Case Study: Telecommunication Churn Prediction

Group Members:

Lavanya Miranam

Vibha Chaudhary

Vaishnavi Gaikwad

Ram Khandare

DataSet Link: [Telco Customer Churn | Kaggle]

What is Customer Churn ?

Customer churn refers to when a customer decides to stop using a service, and switches to a competitor.

What Is Telcom Customer Churn ?

Customer churn in telecommunication referes the customers in telcom switching to other service provider

telecom companies, churn is a major concern as losing customers means losing revenue and market share

Common Causes of Telecom Churn :

- Poor Network Quality (dropped calls, slow internet)
- High Pricing (competitors offering better deals)
- Poor Customer Service (unresolved complaints)
- Lack of Personalization (customers feel undervalued)
- Billing Issues (hidden charges, incorrect bills)

Dataset Information :-

The dataset used in this project was imported from Kaggle.This dataset is about total 7043 records and 21 variables.

Overview of Dataset:-

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupport
5777	6087-YPWHO	Male	0	Yes	No	72	Yes	Yes	DSL	Yes	...	Yes	Yes
1652	3307-TLCUD	Male	0	Yes	No	17	No	No phone service	DSL	Yes	...	Yes	No
4575	8746-OQQRW	Male	0	No	No	4	Yes	Yes	No	No internet service	...	No internet service	No internet service
1192	1579-KLYDT	Male	0	No	No	7	Yes	Yes	Fiber optic	No	...	Yes	No
5658	2371-JQHZZ	Male	0	Yes	No	24	Yes	Yes	Fiber optic	No	...	No	No

5 rows × 21 columns

TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn
Yes	No	No	Two year	No	Mailed check	68.15	4808.7	No
No	No	No	Month-to-month	No	Mailed check	34.40	592.75	No
No internet service	No internet service	No internet service	Month-to-month	No	Mailed check	25.25	101.9	No
No	Yes	No	Month-to-month	Yes	Electronic check	90.45	593.45	Yes
No	Yes	Yes	Month-to-month	Yes	Electronic check	93.00	2248.05	No

Code:- **2. Loading Libraries and Data:-**

```
import pandas as pd
import numpy as np
```

```
tel=pd.read_csv('Telco-Customer-Churn.csv')
```

```
tel.shape
```

```
(7043, 21)
```

Here the DataFrame name we set as tel

3.Columns Description:-

```
tel.columns
```

```
Index(['customerID', 'gender', 'SeniorCitizen', 'Partner', 'Dependents',
       'tenure', 'PhoneService', 'MultipleLines', 'InternetService',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection', 'TechSupport',
       'StreamingTV', 'StreamingMovies', 'Contract', 'PaperlessBilling',
       'PaymentMethod', 'MonthlyCharges', 'TotalCharges', 'Churn'],
      dtype='object')
```

- **Info about Customer realated columns**
  : gender,CutomerID,Partner,Dependent,SeniorCitizen(age range)
- **Services that each customer has signed up for columns**
  : 'PhoneService', 'MultipleLines', 'InternetService', 'OnlineSecurity',

'OnlineBackup', 'DeviceProtection', 'TechSupport','StreamingTV', 'StreamingMovies'

- **Customers who left within the last month :** The Column called churn
- **Customer account information :** 'Contract', 'PaperlessBilling', 'PaymentMethod', 'MonthlyCharges', 'TotalCharges',tenure (no of month they are taking services)

Column Type:-

Code:-

```
]: tel.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
```

- **Numerical Type:** tenure,Monthlycharges,TotalCharges.
- **Categorical:** gender,SeniorCitizen,partner,Dependents,'PhoneService', 'MultipleLines', 'InternetService', 'OnlineSecurity', 'OnlineBackup', 'DeviceProtection', 'TechSupport','StreamingTV', 'StreamingMovies',contract,PaperlessBilling,PaymentMethod,Churn
- **Mixed:** customerID

Analysis

Issues With Dataset Columns Type :-

- Total Charges must be in form of numerical
- Seniorcitizen column having 0 and 1 it must be yes or no a categorical type data

Type Conversion:-

Code:-

1. **Converting to_numeric**

```
tel['TotalCharges']=pd.to_numeric(tel['TotalCharges'],errors='coerce')
```

2. **Setting No at 0 and Yes at 1**

```
tel['SeniorCitizen']=list(tel['SeniorCitizen'].map({0:'No',1:'Yes'}))
```

Output:-

```
tel.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   object 
 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      7032 non-null   float64
 20  Churn             7043 non-null   object 
dtypes: float64(2), int64(1), object(18)
```

Handling Missing Values:-

```
tel.isnull().sum()
```

```
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
```

Here after Transforming TotalCharges object to numeric we can see there is 11 total missing values

Step1: 1) Check how the Total charges Column is formed

By analysis total charges is nothing but (tenure * monthly charges + Extra Charges)

From this we understood how to calculate totalcharges

2) Are there any common factors present in other columns if TotalCharges has NaN values?

```
tel[tel['TotalCharges'].isnull()]
```

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	De
488	4472-LVYGI	Female	No	Yes	Yes	0	No	No phone service	DSL	Yes	...	
753	3115-CZMZD	Male	No	No	Yes	0	Yes	No	No	No internet service	...	
936	5709-LVOEQ	Female	No	Yes	Yes	0	Yes	No	DSL	Yes	...	
1082	4367-NUYAO	Male	No	Yes	Yes	0	Yes	Yes	No	No internet service	...	
1340	1371-DWPAZ	Female	No	Yes	Yes	0	No	No phone service	DSL	Yes	...	
3331	7644-OMVMY	Male	No	Yes	Yes	0	Yes	No	No	No internet service	...	
3826	3213-VVOLG	Male	No	Yes	Yes	0	Yes	Yes	No	No internet service	...	
4380	2520-SGTTA	Female	No	Yes	Yes	0	Yes	No	No	No internet service	...	
5218	2923-ARZLG	Male	No	Yes	Yes	0	Yes	No	No	No internet service	...	
6670	4075-WKNIU	Female	No	Yes	Yes	0	Yes	Yes	DSL	No	...	
6754	2775-SEFEE	Male	No	No	Yes	0	Yes	Yes	DSL	Yes	...	

TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn
Yes	Yes	No	Two year	Yes	Bank transfer (automatic)	52.55	NaN	No
No internet service	No internet service	No internet service	Two year	No	Mailed check	20.25	NaN	No
No	Yes	Yes	Two year	No	Mailed check	80.85	NaN	No
No internet service	No internet service	No internet service	Two year	No	Mailed check	25.75	NaN	No
Yes	Yes	No	Two year	No	Credit card (automatic)	56.05	NaN	No
No internet service	No internet service	No internet service	Two year	No	Mailed check	19.85	NaN	No
No internet service	No internet service	No internet service	Two year	No	Mailed check	25.35	NaN	No
No internet service	No internet service	No internet service	Two year	No	Mailed check	20.00	NaN	No
No internet service	No internet service	No internet service	One year	Yes	Mailed check	19.70	NaN	No
Yes	Yes	No	Two year	No	Mailed check	73.35	NaN	No
Yes	No	No	Two year	Yes	Bank transfer (automatic)	61.90	NaN	No

Here, we can see that the most common contract type is "Two-year" and payment method is mailed check when TotalCharges is NaN and tenure is zero.

To replace Totalcharges nan values we can calculate average tenure and Extra charges whoes contract type is two year and payment method is mailed check

Code:-

Create new dataframe:-

```
df_temp=tel[(tel['Contract']=='Two year') & (tel['PaymentMethod']=='Mailed check')]
```

Drop tenure =0 records to calculate avg:-

```
df_temp.dropna(inplace=True)
```

Create New result Column to calculate total charges

TotalCarges=tenure*monthlycharges

```
df_temp['result']=df_temp['tenure']*df_temp['MonthlyCharges']
```

PaymentMethod	MonthlyCharges	TotalCharges	Churn	result
Mailed check	79.20	2497.2	No	2455.2
Mailed check	19.95	927.1	No	917.7

Now we can see there is some difference in result and Totalcarges the difference is nothing but a Extra charges

Create New differe column to calculate Extra Charges

```
df_temp['differnce']=df_temp['TotalCharges']-df_temp['result']
```

MonthlyCharges	TotalCharges	Churn	result	differnce
79.20	2497.2	No	2455.2	42.0
19.95	927.1	No	917.7	9.4

```
Extra_Charges=df_temp['differnce'].mean()
```

Extra_Charges=2.107

Now Lets fill Tenure=0 records

```
avg_tenure=df_temp['tenure'].mean()
avg_tenure
```

Here we are filling average of tenure where tenure =0

After that we are filling null values of TotalCharges

```
tel['TotalCharges'].fillna(tel['tenure']*tel['MonthlyCharges']+Extra_Charges,inplace=True)
```

```
tel.isnull().sum()
```

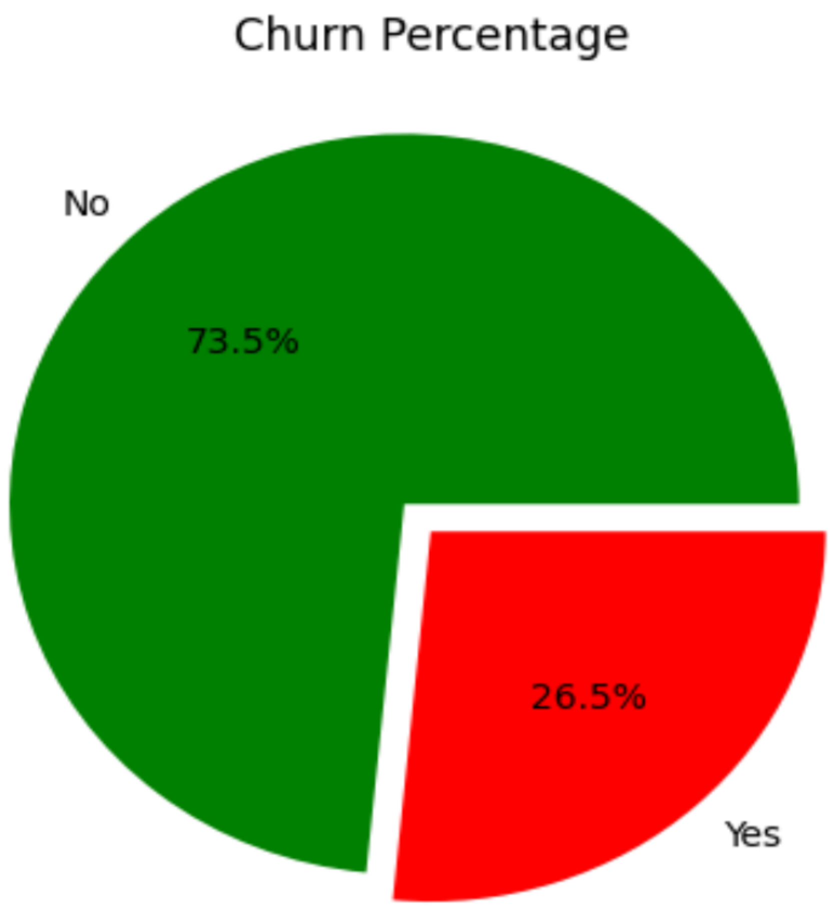
```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV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

Here There is no missing values in our dataset

Cleaning Done

Data Visualization

here the most important column is Churn lets Visualization Churn Rate

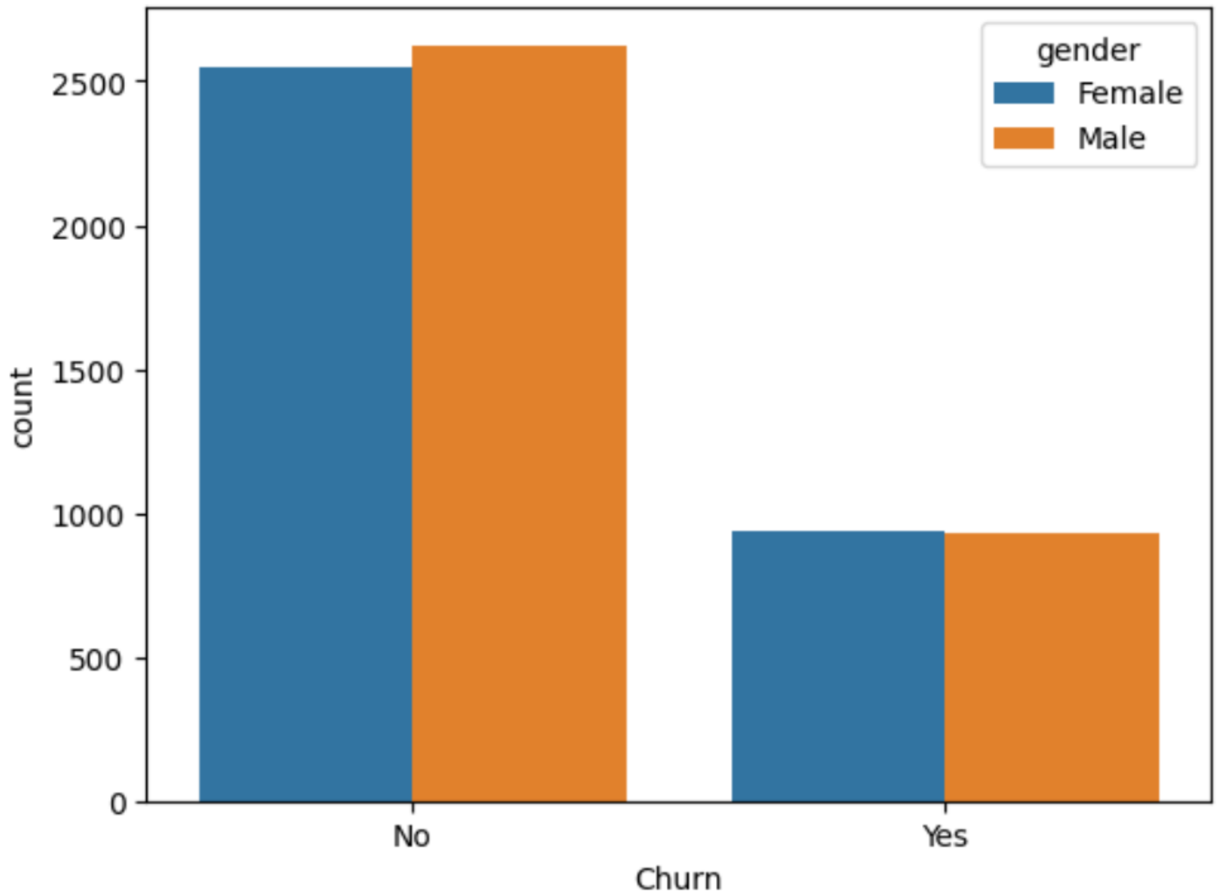


Here we can see 26.5% Customers are in Churn

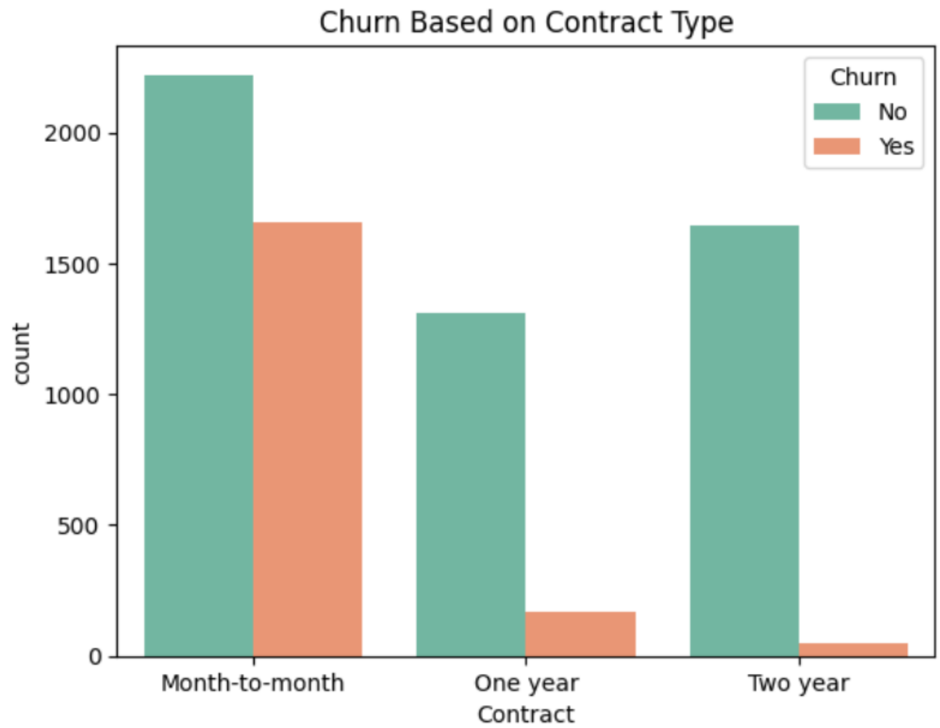
Lets Visualize Churn with all columns

Churn Based On Gender:-

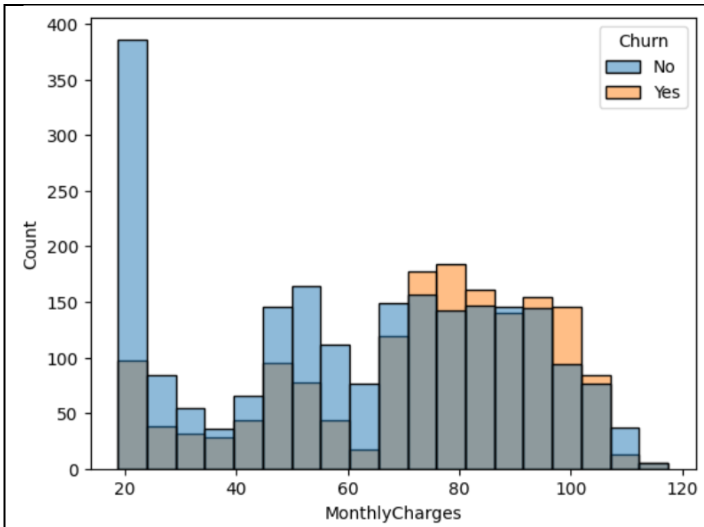
Code



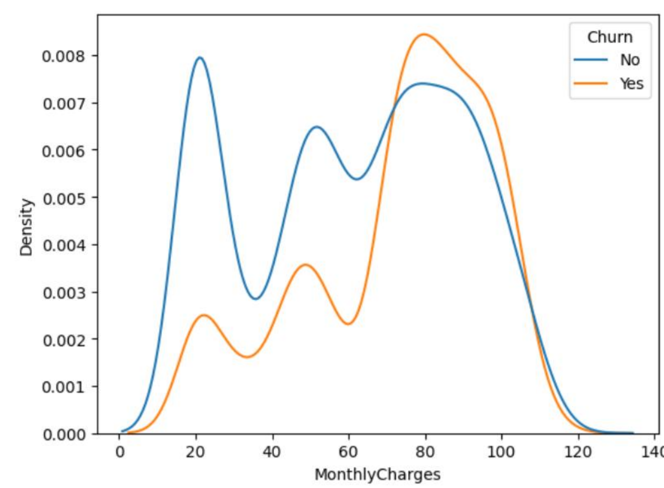
Note: there is no difference in Churn if gender is different



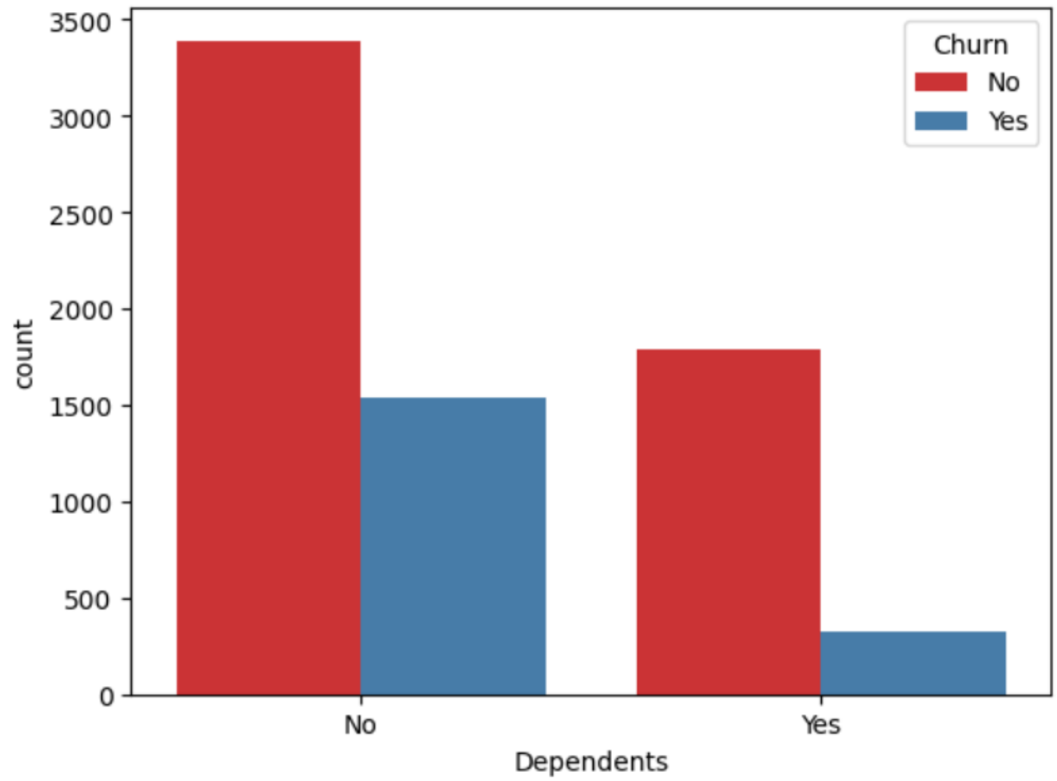
Most of Churn happened in Month-to-month Contract type



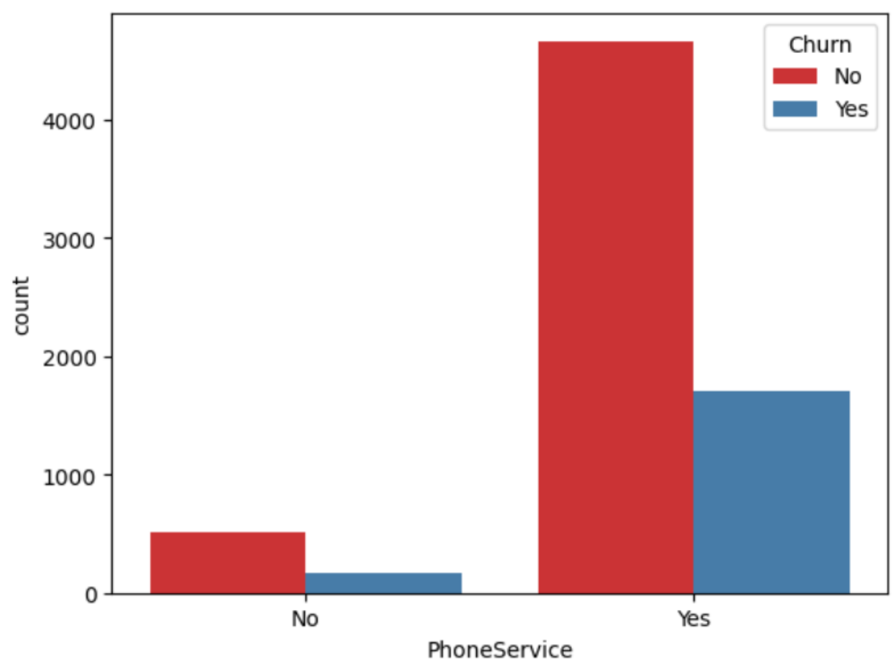

```
sns.kdeplot(data=tel,x=df['MonthlyCharges'],hue='Churn')
```



Most of Churn are having Monthly Charges between 60 tp 120



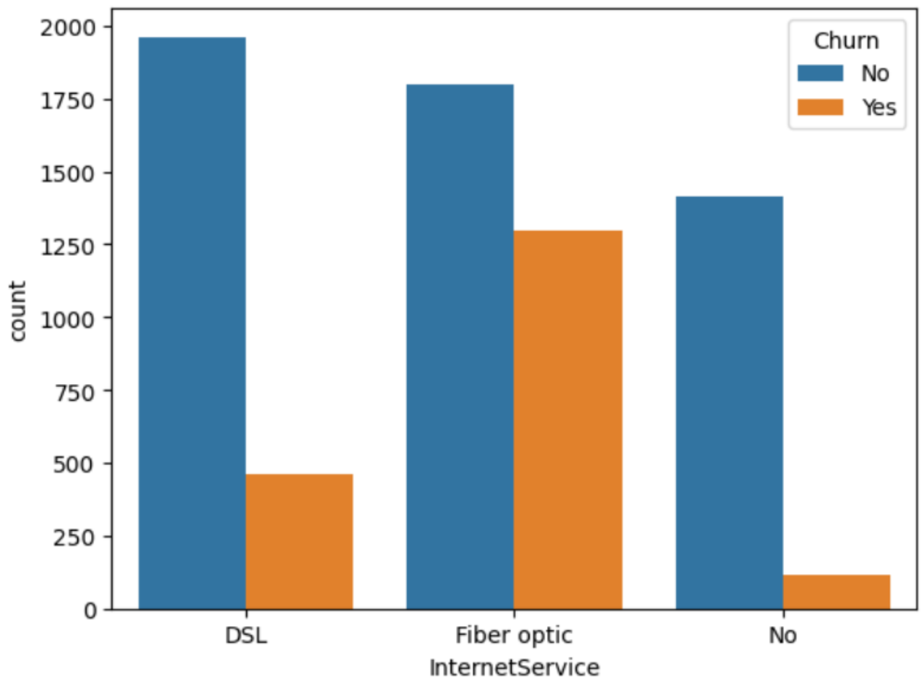
Customers Who are not Depended are in churn

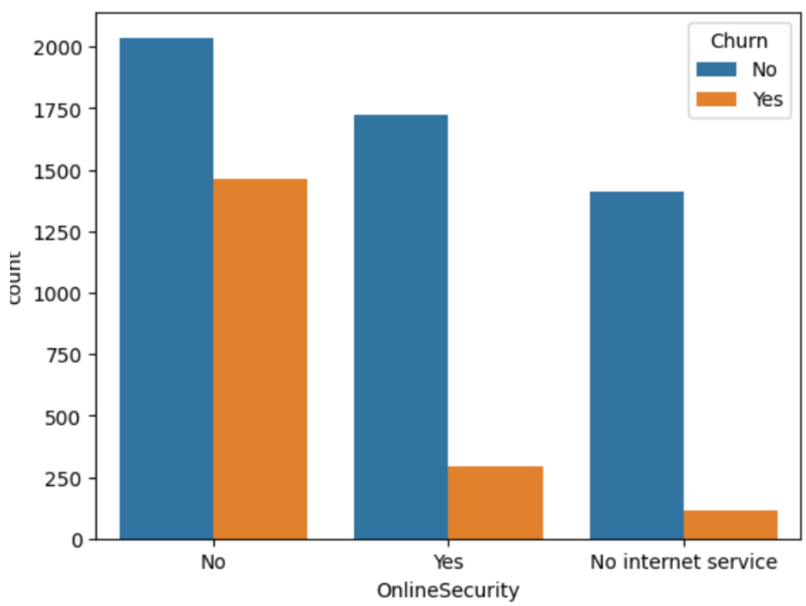


Customers Who are using PhoneService there in churn

We can see

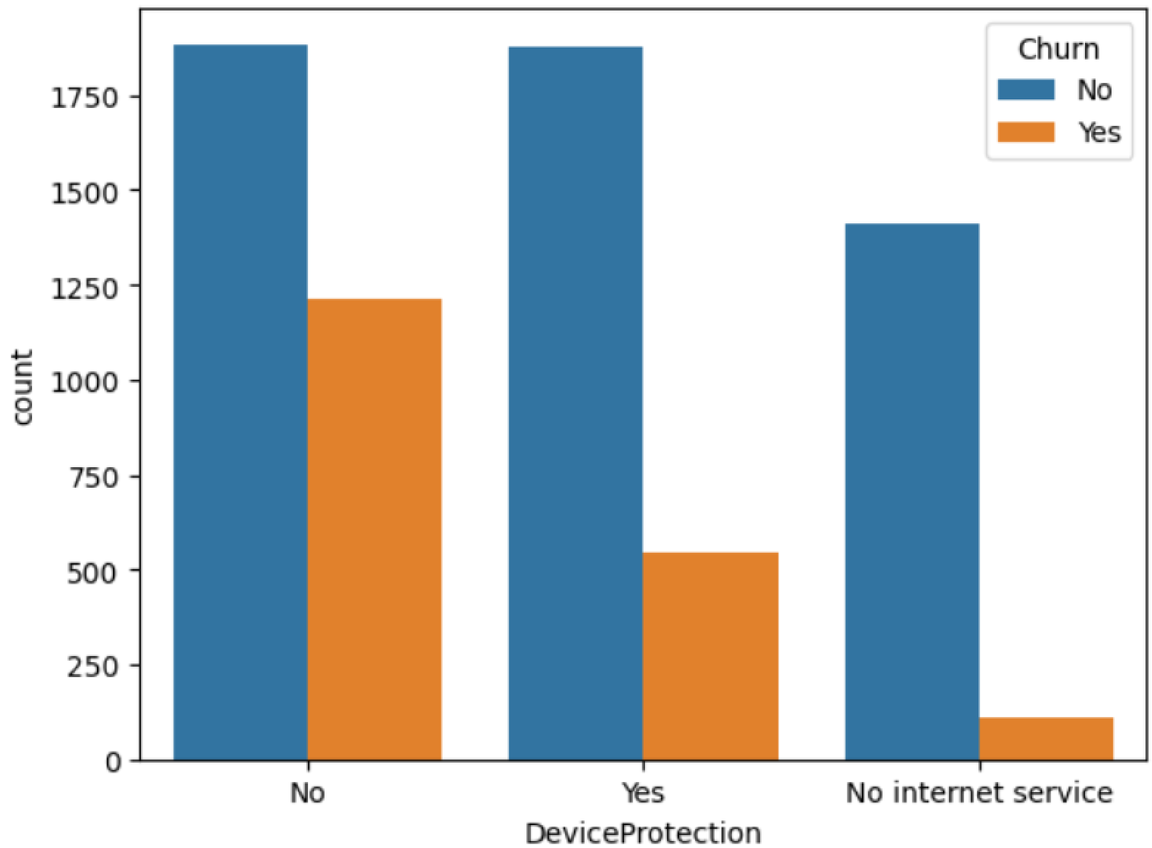
In Internet Service Customers who are using Fier Optic ther are leaving hence we have to take care or stops service of fiber optic





Customers who are not taken any online security are in churn

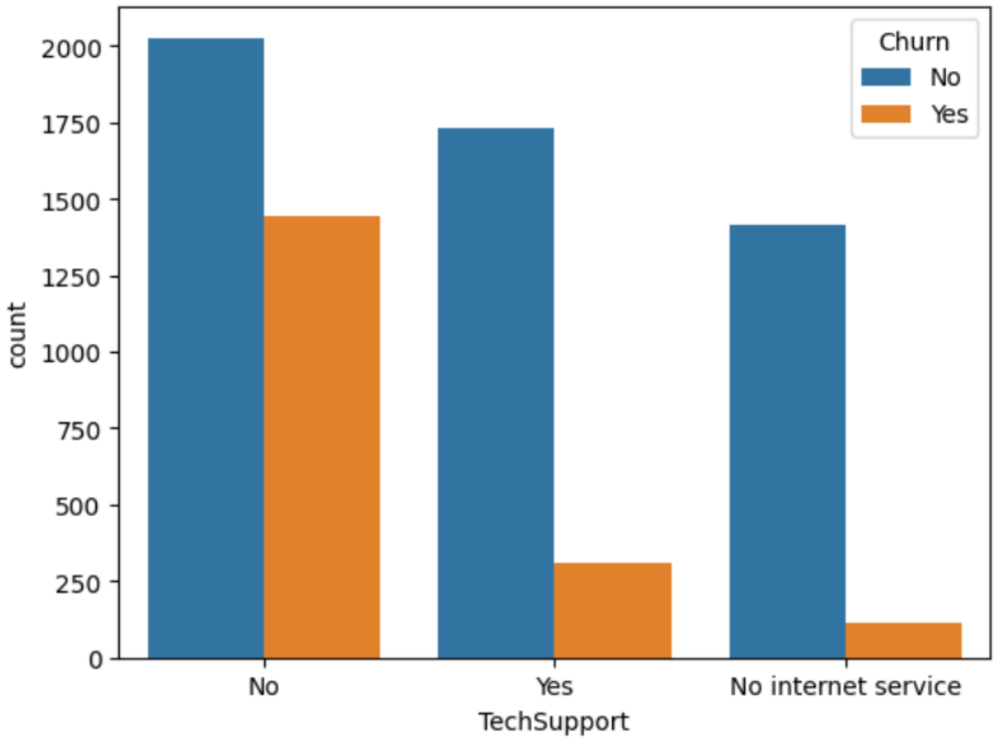
```
sns.countplot(data=tel,x=tel['DeviceProtection'],hue=tel['Churn'])
```



Here also we can see Customers who are not taken DeviceProtection They are in Churn

```
sns.countplot(data=tel,x=tel['TechSupport'],hue=tel['Churn'])
```

```
<Axes: xlabel='TechSupport', ylabel='count'>
```

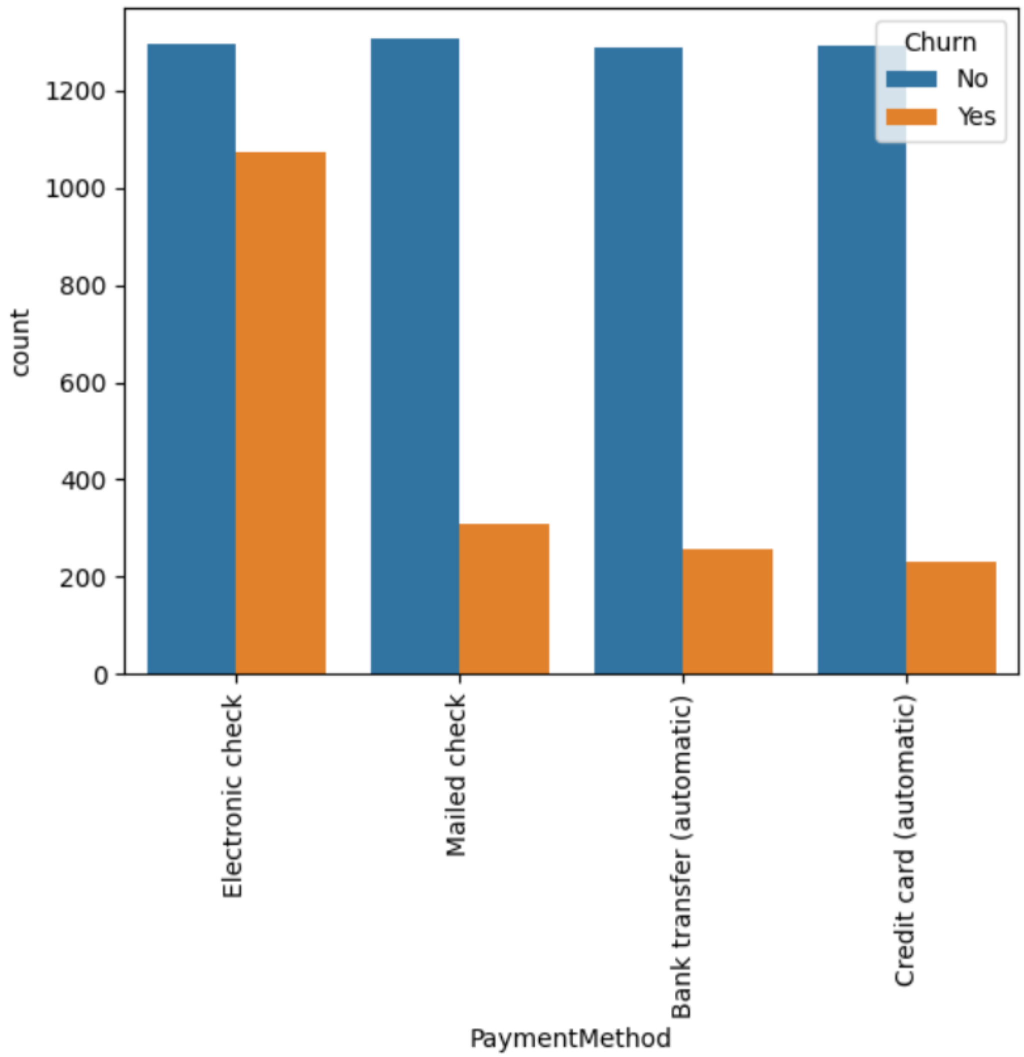


Here a Chustomers who are not having techsupport are in churn

From this technical variable analysis we can conclude that the customers who are using fiber optical and not taking online security,tech support,device protection are in churn

And the customers who are not depended them also in churn

Paymenth Method :-



Here most of customers who are having Electronic check they are in churn by seeing this we can also predict that we have to focus on electronic billing issue may be happing ?

Numerical Columns Correlation:-

```
tdf=tel[['tenure','MonthlyCharges','TotalCharges']]
tcorrelational=tdf.corr()
```

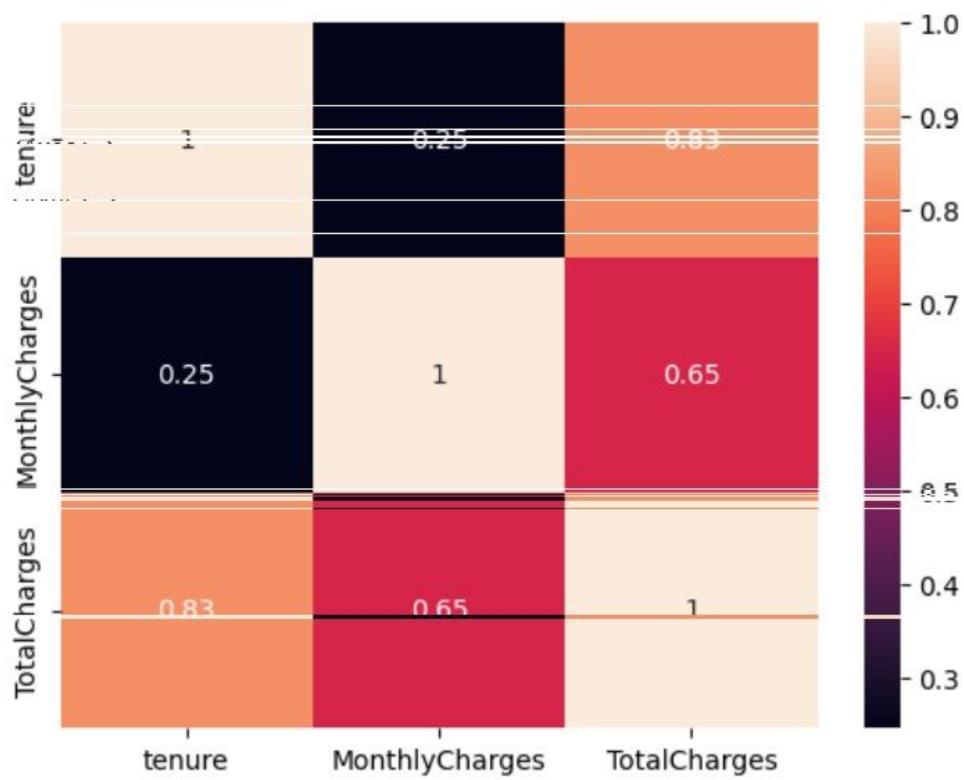
```
tcorrelational
```

	tenure	MonthlyCharges	TotalCharges
tenure	1.000000	0.246035	0.825442
MonthlyCharges	0.246035	1.000000	0.651117
TotalCharges	0.825442	0.651117	1.000000

Here tenure and TotalCharges are having highly positive correlation with each other

```
]: sns.heatmap(tcorrelational,annot=True)
```

```
]: <Axes: >
```



Conclusion:-

Customer Churn Patterns:

- A significant number of customers churn in the first few months of tenure. Customers with longer tenure are less likely to leave.
- Month-to-month contracts have the highest churn rate, while two-year contracts show the least churn.
- Service Usage Insights:

- Customers with Fiber optic internet churn more than those with DSL or no internet service. This may be due to service quality or pricing issues.
- Additional services like online security, tech support, and device protection reduce churn, indicating that customers who invest in these services are more likely to stay.
- Demographic Influence:

- Senior citizens have a higher churn rate than younger customers, possibly due to price sensitivity or service adaptation challenges.
- Customers without dependents or partners are more likely to churn, suggesting family or household stability plays a role in retention.
- Billing & Payment Trends:

- Electronic check payments correlate with higher churn, while credit card and bank transfer payments see lower churn.
- Customers with higher monthly charges tend to leave more often, highlighting potential pricing concerns.
- Correlation Analysis (Heatmap Insights):

- Tenure is negatively correlated with churn, reinforcing that longer-term customers are more loyal.

- **Contract type and payment method have strong associations with churn, emphasizing the importance of structured contracts and payment flexibility.**