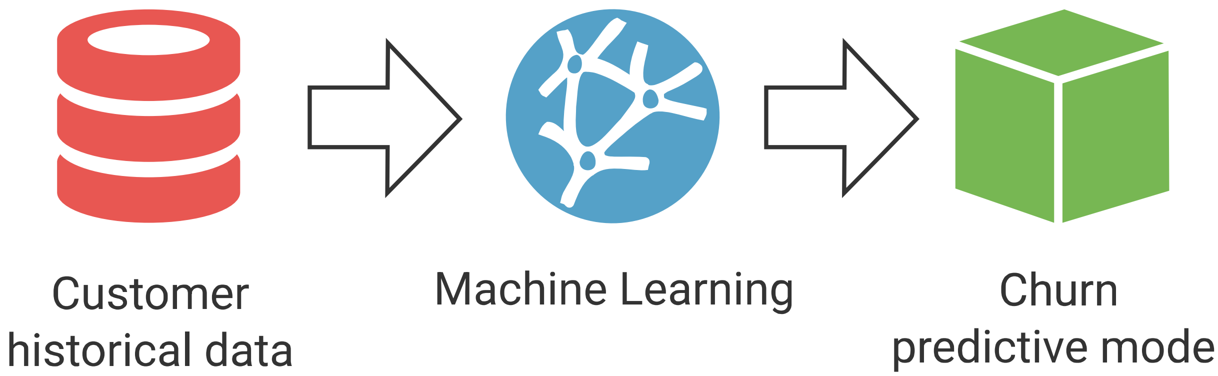


Table of Content:

Introduction

What is Customer Churn?

Customer churn is the rate at which customers discontinue their relationship with a company. In telecommunications, this rate is notably high—typically ranging from 15% to 25% annually—due to intense competition and customer preferences for better pricing, service quality, or additional features.

With millions of subscribers, telecom providers cannot realistically apply personalized retention strategies at scale. However, by leveraging data to identify high-risk customers, companies can target retention efforts more effectively. This strategic focus can reduce churn, strengthen customer loyalty, and enhance overall customer lifetime value (CLV).

Churn is more than a metric—it’s a critical indicator of business health. Retaining an existing customer is far less expensive than acquiring a new one. Companies that minimize churn benefit from lower acquisition costs, stronger brand loyalty, and higher profitability.

Effective churn reduction depends on early detection. This requires a holistic view of customer behavior across in-store activity, product usage, customer service interactions, digital engagement, and social media trends. By acting on these insights, telecom companies can preserve market share and create a foundation for sustainable growth.

Objectives of This Analysis:

This analysis explores customer churn within a telecom dataset and aims to answer the following questions:

- What percentage of customers have churned versus those who remain active?
- Are there identifiable churn patterns based on gender?
- Does churn vary by the type of service subscribed to?
- Which service types are the most profitable?
- What features and services contribute most significantly to profitability?

Loading Libraries and Data

```python
#Importing Libraries
import pandas as pd
import numpy as np
import missingno as msno
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
import math
import warnings
warnings.filterwarnings('ignore')

from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder

from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from xgboost import XGBClassifier
# from catboost import CatBoostClassifier
from sklearn import metrics
from sklearn.metrics import roc_curve
from sklearn.metrics import recall_score, confusion_matrix, precision_score, f1_score, accuracy_score, classification_rep
```

```python
#Importing Data
df = pd.read_csv(r'/work/telco.csv')
df.head()
```

	Customer ID obje...	Gender object	Age int64	Under 30 object	Senior Citizen ob...	Married object	Dependents object	N
0	8779-QRDMV	Male	78	No	Yes	No	No	
1	7495-OOKFY	Female	74	No	Yes	Yes	Yes	
2	1658-BYGOY	Male	71	No	Yes	No	Yes	
3	4598-XLKNJ	Female	78	No	Yes	Yes	Yes	
4	4846-WHAFZ	Female	80	No	Yes	Yes	Yes	

5 rows, 50 cols 10 / page Page 1 of 1

- based on the just a quick look at the data, we can see that there's some cleaning is

Understanding the Data

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 50 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   Customer ID                        7043 non-null   object
 1   Gender                             7043 non-null   object
 2   Age                                7043 non-null   int64
 3   Under 30                           7043 non-null   object
 4   Senior Citizen                     7043 non-null   object
 5   Married                            7043 non-null   object
 6   Dependents                         7043 non-null   object
 7   Number of Dependents               7043 non-null   int64
 8   Country                            7043 non-null   object
 9   State                              7043 non-null   object
 10  City                               7043 non-null   object
 11  Zip Code                           7043 non-null   int64
 12  Latitude                           7043 non-null   float64
 13  Longitude                          7043 non-null   float64
 14  Population                         7043 non-null   int64
 15  Quarter                            7043 non-null   object
 16  Referred a Friend                  7043 non-null   object
 17  Number of Referrals                7043 non-null   int64
 18  Tenure in Months                   7043 non-null   int64
 19  Offer                              3166 non-null   object
 20  Phone Service                      7043 non-null   object
 21  Avg Monthly Long Distance Charges  7043 non-null   float64
 22  Multiple Lines                     7043 non-null   object
 23  Internet Service                   7043 non-null   object
 24  Internet Type                      5517 non-null   object
```

```
display(df.shape)
```

```
(7043, 50)
```

```
display(df.dtypes)
```

```
Customer ID                          object
Gender                               object
Age                                   int64
Under 30                             object
Senior Citizen                       object
Married                              object
Dependents                           object
Number of Dependents                  int64
Country                              object
State                                object
City                                 object
Zip Code                              int64
Latitude                            float64
Longitude                           float64
Population                            int64
Quarter                              object
Referred a Friend                    object
Number of Referrals                   int64
Tenure in Months                      int64
Offer                                object
Phone Service                        object
Avg Monthly Long Distance Charges   float64
Multiple Lines                       object
Internet Service                     object
Internet Type                        object
Avg Monthly GB Download               int64
Online Security                      object
Online Backup                        object
Device Protection Plan               object
Premium Tech Support                 object
Streaming TV                         object
Streaming Movies                     object
```

```
display(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 50 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   Customer ID                        7043 non-null   object
 1   Gender                             7043 non-null   object
 2   Age                                7043 non-null   int64
 3   Under 30                           7043 non-null   object
 4   Senior Citizen                     7043 non-null   object
 5   Married                            7043 non-null   object
 6   Dependents                         7043 non-null   object
 7   Number of Dependents               7043 non-null   int64
 8   Country                            7043 non-null   object
 9   State                              7043 non-null   object
 10  City                               7043 non-null   object
 11  Zip Code                           7043 non-null   int64
 12  Latitude                           7043 non-null   float64
 13  Longitude                          7043 non-null   float64
 14  Population                         7043 non-null   int64
 15  Quarter                            7043 non-null   object
 16  Referred a Friend                  7043 non-null   object
 17  Number of Referrals                7043 non-null   int64
 18  Tenure in Months                   7043 non-null   int64
 19  Offer                              3166 non-null   object
 20  Phone Service                      7043 non-null   object
 21  Avg Monthly Long Distance Charges  7043 non-null   float64
 22  Multiple Lines                     7043 non-null   object
 23  Internet Service                   7043 non-null   object
 24  Internet Type                      5517 non-null   object
```

```
None
```

```
display(df.isnull().sum())
```

```
Customer ID                          0
Gender                               0
Age                                  0
Under 30                             0
Senior Citizen                       0
Married                              0
Dependents                           0
Number of Dependents                 0
Country                              0
State                                0
City                                 0
Zip Code                             0
Latitude                             0
Longitude                            0
Population                           0
Quarter                              0
Referred a Friend                    0
Number of Referrals                  0
Tenure in Months                     0
Offer                             3877
Phone Service                        0
Avg Monthly Long Distance Charges    0
Multiple Lines                       0
Internet Service                     0
Internet Type                     1526
Avg Monthly GB Download              0
Online Security                      0
Online Backup                        0
Device Protection Plan               0
Premium Tech Support                 0
Streaming TV                         0
Streaming Movies                     0
```

```
# Fill missing values for churn-related and service-related columns
fill_values = {
    'Churn Category': 'not churned',
    'Churn Reason': 'not churned',
    'Offer': 'None',
    'Internet Type': 'None'
}
df.fillna(value=fill_values, inplace=True)

# Print shape and preview data
print(df.shape)
df.head()
```

```
(7043, 50)
```

	Customer ID obje...	Gender object	Age int64	Under 30 object	Senior Citizen ob...	Married object	Dependents object	N
0	8779-QRDMV	Male	78	No	Yes	No	No	
1	7495-OOKFY	Female	74	No	Yes	Yes	Yes	
2	1658-BYGOY	Male	71	No	Yes	No	Yes	
3	4598-XLKNJ	Female	78	No	Yes	Yes	Yes	
4	4846-WHAFZ	Female	80	No	Yes	Yes	Yes	

5 rows, 50 cols 10 / page Page 1 of 1

```python
df.shape
df.dtypes
df.info()
df.isnull().sum()
df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 50 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   Customer ID                        7043 non-null   object
 1   Gender                             7043 non-null   object
 2   Age                                7043 non-null   int64
 3   Under 30                           7043 non-null   object
 4   Senior Citizen                     7043 non-null   object
 5   Married                            7043 non-null   object
 6   Dependents                         7043 non-null   object
 7   Number of Dependents               7043 non-null   int64
 8   Country                            7043 non-null   object
 9   State                              7043 non-null   object
 10  City                               7043 non-null   object
 11  Zip Code                           7043 non-null   int64
 12  Latitude                           7043 non-null   float64
 13  Longitude                          7043 non-null   float64
 14  Population                         7043 non-null   int64
 15  Quarter                            7043 non-null   object
 16  Referred a Friend                  7043 non-null   object
 17  Number of Referrals                7043 non-null   int64
 18  Tenure in Months                   7043 non-null   int64
 19  Offer                              7043 non-null   object
 20  Phone Service                      7043 non-null   object
 21  Avg Monthly Long Distance Charges  7043 non-null   float64
 22  Multiple Lines                     7043 non-null   object
 23  Internet Service                   7043 non-null   object
 24  Internet Type                      7043 non-null   object
```

	Customer ID obje...	Gender object	Age int64	Under 30 object	Senior Citizen ob...	Married object	Dependents object	N
0	8779-QRDMV	Male	78	No	Yes	No	No	
1	7495-OOKFY	Female	74	No	Yes	Yes	Yes	
2	1658-BYGOY	Male	71	No	Yes	No	Yes	
3	4598-XLKNJ	Female	78	No	Yes	Yes	Yes	
4	4846-WHAFZ	Female	80	No	Yes	Yes	Yes	

5 rows, 50 cols 10 / page Page 1 of 1

```
df.nunique()
```

```
Customer ID                          7043
Gender                                  2
Age                                    62
Under 30                                2
Senior Citizen                          2
Married                                 2
Dependents                              2
Number of Dependents                   10
Country                                 1
State                                   1
City                                 1106
Zip Code                             1626
Latitude                             1626
Longitude                            1625
Population                           1569
Quarter                                 1
Referred a Friend                       2
Number of Referrals                    12
Tenure in Months                       72
Offer                                   6
Phone Service                           2
Avg Monthly Long Distance Charges    3584
Multiple Lines                          2
Internet Service                        2
Internet Type                           4
Avg Monthly GB Download                50
Online Security                         2
Online Backup                           2
Device Protection Plan                  2
Premium Tech Support                    2
Streaming TV                            2
Streaming Movies                        2
```

The data set includes information about:

- Customers who left within the last month – the column is called Churn
- Services that each customer has signed up for – phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
- Customer account information - how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges
- Demographic info about customers – gender, age range, and if they have partners and dependents
- Based on the checking the data; there are no null values found,

Data Manipulation

```python
# Step-by-step data manipulation and feature engineering

import numpy as np
import pandas as pd




# 1. Define Yes/No conversion function
def yes_no_to_bool(val):
    if pd.isna(val):
        return np.nan
    return str(val).strip().lower() == 'yes'

# 2. Convert 'Yes'/'No' and similar to boolean
yes_no_columns = [
    'Under 30', 'Senior Citizen', 'Married', 'Dependents', 'Referred a Friend',
    'Phone Service', 'Multiple Lines', 'Online Security', 'Online Backup',
    'Device Protection Plan', 'Premium Tech Support', 'Streaming TV',
    'Streaming Movies', 'Streaming Music', 'Unlimited Data', 'Paperless Billing'
]

df[yes_no_columns] = df[yes_no_columns].applymap(yes_no_to_bool)

# 3. Create feature: Total number of add-on services subscribed
addon_services = [
    'Online Security', 'Online Backup', 'Device Protection Plan',
    'Premium Tech Support', 'Streaming TV', 'Streaming Movies', 'Streaming Music'
]

df['Total Addon Services'] = df[addon_services].sum(axis=1)



# 4. Create 'Tenure in Years' by binning 'Tenure in Months'
df['Tenure in Years'] = pd.cut(
    df['Tenure in Months'],
    bins=range(0, 73, 12),
    labels=range(1, 7)
)

# 5. Filter churned customers
churned_df = df[df['Churn Label'] == 'Yes']


# 6. Summary statistics (selected numerical features)
summary_stats = df[['Monthly Charge', 'Total Charges', 'CLTV', 'Tenure in Months']].describe()

# 7. Prepare cleaned dataset for ML
ml_df = df.drop(columns=[
    'Customer ID', 'Country', 'State', 'City', 'Zip Code', 'Latitude', 'Longitude',
    'Churn Reason', 'Churn Category', 'Quarter'
])

# Convert object columns to categorical and handle missing values
for col in ml_df.select_dtypes(include='object').columns:
    ml_df[col] = ml_df[col].astype('category')
    if 'None' not in ml_df[col].cat.categories:
        ml_df[col] = ml_df[col].cat.add_categories('None')
    ml_df[col] = ml_df[col].fillna('None')

# 8. Final info summary
cleaned_shape = df.shape
churned_shape = churned_df.shape
ml_ready_shape = ml_df.shape

# Show outputs
display(summary_stats, cleaned_shape, churned_shape, ml_ready_shape)
```

	Monthly Charge f...	Total Charges flo...	CLTV float64	Tenure in Months f.
cou...	7043	7043	7043	7043
me...	64.76169246	2280.381264	4400.295755	32.386767
std	30.0900471	2266.220462	1183.057152	24.54206101
min	18.25	18.8	2003	1
25%	35.5	400.15	3469	9
50%	70.35	1394.55	4527	29

75%	89.85	3786.6	5380.5	55
max	118.75	8684.8	6500	72

8 rows, 4 cols 10 / page Page 1 of 1

```
(7043, 52)
```

```
(1869, 52)
```

```
(7043, 42)
```

```
ml_df.to_csv("ml_ready_telco.csv", index=False)
```

🔧Transformations:

1. Converted Yes/No columns to boolean (True/False)
2. Engineered Feature: Total Addon Services (sum of optional services)
3. Filtered churned customers → 1,869 out of 7,043
4. Generated Summary Stats (e.g., Monthly Charge avg = ~$64.76)
5. Prepared ML-ready dataset:
   - Removed ID/location columns
   - Handled missing values
   - Cast object fields to categorical

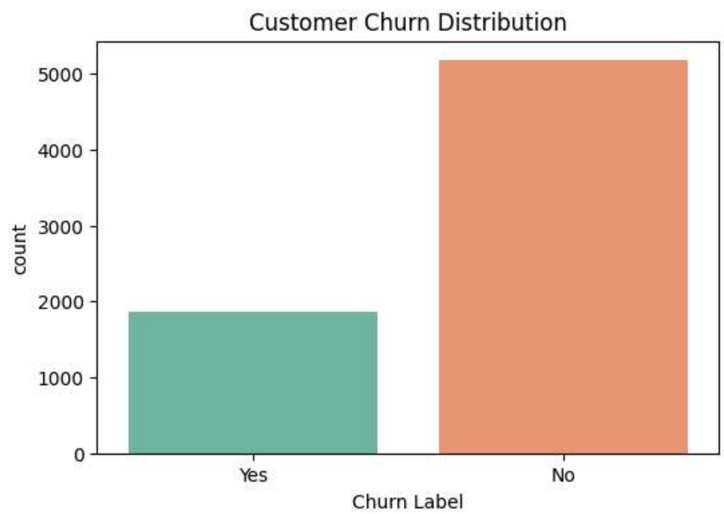
Summary:

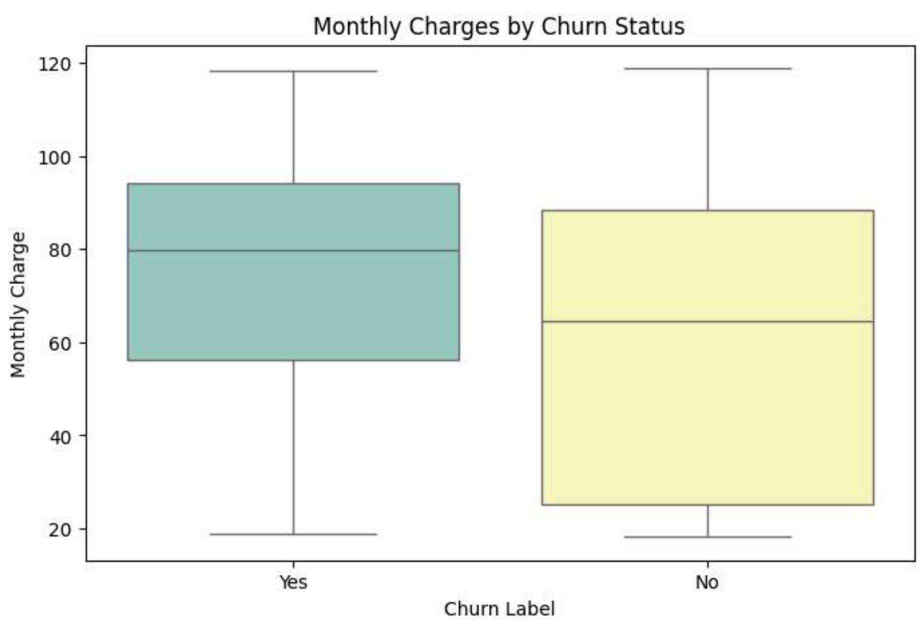
- Original shape: (7043, 50)
- Churned subset: (1869, 52)
- ML-ready shape: (7043, 42)

Data Visualization

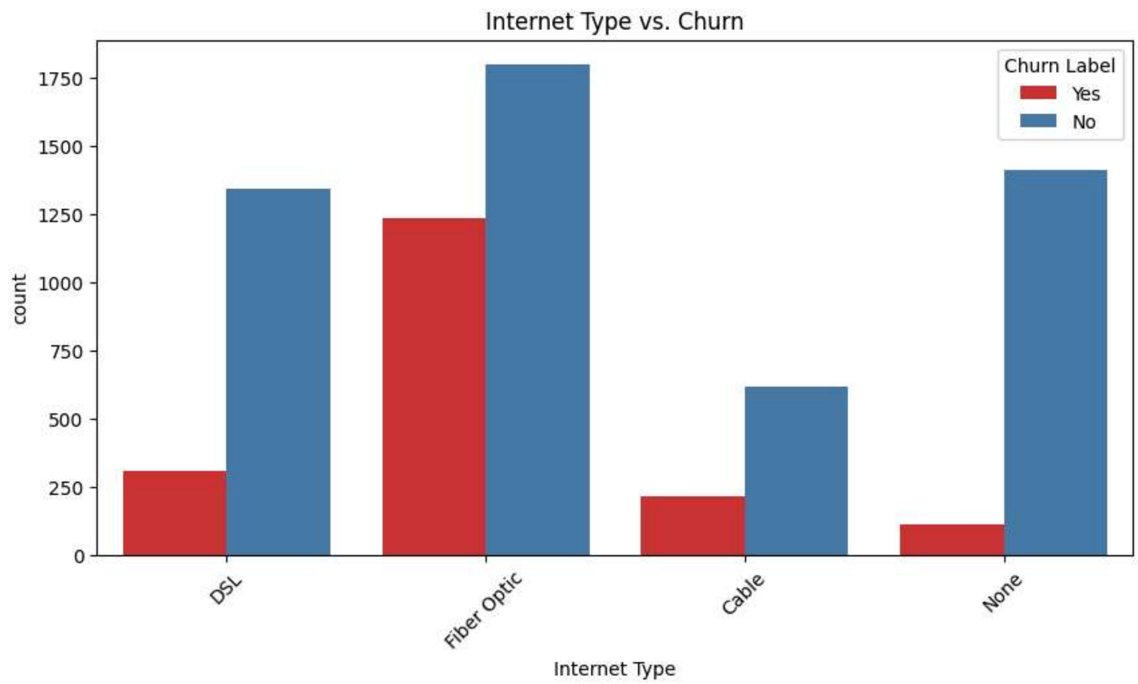
```python
# Visualization: Churn distribution
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x='Churn Label', palette='Set2')
plt.title("Customer Churn Distribution")
plt.show()

# Visualization: Monthly Charges by Churn
plt.figure(figsize=(8, 5))
sns.boxplot(data=df, x='Churn Label', y='Monthly Charge', palette='Set3')
plt.title("Monthly Charges by Churn Status")
plt.show()
```

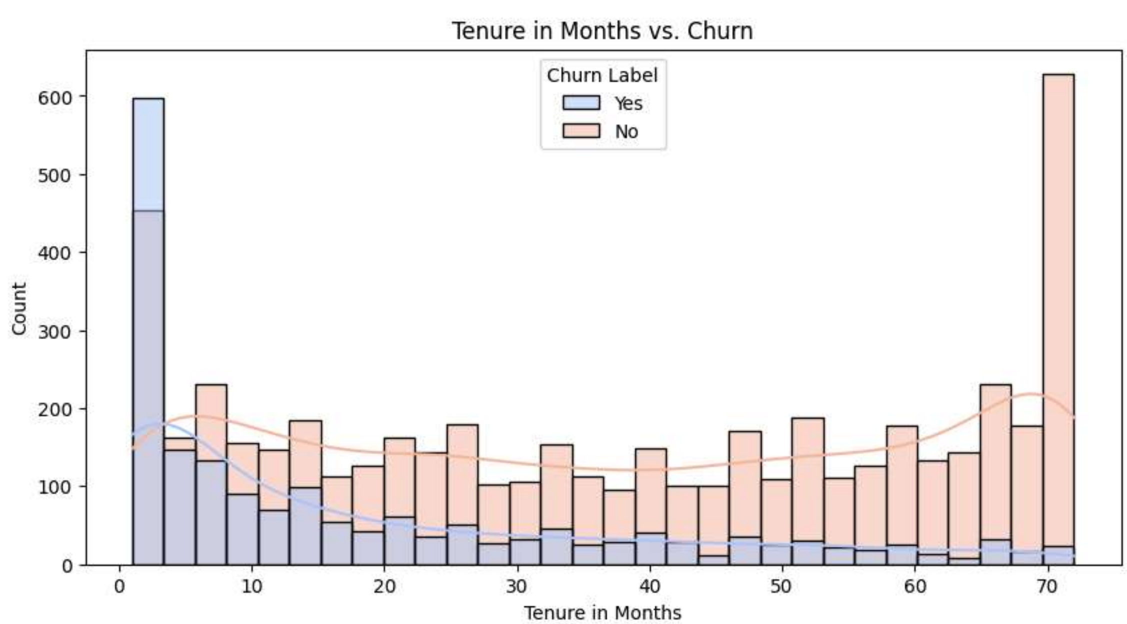




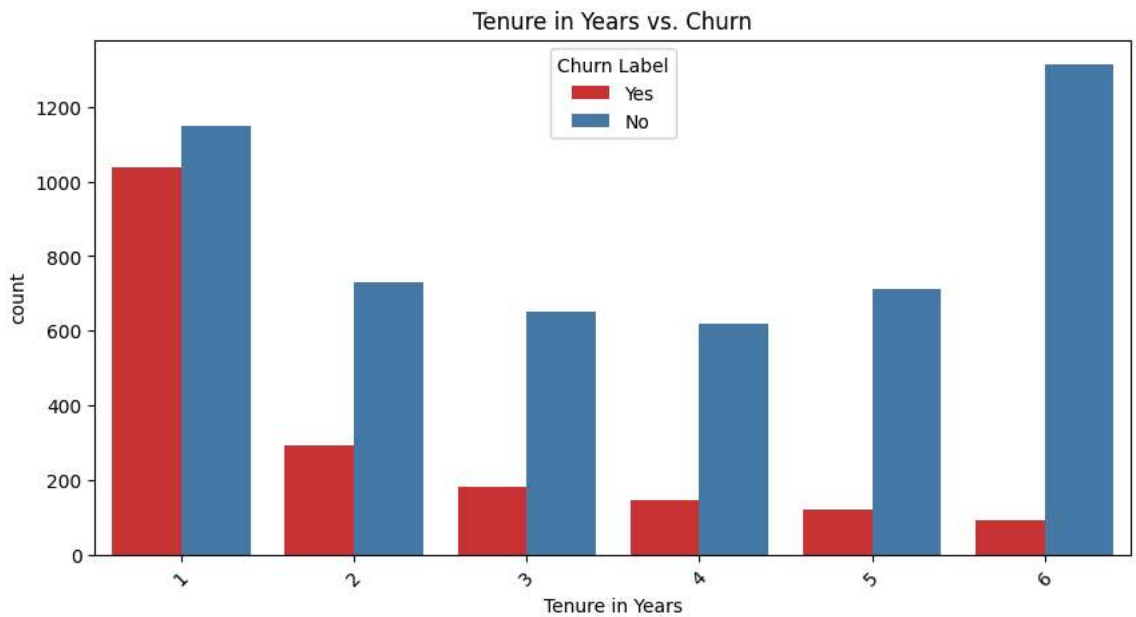
```
# 1. Internet Type distribution by Churn
plt.figure(figsize=(10, 5))
sns.countplot(data=df, x='Internet Type', hue='Churn Label', palette='Set1')
plt.title("Internet Type vs. Churn")
plt.xticks(rotation=45)
plt.show()
```



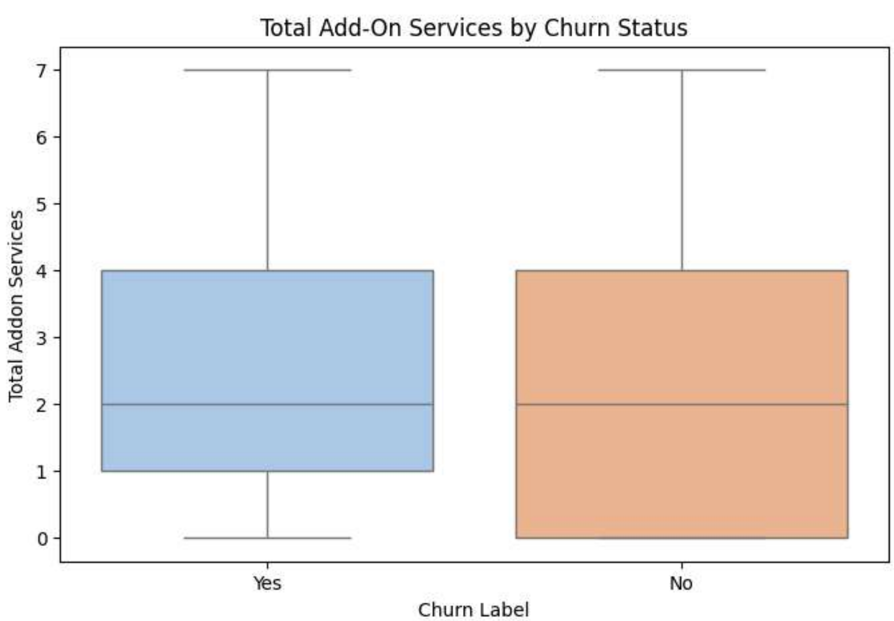

```
# 2. Tenure distribution by churn
plt.figure(figsize=(10, 5))
sns.histplot(data=df, x='Tenure in Months', hue='Churn Label', kde=True, bins=30, palette='coolwarm')
plt.title("Tenure in Months vs. Churn")
plt.show()
```



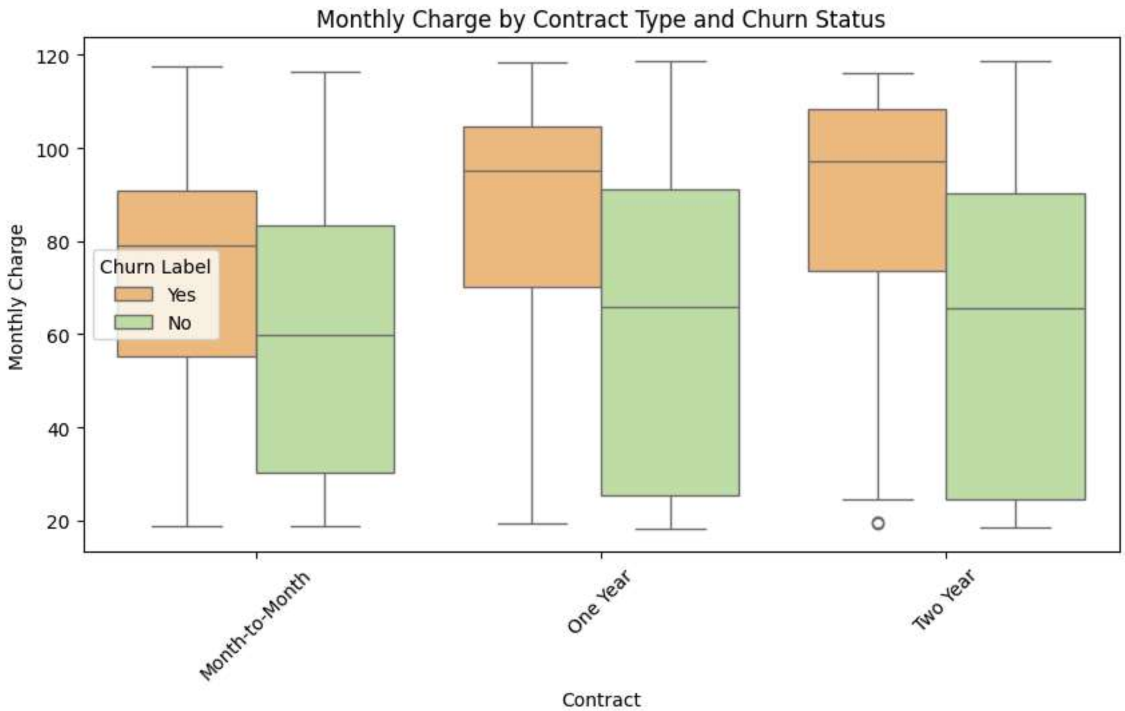
```python
# 1. Internet Type distribution by Churn
plt.figure(figsize=(10, 5))
sns.countplot(data=df, x='Tenure in Years', hue='Churn Label', palette='Set1')
plt.title("Tenure in Years vs. Churn")
plt.xticks(rotation=45)
plt.show()
```



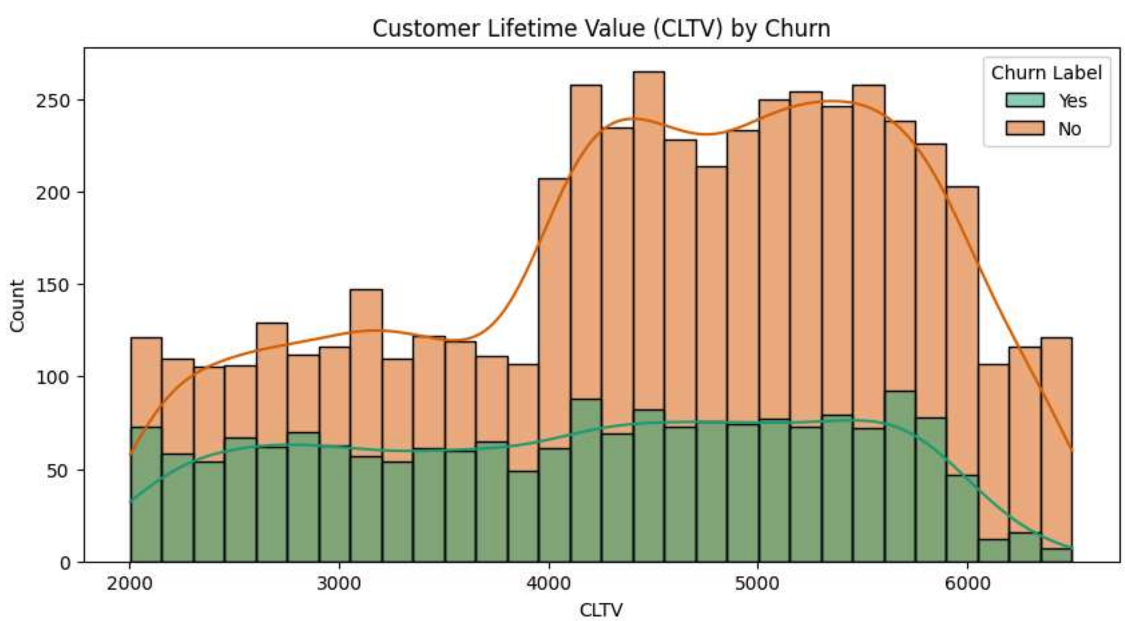

```python
# 4. Addon Services vs Churn
plt.figure(figsize=(8, 5))
sns.boxplot(data=df, x='Churn Label', y='Total Addon Services', palette='pastel')
plt.title("Total Add-On Services by Churn Status")
plt.show()
```



```python
# 5. Monthly Charge by Contract Type and Churn
plt.figure(figsize=(10, 5))
sns.boxplot(data=df, x='Contract', y='Monthly Charge', hue='Churn Label', palette='Spectral')
plt.title("Monthly Charge by Contract Type and Churn Status")
plt.xticks(rotation=45)
plt.show()
```



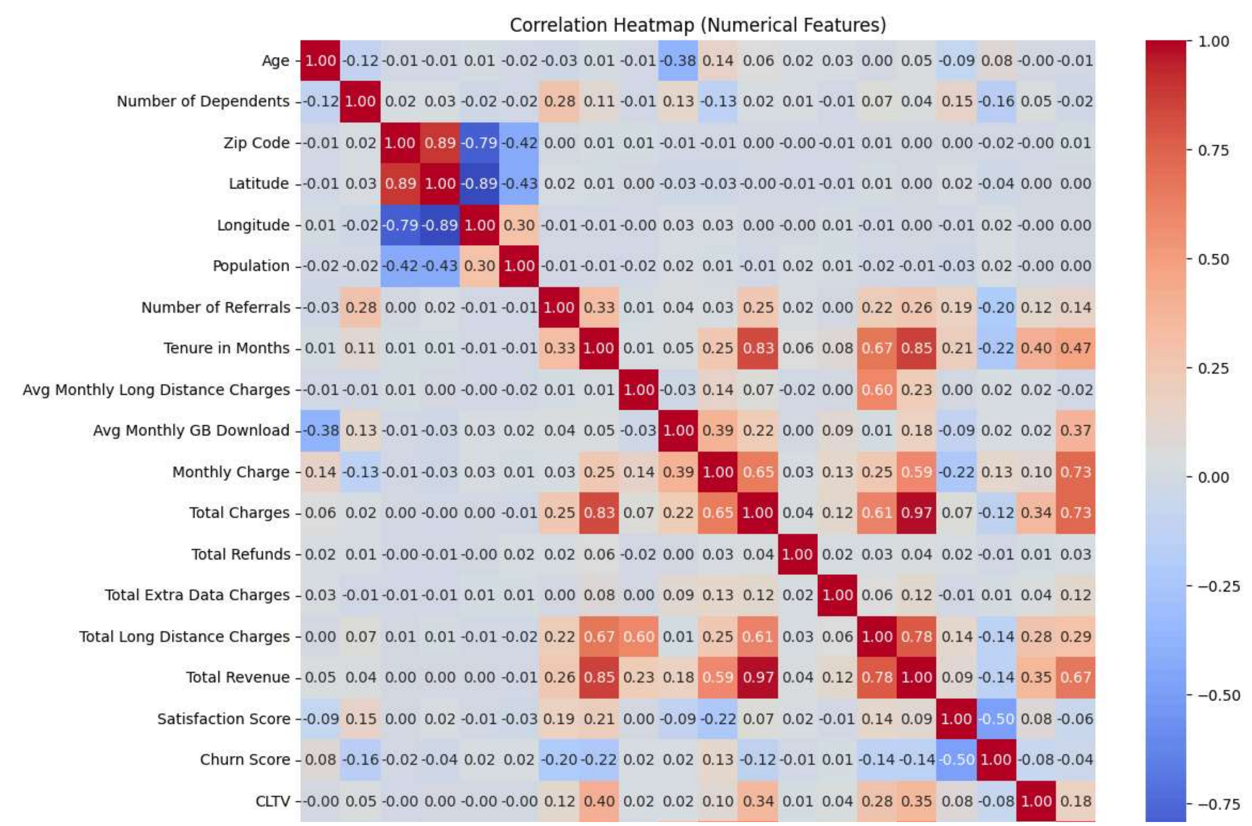

```python
# 6. CLTV (Customer Lifetime Value) by Churn
plt.figure(figsize=(10, 5))
sns.histplot(data=df, x='CLTV', hue='Churn Label', bins=30, kde=True, palette='Dark2')
plt.title("Customer Lifetime Value (CLTV) by Churn")
plt.show()
```



Visual Insights:

1. Internet Type vs. Churn:
   - Customers with Fiber Optic internet churn more often than those with DSL or None.
2. Tenure vs. Churn:
   - Churned customers typically have shorter tenure (often < 20 months).
3. Add-On Services vs. Churn:
   - Those who churn tend to have fewer additional services.
4. Contract Type and Monthly Charges:
   - Month-to-month customers have higher churn, while Two Year contract holders are more stable and have lower churn.
5. Customer Lifetime Value (CLTV):
   - Customers with lower CLTV are more likely to churn.

```python
# 1. Correlation heatmap (numeric features only)
plt.figure(figsize=(12, 10))
corr = df.select_dtypes(include=[np.number]).corr()
sns.heatmap(corr, annot=True, fmt=".2f", cmap="coolwarm")
plt.title("Correlation Heatmap (Numerical Features)")
plt.savefig("corr1.png",dpi=300)
plt.show()
```



🧪 Correlation Heatmap:

```
* Monthly Charge correlates with CLTV and number of Add-On Services.

* Tenure also strongly affects CLTV, as expected.
```

```python
# Plot bar charts (histograms) for numerical columns to show distribution
num_cols = df.select_dtypes(include=['int64', 'float64']).columns
n = len(num_cols)

cols = 3
rows = math.ceil(n / cols)

plt.figure(figsize=(5 * cols, 4 * rows))

for i, col in enumerate(num_cols):
    plt.subplot(rows, cols, i + 1)
    plt.hist(df[col].dropna(), bins=20, color='skyblue', edgecolor='black')
    plt.title(col)
    plt.xlabel(col)
    plt.ylabel("Frequency")

plt.tight_layout()
plt.show()
```