

▼ Project Name -EDA Telecom Churn Analysis Project

Project Type - EDA

Contribution - Team

Ananda M

Amit Negi

▼ Project Summary -

The project involved analyzing a telecom customer churn dataset to gain insights and provide recommendations for reducing churn. The first step was to familiarize ourselves with the data by importing necessary libraries, loading the dataset, and examining its structure and information. The dataset consisted of 23 columns, including variables such as state, account length, international plan, voice mail plan, call durations, charges, customer service calls, and churn status. After understanding the dataset, the next step was to explore and understand the variables. Descriptive statistics were analyzed to get a sense of the distribution and range of values for each variable. Additionally, unique values were examined to identify any anomalies or patterns within the data. Data wrangling was performed to manipulate the dataset for further analysis. Column renaming was done to improve clarity, and new columns were created, such as cost per minute, to better understand price distributions across different states. The analysis of the dataset revealed several interesting insights. The account length distribution indicated

length between 90 and 105 days, with a noticeable drop in retention after 105 days. Certain states, Saved successfully!  had higher account lengths, suggesting a correlation between state and customer loyalty. These states also had a larger customer base, indicating a strong customer presence. Furthermore, states with higher account lengths also had a higher number of customer service calls, implying that these states may be facing more issues requiring resolution. The relatively low churn rate of 16% indicated that the majority of customers had trust in the telecom company. The dataset also provided insights into the usage of international plans and voice mail plans. Around 10% of customers did not use the international plan, suggesting potential areas for improvement in international plan offerings. On the other hand, the data indicated that most customers made use of the voice mail plan, indicating its popularity among the user base. Correlation analysis revealed that the total day minutes, total day calls, and total day charges were highly correlated, indicating that customers tended to make more calls and have longer conversations during the day. Similar patterns were observed for evening and night durations and charges. Interestingly, the usage of international services did not exhibit a strong correlation with the presence of an international plan, suggesting that other factors might influence international usage. The distribution of customer service calls showed that the majority of customers made a low number of calls, but there were outliers who made significantly higher numbers of calls, possibly indicating more complex or unresolved issues. Overall, these insights provided valuable information for the telecom company to focus on improving international plans, addressing customer service concerns in specific states, and potentially enhancing the voice mail plan further. The findings also highlighted the importance of customer engagement during the day and the need to ensure excellent service and issue resolution. By utilizing these insights, the telecom company can develop strategies to reduce customer churn. This may include offering personalized discounts or incentives to at-risk customers, enhancing customer service quality and responsiveness, conducting regular customer surveys to gather feedback, implementing loyalty programs, providing additional value-added services, and adopting targeted marketing campaigns. Furthermore, data-driven approaches can be employed to identify early warning signs of potential churn and take proactive measures to retain customers. In conclusion, the project analysis of the telecom customer churn dataset provided valuable insights and recommendations for reducing churn. By understanding the customer behavior, preferences, and pain points, the telecom company can make informed decisions to enhance customer satisfaction, improve retention, and ultimately drive business growth.

▼ GitHub Link -

<https://github.com/Anand2868/EDA>

▼ Problem Statement

BUSINESS PROBLEM OVERVIEW

Customer churn prediction is extremely important for any business as it recognizes the clients who are likely to stop using their services.

In the telecom industry, customers are able to choose from multiple service providers and actively switch from one operator to another. In this highly competitive market, the telecommunications industry experiences an average of 15-25% annual churn rate. Given the fact that it costs 5-10 times more to acquire a new customer than to retain an existing one, customer retention has now become even more important than customer acquisition.

For many incumbent operators, retaining high profitable customers is the number one business goal. To reduce customer churn, telecom companies need to predict which customers are at high risk of churn. In this project, you will analyse customer-level data of a leading telecom firm, do exploratory data analysis to identify the main indicators why customers are leaving the company.

▼ Define Your Business Objective?

Reducing Customer Churn Rate

▼ General Guidelines :-

1. Well-structured, formatted, and commented code is required.
2. Exception Handling, Production Grade Code & Deployment Ready Code will be a plus. Those students will be awarded some additional credits.

The additional credits will have advantages over other students during Star Student selection.

[Note: - Deployment Ready Code is defined as, the whole .ipynb notebook should be executable in one go without a single error logged.]

3. Each and every logic should have proper comments.

Saved successfully!

charts you want. Make Sure for each and every chart the following format should be answered.

Chart visualization code

- Why did you pick the specific chart?
- What is/are the insight(s) found from the chart?
- Will the gained insights help creating a positive business impact? Are there any insights that lead to negative growth? Justify with specific reason.

5. You have to create at least 20 logical & meaningful charts having important insights.

[Hints : - Do the Visualization in a structured way while following "UBM" Rule.

U - Univariate Analysis,

B - Bivariate Analysis (Numerical - Categorical, Numerical - Numerical, Categorical - Categorical)

M - Multivariate Analysis]

▼ Let's Begin !

▼ 1. Know Your Data

▼ Import Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

▼ Dataset Loading

```
from google.colab import drive #import dataset from google drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
telecom_df=pd.read_csv('/content/drive/MyDrive/EDA Project/Telecom Churn.csv') #load the dataset
```

▼ Dataset First View

<https://colab.research.google.com/drive/1hGihheGLgCUI9CusokxtWSTz0RctzKd5#scrollTo=vncDsAP0Gaoa&printMode=true>

```
telecom_df # Dataset First Look
```

	State	Account length	Area code	International plan	Voice mail plan	Number vmail messages	Total day minutes	Total day calls	Total day charge
0	KS	128	415	No	Yes	25	265.1	110	45.07
1	OH	107	415	No	Yes	26	161.6	123	27.47
2	NJ	137	415	No	No	0	243.4	114	41.38
3	OH	84	408	Yes	No	0	299.4	71	50.90
4	OK	75	415	Yes	No	0	166.7	113	28.34
...
3328	AZ	192	415	No	Yes	36	156.2	77	26.55
3329	WV	68	415	No	No	0	231.1	57	39.29
3330	RI	28	510	No	No	0	180.8	109	30.74
3331	CT	184	510	Yes	No	0	213.8	105	36.35
3332	TN	74	415	No	Yes	25	234.4	113	39.85

3333 rows × 20 columns



Saved successfully!

```
telecom_df.head()
```

	State	Account length	Area code	International plan	Voice mail plan	Number vmail messages	Total day minutes	Total day calls	Total day charge
0	KS	128	415	No	Yes	25	265.1	110	45.07
1	OH	107	415	No	Yes	26	161.6	123	27.47
2	NJ	137	415	No	No	0	243.4	114	41.38
3	OH	84	408	Yes	No	0	299.4	71	50.90
4	OK	75	415	Yes	No	0	166.7	113	28.34

```
telecom_df.tail()
```

	State	Account length	Area code	International plan	Voice mail plan	Number vmail messages	Total day minutes	Total day calls	Total day charge
3328	AZ	192	415	No	Yes	36	156.2	77	26.55
3329	WV	68	415	No	No	0	231.1	57	39.29
3330	RI	28	510	No	No	0	180.8	109	30.74
3331	CT	184	510	Yes	No	0	213.8	105	36.35
3332	TN	74	415	No	Yes	25	234.4	113	39.85

▼ Dataset Rows & Columns count

```
telecom_df.shape # Dataset Rows & Columns count
(3333, 20)
```

▼ Dataset Information

```
telecom_df.info() # Dataset Info
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
```

```
Data columns (total 20 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   State            3333 non-null    object  
 1   Account length   3333 non-null    int64  
 2   Area code         3333 non-null    int64  
 3   International plan 3333 non-null    object  
 4   Voice mail plan  3333 non-null    object  
 5   Number vmail messages 3333 non-null    int64  
 6   Total day minutes 3333 non-null    float64 
 7   Total day calls   3333 non-null    int64  
 8   Total day charge  3333 non-null    float64 
 9   Total eve minutes 3333 non-null    float64 
 10  Total eve calls   3333 non-null    int64  
 11  Total eve charge  3333 non-null    float64 
 12  Total night minutes 3333 non-null    float64 
 13  Total night calls 3333 non-null    int64  
 14  Total night charge 3333 non-null    float64 
 15  Total intl minutes 3333 non-null    float64 
 16  Total intl calls   3333 non-null    int64  
 17  Total intl charge  3333 non-null    float64 
 18  Customer service calls 3333 non-null    int64  
 19  Churn             3333 non-null    bool    
dtypes: bool(1), float64(8), int64(8), object(3)
memory usage: 498.1+ KB
```

▼ Duplicate Values

```
telecom_df[telecom_df.duplicated()].sum()      # Dataset Duplicate Value Count
```

Saved successfully! ×

International plan	0.0
Voice mail plan	0.0
Number vmail messages	0.0
Total day minutes	0.0
Total day calls	0.0
Total day charge	0.0
Total eve minutes	0.0
Total eve calls	0.0
Total eve charge	0.0
Total night minutes	0.0
Total night calls	0.0
Total night charge	0.0
Total intl minutes	0.0
Total intl calls	0.0
Total intl charge	0.0
Customer service calls	0.0
Churn	0.0

dtype: float64

▼ Missing Values/Null Values

```
null_telecom_df=telecom_df.isnull()
```

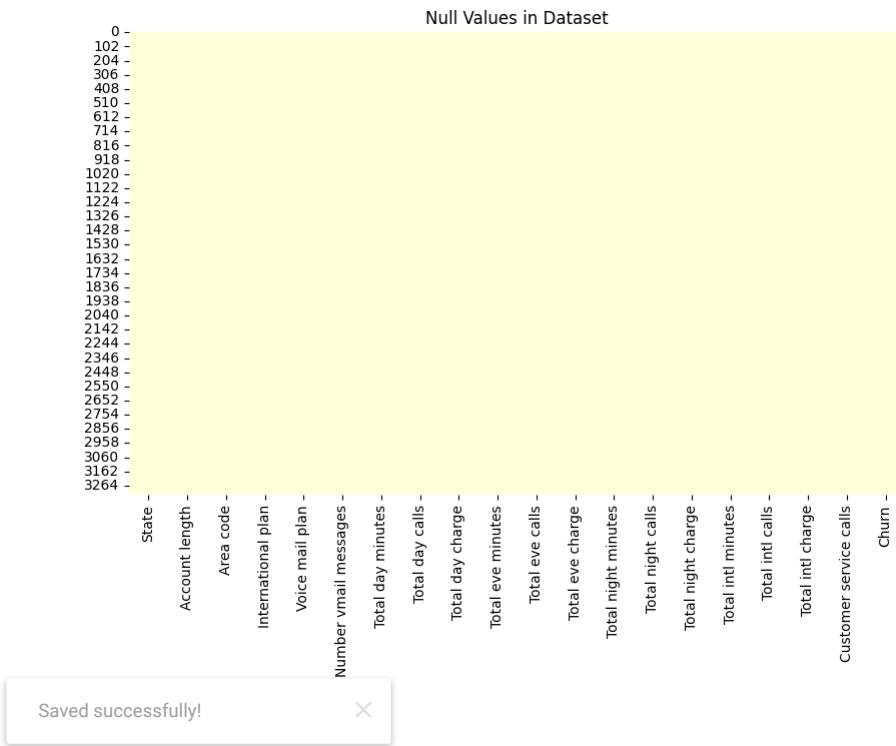
```
null_sum_telecom_df=telecom_df[telecom_df.isnull()].sum()      # Missing Values/Null Values Count
null_sum_telecom_df
```

State	0
Account length	0.0
Area code	0.0
International plan	0
Voice mail plan	0
Number vmail messages	0.0
Total day minutes	0.0
Total day calls	0.0
Total day charge	0.0
Total eve minutes	0.0
Total eve calls	0.0
Total eve charge	0.0
Total night minutes	0.0
Total night calls	0.0
Total night charge	0.0
Total intl minutes	0.0
Total intl calls	0.0
Total intl charge	0.0
Customer service calls	0.0
Churn	0

dtype: object

```
plt.figure(figsize=(10, 6))
sns.heatmap(null_telecom_df, cmap='YlGnBu', cbar=False)
```

```
plt.title('Null Values in Dataset')
plt.show() #Visualizing the missing values
```



Saved successfully! ×

▼ What did you know about your dataset?

The dataset given is a dataset from Telecommunication industry, and we have to analysis the churn of customers and the insights behind it. Churn prediction is analytical studies on the possibility of a customer abandoning a product or service. The goal is to understand and take steps to change it before the costumer gives up the product or service.

Info- As per till now we can see there are 3333 entries in the Telecom Dataset and there are 20 columns. There is 1 Boolean Datatype, 8 Float datatypes, 8 interger datatypes, and 3 text/object datatypes.

Duplicates- There is no duplicate values in this Dataset.

Missing/Null Values- There is no null values in the Dataset which help us to get the insights in a good manner

▼ 2. Understanding Your Variables

```
telecom_df.columns # Dataset Columns

Index(['State', 'Account length', 'Area code', 'International plan',
       'Voice mail plan', 'Number vmail messages', 'Total day minutes',
       'Total day calls', 'Total day charge', 'Total eve minutes',
       'Total eve calls', 'Total eve charge', 'Total night minutes',
       'Total night calls', 'Total night charge', 'Total intl minutes',
       'Total intl calls', 'Total intl charge', 'Customer service calls',
       'Churn'],
      dtype='object')
```

```
telecom_df.describe(include='all') # Dataset Describe
```

	State	Account length	Area code	International plan	Voice mail plan	Number vmail messages	Total minu
count	3333	3333.000000	3333.000000	3333	3333	3333.000000	3333.000
unique	51	Nan	Nan	2	2	Nan	1
top	WV	Nan	Nan	No	No	Nan	1
freq	106	Nan	Nan	3010	2411	Nan	1
mean	Nan	101.064806	437.182418	Nan	Nan	8.099010	179.775
std	Nan	39.822106	42.371290	Nan	Nan	13.688365	54.467
min	Nan	1.000000	408.000000	Nan	Nan	0.000000	0.000

Variables Description

The variable descriptions for the columns in the `telecom_df` DataFrame:

1. State: The state in the United States where the customer resides.
 2. Account length: The number of days the customer has been an account holder.
 3. Area code: The three-digit area code of the customer's phone number.
 4. International plan: Whether the customer has an international calling plan (yes/no).
 5. Voice mail plan: Whether the customer has a voicemail plan (yes/no).
 6. Number vmail messages: The number of voicemail messages the customer has.
7. Total day minutes: The total number of minutes the customer used during the day.
8. Total day calls: The total number of calls the customer made during the day.
9. Total day charge: The total charge in dollars for the customer's day usage.
 10. Total eve minutes: The total number of minutes the customer used during the evening.
 11. Total eve calls: The total number of calls the customer made during the evening.
 12. Total eve charge: The total charge in dollars for the customer's evening usage.
 13. Total night minutes: The total number of minutes the customer used during the night.
 14. Total night calls: The total number of calls the customer made during the night.
 15. Total night charge: The total charge in dollars for the customer's night usage.
 16. Total intl minutes: The total number of international minutes the customer used.
 17. Total intl calls: The total number of international calls the customer made.
 18. Total intl charge: The total charge in dollars for the customer's international usage.
 19. Customer service calls: The number of customer service calls made by the customer.
 20. Churn: Whether the customer churned (cancelled the service) (True/False).

Check Unique Values for each variable.

```
telecom_df.nunique() #Check Unique Values for each variable.
```

State	51
Account length	212
Area code	3
International plan	2
Voice mail plan	2
Number vmail messages	46
Total day minutes	1667
Total day calls	119
Total day charge	1667
Total eve minutes	1611
Total eve calls	123
Total eve charge	1440
Total night minutes	1591
Total night calls	120
Total night charge	933
Total intl minutes	162
Total intl calls	21
Total intl charge	162
Customer service calls	10
Churn	2

dtype: int64

```
telecom_df['State'].unique()
```

```
array(['KS', 'OH', 'NJ', 'OK', 'AL', 'MA', 'MO', 'LA', 'WV', 'IN', 'RI',
       'IA', 'MT', 'NY', 'ID', 'VT', 'VA', 'TX', 'FL', 'CO', 'AZ', 'SC',
       'NE', 'WY', 'HI', 'IL', 'NH', 'GA', 'AK', 'MD', 'AR', 'WI', 'OR',
```

```
'MI', 'DE', 'UT', 'CA', 'MN', 'SD', 'NC', 'WA', 'NM', 'NV', 'DC',
'KY', 'ME', 'MS', 'TN', 'PA', 'CT', 'ND'], dtype=object)
```

```
telecom_df['Account length'].unique()
```

```
array([128, 107, 137, 84, 75, 118, 121, 147, 117, 141, 65, 74, 168,
95, 62, 161, 85, 93, 76, 73, 77, 130, 111, 132, 174, 57,
54, 20, 49, 142, 172, 12, 72, 36, 78, 136, 149, 98, 135,
34, 160, 64, 59, 119, 97, 52, 60, 10, 96, 87, 81, 68,
125, 116, 38, 40, 43, 113, 126, 150, 138, 162, 90, 50, 82,
144, 46, 70, 55, 106, 94, 155, 80, 104, 99, 120, 108, 122,
157, 103, 63, 112, 41, 193, 61, 92, 131, 163, 91, 127, 110,
140, 83, 145, 56, 151, 139, 6, 115, 146, 185, 148, 32, 25,
179, 67, 19, 170, 164, 51, 208, 53, 105, 66, 86, 35, 88,
123, 45, 100, 215, 22, 33, 114, 24, 101, 143, 48, 71, 167,
89, 199, 166, 158, 196, 209, 16, 39, 173, 129, 44, 79, 31,
124, 37, 159, 194, 154, 21, 133, 224, 58, 11, 109, 102, 165,
18, 30, 176, 47, 190, 152, 26, 69, 186, 171, 28, 153, 169,
13, 27, 3, 42, 189, 156, 134, 243, 23, 1, 205, 200, 5,
9, 178, 181, 182, 217, 177, 210, 29, 180, 2, 17, 7, 212,
232, 192, 195, 197, 225, 184, 191, 201, 15, 183, 202, 8, 175,
4, 188, 204, 221])
```

```
telecom_df['Area code'].unique()
```

```
array([415, 408, 510])
```

```
telecom_df['International plan'].unique()
```

```
array(['No', 'Yes'], dtype=object)
```

Saved successfully!

```
telecom_df['Voice mail plan'].unique()
```

```
array(['Yes', 'No'], dtype=object)
```

```
telecom_df['Total day minutes'].unique()
```

```
array([265.1, 161.6, 243.4, ..., 321.1, 231.1, 180.8])
```

```
telecom_df['Total day calls'].unique()
```

```
array([110, 123, 114, 71, 113, 98, 88, 79, 97, 84, 137, 127, 96,
70, 67, 139, 66, 90, 117, 89, 112, 103, 86, 76, 115, 73,
109, 95, 105, 121, 118, 94, 80, 128, 64, 106, 102, 85, 82,
77, 120, 133, 135, 108, 57, 83, 129, 91, 92, 74, 93, 101,
146, 72, 99, 104, 125, 61, 100, 87, 131, 65, 124, 119, 52,
68, 107, 47, 116, 151, 126, 122, 111, 145, 78, 136, 140, 148,
81, 55, 69, 158, 134, 130, 63, 53, 75, 141, 163, 59, 132,
138, 54, 58, 62, 144, 143, 147, 36, 40, 150, 56, 51, 165,
30, 48, 60, 42, 0, 45, 160, 149, 152, 142, 156, 35, 49,
157, 44])
```

```
telecom_df['Total day charge'].unique()
```

```
array([45.07, 27.47, 41.38, ..., 54.59, 39.29, 30.74])
```

```
telecom_df['Total eve minutes'].unique()
```

```
array([197.4, 195.5, 121.2, ..., 153.4, 288.8, 265.9])
```

```
telecom_df['Total eve calls'].unique()
```

```
array([ 99, 103, 110, 88, 122, 101, 108, 94, 80, 111, 83, 148, 71,
75, 76, 97, 90, 65, 93, 121, 102, 72, 112, 100, 84, 109,
63, 107, 115, 119, 116, 92, 85, 98, 118, 74, 117, 58, 96,
66, 67, 62, 77, 164, 126, 142, 64, 104, 79, 95, 86, 105,
81, 113, 106, 59, 48, 82, 87, 123, 114, 140, 128, 60, 78,
125, 91, 46, 138, 129, 89, 133, 136, 57, 135, 139, 51, 70,
151, 137, 134, 73, 152, 168, 68, 120, 69, 127, 132, 143, 61,
124, 42, 54, 131, 52, 149, 56, 37, 130, 49, 146, 147, 55,
12, 50, 157, 155, 45, 144, 36, 156, 53, 141, 44, 153, 154,
150, 43, 0, 145, 159, 170])
```

```
telecom_df['Total eve charge'].unique()
```

```
array([16.78, 16.62, 10.3, ..., 13.04, 24.55, 22.6])
```

```
telecom_df['Total night minutes'].unique()
```

```
array([244.7, 254.4, 162.6, ..., 280.9, 120.1, 279.1])

telecom_df['Total intl minutes'].unique()

array([10. , 13.7, 12.2, 6.6, 10.1, 6.3, 7.5, 7.1, 8.7, 11.2, 12.7,
       9.1, 12.3, 13.1, 5.4, 13.8, 8.1, 13. , 10.6, 5.7, 9.5, 7.7,
      10.3, 15.5, 14.7, 11.1, 14.2, 12.6, 11.8, 8.3, 14.5, 10.5, 9.4,
     14.6, 9.2, 3.5, 8.5, 13.2, 7.4, 8.8, 11. , 7.8, 6.8, 11.4,
    9.3, 9.7, 10.2, 8. , 5.8, 12.1, 12. , 11.6, 8.2, 6.2, 7.3,
   6.1, 11.7, 15. , 9.8, 12.4, 8.6, 10.9, 13.9, 8.9, 7.9, 5.3,
  4.4, 12.5, 11.3, 9. , 9.6, 13.3, 20. , 7.2, 6.4, 14.1, 14.3,
  6.9, 11.5, 15.8, 12.8, 16.2, 0. , 11.9, 9.9, 8.4, 10.8, 13.4,
 10.7, 17.6, 4.7, 2.7, 13.5, 12.9, 14.4, 10.4, 6.7, 15.4, 4.5,
 6.5, 15.6, 5.9, 18.9, 7.6, 5. , 7. , 14. , 18. , 16. , 14.8,
 3.7, 2. , 4.8, 15.3, 6. , 13.6, 17.2, 17.5, 5.6, 18.2, 3.6,
16.5, 4.6, 5.1, 4.1, 16.3, 14.9, 16.4, 16.7, 1.3, 15.2, 15.1,
15.9, 5.5, 16.1, 4. , 16.9, 5.2, 4.2, 15.7, 17. , 3.9, 3.8,
 2.2, 17.1, 4.9, 17.9, 17.3, 18.4, 17.8, 4.3, 2.9, 3.1, 3.3,
 2.6, 3.4, 1.1, 18.3, 16.6, 2.1, 2.4, 2.5])
```

```
telecom_df['Total intl calls'].unique()

array([ 3,  5,  7,  6,  4,  2,  9, 19,  1, 10, 15,  8, 11,  0, 12, 13, 18,
       14, 16, 20, 17])
```

```
telecom_df['Total intl charge'].unique()

array([2.7 , 3.7 , 3.29, 1.78, 2.73, 1.7 , 2.03, 1.92, 2.35, 3.02, 3.43,
       2.46, 3.32, 3.54, 1.46, 3.73, 2.19, 3.51, 2.86, 1.54, 2.57, 2.08,
      Saved successfully! × 3.83, 3.4 , 3.19, 2.24, 3.92, 2.84, 2.54,
  1.57, 3.27, 3.24, 3.13, 2.21, 1.67, 1.97,
  1.65, 3.16, 4.05, 2.65, 3.35, 2.32, 2.94, 3.75, 2.4 , 2.13, 1.43,
  1.19, 3.38, 3.05, 2.43, 2.59, 3.59, 5.4 , 1.94, 1.73, 3.81, 3.86,
  1.86, 3.11, 4.27, 3.46, 4.37, 0. , 3.21, 2.67, 2.27, 2.92, 3.62,
  2.89, 4.75, 1.27, 0.73, 3.65, 3.48, 3.89, 2.81, 1.81, 4.16, 1.22,
  1.76, 4.21, 1.59, 5.1 , 2.05, 1.35, 1.89, 3.78, 4.86, 4.32, 4. ,
  1. , 0.54, 1.3 , 4.13, 1.62, 3.67, 4.64, 4.73, 1.51, 4.91, 0.97,
  4.46, 1.24, 1.38, 1.11, 4.4 , 4.02, 4.43, 4.51, 0.35, 4.1 , 4.08,
  4.29, 1.49, 4.35, 1.08, 4.56, 1.4 , 1.13, 4.24, 4.59, 1.05, 1.03,
  0.59, 4.62, 1.32, 4.83, 4.67, 4.97, 4.81, 1.16, 0.78, 0.84, 0.89,
  0.7 , 0.92, 0.3 , 4.94, 4.48, 0.57, 0.65, 0.68])
```

```
telecom_df['Customer service calls'].unique()
```

```
array([1, 0, 2, 3, 4, 5, 7, 9, 6, 8])
```

```
telecom_df['Churn'].unique()
```

```
array([False, True])
```

▼ 3. Data Wrangling

Data Wrangling is the process of gathering, collecting, and transforming Raw data into another format for better understanding, decision-making, accessing, and analysis in less time.

▼ Data Wrangling Code

```
telecom_df['Account length'].value_counts().head() #total count account length

105    43
87     42
101    40
93     40
90     39
Name: Account length, dtype: int64
```

```
telecom_df['Account length'].value_counts().tail() #total count account length for lowest 5 states

243    1
200    1
232    1
5      1
221    1
Name: Account length, dtype: int64
```

```
telecom_df.groupby(['State']).agg({'Account length':'sum'}).sort_values(by='Account length', ascending=False).head(10) #total account length
```

Account length

State	Account length
WV	10116
MN	8532
NY	8358
VA	8157
WY	8142
OH	7968
OR	7891
AL	7842
WI	7713
ID	7628

```
x=telecom_df.groupby(['State']).agg({'State':'count'}) #most top 10 states where orange telecomm's is used
x['State'].sort_values(ascending=False).head(10)
```

State

View 100

Saved successfully! X

AL	80
OR	78
WI	78
OH	78
VA	77
WY	77
CT	74

Name: State, dtype: int64

```
telecom_df.groupby(['State']).agg({'Customer service calls':'count'}).sort_values(by='Customer service calls', ascending=False).head(10)
```

Customer service calls

State	Customer service calls
WV	106
MN	84
NY	83
AL	80
OR	78
WI	78
OH	78
VA	77
WY	77
CT	74

```
telecom_df['Churn'].value_counts() #total no. of churn
```

False	2850
True	483

Name: Churn, dtype: int64

```
telecom_df['International plan'].value_counts() #counting the international plan used by the customers
```

No	3010
Yes	323

Name: International plan, dtype: int64

```
telecom_df['Voice mail plan'].value_counts() #counting the voice mail plan used by the customers
```

No	2411
Yes	922

Name: Voice mail plan, dtype: int64

```
st_ch_intpln=telecom_df.groupby(['State','Churn','International plan']).agg({'Churn': 'count'}) #to the count of churn group wise using

st_ch_intpln['Churn'].sort_values(ascending=False) #sorting of variable descending wise

State  Churn  International plan
WV    False   No                91
AL    False   No                67
MN    False   No                66
VA    False   No                66
WI    False   No                66
..    
WA    False   Yes               1
GA    False   Yes               1
LA    False   Yes               1
OR    True    Yes               1
MT    False   Yes               1
Name: Churn, Length: 195, dtype: int64
```

```
telecom_df.groupby(['State']).agg({'Total day charge': 'mean','Total eve charge':'mean','Total night charge':'mean','Total intl charge':'
```

Saved successfully! ×

State	Total day charge	Total eve charge	Total night charge	Total intl charge
AK	30.325769	15.664231	8.654808	2.705385
AL	31.621500	16.614500	8.427875	2.779000
AR	29.940000	17.089636	9.245818	2.800727
AZ	29.173125	15.958906	8.730625	2.870625
CA	31.206176	16.913529	8.932941	2.665588
CO	30.381818	17.584848	8.546061	2.665758
CT	29.774324	17.325946	9.270405	2.713378
DC	29.135370	16.684074	9.285741	2.864630
DE	29.680164	17.701311	9.175574	2.778361
FL	30.521429	17.873651	8.826508	2.713175
GA	31.455185	17.352222	8.718519	2.775185
HI	29.913019	16.263774	9.166792	2.715849
IA	30.195455	17.544318	8.617955	2.612500
ID	30.365753	16.542055	9.116986	2.835616

```
telecom_df.corr() #to find the correlation between the columns
```

Saved successfully! ×

```
<ipython-input-57-beff8cdca78c>:1: FutureWarning: The default value of numeric_c
telecom df.corr() #to find the correlation between the columns
```

Renaming the Columns

```
length      code  messages  minutes_  calls_  charge_  minutes_
telecom_df.rename(columns={'Number vmail messages':'Number Of Mail Messages'},inplace=True) #renaming the number vmail messages
length      1.000000 -0.012403 -0.004020  0.000210  0.000470  0.000214 -0.000751
telecom_df.columns #to check the columns name

Index(['State', 'Account length', 'Area code', 'International plan',
       'Voice mail plan', 'Number Of Mail Messages', 'Total day minutes',
       'Total day calls', 'Total day charge', 'Total eve minutes',
       'Total eve calls', 'Total eve charge', 'Total night minutes',
       'Total night calls', 'Total night charge', 'Total intl minutes',
       'Total intl calls', 'Total intl charge', 'Customer service calls',
       'Churn'],
      dtype='object')
```

Feature Engineering

```
telecom_df.head()
```

	State	Account length	Area code	International plan	Voice mail plan	Number Of Mail Messages	Total day minutes	Total day calls	Total day charge
0	KS	128	415	No	Yes	25	265.1	110	45.07
1	CA	107	415	No	Yes	26	161.6	123	27.47
Saved successfully!									
3	OH	84	408	Yes	No	0	299.4	71	50.90
4	OK	75	415	Yes	No	0	166.7	113	28.34

```
telecom_df['Day cost per minute']=round(telecom_df['Total day minutes']/telecom_df['Total day charge'],2) #created a column day cost per
minutes      0.000017 -0.012403  0.000200 -0.004020  0.000100 -0.000470  0.000214 -0.000751
telecom_df['Day cost per minute'].value_counts() #to check the column value counts
```

```
5.88    3319
5.89     7
5.87     2
5.90     1
5.91     1
5.86     1
Name: Day cost per minute, dtype: int64
```

```
telecom_df['Eve cost per minute']=round(telecom_df['Total eve minutes']/telecom_df['Total eve charge'],2) #created a column eve cost per
minutes      0.016511  0.006171  0.000700  0.005151  0.010150  0.005151  0.002706
telecom_df['Eve cost per minute'].value_counts() #to check the value counts
```

```
11.76    1858
11.77    1471
11.75     3
Name: Eve cost per minute, dtype: int64
```

```
telecom_df['Night cost per minute']=round(telecom_df['Total night minutes']/telecom_df['Total night charge'],2) #created a column night c
```

```
telecom_df['Night cost per minute'].value_counts() #to check the value counts of the column
```

```
22.22    1357
22.23    1148
22.21     653
22.24     123
22.20      37
22.25      8
22.19      2
22.27      2
22.17      1
22.18      1
22.31      1
Name: Night cost per minute, dtype: int64
```

```
telecom_df[telecom_df['Day cost per minute'].isna()] #to check the null values in the new column
```

	State	Account length	Area code	International plan	Voice mail plan	Number Of Mail Messages	Total day minutes	Total day calls	Total day charge
1345	SD	98	415	No	No	0	0.0	0	0.0
1397	VT	101	510	No	No	0	0.0	0	0.0

2 rows × 23 columns

```
x=0 #after analysing the column filled the null values with 0
telecom_df['Day cost per minute'].fillna(x,inplace=True)
```

```
telecom_df.info() #to check the null values
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
Data columns (total 23 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   State            3333 non-null    object  
 1   Account length   3333 non-null    int64  
 2   Area code         3333 non-null    int64  
 3   International plan 3333 non-null    object  
 4   Voice mail plan  3333 non-null    object  
 5   Number Of Mail Messages 3333 non-null    int64  
 6   Total day minutes 3333 non-null    float64 
 7   Total day calls   3333 non-null    int64  
 8   Total day charge  3333 non-null    float64 
 9   Total night minutes 3333 non-null    float64 
 10  Total night calls  3333 non-null    int64  
 11  Total night charge 3333 non-null    float64 
 12  Total intl minutes 3333 non-null    float64 
 13  Total intl calls   3333 non-null    int64  
 14  Total intl charge  3333 non-null    float64 
 15  Customer service calls 3333 non-null    int64  
 16  Churn             3333 non-null    bool    
 17  Day cost per minute 3333 non-null    float64 
 18  Eve cost per minute 3332 non-null    float64 
 19  Night cost per minute 3333 non-null    float64 
dtypes: bool(1), float64(11), int64(8), object(3)
memory usage: 576.2+ KB
```

```
telecom_df[telecom_df['Eve cost per minute'].isna()] #to check the null values in the new column
```

	State	Account length	Area code	International plan	Voice mail plan	Number Of Mail Messages	Total day minutes	Total day calls	Total day charge
2932	UT	97	415	No	No	0	209.2	134	35.56

1 rows × 23 columns

```
x=0 #after analysing the column filled the null values with 0
telecom_df['Eve cost per minute'].fillna(x,inplace=True)
```

```
telecom_df.info() #to check the null values
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
Data columns (total 23 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   State            3333 non-null    object  
 1   Account length   3333 non-null    int64  
 2   Area code         3333 non-null    int64  
 3   International plan 3333 non-null    object  
 4   Voice mail plan  3333 non-null    object  
 5   Number Of Mail Messages 3333 non-null    int64  
 6   Total day minutes 3333 non-null    float64 
 7   Total day calls   3333 non-null    int64  
 8   Total day charge  3333 non-null    float64 
 9   Total eve minutes 3333 non-null    float64 
 10  Total eve calls   3333 non-null    int64  
 11  Total eve charge  3333 non-null    float64 
 12  Total night minutes 3333 non-null    float64 
 13  Total night calls  3333 non-null    int64
```

```

14 Total night charge      3333 non-null   float64
15 Total intl minutes     3333 non-null   float64
16 Total intl calls       3333 non-null   int64
17 Total intl charge      3333 non-null   float64
18 Customer service calls 3333 non-null   int64
19 Churn                  3333 non-null   bool
20 Day cost per minute    3333 non-null   float64
21 Eve cost per minute    3333 non-null   float64
22 Night cost per minute  3333 non-null   float64
dtypes: bool(1), float64(11), int64(8), object(3)
memory usage: 576.2+ KB

```

▼ What all manipulations have you done and insights you found?

The manipulation we have done in this dataset that we renamed the column name and created a new columns cost per minute wise which will help us to identify the distributions of price among different states.

The analysis of the dataset reveals interesting insights. The account length distribution suggests that most customers have an account length between 90-105 days, with a significant drop in retention after 105 days. States such as WV, MN, NY, VA, and WY have higher account lengths, indicating a correlation between state and customer loyalty. These states also have a higher number of users, indicating a strong customer base.

Furthermore, the states with higher account lengths also have a higher number of customer service calls, suggesting that these states may be facing more issues that require resolution. The relatively low churn rate of 16% indicates that the majority of customers trust the telecom company.

Additionally, around 10% of customers do not use the international plan, indicating potential areas for improvement in international plan

Saved successfully! ✖ suggests that customers generally make use of the voice mail plan, highlighting its popularity.

These insights provide valuable information for the telecom company to focus on improving international plans, addressing customer service concerns in specific states, and potentially enhancing the voice mail plan further. Upon further analysis of the dataset, it is evident that the total day minutes, total day calls, and total day charge columns are highly correlated, indicating that customers tend to make more calls and have longer conversations during the day. A similar pattern can be observed for the evening and night durations and charges.

Interestingly, the total international minutes and total international calls do not exhibit a strong correlation with the international plan column, suggesting that usage of international services may not solely depend on having an international plan.

The distribution of customer service calls indicates that the majority of customers make a low number of calls. However, there are a few outliers who make a significantly higher number of calls, potentially indicating more complex or unresolved issues.

Overall, these additional insights shed light on the calling patterns, international usage, and customer service interactions within the telecom company. Further exploration and analysis of these factors could provide valuable information for refining service offerings, addressing customer needs, and improving overall customer satisfaction.

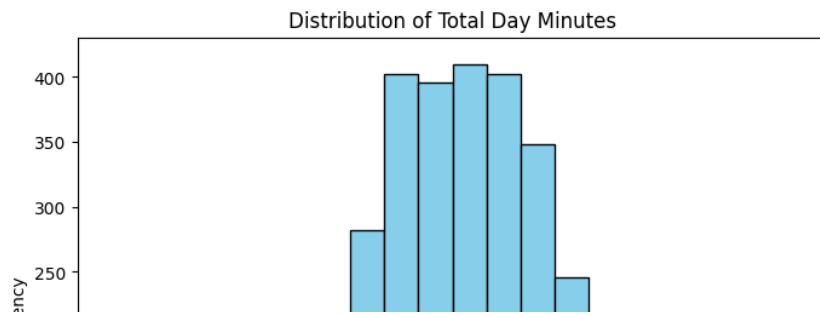
4. Data Vizualization, Storytelling & Experimenting with charts : Understand the relationships between variables

Histogram Chart-Distribution of Total Day Minutes(Univariate)

```

plt.figure(figsize=(8, 6)) #visualisation code
plt.hist(telecom_df['Total day minutes'], bins=20, color='skyblue', edgecolor='black')
plt.title('Distribution of Total Day Minutes')
plt.xlabel('Total Day Minutes')
plt.ylabel('Frequency')
plt.show()

```



- 1. Why did you pick the specific chart?

| |

The reason behind to choose this histogram is to show the distribution of the total day minutes.

--- | |

- 2. What is/are the insight(s) found from the chart?

50 | |

The insight found from the chart is that total day minutes is high in between 150 and 200 so we can assume that in day cost per min is less so people prefer calls in day more as compared to others.

Total Day Minutes

- 3. Will the gained insights help creating a positive business impact?

Saved successfully!

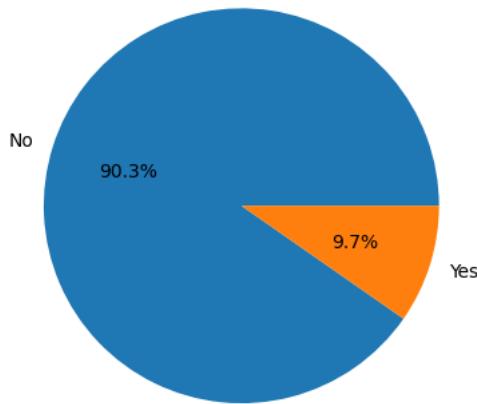
Business impact.



Pie Chart-Distribution of International Plan(Univariate)

```
telecom_df['International plan'].value_counts().plot(kind='pie', autopct='%1.1f%%') #visualisation code
plt.title('Distribution of International Plan')
plt.ylabel('')
plt.show()
```

Distribution of International Plan



- 1. Why did you pick the specific chart?

The reason behind the specific chart is to show the proportionality of the Distribution of International plan if it is YES and NO.

- 2. What is/are the insight(s) found from the chart?

As we can see in the chart 9.7% of the people choose International Plan and 90.3% of the people don't choose any international plan.

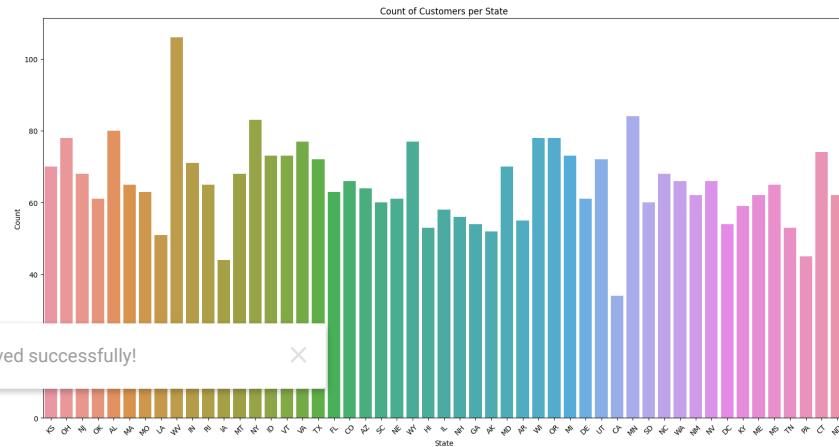
- 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

This insights will lead to postive growth so we can change the plan for the customer if they are intrested in international calls by giving them more offer or charge less on a cost per minute.

▼ Bar Chart-Count of Customers per State(Univariate)

```
#visualization code
plt.figure(figsize=(20,10)) # Adjust the figure size as needed
sns.countplot(data=telecom_df, x='State')
plt.title('Count of Customers per State')
plt.xlabel('State')
plt.ylabel('Count')
plt.xticks(rotation=45)
plt.show()
```



▼ 1. Why did you pick the specific chart?

The reason behind to choose this chart is to show in which states the no of customers are there which help us to analyse in which state where we have to focus more

▼ 2. What is/are the insight(s) found from the chart?

As we can see in the chart in state "WV" there is high customer as compared to others and in "CA" is very less customers.

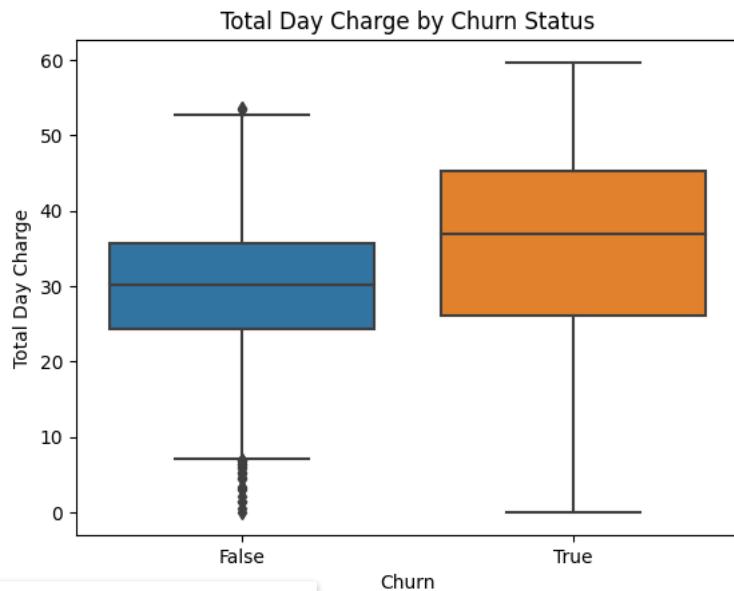
▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

The insights gained from analyzing the top states where orange telecomm's service is used the most can potentially create a positive business impact. By focusing on these states, the company can allocate resources, tailor marketing strategies, and provide localized offerings to effectively target the customer base. However, it is important to consider factors such as market saturation, competition, customer satisfaction, and external influences that could lead to negative growth. Careful analysis and proactive measures are necessary to address these challenges and ensure sustained positive growth in the telecom market.

▼ Box Plot-Total Day Charge by Churn Status(Univariate)

```
sns.boxplot(data=telecom_df, x='Churn', y='Total day charge') #visualization code
plt.title('Total Day Charge by Churn Status')
plt.xlabel('Churn')
plt.ylabel('Total Day Charge')
plt.show()
```



Saved successfully! X

▼ 1. Why did you pick the specific chart?

The reason behind using Box Plot is to gain insights about the distribution, central tendency, variability, and potential outliers in the data.

▼ 2. What is/are the insight(s) found from the chart?

A symmetrical box indicates a relatively balanced distribution, while a skewed box may suggest an asymmetrical distribution. The position of the median line within the box provides information about the central tendency of the data. If the median is closer to the bottom of the box, the distribution is positively skewed, and if it is closer to the top, the distribution is negatively skewed.

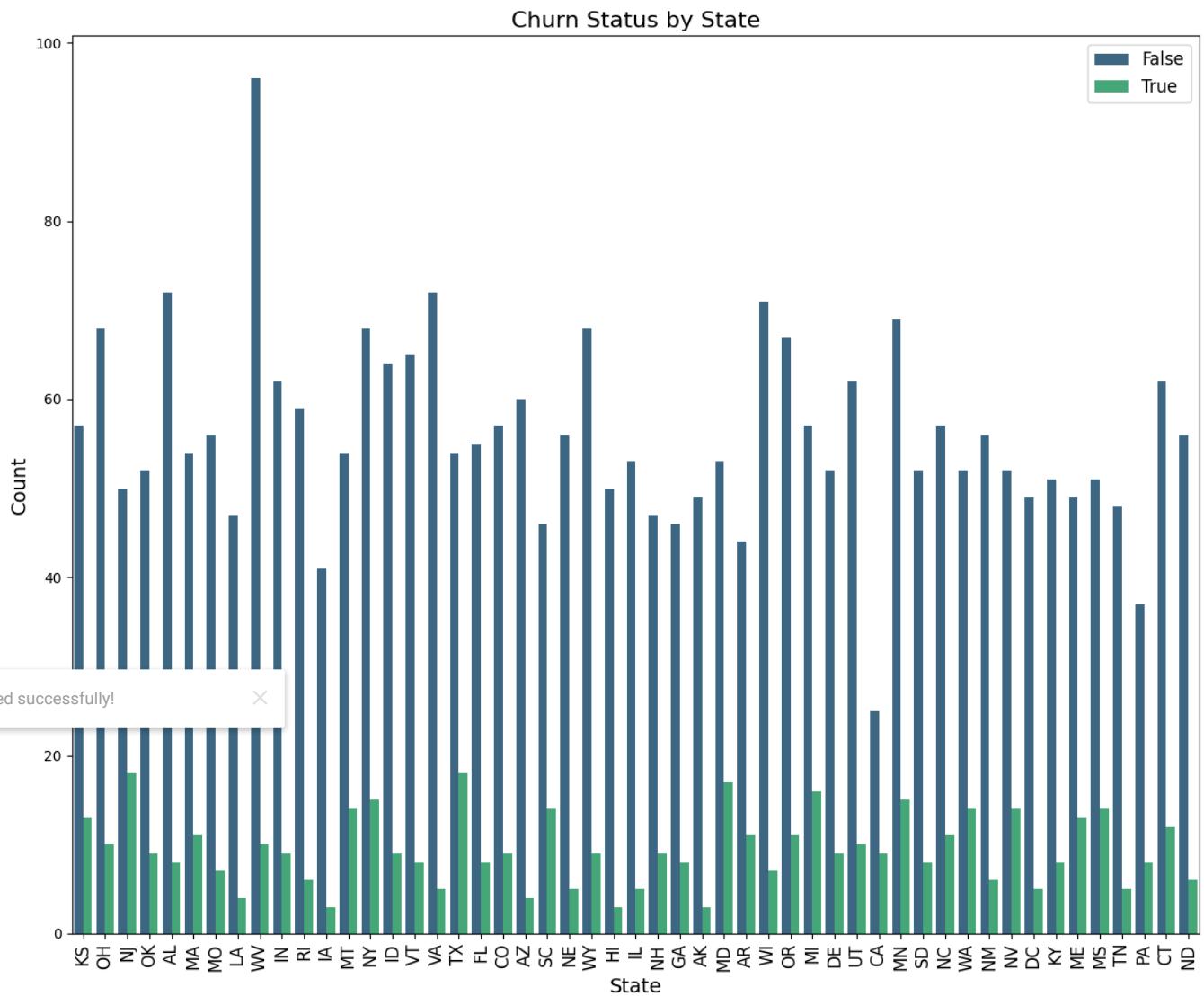
▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

The skewed distribution of Total Day Charge by Churn Status, specifically for customers with a "False" churn status, suggests a potential negative impact on growth. Higher day charges for these customers may indicate dissatisfaction or increased costs, potentially leading to a higher churn rate. It is crucial to investigate the underlying reasons behind the skewed distribution and take corrective actions. By addressing concerns related to service quality, pricing, or customer support, the company can mitigate the risk of negative growth and work towards creating a positive business impact.

▼ Bar Chart-Churn Status by State(Bivariate)

```
plt.figure(figsize=(12, 10))
sns.countplot(data=telecom_df, x='State', hue='Churn', palette='viridis')
plt.title('Churn Status by State', fontsize=16)
plt.xlabel('State', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=90, fontsize=12)
plt.legend(['False', 'True'], loc='upper right', fontsize=12)
plt.tight_layout()
plt.show()
```



▼ 1. Why did you pick the specific chart?

The specific reason to pick the chart is to compare the churn status state wise.

▼ 2. What is/are the insight(s) found from the chart?

From the chart showing the churn count by state, one insight that can be observed is that the churn count is relatively lower in the state of Iowa (IA) compared to other states. This suggests that customers from Iowa have a relatively lower likelihood of churning compared to customers from other states. This insight can be valuable for the business to understand the factors or strategies that contribute to higher customer retention in Iowa, which can then be leveraged to improve customer retention in other states.

▼ 3. Will the gained insights help creating a positive business impact?

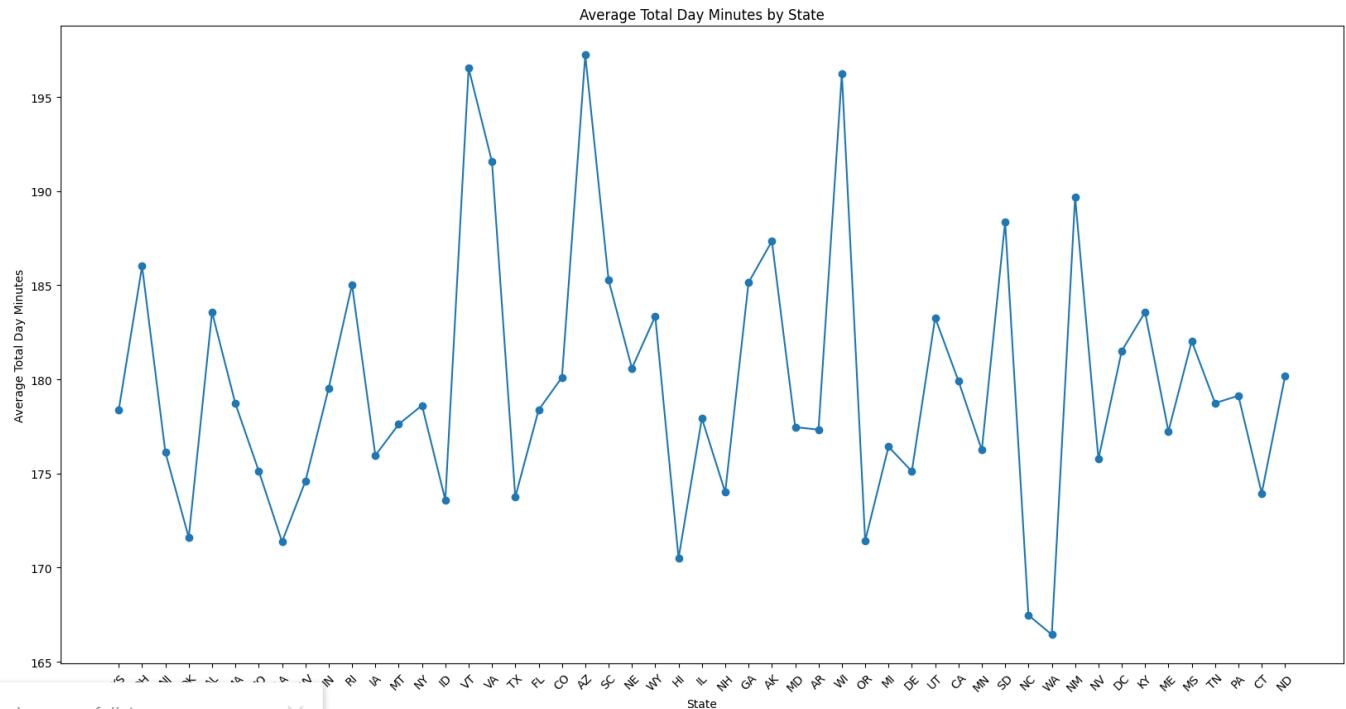
Are there any insights that lead to negative growth? Justify with specific reason.

The insight that customers from Iowa have a lower churn count can potentially create a positive business impact by understanding factors for higher customer retention. However, without further analysis, it is challenging to identify specific insights leading to negative growth. Multiple factors like customer satisfaction, service quality, and pricing need consideration to determine if any insights from the chart would have a negative impact on growth.

▼ Line Chart-Average Total Day Minutes by State(Bivariate)

```
states = telecom_df['State'].unique() #code for the line chart
plt.figure(figsize=(20,10))
telecom_df.groupby('State')['Total day minutes'].mean().plot(kind='line', marker='o')
plt.title('Average Total Day Minutes by State')
plt.xlabel('State')
plt.ylabel('Average Total Day Minutes')
```

```
plt.xticks(np.arange(len(states)), states, rotation=45)
plt.show()
```



Saved successfully! ×

▼ 1. Why did you pick the specific chart?

A line plot was chosen to visualize the average total day minutes by state because it allows for easy comparison between states, shows variations in values, and can reveal trends or patterns.

▼ 2. What is/are the insight(s) found from the chart?

Insights from the chart indicate that Vermont (VT), Arizona (AZ), and Wisconsin (WI) have higher average total day minutes, while North Carolina (NC) and Washington (WA) have lower average total day minutes. These variations in call durations across states can inform targeted marketing strategies and service offerings to cater to customer preferences in each region.

▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

The gained insights on variations in average total day minutes by state can positively impact the business by enabling targeted strategies.

▼ Scatter Plot-Total Day Minutes vs. Total Eve Minutes(Bivariate)

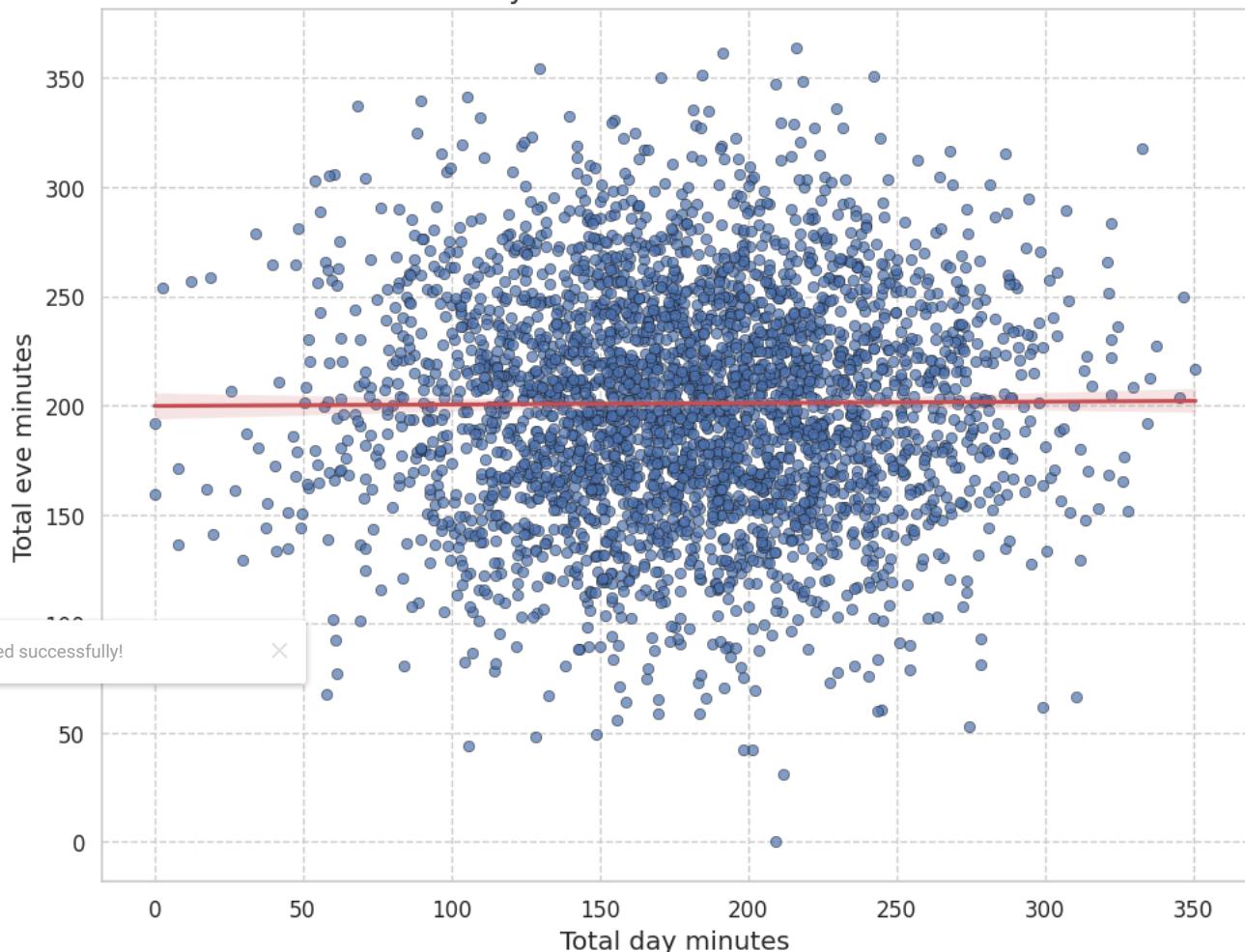
```
sns.set(style='whitegrid', font_scale=1.2)
# Create the scatter plot with customized settings
plt.figure(figsize=(10, 8)) # Adjust the figure size as needed
sns.scatterplot(data=telecom_df, x='Total day minutes', y='Total eve minutes', alpha=0.7, edgecolor='k')
plt.title('Total Day Minutes vs. Total Eve Minutes', fontsize=16)
plt.xlabel('Total Day Minutes')
plt.ylabel('Total Eve Minutes')
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.grid(True, linestyle='--')

# Add a regression line
sns.regressionplot(data=telecom_df, x='Total day minutes', y='Total eve minutes', scatter=False, color='r')

# Show the plot
```

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

Total Day Minutes vs. Total Eve Minutes



▼ 1. Why did you pick the specific chart?

The scatter plot with a horizontal line was chosen to visualize the relationship between 'Total Day Minutes' and 'Total Eve Minutes' and assess any outliers. The plot helps identify any patterns or correlations between the variables, and the added mean line provides a reference point for comparison. This chart enables insights into usage patterns, outlier detection, and decision-making regarding service offerings or customer engagement strategies.

▼ 2. What is/are the insight(s) found from the chart?

The scatter plot comparing 'Total Day Minutes' and 'Total Eve Minutes' indicates a positive correlation between the two variables. The red horizontal line represents the mean value of 'Total Eve Minutes'. Outliers can be identified, and the plot provides insights into the distribution and relationship between the variables, helping understand the usage patterns and potential deviations from the average evening minutes.

▼ 3. Will the gained insights help creating a positive business impact?

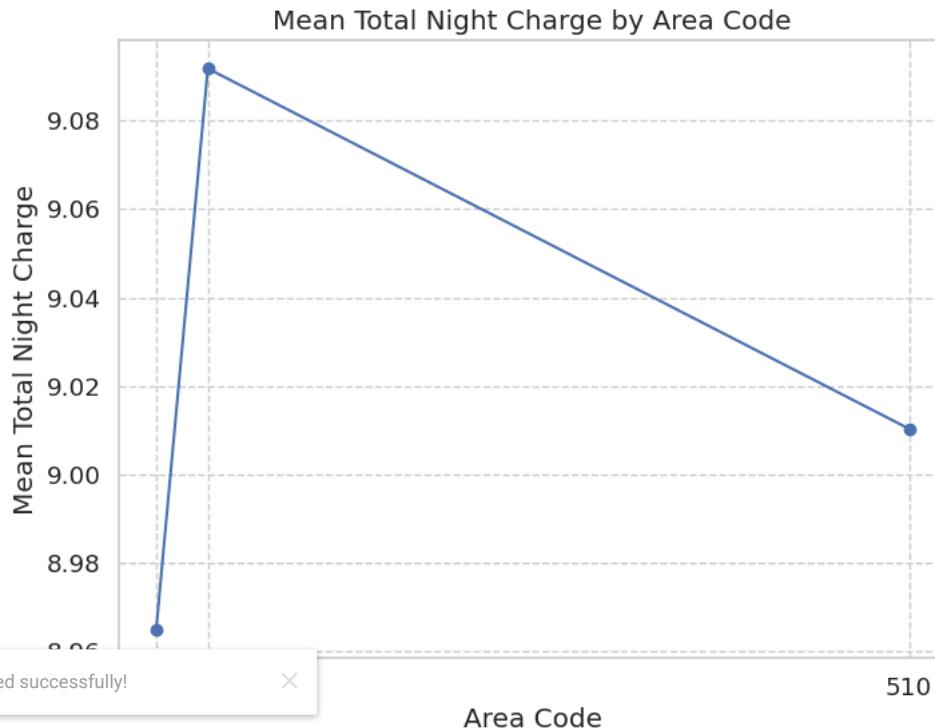
Are there any insights that lead to negative growth? Justify with specific reason.

yes it helps to gain the insights. from the chart we can see both are correlated with each other so we have to consider both before giving the plans.

Line Chart-Mean Total Night Charge by Area Code(Bivariate)

```
plt.figure(figsize=(8, 6)) #data visualisation
mean_night_charge = telecom_df.groupby('Area code')['Total night charge'].mean()
plt.plot(mean_night_charge.index, mean_night_charge, marker='o')
plt.title('Mean Total Night Charge by Area Code')
plt.xlabel('Area Code')
plt.ylabel('Mean Total Night Charge')
```

```
plt.xticks(mean_night_charge.index)
plt.grid(True, linestyle='--')
plt.show()
```



▼ 1. Why did you pick the specific chart?

The main motive to choose this line chart to show the mean total night charge area code to know the trend in different area code.

▼ 2. What is/are the insight(s) found from the chart?

The chart reveals that the mean total night charge for area code 415 is higher compared to area codes 510 and 408, indicating an imbalance in pricing or usage patterns. This suggests the need to equalize the mean total night charge among the area codes to ensure fairness. Further analysis can be done to understand the factors contributing to the disparity and develop strategies to achieve pricing consistency across all area codes.

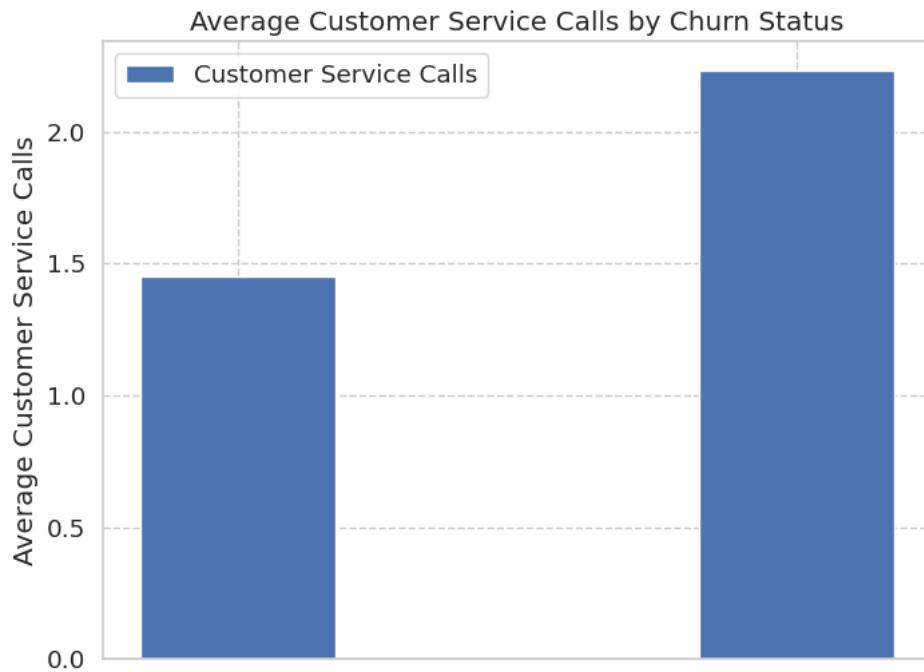
▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

The gained insight of the disparity in mean total night charges can positively impact the business by promoting pricing fairness and enhancing customer satisfaction. However, without considering other factors, it is challenging to identify specific insights that may lead to negative growth. A comprehensive analysis is necessary to assess potential negative growth implications, considering various factors like market dynamics, customer demand, and competitive landscape.

▼ Bar Chart-Average Customer Service Calls by Churn Status(Bivariate)

```
plt.figure(figsize=(8, 6)) #data visualisation code
churn_counts = telecom_df.groupby('Churn')['Customer service calls'].mean()
bar_width = 0.35
index = np.arange(len(churn_counts.index))
plt.bar(index, churn_counts, width=bar_width, label='Customer Service Calls')
plt.title('Average Customer Service Calls by Churn Status')
plt.xlabel('Churn')
plt.ylabel('Average Customer Service Calls')
plt.xticks(index, churn_counts.index)
plt.legend()
plt.grid(True, linestyle='--')
plt.show()
```



▼ 1. Why did you pick the specific chart?

Saved successfully! X

Facilitate easy comparison of the average customer service calls between different churn statuses. This chart effectively represents the categorical churn status variable and the numerical average customer service calls. Its simplicity and familiar format allow for easy interpretation and understanding by a wide range of audiences.

▼ 2. What is/are the insight(s) found from the chart?

The insight from the chart is that, on average, customers who churn (churn=True) have higher customer service call volumes compared to customers who do not churn (churn=False). However, the difference in average customer service calls between the two churn statuses is relatively moderate, with a range of only 1.7.

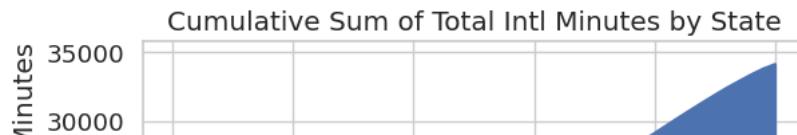
▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Insights of higher customer service calls for churned customers can help improve retention but no specific negative growth insights identified.

Area Chart-Cumulative Sum of Total Intl Minutes by State(Bivariate)

```
df = telecom_df.groupby('State')['Total intl minutes'].sum().sort_values(ascending=False) # visualization code
df.cumsum().plot(kind='area')
plt.title('Cumulative Sum of Total Intl Minutes by State')
plt.xlabel('States')
plt.ylabel('Cumulative Sum of Total Intl Minutes')
plt.show()
```



▼ 1. Why did you pick the specific chart?



The area plot was chosen to visualize the cumulative sum of total international minutes by state as it effectively showcases the progression and accumulation of minutes over the different states. The filled area under the curve emphasizes the overall magnitude and allows for easy comparison of the cumulative sums between states.

▼ 2. What is/are the insight(s) found from the chart?



The insight from the chart is that states like CA, DC, NE, MS, MD, and WV have a higher cumulative sum of total international minutes. This indicates potential higher demand or usage of international calling services in these states compared to others.

▼ 3. Will the gained insights help creating a positive business impact?

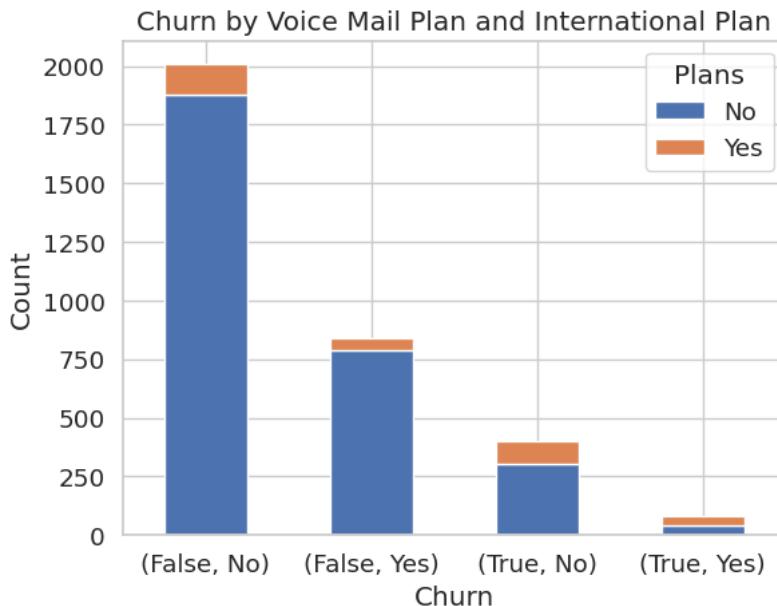
Are there any insights that lead to negative growth? Justify with specific reason.

The gained insight of higher cumulative international minutes in certain states can potentially help create a positive business impact by highlighting regions with higher demand. No specific insights indicating negative growth are identified from the given information.

Saved successfully! X

Churn by Voice Mail Plan and International Plan(Multivariate)

```
df = telecom_df.groupby(['Churn', 'Voice mail plan', 'International plan']).size().unstack() #visualization code
df.plot(kind='bar', stacked=True)
plt.title('Churn by Voice Mail Plan and International Plan')
plt.xlabel('Churn')
plt.ylabel('Count')
plt.legend(title='Plans', loc='upper right')
plt.xticks(rotation=0)
plt.show()
```



▼ 1. Why did you pick the specific chart?

The specific chart, a stacked bar chart comparing churn by voice mail plan and international plan, was chosen to visually depict the count of each plan combination and their distribution across churn status. It effectively illustrates the relationships between churn and the presence/absence of voice mail and international plans.

▼ 2. What is/are the insight(s) found from the chart?

Firstly, the majority of customers, indicated by the highest count, do not have a voice mail plan or an international plan (False, No). This suggests a significant portion of customers opt not to subscribe to these additional services. Secondly, the count decreases progressively from (False, Yes) to (True, No) to (True, Yes), indicating varying levels of adoption for voice mail and international plans among churned and non-churned customers. Overall, the chart highlights the distribution of plan combinations and their association with churn status, providing valuable insights into customer preferences and potential factors influencing churn rates.

- ▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

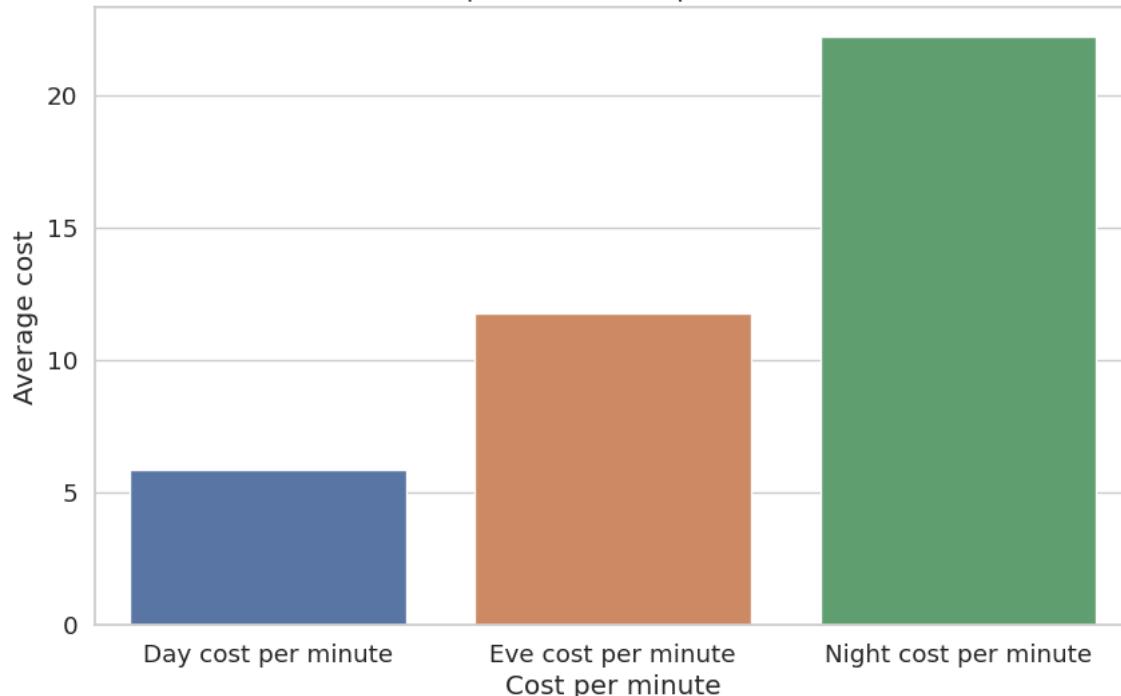
The gained insights can help create a positive business impact by identifying plan preferences.

Bar Chart-Comparison of Cost per Minute(Multivariate)

```
cost_per_minute = ['Day cost per minute', 'Eve cost per minute', 'Night cost per minute'] # Data for comparison
values = [telecom_df['Day cost per minute'].mean(),
          telecom_df['Eve cost per minute'].mean(),
          telecom_df['Night cost per minute'].mean()]
# Setting the figure size
plt.figure(figsize=(10, 6))
# Creating the bar chart using Seaborn
sns.barplot(x=cost_per_minute, y=values)
plt.xlabel('Cost per minute')
plt.ylabel('Average cost')
```

Saved successfully! X

Comparison of Cost per Minute



- ▼ 1. Why did you pick the specific chart?

The specific bar chart was chosen to compare the average cost per minute for different time periods (day, evening, and night) in a visually clear and concise manner.

- ▼ 2. What is/are the insight(s) found from the chart?

The insight from the chart is that the average cost per minute follows a pattern, with the night cost per minute being the highest, followed by the evening cost per minute, and the day cost per minute being the lowest. This indicates that the cost of phone calls tends to be higher during the night hours compared to the evening and day hours.

- ▼ 3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

The gained insights regarding the varying average cost per minute for different time periods can potentially help create a positive business impact. Businesses can strategically adjust pricing or promotional offers based on these insights to optimize revenue and customer satisfaction.

Line Chart-Total day calls,Total eve calls,Total night calls for top 10 States

```
fig, axes = plt.subplots(3, 1, figsize=(12, 10))

# Selecting the top 10 states based on total call counts
top_10_states = telecom_df.groupby('State')[['Total day calls', 'Total eve calls', 'Total night calls']].sum().sum(axis=1).nlargest(10).index

# Filtering the data for the top 10 states
top_10_data = telecom_df.loc[telecom_df['State'].isin(top_10_states)]

# Grouping the filtered data by state and calculating the mean of call counts
call_counts = top_10_data.groupby('State')[['Total day calls', 'Total eve calls', 'Total night calls']].mean()

# Plotting the line charts for each call type
for i, call_type in enumerate(call_counts.columns):
    sns.lineplot(data=call_counts, x=call_counts.index, y=call_type, ax=axes[i])
    axes[i].set_title(f'Average {call_type} by State (Top 10 States)')
    axes[i].set_xlabel('State')
    axes[i].set_ylabel('Average Call Count')
    axes[i].set_xticks(range(len(call_counts.index)))
    axes[i].set_xticklabels(call_counts.index, rotation=45)
```

Saved successfully! ×

Average Total day calls by State (Top 10 States)



1. Why did you pick the specific chart?

I picked the specific chart, which is a line chart, because it allows for the comparison of trends or changes over time or different categories. In this case, the line chart is used to compare the average call counts across different call types (total day calls, total eve calls, and total night calls) for the top 10 states. It helps visualize any variations or patterns in call counts for each call type across the selected states.

2. What is/are the insight(s) found from the chart?

Average Total eve calls by State (Top 10 States)

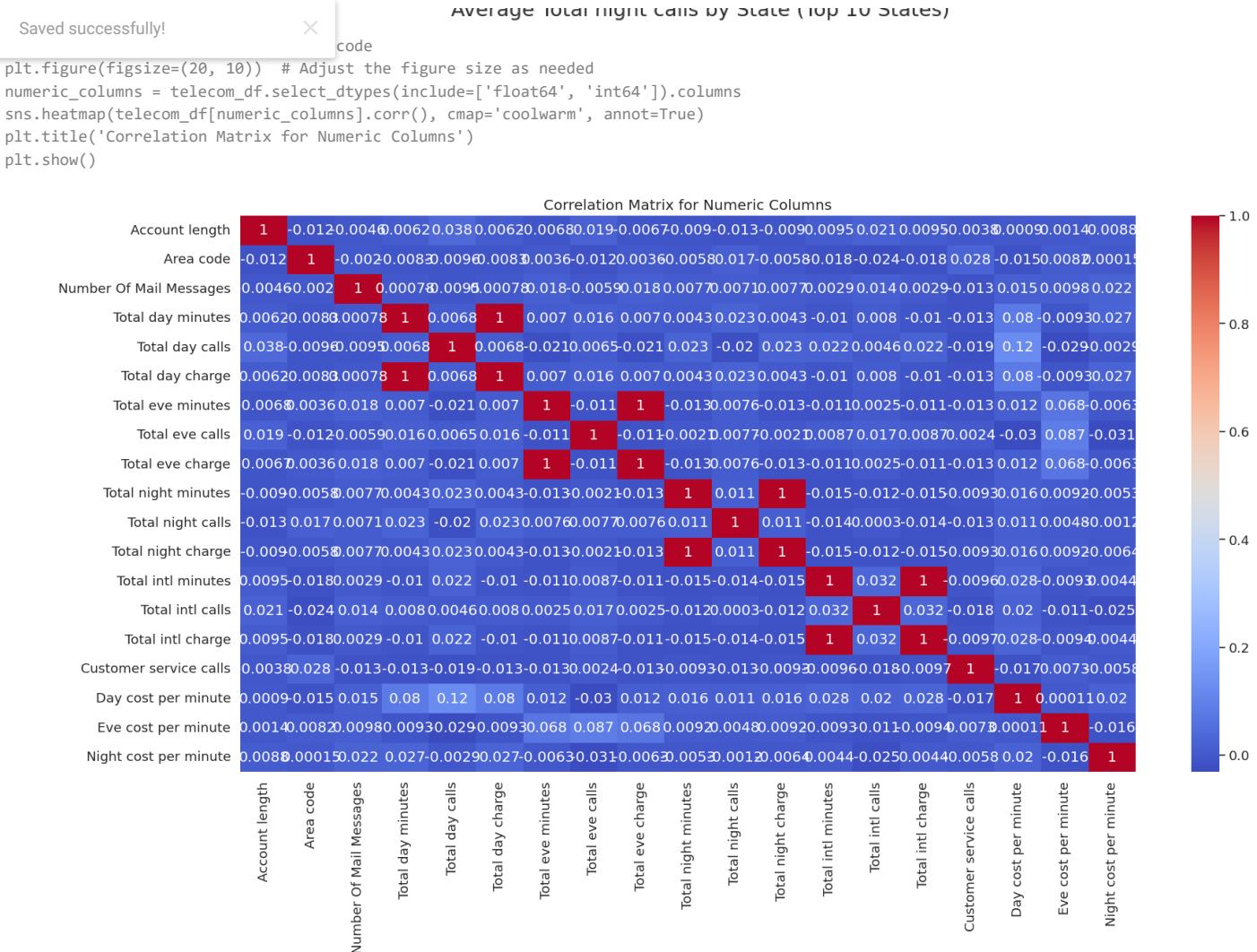
The insights from the chart show the average call counts for 'Total day calls', 'Total eve calls', and 'Total night calls' across the top 10 states, allowing for comparisons of call activity between different call types and states.

3. Will the gained insights help creating a positive business impact?

Are there any insights that lead to negative growth? Justify with specific reason.

Yes, Positive impact Insights can inform targeted strategies for call optimization.

Correlation Heatmap-Correlation Matrix for Numeric Columns(Multivariate)



▼ 1. Why did you pick the specific chart?

The correlation matrix provides an overview of the relationships between variables in the dataset. It helps identify patterns and dependencies.

▼ 2. What is/are the insight(s) found from the chart?

From the correlation matrix, we can observe the strength and direction of the relationships between different variables. For example, there is a positive correlation between total day minutes and total day charge, indicating that longer calls result in higher charges. Additionally, there is a negative correlation between customer service calls and churn, suggesting that higher customer service calls are associated with lower churn rates.

▼ Pair Plot-Pairwise Relationships of Numeric Columns(Multivariate)

```
numeric_columns = telecom_df.select_dtypes(include=['float64', 'int64']).columns

# Set style and context for improved aesthetics
sns.set(style='ticks', font_scale=1.2)

# Create the pair plot with customized settings
g = sns.pairplot(telecom_df[numeric_columns], diag_kind='kde', plot_kws={'alpha': 0.6, 's': 30}, corner=True)

# Customize plot titles and labels
g.fig.suptitle('Pairwise Relationships of Numeric Columns', y=1.03, fontsize=16)

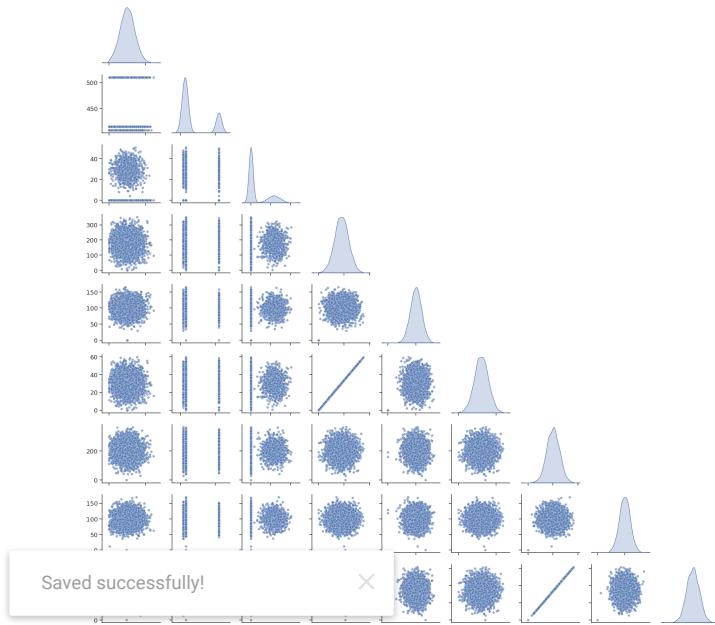
Saved successfully! 
# Adjust subplot spacing
g.fig.subplots_adjust(top=0.92, bottom=0.08, left=0.08, right=0.92, hspace=0.2, wspace=0.2)

# Save the plot as a PNG image
g.savefig('pairplot.png', dpi=300)

# Show a message indicating the image is saved
print("Pair plot image saved as 'pairplot.png'")
```

Pair plot image saved as 'pairplot.png'

Pairwise Relationships of Numeric Columns



▼ 1. Why did you pick the specific chart?



The pair plot was chosen because it allows for visual exploration of pairwise relationships between multiple numeric variables, helping to identify potential correlations and patterns in the data.



▼ 2. What is/are the insight(s) found from the chart?



The pair plot provides insights into the relationships between different numeric variables in the dataset. It helps identify correlations, such as a positive relationship between total day minutes and total day charge, and can reveal patterns or clusters that may exist within the data.

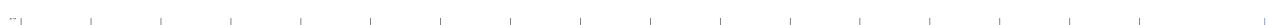


▼ 5. Solution to Business Objective



▼ What do you suggest the client to achieve Business Objective ?

Explain Briefly.



Solution to Reduce Customer Churn

1. Offer personalized discounts or incentives to customers who are at a higher risk of churning.
2. Enhance the quality and reliability of customer service to ensure prompt issue resolution and customer satisfaction.
3. Conduct customer surveys and implement improvements based on feedback to address pain points and enhance overall experience.
4. Implement a loyalty program to reward long-term customers and encourage their continued loyalty.
5. Provide additional value-added services or features that differentiate your offering from competitors.
6. Offer flexible pricing plans or options that cater to the diverse needs and budgets of customers.
7. Implement targeted marketing campaigns to re-engage inactive or at-risk customers.
8. Develop a customer retention team dedicated to monitoring and proactively addressing customer churn.
9. Leverage data analytics to identify early warning signs of potential churn and take preventive measures.
10. Foster a customer-centric culture within the organization, focusing on building strong relationships and delivering exceptional customer experiences.

These strategies can help reduce customer churn by improving customer satisfaction, addressing pain points, and providing added value to retain customers in a highly competitive market.

▼ Conclusion