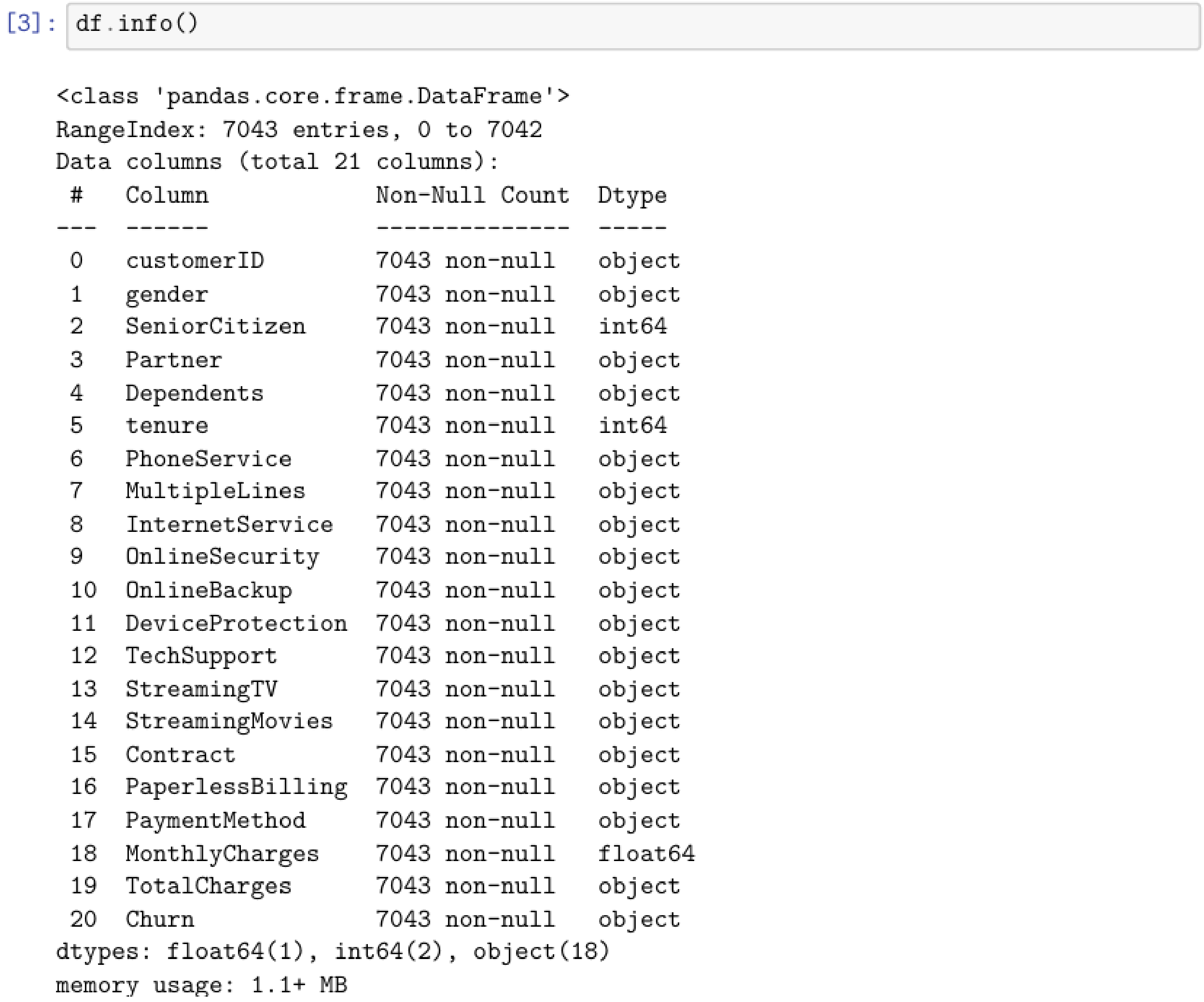
The dataset consists of **customer demographic details, subscription information, billing details, and churn status**. Below is a summary of the key variables:

Column Name	Description	Type
customerID	Unique identifier for each customer	Categorical
gender	Gender of the customer (Male/Female)	Categorical
SeniorCitizen	Whether the customer is a senior citizen (0/1)	Numerical
Partner	Whether the customer has a partner (Yes/No)	Categorical
Dependents	Whether the customer has dependents (Yes/No)	Categorical
tenure	Number of months the customer has stayed	Numerical
PhoneService	Whether the customer has phone service (Yes/No)	Categorical
MultipleLines	Whether the customer has multiple lines (Yes/No/No phone service)	Categorical
InternetService	Type of internet service (DSL, Fiber optic, No)	Categorical
OnlineSecurity	Whether the customer has online security (Yes/No/No internet)	Categorical

OnlineBackup	Whether the customer has online backup (Yes/No/No internet)	Categorical
DeviceProtection	Whether the customer has device protection (Yes/No/No internet)	Categorical
TechSupport	Whether the customer has tech support (Yes/No/No internet)	Categorical
StreamingTV	Whether the customer has streaming TV (Yes/No/No internet)	Categorical
StreamingMovies	Whether the customer has streaming movies (Yes/No/No internet)	Categorical
Contract	Type of contract (Month-to-month, One year, Two year)	Categorical
PaperlessBilling	Whether the customer has paperless billing (Yes/No)	Categorical
PaymentMethod	Payment method (Electronic check, Mailed check, Bank transfer, Credit card)	Categorical
MonthlyCharges	The amount charged per month	Numerical
TotalCharges	The total amount charged during tenure	Numerical
Churn	Whether the customer churned (Yes/No)	Categorical

3. Data Cleaning and Preprocessing

Before beginning the analysis, it is essential to clean the dataset and handle issues such as missing values, incorrect data types, and irrelevant or redundant features.



```
[3]: df.info()
```

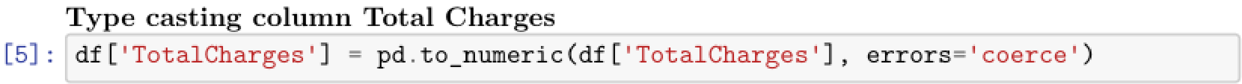
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   customerID        7043 non-null   object 
 1   gender            7043 non-null   object 
 2   SeniorCitizen     7043 non-null   int64  
 3   Partner           7043 non-null   object 
 4   Dependents        7043 non-null   object 
 5   tenure            7043 non-null   int64  
 6   PhoneService      7043 non-null   object 
 7   MultipleLines     7043 non-null   object 
 8   InternetService   7043 non-null   object 
 9   OnlineSecurity    7043 non-null   object 
 10  OnlineBackup      7043 non-null   object 
 11  DeviceProtection  7043 non-null   object 
 12  TechSupport       7043 non-null   object 
 13  StreamingTV       7043 non-null   object 
 14  StreamingMovies   7043 non-null   object 
 15  Contract          7043 non-null   object 
 16  PaperlessBilling  7043 non-null   object 
 17  PaymentMethod     7043 non-null   object 
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object 
 20  Churn             7043 non-null   object 
dtypes: float64(1), int64(2), object(18)
memory usage: 1.1+ MB
```

3.1 Handling Missing Values

The dataset appears **complete**, but some values in the **TotalCharges** column are stored as empty strings, which need to be converted to numeric values.

Steps Taken:

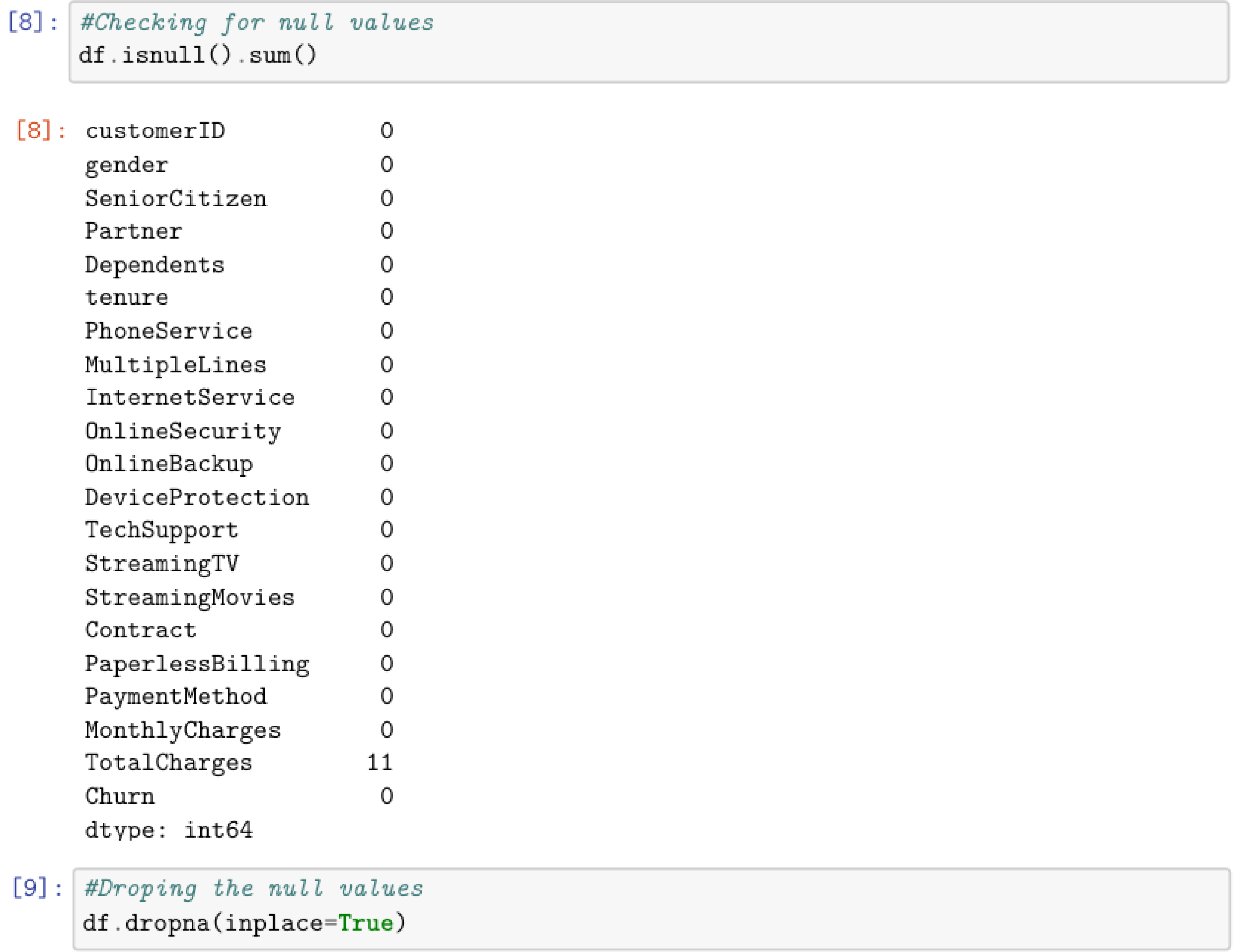
- Converted empty values in **TotalCharges** to NaN and imputed them with the median.



Type casting column Total Charges

```
[5]: df['TotalCharges'] = pd.to_numeric(df['TotalCharges'], errors='coerce')
```

- Verified that no missing values remain in the dataset.



```
[8]: #Checking for null values
     df.isnull().sum()
```

```
[8]: customerID           0
     gender               0
     SeniorCitizen        0
     Partner              0
     Dependents           0
     tenure               0
     PhoneService         0
     MultipleLines        0
     InternetService      0
     OnlineSecurity       0
     OnlineBackup         0
     DeviceProtection     0
     TechSupport          0
     StreamingTV          0
     StreamingMovies      0
     Contract             0
     PaperlessBilling     0
     PaymentMethod        0
     MonthlyCharges       0
     TotalCharges        11
     Churn                0
     dtype: int64
```

```
[9]: #Droping the null values
     df.dropna(inplace=True)
```

3.2 Handling Duplicates

Since the dataset contains a **unique customerID**, no duplicate records are expected. We verified and confirmed there are **no duplicate rows**.

```
[11]: #Removing the customerID column
      df.drop(columns='customerID', inplace=True)
```

```
[12]: #Checking for duplicate values
      df.duplicated().sum()
```

```
[12]: 22
```

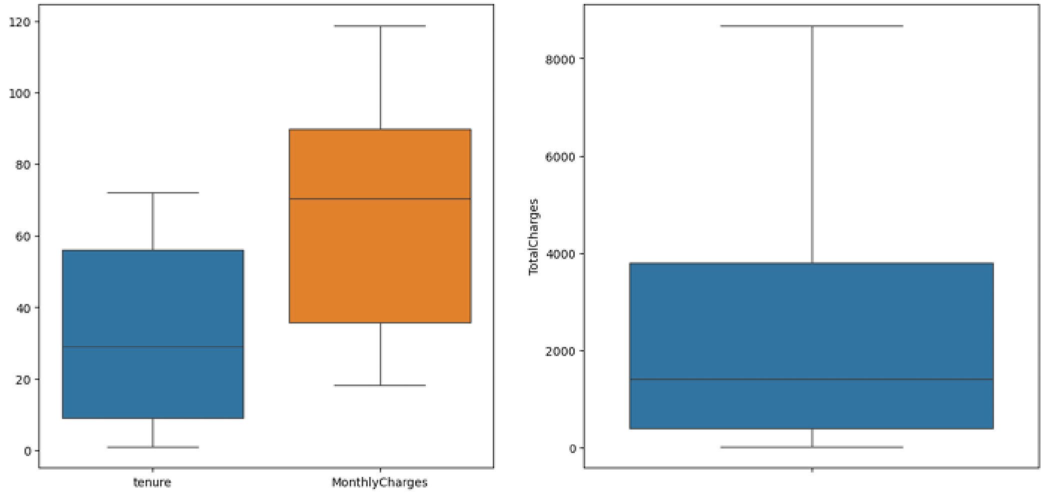
```
[13]: #Removing the duplicate values
      df.drop_duplicates(inplace=True)
```

```
[14]: df.duplicated().sum()
```

```
[14]: 0
```

3.3 Outlier Removal

```
[20]: plt.figure(figsize=(15,7))

      plt.subplot(1,2,1)
      sns.boxplot(df.drop(columns=['TotalCharges','SeniorCitizen']))

      plt.subplot(1,2,2)
      sns.boxplot(df['TotalCharges'])

      plt.show()
```



[21]:

```
df.shape
```

[21]:

```
(7010, 20)
```

[22]:

```
#Columns for outlier removal
cols = ['tenure', 'MonthlyCharges', 'TotalCharges']

#Using IQR method to remove outliers
Q1 = df[cols].quantile(0.25)
Q3 = df[cols].quantile(0.75)
IQR = Q3 - Q1

#Removing the outliers
# df = df[(df[cols] < (Q1 - 1.5 * IQR)) |(df[cols] > (Q3 + 1.5 * IQR))]
df = df[~((df[cols] < (Q1 - 1.5 * IQR)) |(df[cols] > (Q3 + 1.5 * IQR))).
 ↪any(axis=1)]
```

[23]:

```
df.shape
```

[23]:

```
(7010, 20)
```

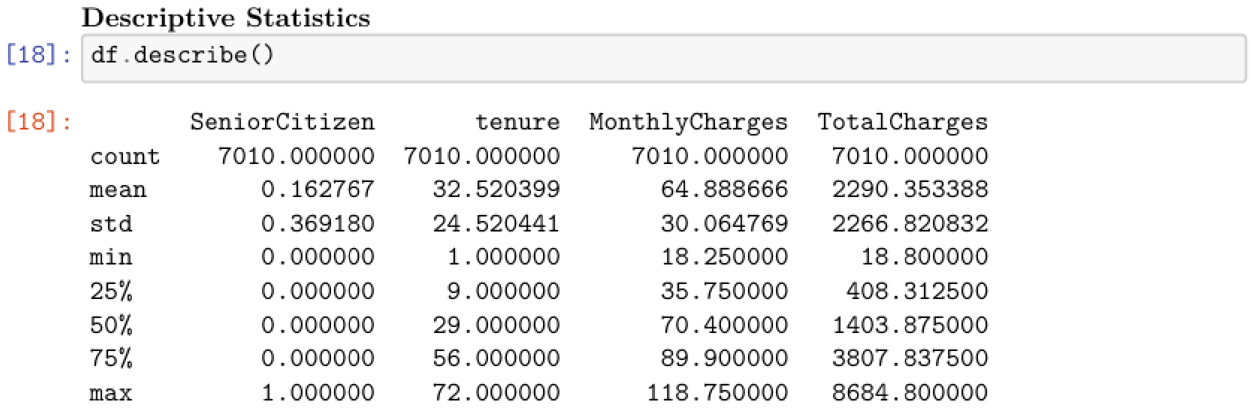
There is no outliers in the dataset.

4. Data Analysis and Data Visualization

This section explores the **Telco Customer Churn dataset** to identify key trends and patterns. We analyze the distribution of variables, relationships between features, and their impact on customer churn.

4.1 Summary Statistics

A quick look at the numerical features provides insights into the dataset's structure and spread.



Descriptive Statistics

```
[18]: df.describe()
```

```
[18]:        SeniorCitizen       tenure  MonthlyCharges  TotalCharges
      count    7010.000000  7010.000000     7010.000000   7010.000000
      mean        0.162767    32.520399       64.888666   2290.353388
      std         0.369180    24.520441       30.064769   2266.820832
      min         0.000000     1.000000       18.250000     18.800000
      25%         0.000000     9.000000       35.750000    408.312500
      50%         0.000000    29.000000       70.400000   1403.875000
      75%         0.000000    56.000000       89.900000   3807.837500
      max         1.000000    72.000000      118.750000   8684.800000
```

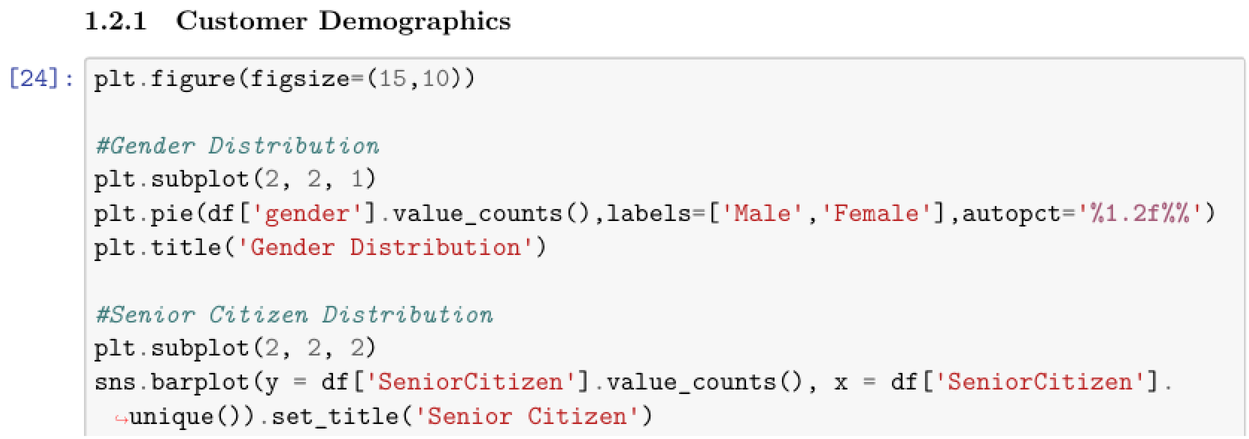
Key Observations:

- The **tenure** ranges from **1 to 72 months**, indicating both new and long-term customers.
- The **MonthlyCharges** vary between **$18.25 and $118.75**, showing different subscription levels.
- The **TotalCharges** column initially had missing values, which were handled earlier.

In the exploratory data analysis, I will be visualizing the data to get a better understanding of the data and to see if there are any trends or pattern in the data.

4.2 Univariate Analysis

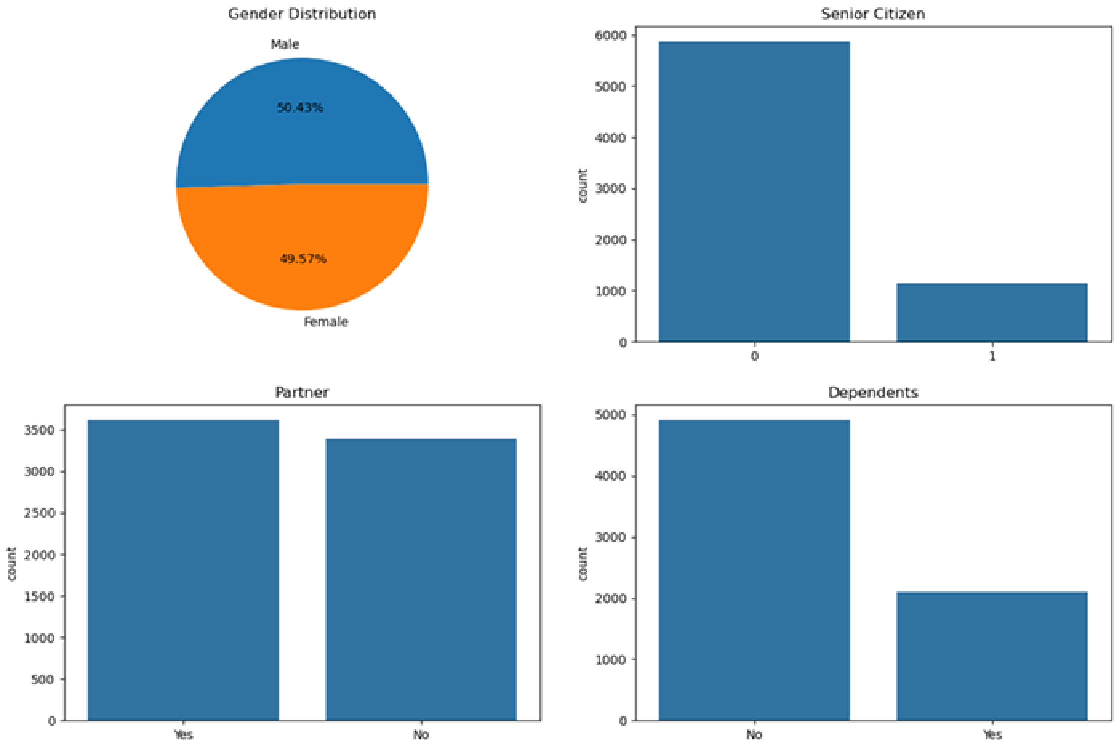
First, I will begin with looking at the distribution of the data and then I will look at the relationship between the independent variables and the target variable.



1.2.1 Customer Demographics

```
[24]: plt.figure(figsize=(15,10))

      #Gender Distribution
      plt.subplot(2, 2, 1)
      plt.pie(df['gender'].value_counts(),labels=['Male','Female'],autopct='%1.2f%%')
      plt.title('Gender Distribution')

      #Senior Citizen Distribution
      plt.subplot(2, 2, 2)
      sns.barplot(y = df['SeniorCitizen'].value_counts(), x = df['SeniorCitizen'].
       ↪unique()).set_title('Senior Citizen')
```

```
#Partner Distribution
plt.subplot(2, 2, 3)
sns.barplot(y = df['Partner'].value_counts(), x = df['Partner'].unique()).
 ↪set_title('Partner')

#Dependents Distribution
plt.subplot(2, 2, 4)
sns.barplot(y = df['Dependents'].value_counts(), x = df['Dependents'].unique()).
 ↪set_title('Dependents')


plt.show()
```

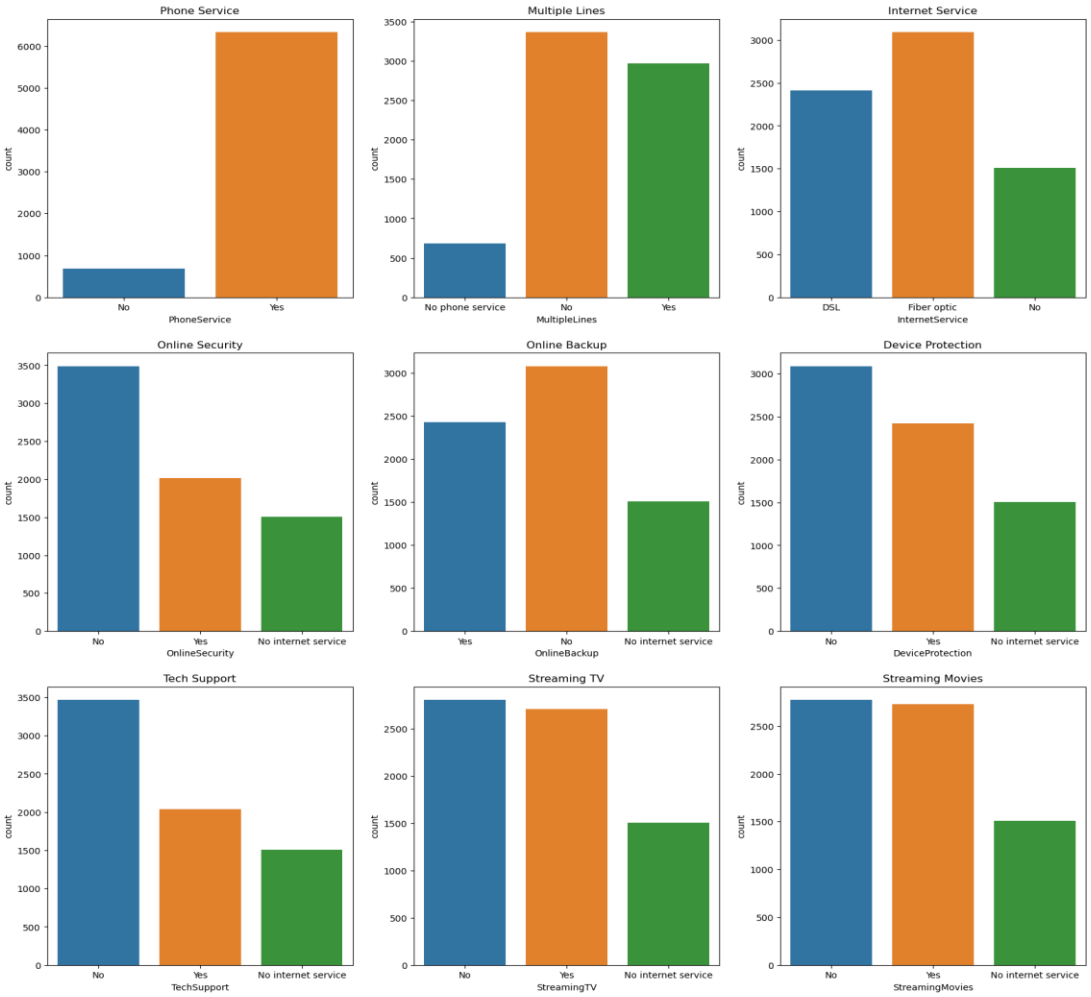


These graphs shows the customer demographics. The number of males and females is almost same, with few more males than females in the dataset. Majority of them are not senior citizen. Nearly 3500, customers have a partner and similar number of customers don't. Majority of the customers don't have dependents, but still a significant number does have dependents. From these graphs, we get know about the customers demographics, which help us to get an idea of their psychology based on their age, relationship status, and dependents.

1.3 Services

[25]:

```
plt.figure(figsize=(20, 20))

#phone service
plt.subplot(3,3,1)
sns.countplot(x=df['PhoneService'],hue=df['PhoneService']).set_title("Phone␣
↪Service")

#Multiple Lines
plt.subplot(3,3,2)
sns.countplot(x=df['MultipleLines'],hue=df['MultipleLines']).
↪set_title("Multiple Lines")

#Internet Service
plt.subplot(3,3,3)
sns.countplot(x=df['InternetService'],hue=df['InternetService']).
↪set_title("Internet Service")

#Online Security
plt.subplot(3,3,4)
sns.countplot(x=df['OnlineSecurity'],hue=df['OnlineSecurity']).
↪set_title("Online Security")

#Online Backup
plt.subplot(3,3,5)
sns.countplot(x=df['OnlineBackup'],hue=df['OnlineBackup']).set_title("Online␣
↪Backup")

#Device Protection
plt.subplot(3,3,6)
sns.countplot(x=df['DeviceProtection'],hue=df['DeviceProtection']).
↪set_title("Device Protection")

#Tech Support
plt.subplot(3,3,7)
sns.countplot(x=df['TechSupport'],hue=df['TechSupport']).set_title("Tech␣
↪Support")

#Streaming TV
plt.subplot(3,3,8)
sns.countplot(x=df['StreamingTV'],hue=df['StreamingTV']).set_title("Streaming␣
↪TV")

#Streaming Movies
plt.subplot(3,3,9)
sns.countplot(x=df['StreamingMovies'],hue=df['StreamingMovies']).
↪set_title("Streaming Movies")

plt.show()
```



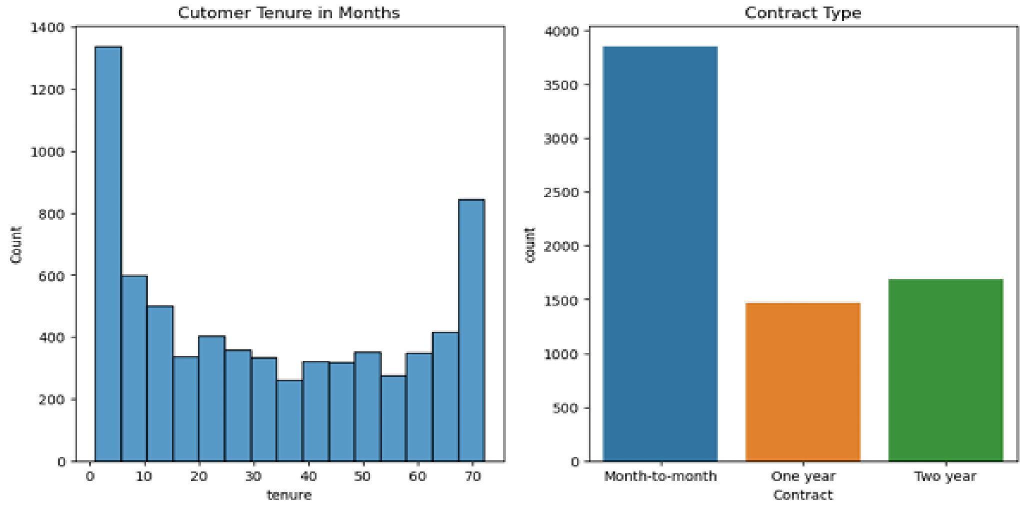
The above graphs visualize the services taken by the customers from the telecom company. Nearly 6000 customers have taken phone service. However, nearly half of the customers have taken multiple lines from the company. Almost 5500, have taken internet services from the company, where 3000 customers opted fibre optics and rest of them opted DSL which could possible for business purposes. From these three major services related to telecom, the phone services and the internet services are the most popular services among the customers.

Coming to other services which includes- Online Security, Online Backup, Device Protection, Tech Support, and Streaming Services. The online backup and device protection service is opted by almost 2500 customers, which highlights the customers concern regarding their device safety and data protection. The online security and tech support is opted by almost 2000 customers which are least opted services among the customers. The streaming services are the most popular services, with more than 2500 customers opting for it.

From this, I conclude that part from the internet and phone services, the streaming services are most opted ones. Therefore, the company should focus on providing better streaming services to the customers.

1.3.1 Tenure and Contract

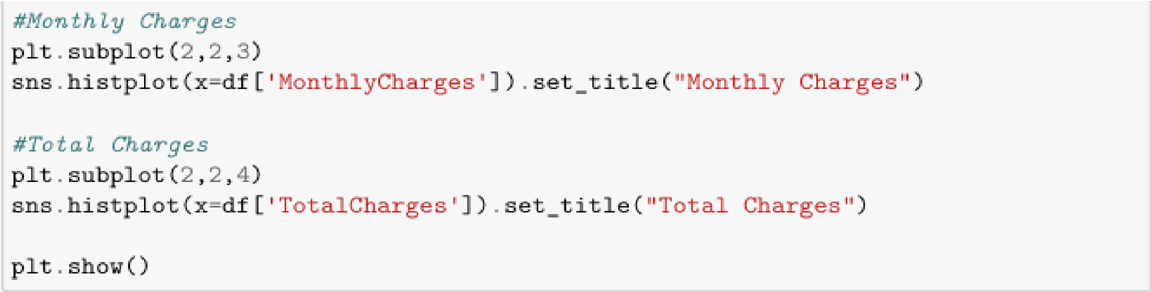
[26]:
```
plt.figure(figsize=(12,6))

plt.subplot(1,2,1)
sns.histplot(x = 'tenure', data = df).set_title('Cutomer Tenure in Months')

plt.subplot(1,2,2)
sns.countplot(x = 'Contract', data = df,hue='Contract').set_title('Contract␣
↪Type')

plt.show()
```



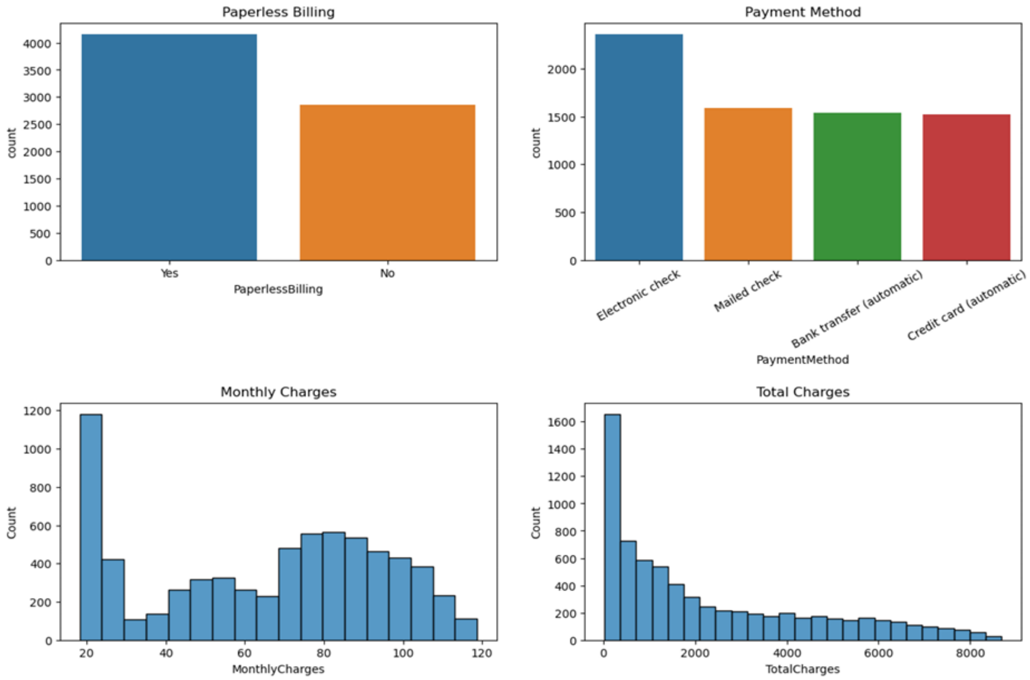
In the above graphs we can see the distribution of customer tenure with the company and the count of the type of contract the company had with customer. Here, most if the customers had tenure less than a month, and most of the customers had a month-to-month contract with the company. Therefore, the customers with shorter tenure have month-to-month contract with the company. In addition to that, a significant number of customers have tenure of nearly 70 months, which highlights the loyalty of the customers towards the company. Moreover, after month-to month contract, the second most popular contract is two-year contract, which is opted by almost 1700 customers. Rest of the customers have tenure between 1-5 years.

1.3.2 Billing and Charges

[27]:
```
plt.figure(figsize=(15,10))
plt.subplots_adjust(hspace=.6)

#papaerless billing
plt.subplot(2,2,1)
sns.countplot(x = df['PaperlessBilling'],hue=df['PaperlessBilling']).
↪set_title('Paperless Billing')

#Payment Method
plt.subplot(2,2,2)
sns.countplot(x = df['PaymentMethod'],hue=df['PaymentMethod']).
↪set_title('Payment Method')
plt.xticks(rotation=30)
```



```
#Monthly Charges
plt.subplot(2,2,3)
sns.histplot(x=df['MonthlyCharges']).set_title("Monthly Charges")

#Total Charges
plt.subplot(2,2,4)
sns.histplot(x=df['TotalCharges']).set_title("Total Charges")

plt.show()
```

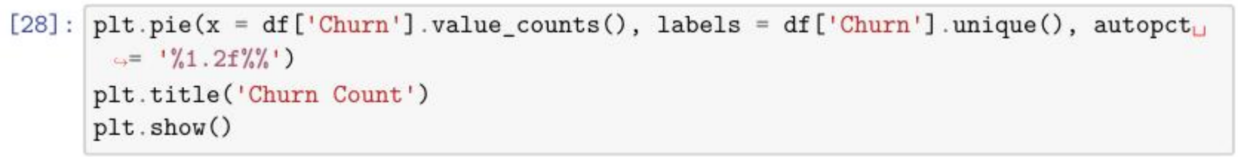


These graphs shows the method of billing and the bill amounts. Most of the customers, nearly 4000 prefer paperless billing, however, a little bit over half of them pays through electronic check. But still a significant number of customers prefer paper bills. Apart from electronic checks, the other modes of payment accepted by the company includes- mailed checks, bank transfer and credit cards. Nearly 4500 customers altogether prefer these modes of payment.

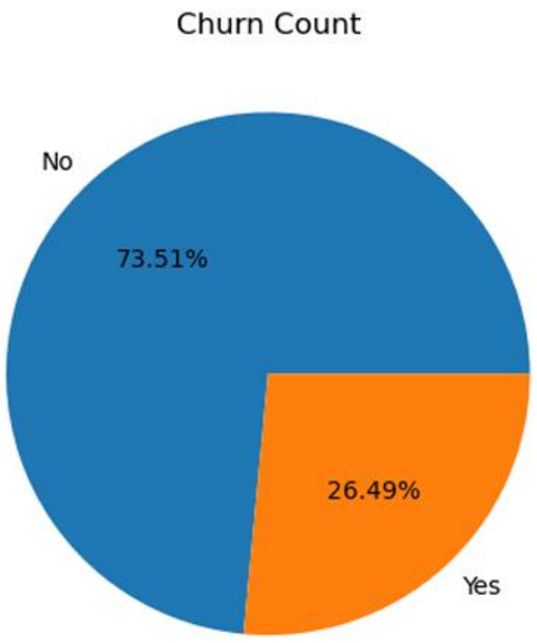
Now, for the monthly charges, huge number of customers pays near 20 dollars for the monthly services and majority of the customer having total charges less than 2000 dollars. However, there are considerable number of customers having monthly charges between 70 to 100 dollars. Interestingly, If we look at the total charges graph, we can see that some customers have a total bill more than 4000 and even 8000 as well. This could be possible, if the customer has a long tenure or uses a lot of services.

Now, I conclude that company mainly have customers with low charges, which means company should focus on these customers by providing even more affordable services.

1.3.3 Churn Count



```
[28]: plt.pie(x = df['Churn'].value_counts(), labels = df['Churn'].unique(), autopct␣
       ↪= '%1.2f%%')
      plt.title('Churn Count')
      plt.show()
```

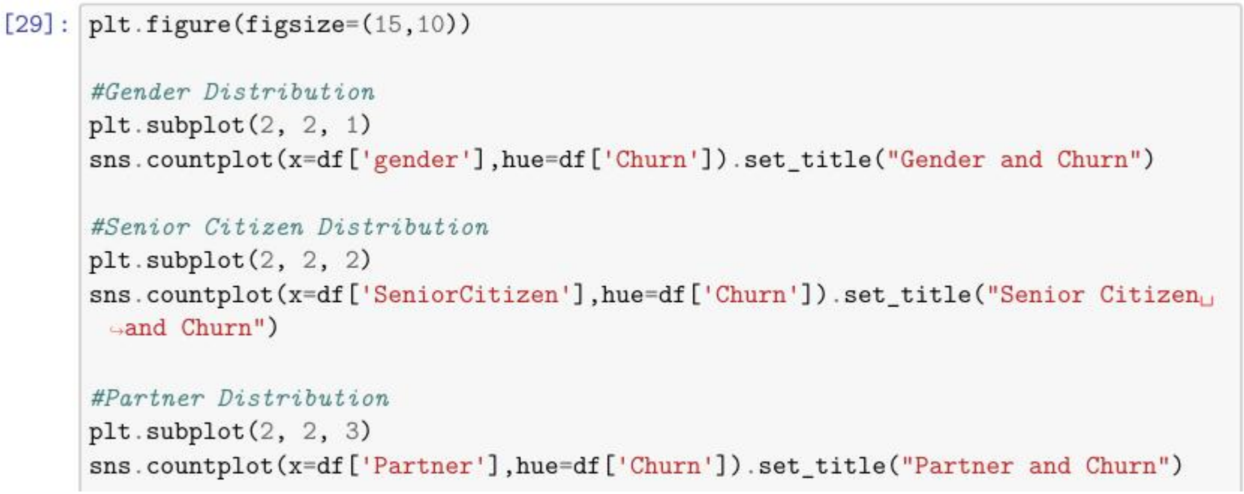


In the dataset, the number of churning customers is very less as compared to non-churning. Only 26.49% churned from the telecom company. This could be a potential proof, that company is quite good at retaining its customers.

4.3 Bivariate Analysis

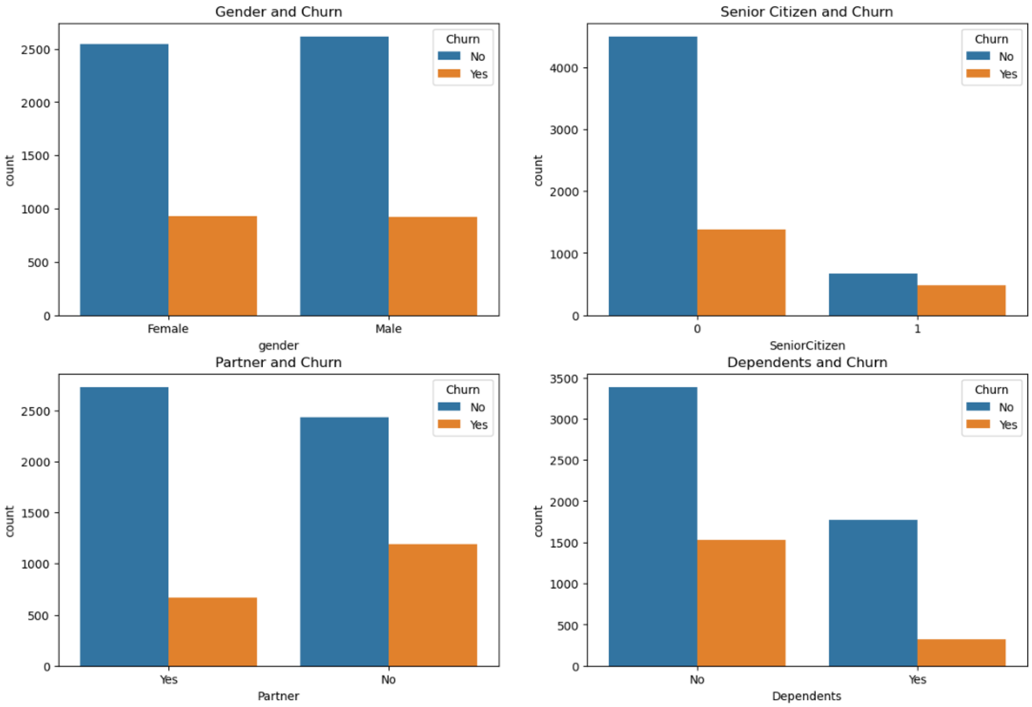
Till now, I have visualized the data and got a better understanding of the data. Now, I will look at the relationship between the independent variables and the target variable.

1.3.4 Customer Demogrpahics and Churn



```
[29]: plt.figure(figsize=(15,10))

      #Gender Distribution
      plt.subplot(2, 2, 1)
      sns.countplot(x=df['gender'],hue=df['Churn']).set_title("Gender and Churn")

      #Senior Citizen Distribution
      plt.subplot(2, 2, 2)
      sns.countplot(x=df['SeniorCitizen'],hue=df['Churn']).set_title("Senior Citizen␣
       ↪and Churn")

      #Partner Distribution
      plt.subplot(2, 2, 3)
      sns.countplot(x=df['Partner'],hue=df['Churn']).set_title("Partner and Churn")
```

```
#Dependents Distribution
plt.subplot(2, 2, 4)
sns.countplot(x=df['Dependents'],hue=df['Churn']).set_title("Dependents and␣
 ↪Churn")

plt.show()
```



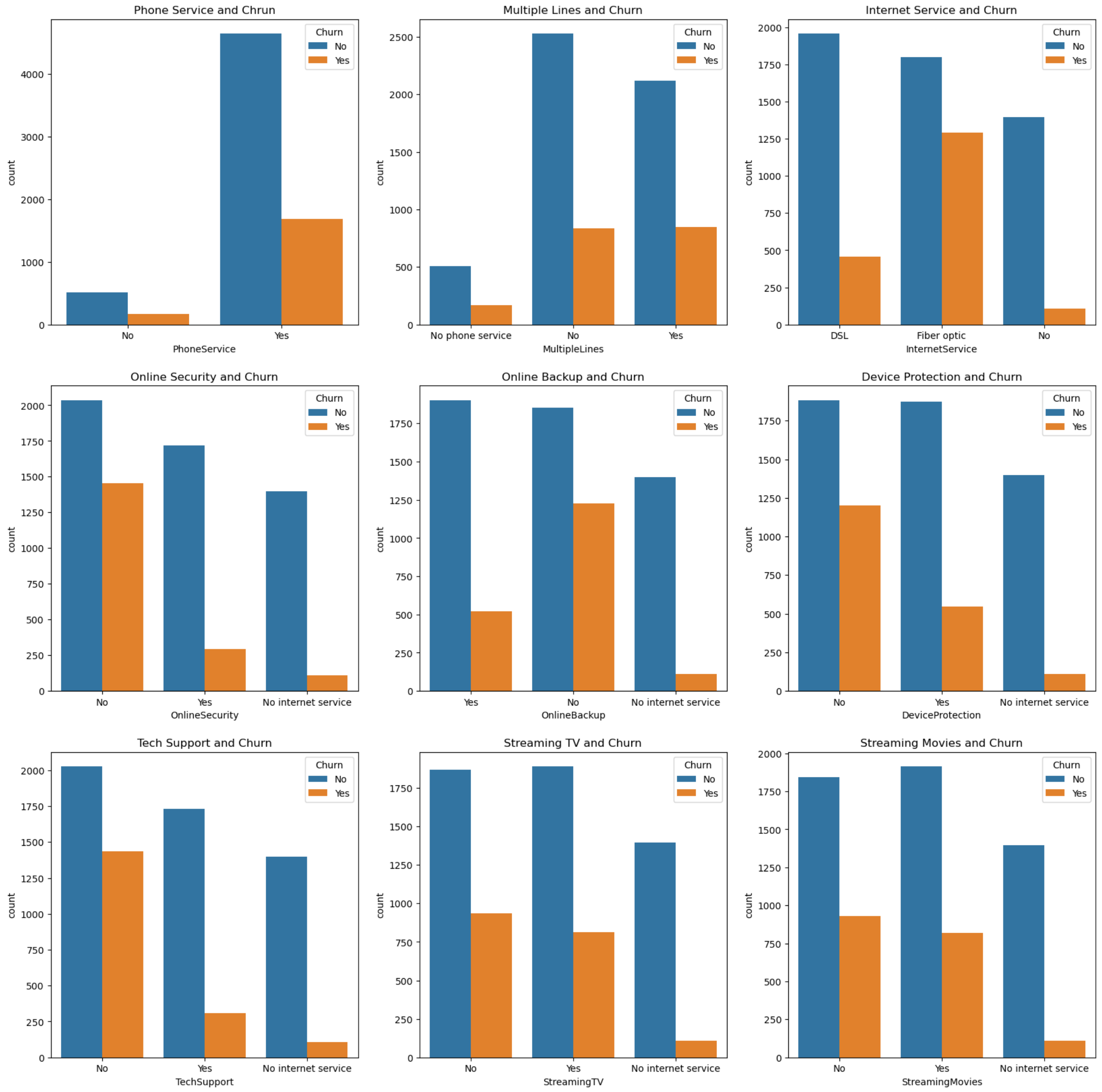
From these graphs, we can get know about the relation between customer demographics and customer churn. Both makes and females have equal number of churn count, so there is not relation between gender and customer churn. However, the senior citizens have a lesser churn count as compared to non-senior citizens, which may be because their age and they don’t want to hasle with the process of changing the telecom company. The customers with no partners have higher churn count as compared to customers with partners. Similarly, customers with no dependents have higher churn count as compared to customers with dependents.

From this I conclude that customers whom are single with no partner or have no dependents have higher churn count and senior citizens have lower churn count.

1.3.5 Services and Churn

[30]:

```
plt.figure(figsize=(20, 20))

#phone service
plt.subplot(3,3,1)
sns.countplot(x=df['PhoneService'],hue=df['Churn']).set_title("Phone Service and Chrun")

#Multiple Lines
plt.subplot(3,3,2)
sns.countplot(x=df['MultipleLines'],hue=df['Churn']).set_title("Multiple Lines and Churn")

#Internet Service
plt.subplot(3,3,3)
sns.countplot(x=df['InternetService'],hue=df['Churn']).set_title("Internet Service and Churn")

#Online Security
plt.subplot(3,3,4)
sns.countplot(x=df['OnlineSecurity'],hue=df['Churn']).set_title("Online Security and Churn")

#Online Backup
plt.subplot(3,3,5)
sns.countplot(x=df['OnlineBackup'],hue=df['Churn']).set_title("Online Backup and Churn")

#Device Protection
plt.subplot(3,3,6)
sns.countplot(x=df['DeviceProtection'],hue=df['Churn']).set_title("Device Protection and Churn")

#Tech Support
plt.subplot(3,3,7)
sns.countplot(x=df['TechSupport'],hue=df['Churn']).set_title("Tech Support and Churn")

#Streaming TV
plt.subplot(3,3,8)
sns.countplot(x=df['StreamingTV'],hue=df['Churn']).set_title("Streaming TV and Churn")

#Streaming Movies
plt.subplot(3,3,9)
sns.countplot(x=df['StreamingMovies'],hue=df['Churn']).set_title("Streaming Movies and Churn")
plt.show()
```

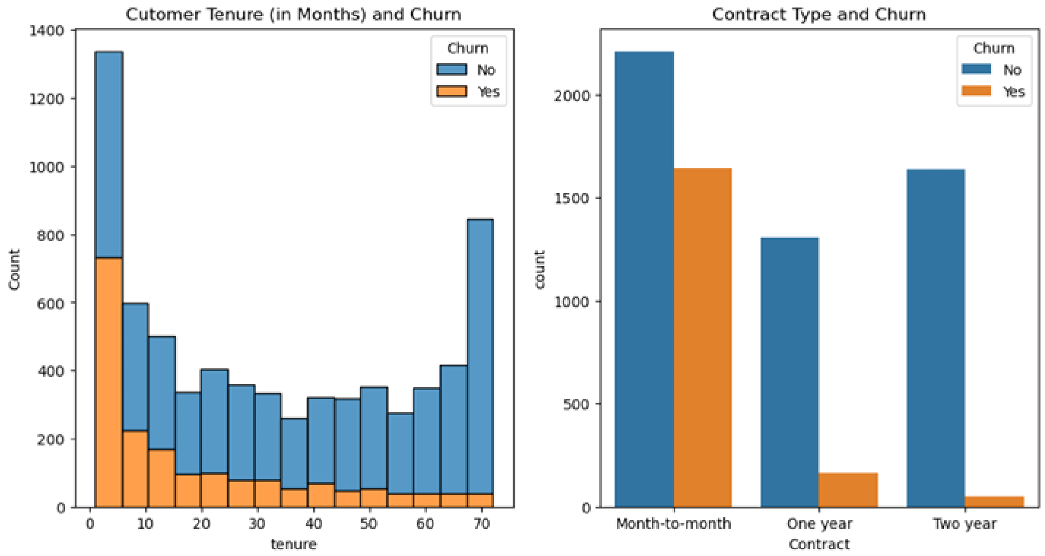


These graphs visualizes the relation between customer churn based on services opted by the customer. In the phone and internet service, there is no relation between churn and service opted, however the churn count is higher for the customers, who have taken multiple lines. Coming to other services, where customers who have not taken Online backup or Device Protection service has higher churn count, than those who have opted. Moreover, the customers with streaming services have lower churn count as compared to those who have not opted for it.

Therefore, certain services have relation with the customer churn, which are multiple lines, Online Backup, Device Protection, and Streaming Services.

1.3.6 Tenure/Contract and Churn

[31]:
```
plt.figure(figsize=(12,6))

plt.subplot(1,2,1)
sns.histplot(x = 'tenure', data = df, hue='Churn', multiple = 'stack').
 ↪set_title('Cutomer Tenure (in Months) and Churn')

plt.subplot(1,2,2)
sns.countplot(x = 'Contract', data = df,hue='Churn').set_title('Contract Type␣
 ↪and Churn')

plt.show()
```

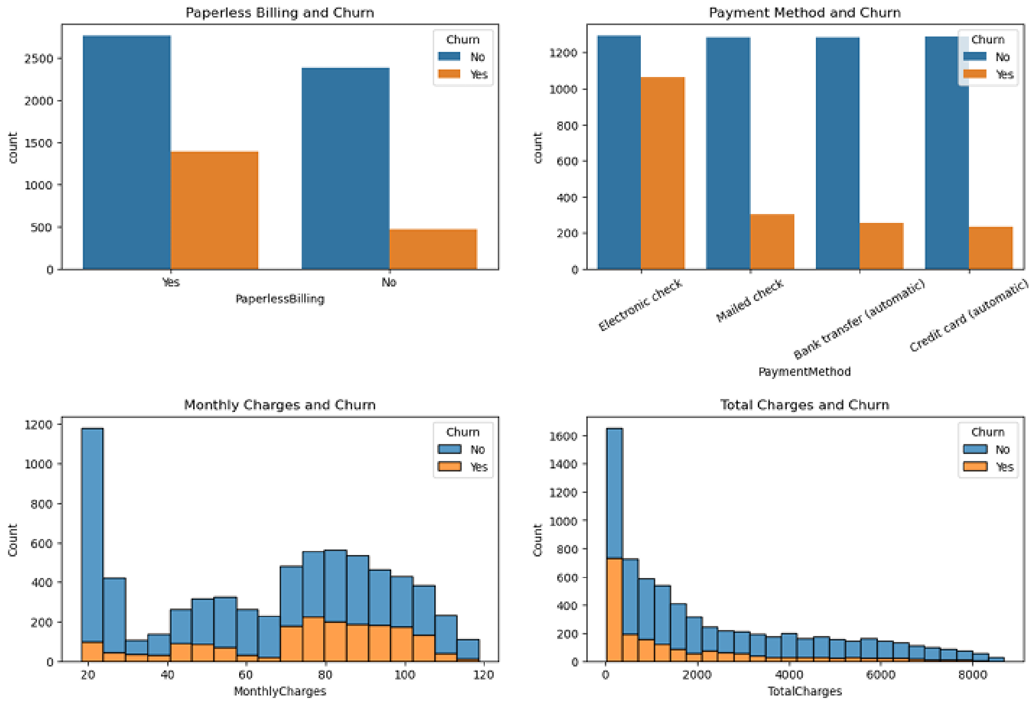


Looks like the customer tenure and contract has an inverse relation. The customers with shorter tenure or tenure less than 5 months have higher churn count. The churn count decreases with increase in tenure. Moreover, the customers with month-to-month contract have higher churn count as compared to those with one or two year contract which also proves that customer who have longer contract with the company have lower churn count.

1.3.7 Billing/Charges and Churn¶

[32]:
```
plt.figure(figsize=(15,10))
plt.subplots_adjust(hspace=.6)

#papaerless billing
plt.subplot(2,2,1)
```

```
sns.countplot(x = df['PaperlessBilling'],hue=df['Churn']).set_title('Paperless Billing and Churn')

#Payment Method
plt.subplot(2,2,2)
sns.countplot(x = df['PaymentMethod'],hue=df['Churn']).set_title('Payment Method and Churn')
plt.xticks(rotation=30)

#Monthly Charges
plt.subplot(2,2,3)
sns.histplot(x=df['MonthlyCharges'], hue = df['Churn'], multiple= 'stack').set_title("Monthly Charges and Churn")

#Total Charges
plt.subplot(2,2,4)
sns.histplot(x=df['TotalCharges'], hue = df['Churn'], multiple= 'stack').set_title("Total Charges and Churn")

plt.show()
```

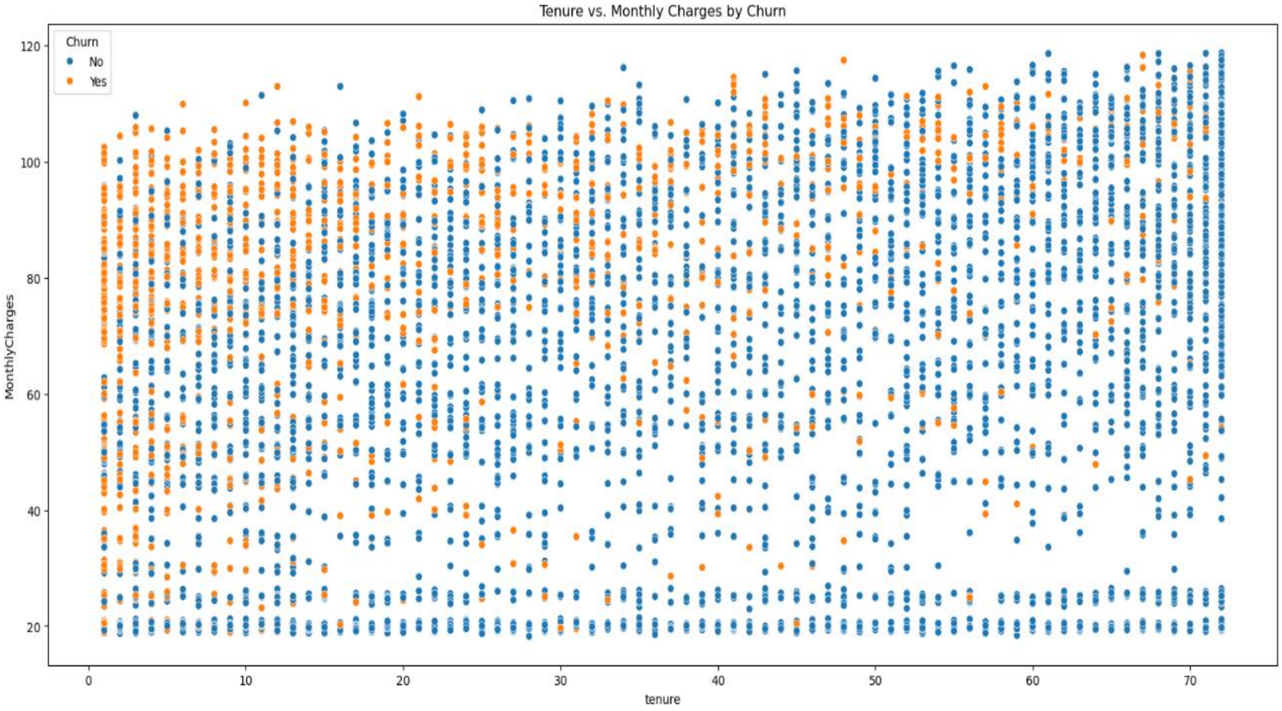


The paperless billing and payment method have not significant relation with the customer churn. However, the monthly and total charges do have a interesting relation with the customer churn. The customers with higher monthly charges have higher churn count, which is quite obvious. But, the customers with higher total charges have lower churn count, which is quite interesting. This could be possible, if the customer has a long tenure or uses a lot of services. Therefore, the company should focus on lowering the monthly charges for the customers in order to reduce the churn count.

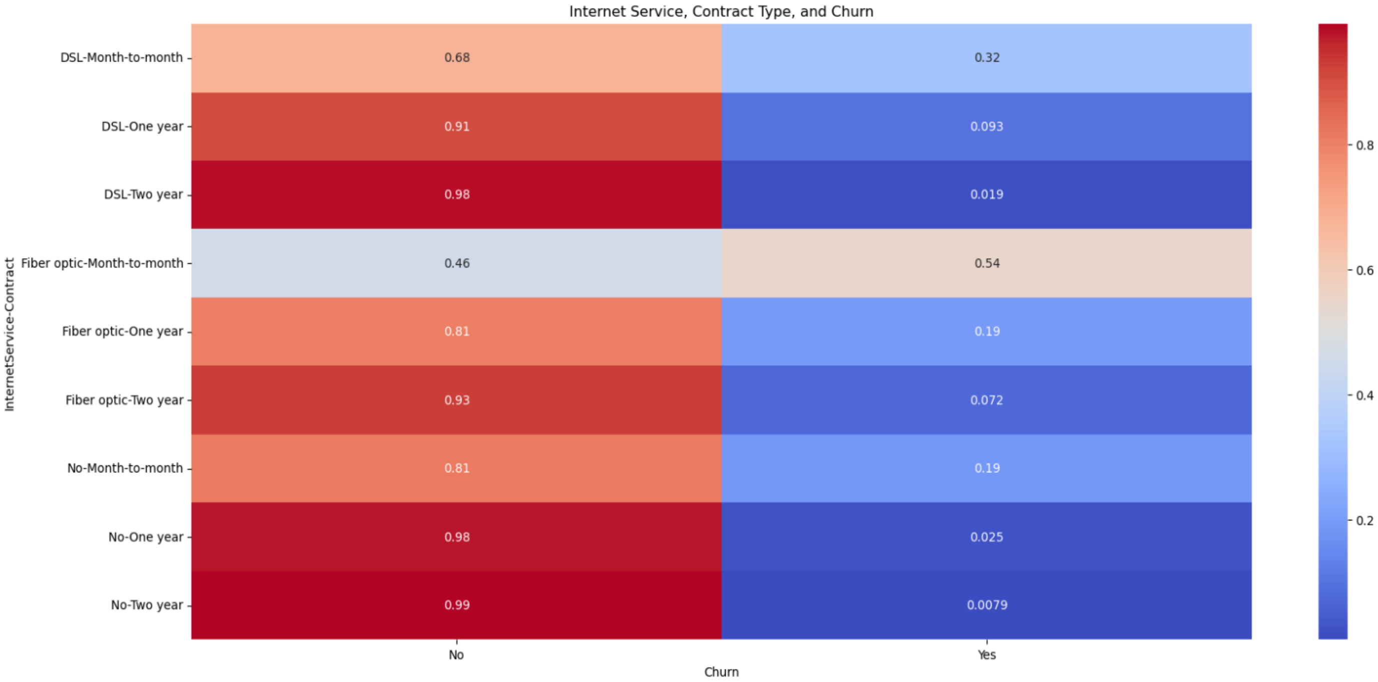
4.4 Multivariate Analysis

Till now, I have analyzed the relationships between independent variables and the target variable individually. Now, I will examine how multiple independent variables interact with each other and their combined effect on the target variable.

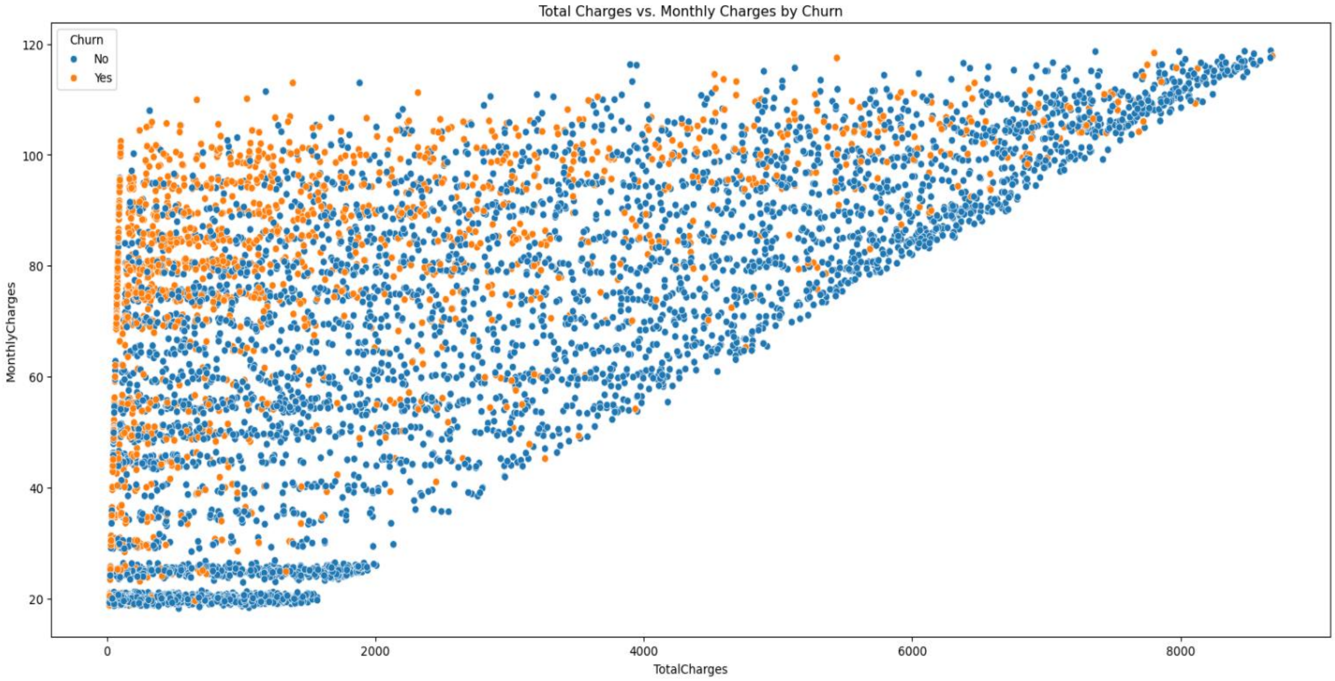
```
[33]: plt.figure(figsize=(20, 30))

      plt.subplot(3, 1, 1)
      sns.scatterplot(x=df["tenure"], y=df["MonthlyCharges"], hue=df["Churn"])
      plt.title("Tenure vs. Monthly Charges by Churn")

      plt.subplot(3, 1, 2)
      sns.heatmap(pd.crosstab(index=[df['InternetService'], df['Contract']],␣
       ↪columns=df['Churn'], normalize='index'), annot=True, cmap="coolwarm")
      plt.title("Internet Service, Contract Type, and Churn")

      plt.subplot(3, 1, 3)
      sns.scatterplot(x=df["TotalCharges"], y=df["MonthlyCharges"], hue=df["Churn"])
      plt.title("Total Charges vs. Monthly Charges by Churn")

      plt.show()
```



This scatter plot shows how customer tenure and monthly charges are related to churn. Customers with a shorter tenure and higher monthly charges tend to have a higher churn rate (represented by more orange points in that region). Customers with a longer tenure and lower monthly charges have a lower churn rate, indicating they are more likely to stay.



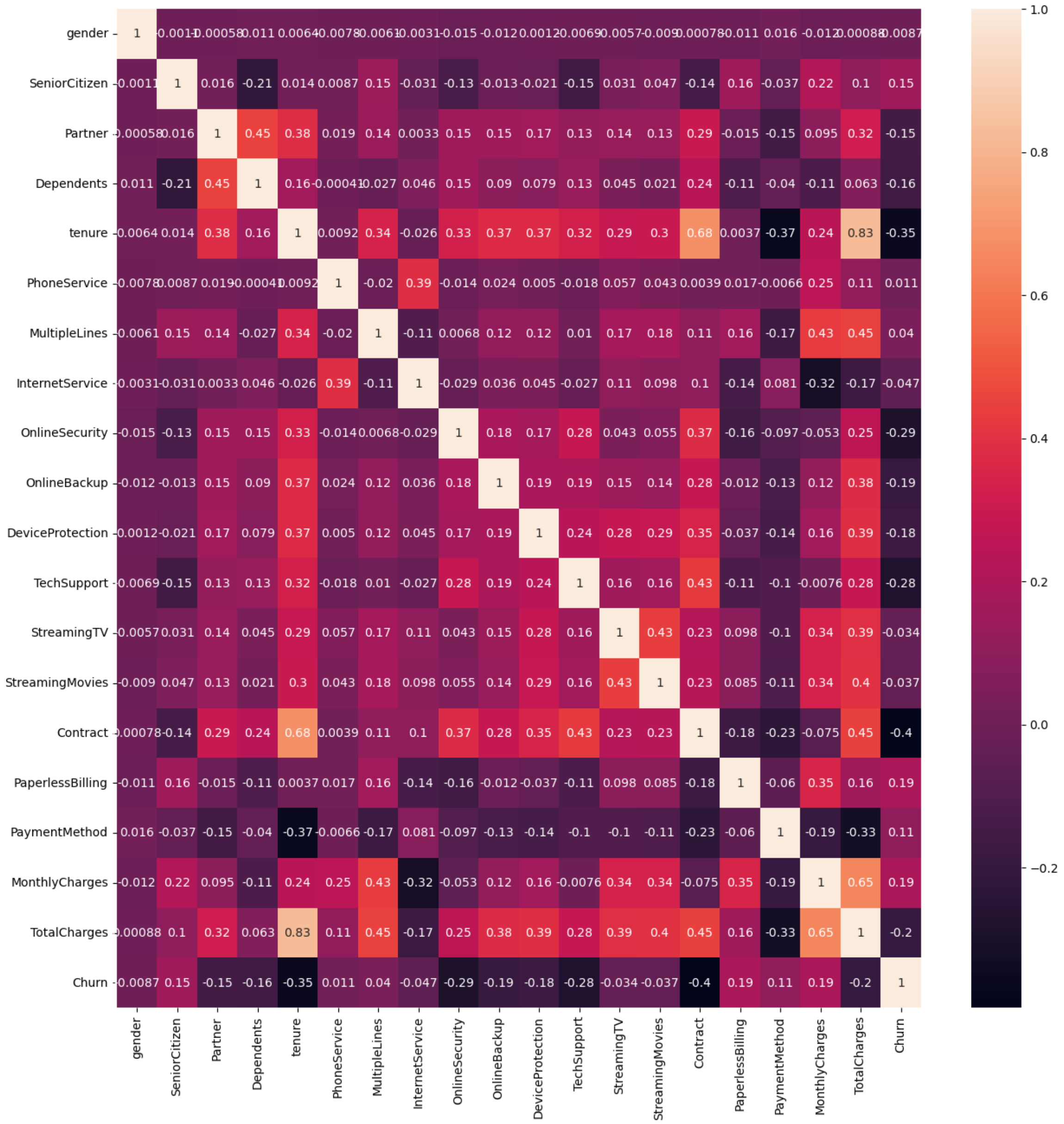
This heatmap represents the relationship between internet service type, contract type, and customer churn. Customers on month-to-month contracts, especially those with Fiber optic internet, have a significantly higher churn rate compared to those with one-year or two-year contracts. This suggests that longer contract durations help reduce churn.



This scatter plot examines the relationship between total charges, monthly charges, and churn. Customers with high monthly charges tend to churn more frequently, while those with high total charges (who have likely been customers for a long time) tend to have lower churn rates. This suggests that long-term customers who accumulate high total charges are more loyal, whereas new customers with high monthly charges are at a greater risk of leaving.

1.6 Correlation Matrix Heatmap

```
[38]: plt.figure(figsize=(15, 15))
      sns.heatmap(df.corr(), annot=True)
      plt.show()
```

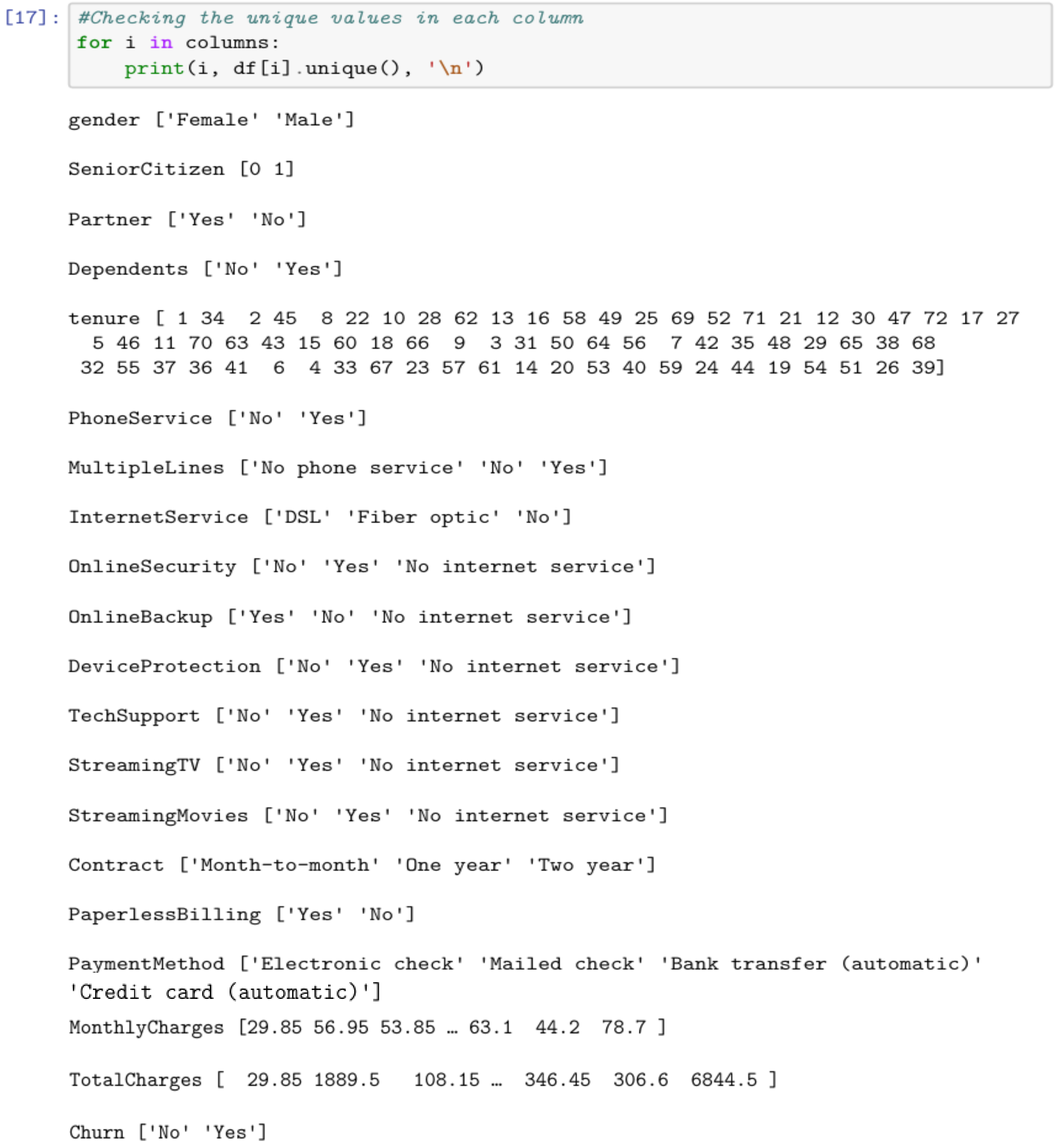


The heatmap shows the correlation between different features in the dataset. Customers with **longer tenure and long-term contracts** are less likely to churn, while those on **month-to-month contracts** have a higher churn rate. **Tech support and online security services** reduce churn, suggesting they improve customer retention. **Higher monthly charges slightly increase churn**, but customers with **higher total charges tend to stay longer**, likely due to extended tenure or additional services. The company should focus on **offering long-term contracts, improving support services, and managing monthly charges** to reduce churn.

5. Feature Engineering

Feature engineering involves transforming raw data to improve model performance. In this step, **Label Encoding** is applied to convert categorical variables into numerical values, making them suitable for machine learning models. Then, **Feature Scaling** is performed using StandardScaler to standardize numerical features (tenure, MonthlyCharges, and TotalCharges). This ensures all features have a mean of 0 and a standard deviation of 1, preventing larger values from dominating the model. These transformations enhance model accuracy and stability.

5.1 Label Encoding



```
[17]: #Checking the unique values in each column
      for i in columns:
          print(i, df[i].unique(), '\n')
```

```
gender ['Female' 'Male']

SeniorCitizen [0 1]

Partner ['Yes' 'No']

Dependents ['No' 'Yes']

tenure [ 1 34  2 45  8 22 10 28 62 13 16 58 49 25 69 52 71 21 12 30 47 72 17 27
  5 46 11 70 63 43 15 60 18 66  9  3 31 50 64 56  7 42 35 48 29 65 38 68
 32 55 37 36 41  6  4 33 67 23 57 61 14 20 53 40 59 24 44 19 54 51 26 39]

PhoneService ['No' 'Yes']

MultipleLines ['No phone service' 'No' 'Yes']

InternetService ['DSL' 'Fiber optic' 'No']

OnlineSecurity ['No' 'Yes' 'No internet service']

OnlineBackup ['Yes' 'No' 'No internet service']

DeviceProtection ['No' 'Yes' 'No internet service']

TechSupport ['No' 'Yes' 'No internet service']

StreamingTV ['No' 'Yes' 'No internet service']

StreamingMovies ['No' 'Yes' 'No internet service']

Contract ['Month-to-month' 'One year' 'Two year']

PaperlessBilling ['Yes' 'No']

PaymentMethod ['Electronic check' 'Mailed check' 'Bank transfer (automatic)'
'Credit card (automatic)']
MonthlyCharges [29.85 56.95 53.85 … 63.1  44.2  78.7 ]

TotalCharges [  29.85 1889.5   108.15 …  346.45  306.6  6844.5 ]

Churn ['No' 'Yes']
```

[34]:

```
from sklearn.preprocessing import LabelEncoder
```

[35]:

```
#colums for label encoding
cols = df.columns[df.dtypes == 'object']

#Label encoder object
le = LabelEncoder()

#Label encoding the columns
for i in cols:
    le.fit(df[i])
    df[i] = le.transform(df[i])
    print(i, df[i].unique(), '\n')
```

```
gender [0 1]

Partner [1 0]

Dependents [0 1]

PhoneService [0 1]

MultipleLines [1 0 2]

InternetService [0 1 2]

OnlineSecurity [0 2 1]

OnlineBackup [2 0 1]

DeviceProtection [0 2 1]

TechSupport [0 2 1]

StreamingTV [0 2 1]

StreamingMovies [0 2 1]

Contract [0 1 2]
PaperlessBilling [1 0]

PaymentMethod [2 3 0 1]

Churn [0 1]
```

5.2 Feature Scaling

Feature Scaling

```
[36]: from sklearn.preprocessing import StandardScaler
```

```
[37]: #Standardizing the data
      sc = StandardScaler()
      df[['tenure', 'MonthlyCharges', 'TotalCharges']] = sc.
       ↪fit_transform(df[['tenure', 'MonthlyCharges', 'TotalCharges']])

      df.head(3)
```

```
[37]:    gender  SeniorCitizen  Partner  Dependents    tenure  PhoneService  \
      0       0              0        1           0 -1.285566             0
      1       1              0        0           0  0.060346             1
      2       1              0        0           0 -1.244781             1

         MultipleLines  InternetService  OnlineSecurity  OnlineBackup  \
      0              1                0               0             2
      1              0                0               2             0
      2              0                0               2             2

         DeviceProtection  TechSupport  StreamingTV  StreamingMovies  Contract  \
      0                 0            0            0                0         0
      1                 2            0            0                0         1
      2                 0            0            0                0         0

         PaperlessBilling  PaymentMethod  MonthlyCharges  TotalCharges  Churn
      0                 1              2       -1.165523     -0.997284      0
      1                 0              3       -0.264071     -0.176848      0
      2                 1              3       -0.367189     -0.962740      1
```

6. Insights

1. **Customer Tenure and Churn:**

   Customers with **longer tenure** tend to have a **lower churn rate**, indicating that retaining customers for a longer period reduces the likelihood of churn.

2. **Monthly Charges Impact:**

   Customers with **higher monthly charges** show a **higher churn rate**, suggesting that expensive plans may contribute to customer dissatisfaction and cancellations.

3. **Total Charges vs. Churn:**

   Despite high monthly charges increasing churn, customers with **higher total charges** tend to stay longer, implying that **long-term customers are more loyal** and less likely to churn.

4. **Contract Type Influence:**

   Customers on **monthly contracts** have a **significantly higher churn rate** compared to those with **one-year or two-year contracts**, indicating that offering long-term plans may help reduce churn.

5. **Internet Service and Churn:**

   Customers using **Fiber optic internet** experience **higher churn rates** than DSL users, possibly due to cost or service-related issues.

6. **Impact of Online Services:**

   Customers who **do not subscribe to additional services** like **Online Security, Tech Support, and Streaming Services** tend to churn more, indicating that bundling services may improve customer retention.

7. Conclusion

1. **Churn Rate by Customer Type:**
   - **Senior citizens** have a **lower churn** rate.
   - **Single customers** or those **without dependents** tend to **churn more**.
2. **Service Satisfaction and Churn:**
   - Customers using **streaming services** have **lower churn** compared to those using **Online Backup** and **Device Protection**.
   - **Less satisfaction with Online Backup and Device Protection** may contribute to higher churn.
3. **Tenure and Churn Relationship:**
   - **Tenure** has an **inverse relation** with **churn.**
   - Customers with a **tenure shorter than 5 months** have a **higher churn rate**.
4. **Contract Type and Churn:**
   - **Month-to-month contract customers churn more** than those with **1-year or 2-year contracts**.
   - **Long-term contracts** help retain customers.
5. **Pricing and Churn Behaviour:**
   - **Higher monthly charges** lead to **higher churn.**
   - Customers with **higher total charges** have **lower churn** (suggesting long-term customers are more stable).
   - **Lowering monthly charges** could help **reduce churn.**
6. **Feature Importance for Predicting Churn:**
   - The **most important factors affecting churn** are:
     - **Tenure**
     - **Contract Type**
     - **Monthly Charges**
     - **Total Charges**
   - The company should **focus on these features** to reduce customer churn.