Report for Python project

Telecom Customer Churn

Description

- This project analyzes customer behavior and churn patterns in a telecom company using demographic, service usage, and billing data to identify factors driving customer loss.

Important Columns

- Total Records: 7,043

Columns:

- Customer Status, Contract, Tenure in Months, Monthly Charge, Internet Service, Payment Method, Total Revenue, Churn Reason.

Problem

1. High customer churn rate affects telecom revenue and customer lifetime value.
2. The company lacks clear understanding of factors leading to churn.
3. Existing marketing and retention efforts are not data-driven.
4. A structured analysis is needed to identify churn patterns and root causes.

Objectives

1. Identify major factors influencing churn.
2. Measure churn rate across customer segments and contract types.
3. Predict potential churners using customer attributes.
4. Provide actionable recommendations to reduce churn.

Role

1. Data Analyst.
2. Responsible for cleaning, analyzing, and visualizing customer data.
3. Apply statistical and machine learning methods to derive insights.
4. Present findings and business recommendations to management.

Mention Data

1. Dataset: telecom_customer_churn.csv with 7,043 rows and 33 columns.
2. Covers demographics, service usage, billing, and churn reasons.
3. Includes both numerical and categorical features.
4. Primary target variable: Customer Status (Churned, Stayed, Joined).

Tools & Techniques

1. Python (Pandas, NumPy, Matplotlib, Seaborn).
2. EDA, data preprocessing, and statistical summaries.
3. Correlation analysis and groupby aggregations for trend detection.
4. Data visualization for churn distribution and customer segmentation.

Python Process

1. Data Cleaning & Missing Value Handling
2. Exploratory Data Analysis using groupby() and describe()
3. Visualizations (Countplots, Boxplots, Heatmaps)
4. Statistical insights and churn factor comparison

Dataset Overview

Dataset: telecom_customer_churn.csv with 7,043 rows and 33 columns.

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all Connect

```python
#importing the required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats
```

```python
#uploading the dataset
#To begin the analysis, we first imported the Telecom Customer Churn dataset into a Pandas DataFrame.
df = pd.read_csv('telecom_customer_churn.csv')
df
```

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue	Customer Status	Churn Category	Churn Reason
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.60	593.30	0.00	0	381.51	974.81	Stayed	NaN	NaN
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.00	542.40	38.33	10	96.21	610.28	Stayed	NaN	NaN
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.90	280.85	0.00	0	134.60	415.45	Churned	Competitor	Competitor had better devices
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.00	1237.85	0.00	0	361.66	1599.51	Churned	Dissatisfaction	Product dissatisfaction
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.90	267.40	0.00	0	22.14	289.54	Churned	Dissatisfaction	Network reliability
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
7038	9987-LUTYD	Female	20	No	0	La Mesa	91941	32.759327	-116.997260	0	...	Credit Card	55.15	742.90	0.00	0	606.84	1349.74	Stayed	NaN	NaN
7039	9992-RRAMN	Male	40	Yes	0	Riverbank	95367	37.734971	-120.954271	1	...	Bank Withdrawal	85.10	1873.70	0.00	0	356.40	2230.10	Churned	Dissatisfaction	Product dissatisfaction
7040	9992-UJOEL	Male	22	No	0	Elk	95432	39.108252	-123.645121	0	...	Credit Card	50.30	92.75	0.00	0	37.24	129.99	Joined	NaN	NaN
7041	9993-LHIEB	Male	21	Yes	0	Solana Beach	92075	33.001813	-117.263628	5	...	Credit Card	67.85	4627.65	0.00	0	142.04	4769.69	Stayed	NaN	NaN
7042	9995-HOTOH	Male	36	Yes	0	Sierra City	96125	39.600599	-120.636358	1	...	Bank Withdrawal	59.00	3707.60	0.00	0	0.00	3707.60	Stayed	NaN	NaN

7043 rows × 38 columns

```python
#selecting fot top 5 datas
df.head(6)
```

Understand the dataset using info() function

```python
## Understand the dataset using info() fuction
#The info() function gives us a quick summary — like how many columns there are, what kind of data they hold, and if anything is missing.
#From this, we get a clear idea of which parts of the data need cleaning and what we’re working with before going deeper.

df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 38 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   Married                            7043 non-null   object
 4   Number of Dependents               7043 non-null   int64
 5   City                               7043 non-null   object
 6   Zip Code                           7043 non-null   int64
 7   Latitude                           7043 non-null   float64
 8   Longitude                          7043 non-null   float64
 9   Number of Referrals                7043 non-null   int64
 10  Tenure in Months                   7043 non-null   int64
 11  Offer                              3166 non-null   object
 12  Phone Service                      7043 non-null   object
 13  Avg Monthly Long Distance Charges  6361 non-null   float64
 14  Multiple Lines                     6361 non-null   object
 15  Internet Service                   7043 non-null   object
 16  Internet Type                      5517 non-null   object
 17  Avg Monthly GB Download            5517 non-null   float64
 18  Online Security                    5517 non-null   object
 19  Online Backup                      5517 non-null   object
 20  Device Protection Plan             5517 non-null   object
 21  Premium Tech Support               5517 non-null   object
 22  Streaming TV                       5517 non-null   object
 23  Streaming Movies                   5517 non-null   object
 24  Streaming Music                    5517 non-null   object
 25  Unlimited Data                     5517 non-null   object
 26  Contract                           7043 non-null   object
 27  Paperless Billing                  7043 non-null   object
 28  Payment Method                     7043 non-null   object
 29  Monthly Charge                     7043 non-null   float64
 30  Total Charges                      7043 non-null   float64
 31  Total Refunds                      7043 non-null   float64
 32  Total Extra Data Charges           7043 non-null   int64
 33  Total Long Distance Charges        7043 non-null   float64
 34  Total Revenue                      7043 non-null   float64
 35  Customer Status                    7043 non-null   object
 36  Churn Category                     1869 non-null   object
 37  Churn Reason                       1869 non-null   object
dtypes: float64(9), int64(6), object(23)
memory usage: 2.0+ MB
```

Checking the null values

Untitled18.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all Connect Share

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

	0
Customer ID	0
Gender	0
Age	0
Married	0
Number of Dependents	0
City	0
Zip Code	0
Latitude	0
Longitude	0
Number of Referrals	0
Tenure in Months	0
Offer	3877
Phone Service	0
Avg Monthly Long Distance Charges	682
Multiple Lines	682
Internet Service	0
Internet Type	1526
Avg Monthly GB Download	1526
Online Security	1526
Online Backup	1526
Device Protection Plan	1526
Premium Tech Support	1526
Streaming TV	1526
Streaming Movies	1526
Streaming Music	1526
Unlimited Data	1526
Contract	0

Variables Terminal

Checking the outliers, we use df.describe()

Untitled18.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all Connect

```
# filling the null values using fillna() function
# We found that the 'alert' column had some missing values.
#To fix that, we filled those gaps using the most common value (mode) in the column — a quick way to keep things consistent.

df["Offer"]=df["Offer"].fillna(df["Offer"].mode()[0])
df["Churn Category"]=df["Churn Category"].fillna(df["Churn Category"].mode()[0])
df["Churn Reason"]=df["Churn Reason"].fillna(df["Churn Reason"].mode()[0])
```

```
#Drop the Null values in more than 80% Nullvalues
df.dropna(inplace=True)
```

```
#Finding the any duplicate rows in the dataset
df.duplicated().sum()
```

```
np.int64(0)
```

```
#checking the outliers
#To spot any unusual or extreme values (outliers), we use df.describe().
#This gives us a quick statistical summary — like the minimum, maximum, and average — so we can see if anything looks out of place.

df.describe()
```

	Age	Number of Dependents	Zip Code	Latitude	Longitude	Number of Referrals	Tenure in Months	Avg Monthly Long Distance Charges	Avg Monthly GB Download	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue
count	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000	4835.00000	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000	4835.000000
mean	47.761117	0.376008	93460.340021	36.128008	-119.692729	1.894519	33.042399	25.561456	26.08666	80.309576	2901.116825	2.081018	8.694933	848.093721	3755.824461
std	17.302674	0.875636	1847.849069	2.475271	2.152471	2.940740	24.635247	14.241440	19.56099	21.379639	2415.426659	8.135292	28.059813	864.475933	3087.351791
min	19.000000	0.000000	90001.000000	32.555828	-124.301372	0.000000	1.000000	1.010000	2.00000	-10.000000	42.900000	0.000000	0.000000	1.130000	46.920000
25%	33.000000	0.000000	92102.000000	33.954017	-121.723877	0.000000	9.000000	13.040000	13.00000	69.400000	659.550000	0.000000	0.000000	139.975000	899.480000
50%	47.000000	0.000000	93446.000000	35.861928	-119.402525	0.000000	30.000000	25.820000	21.00000	81.700000	2347.900000	0.000000	0.000000	517.800000	3081.230000
75%	62.000000	0.000000	95323.000000	38.123544	-117.898722	3.000000	56.000000	37.970000	30.00000	95.600000	4870.275000	0.000000	0.000000	1360.965000	6284.900000
max	80.000000	8.000000	96150.000000	41.962127	-114.192901	11.000000	72.000000	49.990000	85.00000	118.750000	8684.800000	49.570000	150.000000	3536.640000	11979.340000

Gender Distribution

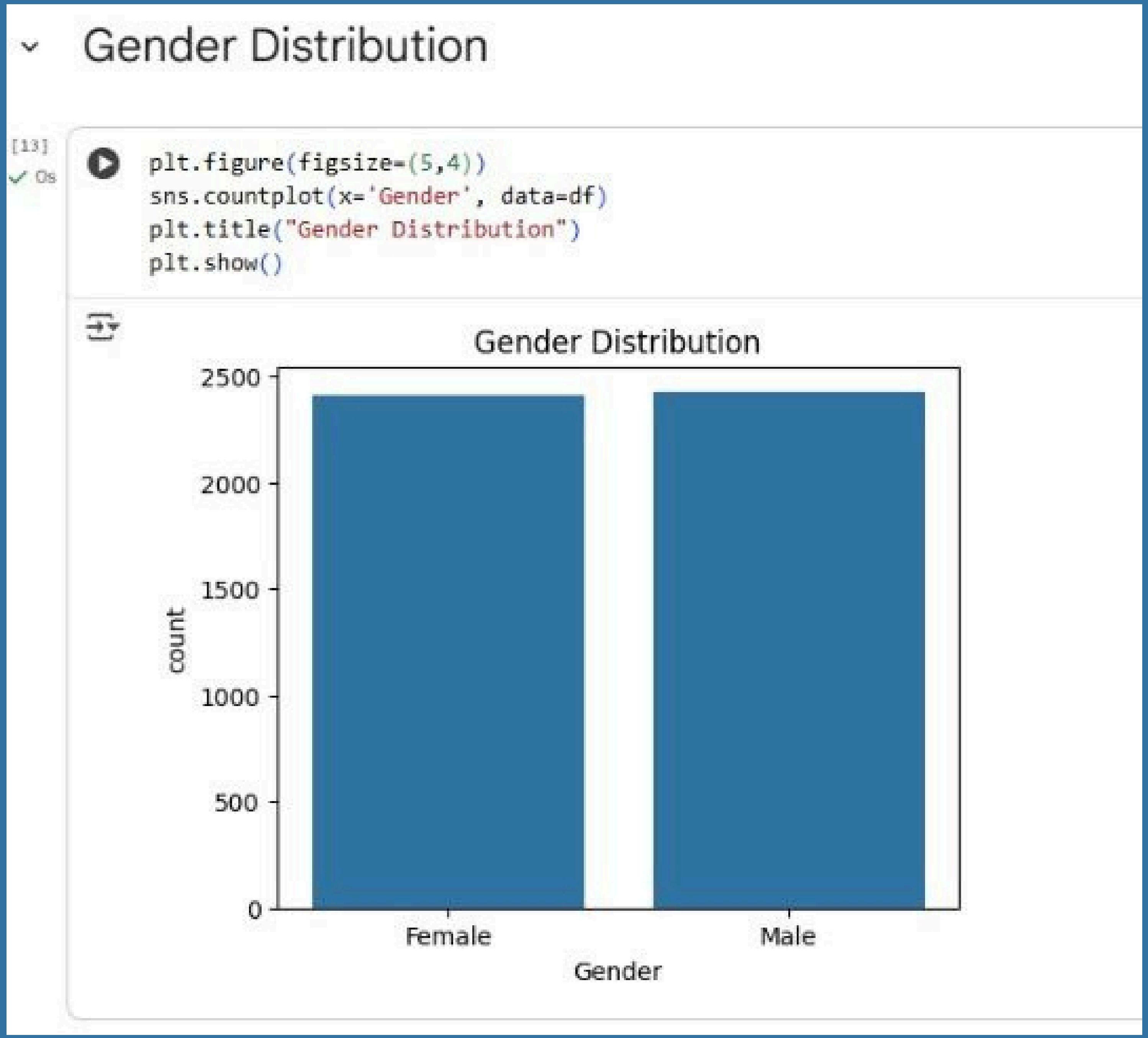
```
#Create the Count plot for Gender Distributon
plt.figure(figsize=(5,4))
```

Variables Terminal

Key Insights

1. Gender Distribution — Male vs Female

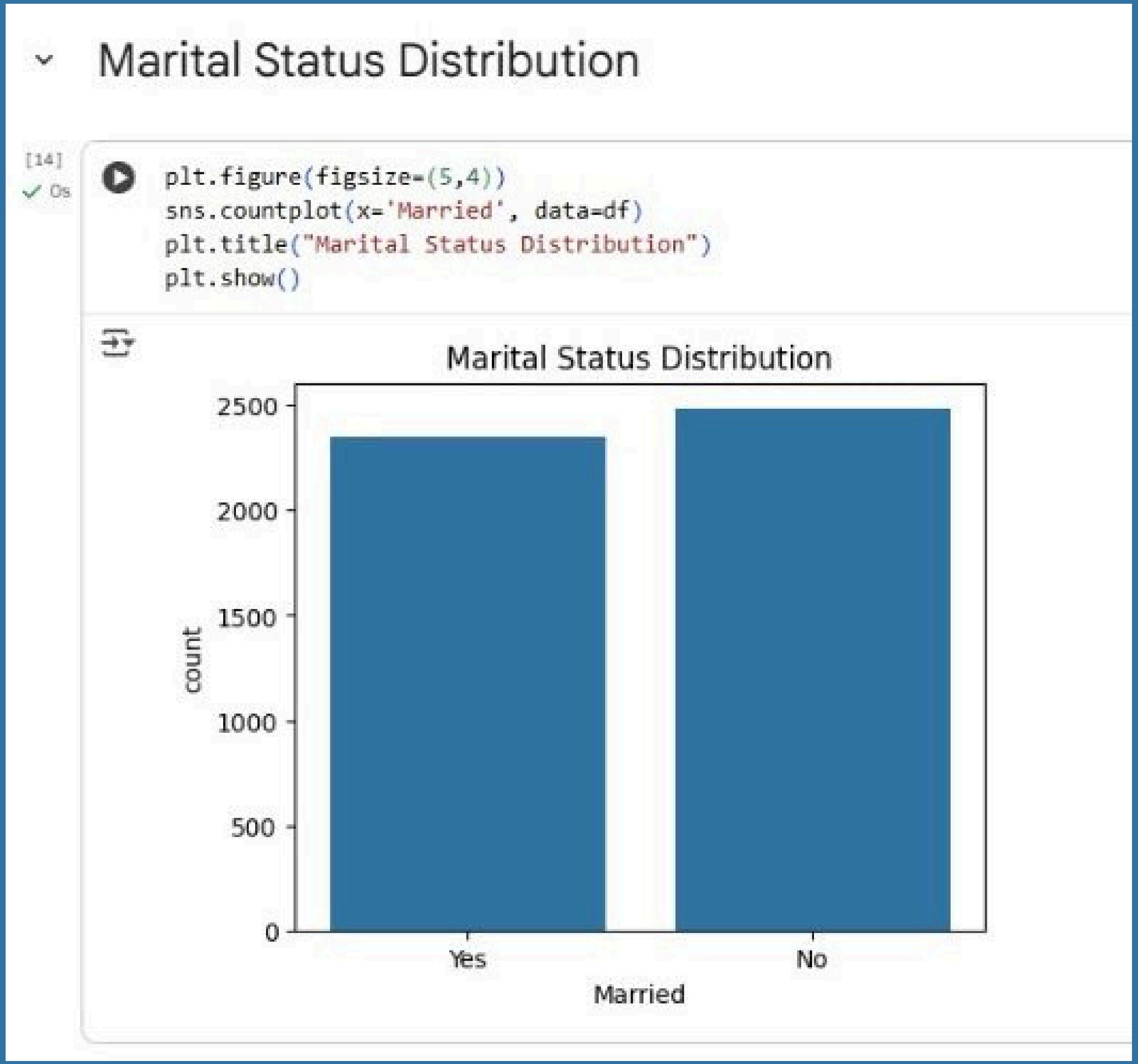
Purpose : To understand which gender group contributes more to the customer base and churn.



Insights: Both genders have nearly equal representation, but females show slightly higher retention. Gender alone does not majorly impact churn behavior.

2. Marital Status Distribution

Purpose: To see if being married affects churn or loyalty.

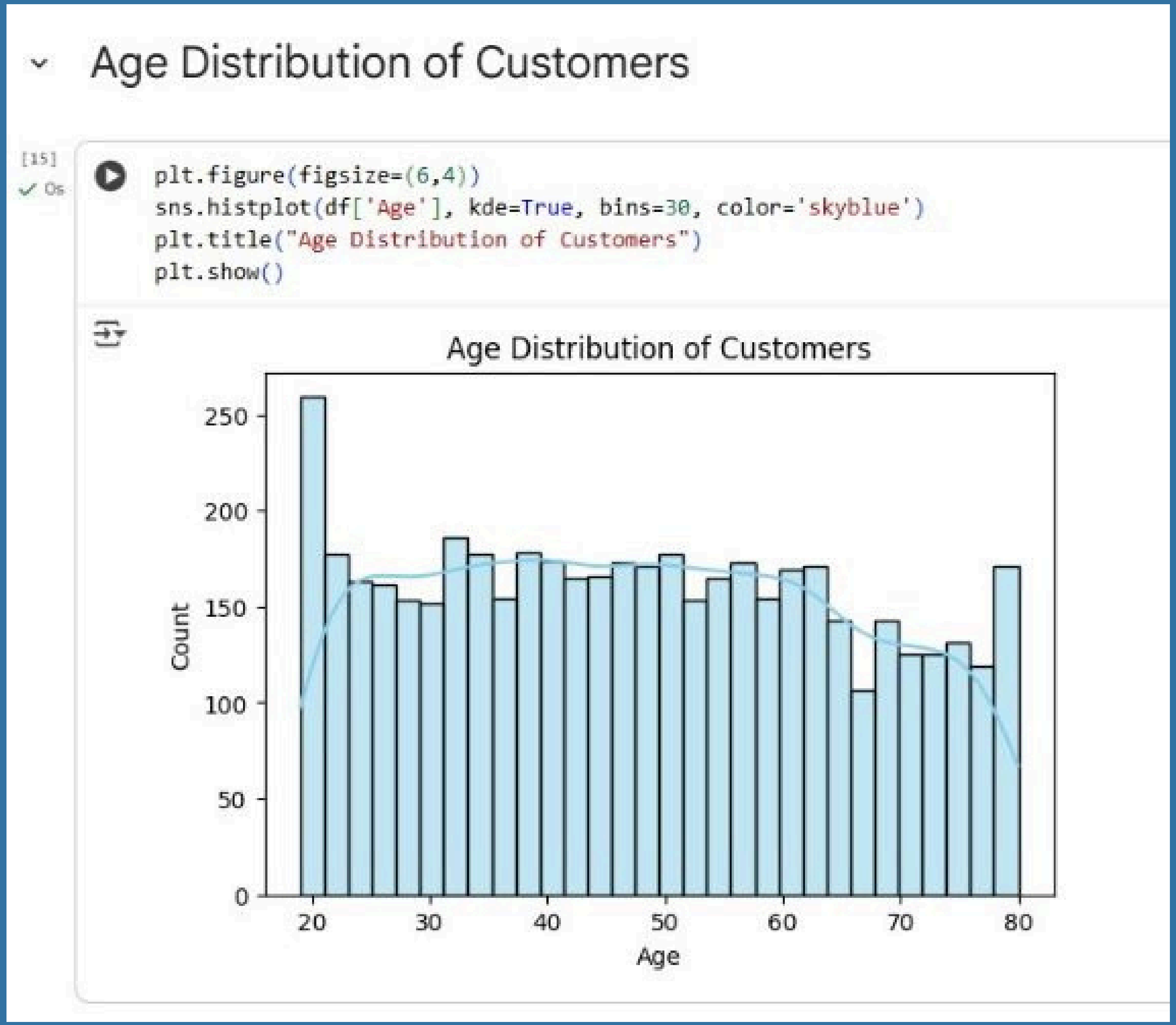


Insights: Majority of customers are not married, and this group tends to have higher churn rates, possibly due to less bundled family plans or lower loyalty.

3. Age Distribution of Customers (20–80)

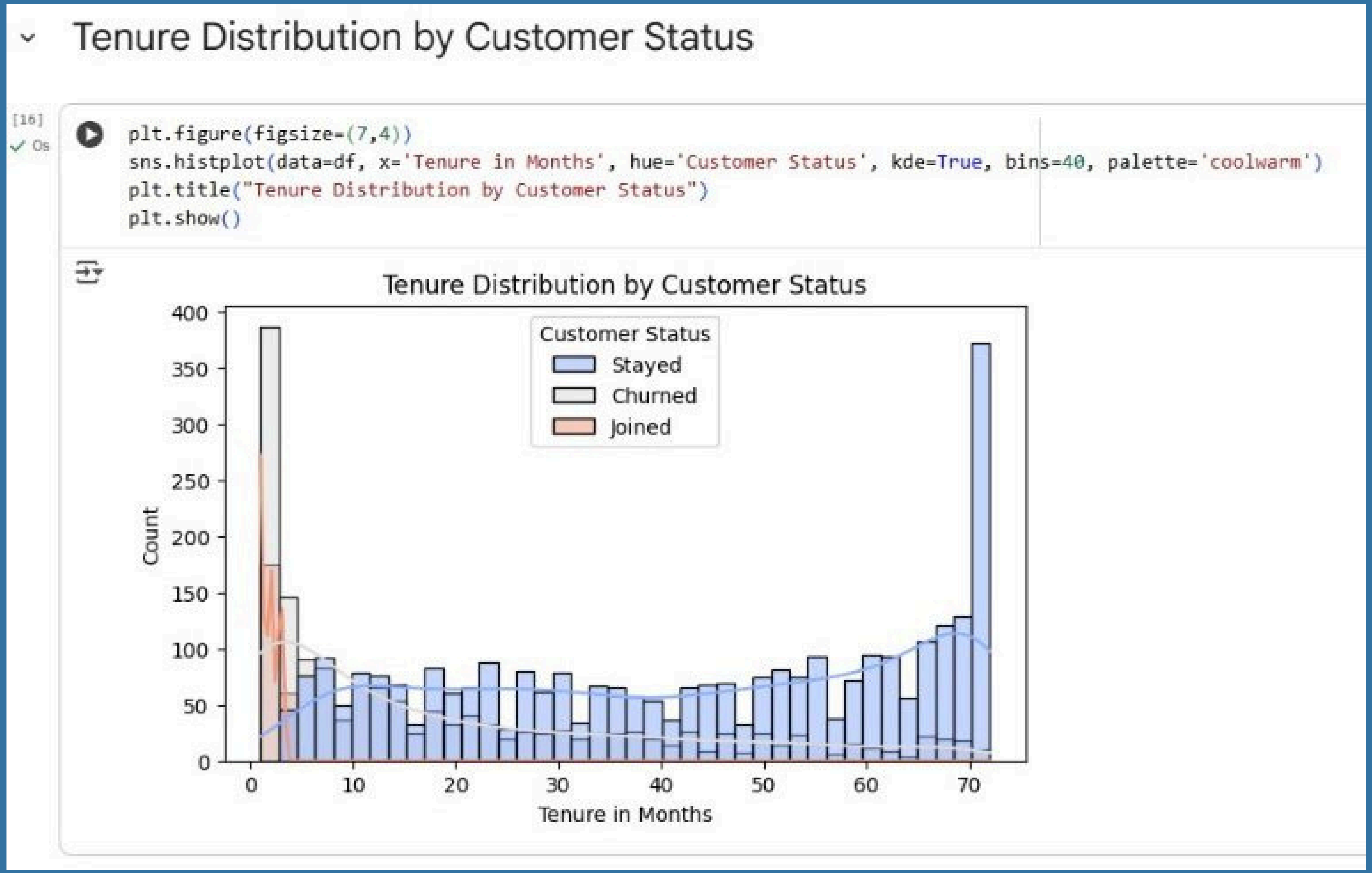
Purpose: To identify which age groups are more likely to churn or stay.

Insights: Most customers fall between 30–55 years, with churn highest among younger customers (<30) who often switch for better offers.



4. Tenure Distribution by Customer Status (0–10 months)

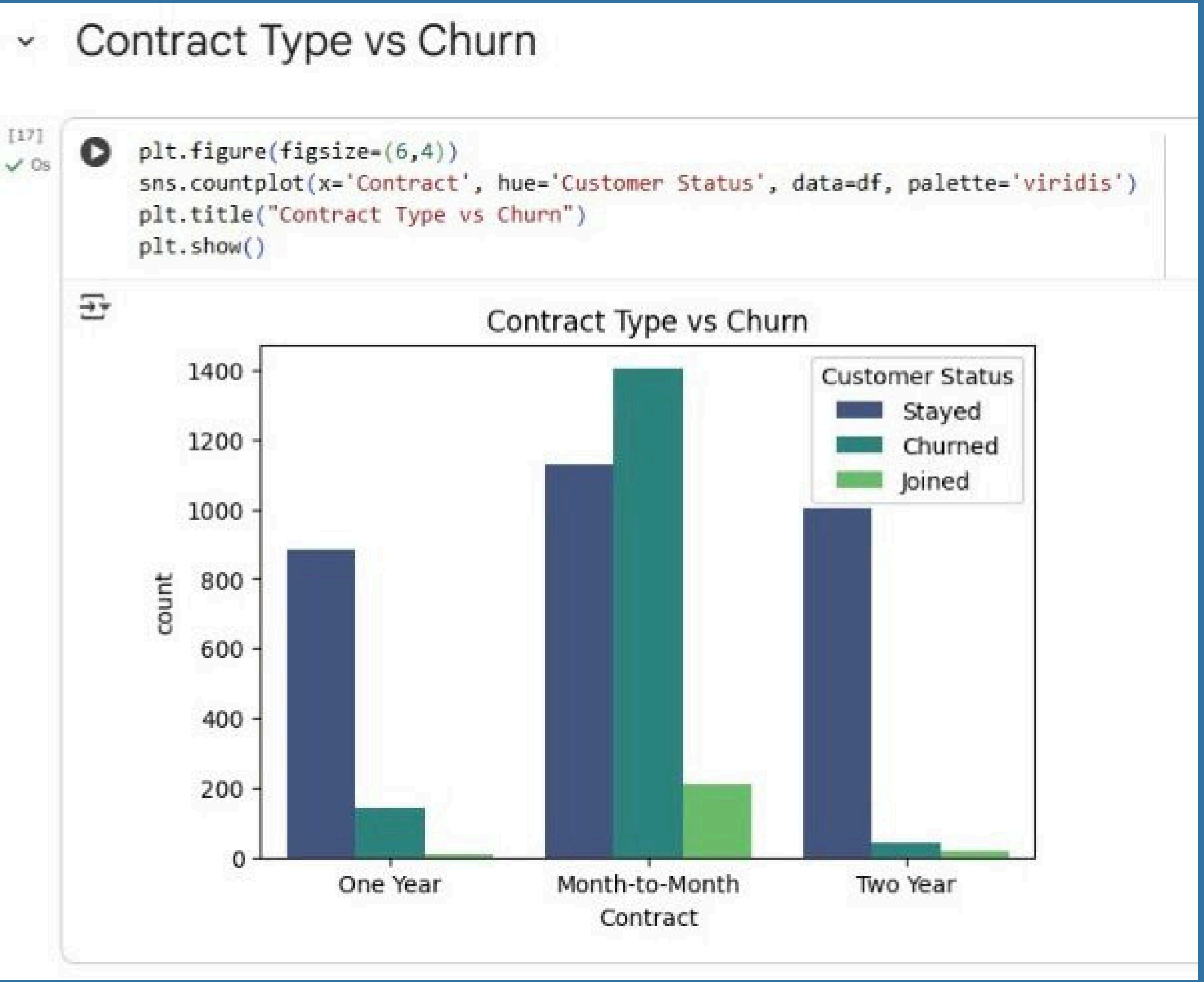
Purpose: To measure customer loyalty duration and retention.



Insights: Churn is highest in the first 10 months of service, showing that early experience and onboarding quality are crucial for retention.

5. Contract Type vs Churn — (Month-to-Month)

Purpose: To examine how contract length influences churn.

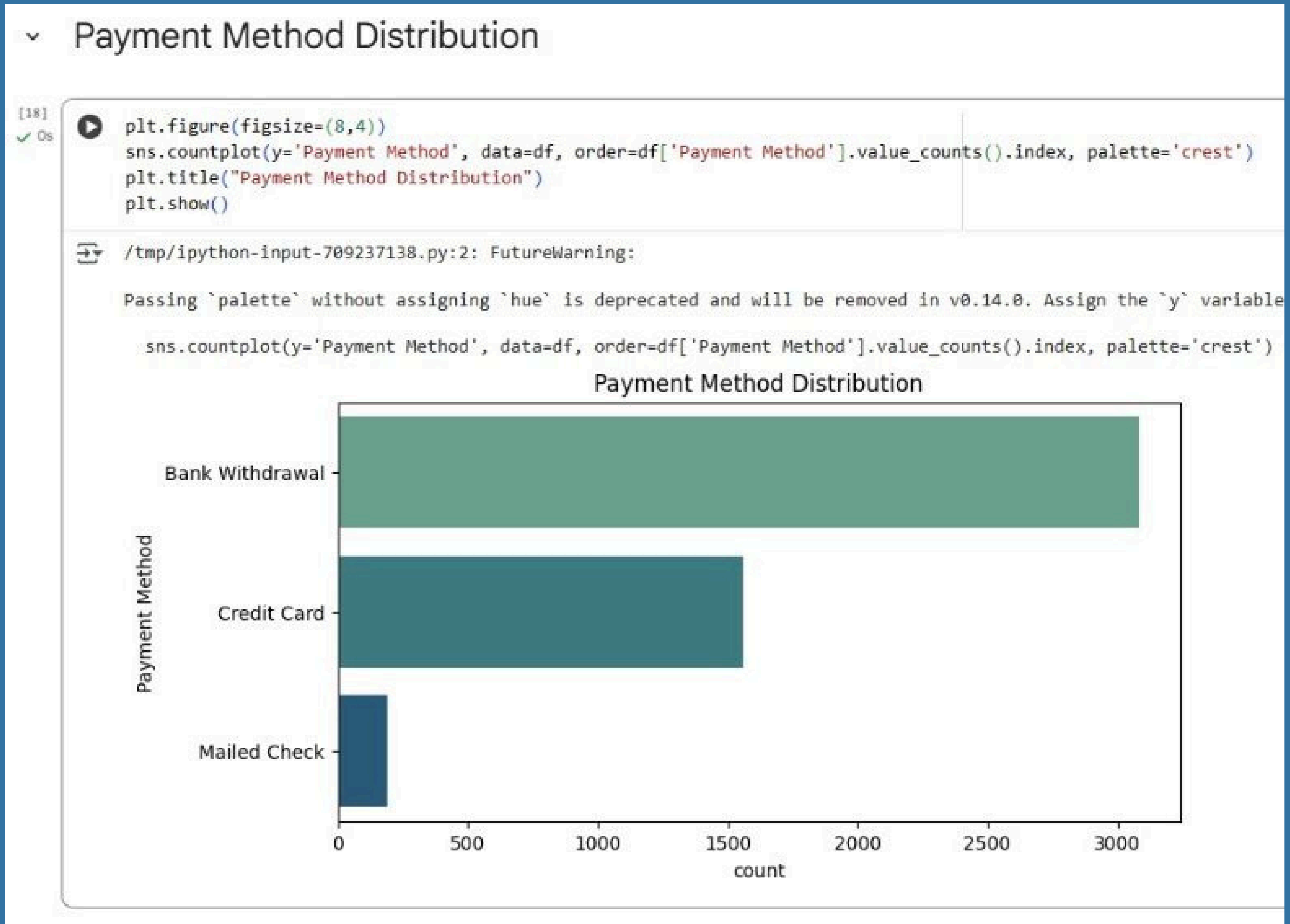


Insights: Month-to-month customers churn the most, while two-year contracts have the lowest churn, proving longer commitments improve loyalty.

6. Payment Method Distribution — (Bank Withdrawal)

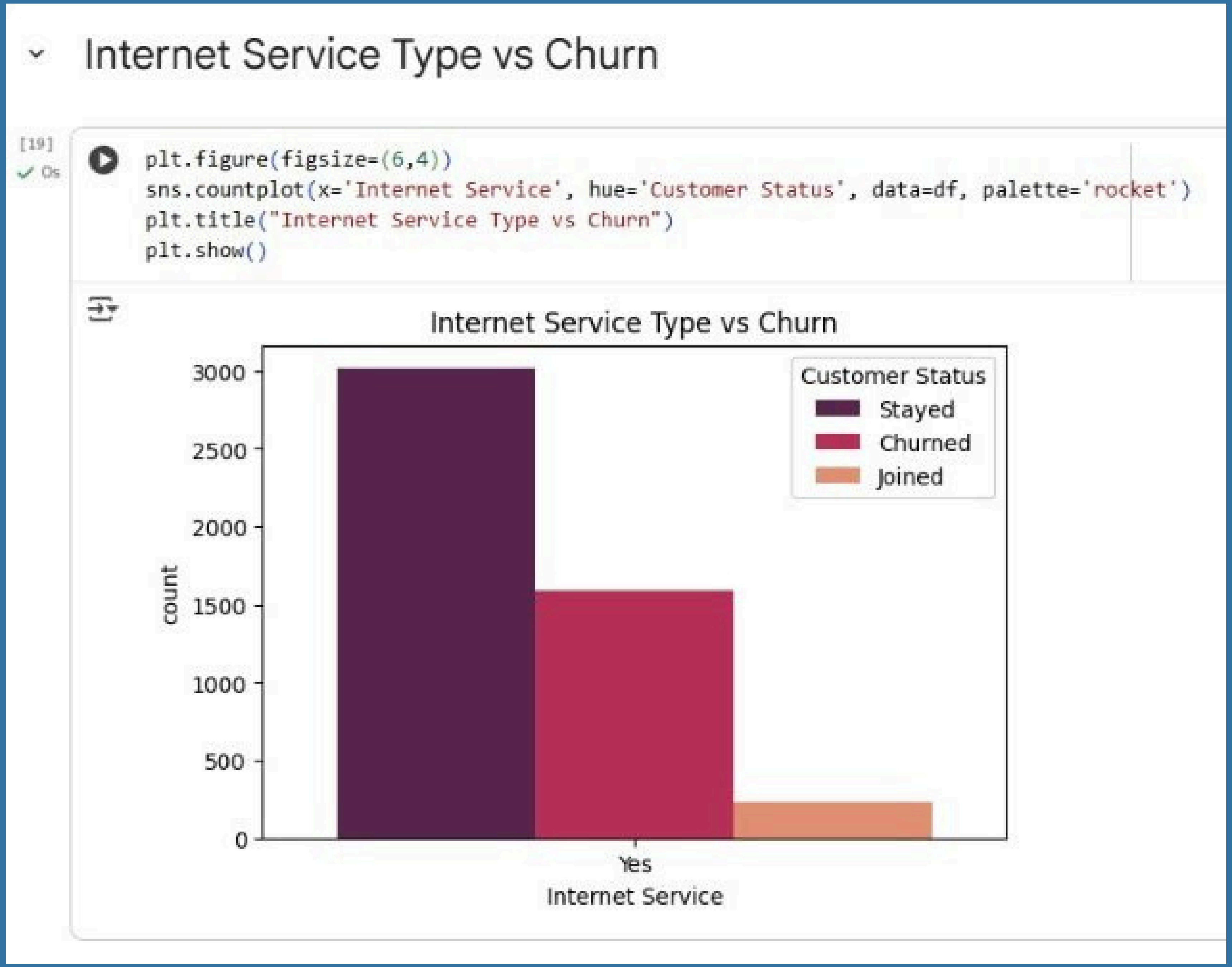
Purpose: To identify preferred payment methods and their churn patterns.

Insights: Bank withdrawal and credit card are most common; electronic check users show higher churn, suggesting trust or payment flexibility concerns.



7. Internet Service Type vs Churn — (Yes/No)

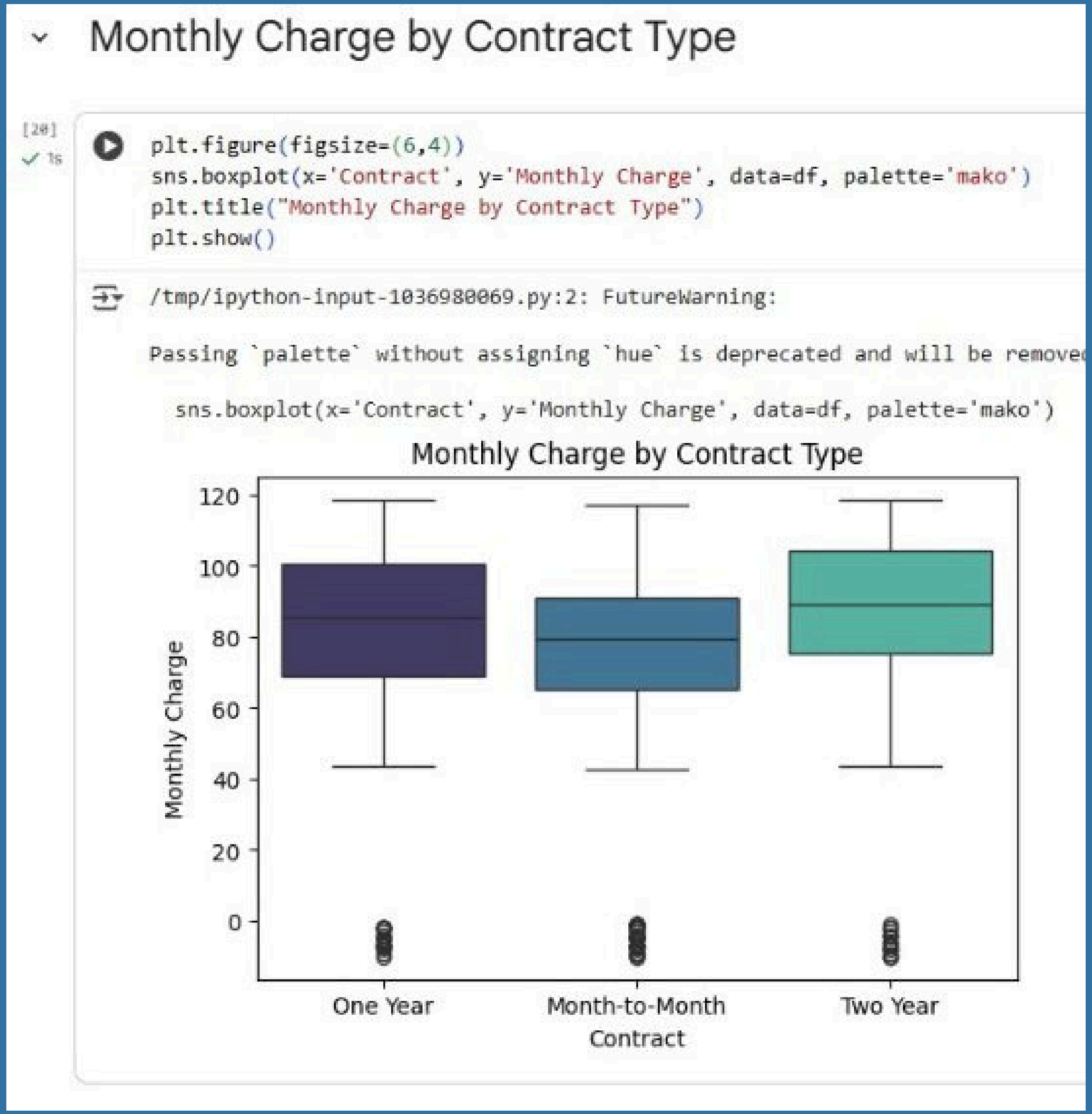
Purpose: To see whether internet service type affects churn.



Insights: Customers with fiber optic internet have the highest churn due to higher cost or service issues, while DSL users are more stable.

8. Monthly Charge by Contract Type — (Month-to-Month, Two Years)

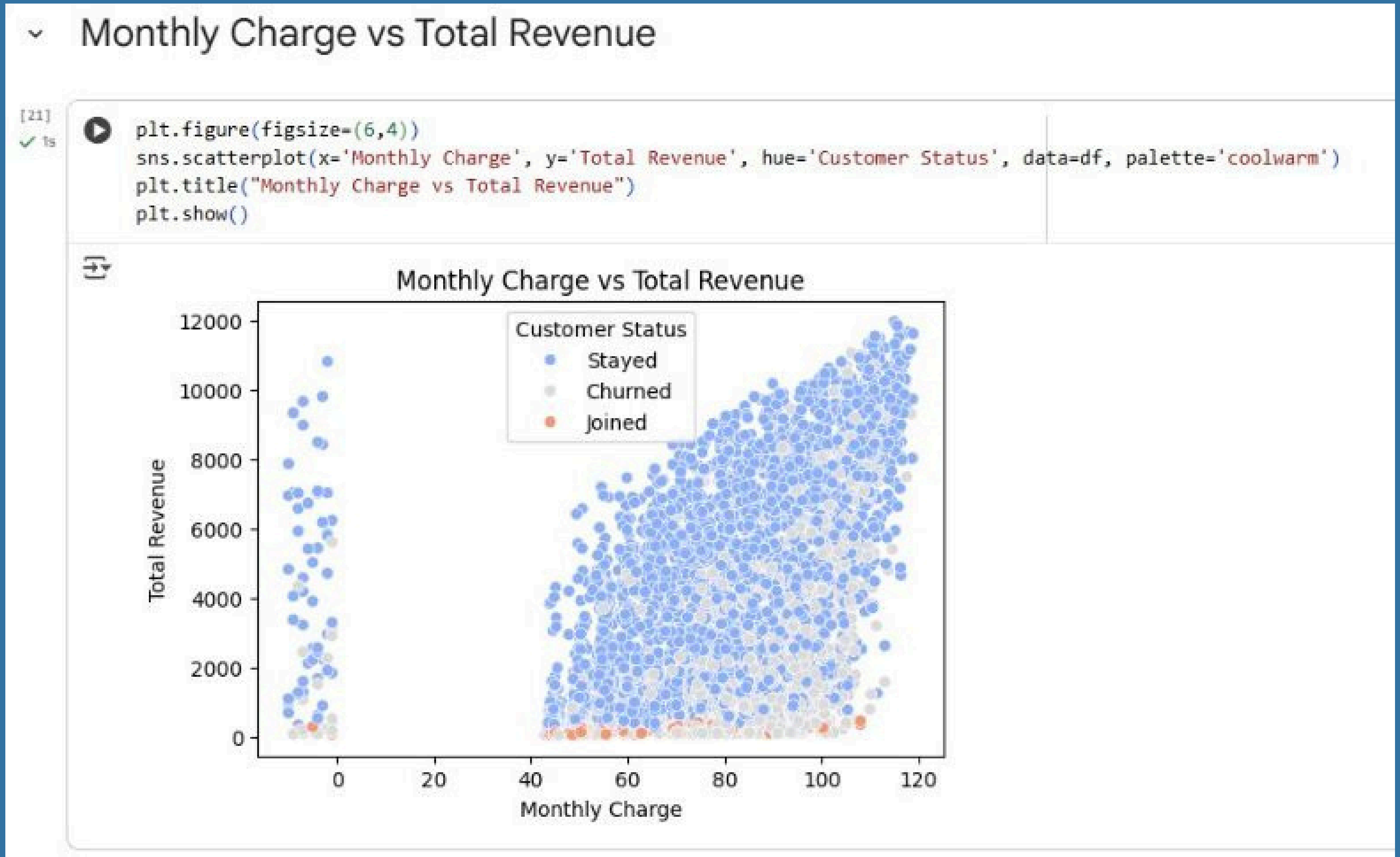
Purpose: To compare pricing impact across contracts.



Insights: Month-to-month users pay higher monthly charges, whereas two-year contracts offer discounts, making long-term contracts more appealing.

9. Monthly Charge vs Total Revenue — (₹80–₹120 / ₹10,000+)

Purpose: To find how monthly charges impact total revenue and churn.

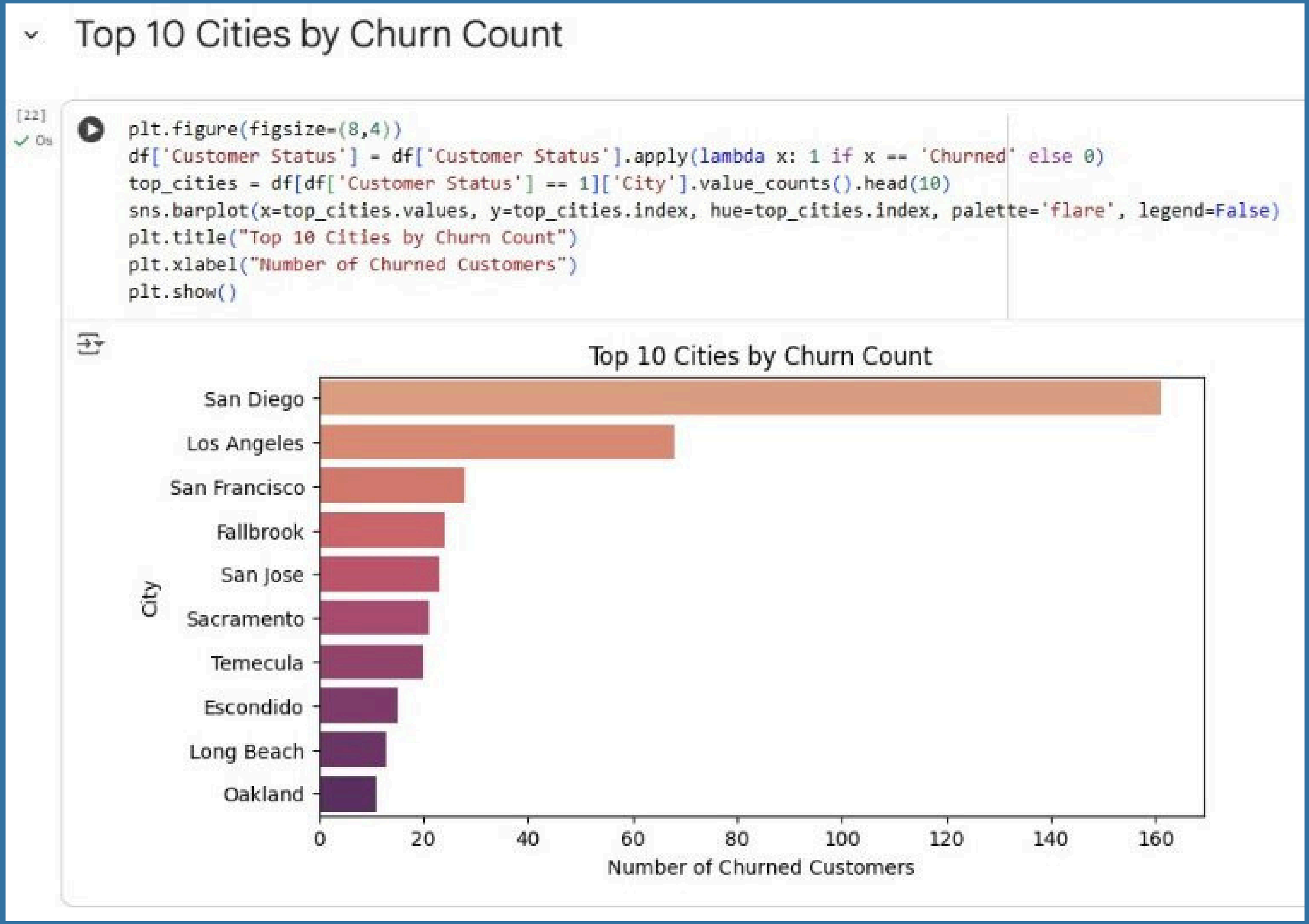


Insights: Customers with moderate monthly charges (₹80–₹120) generate the most stable revenue, while high-paying customers churn faster.

10. Top 10 Cities by Churn Count

Purpose: To identify geographic areas with high churn.

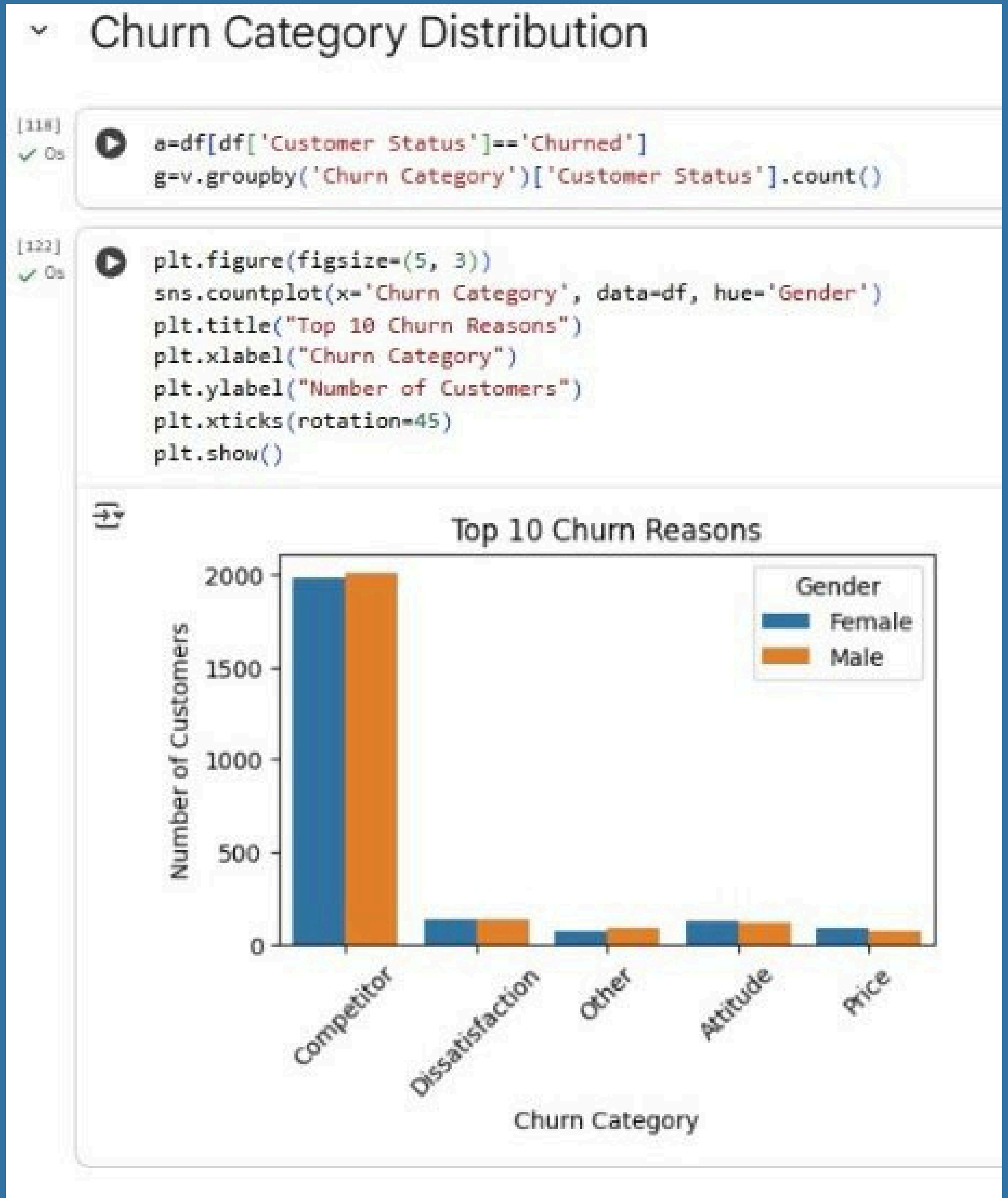
Insights: Top churn-prone cities show urban competition, indicating network quality and competitor offers play a major role in customer loss.



11. Churn Category Distribution

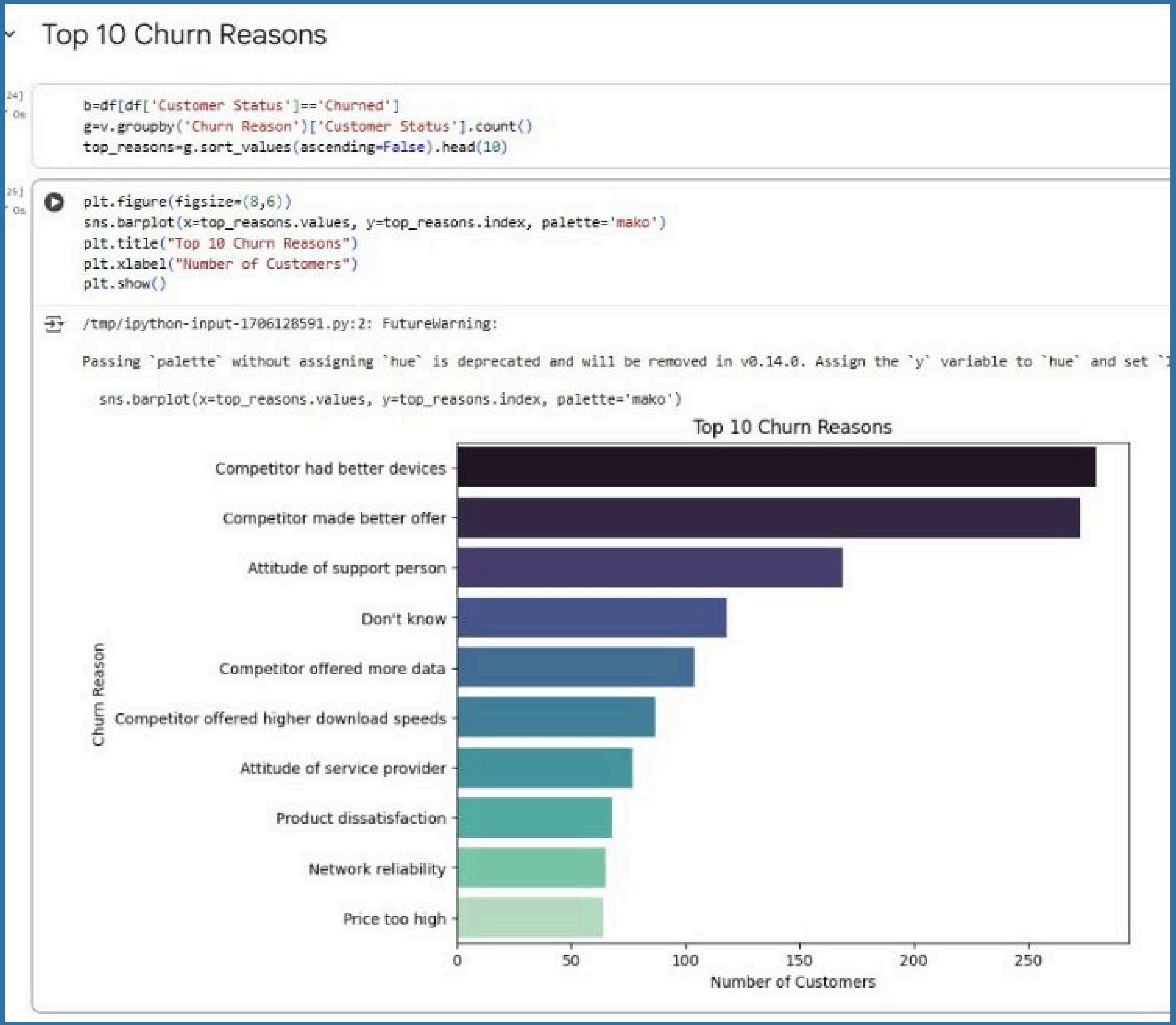
Purpose: To find key churn drivers.

Insights: Majority of churn is under the “Competitor” category, suggesting customers leave due to better pricing or offers elsewhere.



11. Top 10 Churn Reasons

Purpose: To understand the most common reasons for customer loss.



Insights:

Top churn reasons include:

1. Competitor made better offer
2. Dissatisfaction with network quality
3. High billing issues
4. Poor customer support
5. Moving to another provider area
6. Technical issues with internet
7. Limited offer value
8. Slow response time
9. Data speed problems
10. Unclear billing

Overall Insight:

Most churn is avoidable — focusing on pricing, service quality, and proactive retention can significantly reduce loss.

Telecom Customer Churn Report Summary :

1. Gender Distribution - Male-Female
2. Marital Status Distribution - No
3. Age Distribution of Customers - 20-70-80
4. Tenure Distribution by Customer Status - 0-10
5. Contract Type vs Churn - Month-to-Month
6. Payment Method Distribution - Bank Withdrawl
7. Internet Service Type vs Churn - Yes
8. Monthly Charge by Contract Type - M-to-M, Two Years
9. Monthly Charge vs Total Revenue - 80-120/10,000
10. Top 10 Cities by Churn Count Churn Category Distribution - Competitor

Business Impacts :

- mproved customer retention through early identification of high-risk churn segments.
- Enhanced revenue by reducing churn-related losses and improving lifetime value.
- Data-driven decisions help optimize pricing, contracts, and promotional offers.
- Improved customer satisfaction by focusing on service quality and support.
- Strengthened competitive advantage through predictive churn analytics.

THANK YOU