CUSTOMER CHURN PREDICTION

W W W . R E A L L Y G R E A T S I T E . C O M

Executor: Intern
Le Trung Kien

Topic

Customer churn prediction is a critical aspect of business management, particularly for industries like telecommunications, internet service providers, pay TV companies, insurance firms, and alarm monitoring services. It involves understanding and addressing customer attrition, which refers to the loss of clients or customers.
For businesses in these sectors, measuring customer attrition is a vital business metric. This is because retaining an existing customer is significantly more cost-effective than acquiring a new one. As a result, these companies often have customer service branches dedicated to re-engaging customers who are considering leaving. This is because the long-term value of recovered customers far outweighs that of newly acquired ones.

Solution

To address customer churn, predictive analytics comes into play, churn prediction models to assess the likelihood of customers leaving. These models prioritize a small list of potential defectors, enabling businesses to concentrate their customer retention efforts on those who are most at risk of churning.

Objective

Customer churn prediction is a critical aspect of business management, particularly for industries like telecommunications, internet service providers, pay TV companies, insurance firms, and alarm monitoring services. It involves understanding and addressing customer attrition, which refers to the loss of clients or customers.
For businesses in these sectors, measuring customer attrition is a vital business metric. This is because retaining an existing customer is significantly more cost-effective than acquiring a new one. As a result, these companies often have customer service branches dedicated to re-engaging customers who are considering leaving. This is because the long-term value of recovered customers far outweighs that of newly acquired ones.

Contents

CHAPTER I: DATA CLEANING

CHAPTER II: EDA-REPRESENTS THE CORRELATION BETWEEN THE TARGET DATA COLUMN AND THE REMAINING COLUMNS

CHAPTER III: APPLY THE MACHINE LEARNING RANDOM FOREST MODEL TO PREDICT POTENTIAL CUSTOMER CHURN

DATA CLEANING

Use functions from the pandas library to process and clean the data.

DATA CLEANING

Import required libraries

```
1   import pandas as pd
2   import matplotlib.pyplot as plt
3   from sklearn.preprocessing import LabelEncoder
4   from sklearn.tree import DecisionTreeClassifier
5   from sklearn import metrics
6   from sklearn.metrics import accuracy_score, auc, confusion_matrix, roc_auc_score, roc_curve, recall_score
```

Reading dataset

```
 9   # Reading dataset
10   train_data = pd.read_csv('data/churn-bigml-80.csv')
11   test_data = pd.read_csv('data/churn-bigml-20.csv')
```

Check and delete rows missing value

```
17   # Delete rows missing values
18   train_data.dropna()
19   test_data.dropna()
20
21   # Remove duplicate rows
22   train_data = train_data.drop_duplicates()
23   test_data = test_data.drop_duplicates()
```

EDA-Represents the correlation between the target data column and the remaining columns

Analyze data, utilize charts, to derive correlations between the target column and the remaining columns.

EDA-Represents the correlation between the target data column and the remaining columns

Checking the information of the data

```
Data columns (total 20 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   State                   2666 non-null   object 
 1   Account length          2666 non-null   int64  
 2   Area code               2666 non-null   int64  
 3   International plan      2666 non-null   object 
 4   Voice mail plan         2666 non-null   object 
 5   Number vmail messages   2666 non-null   int64  
 6   Total day minutes       2666 non-null   float64
 7   Total day calls         2666 non-null   int64  
 8   Total day charge        2666 non-null   float64
 9   Total eve minutes       2666 non-null   float64
 10  Total eve calls         2666 non-null   int64  
 11  Total eve charge        2666 non-null   float64
 12  Total night minutes     2666 non-null   float64
 13  Total night calls       2666 non-null   int64  
 14  Total night charge      2666 non-null   float64
 15  Total intl minutes      2666 non-null   float64
 16  Total intl calls        2666 non-null   int64  
 17  Total intl charge       2666 non-null   float64
 18  Customer service calls  2666 non-null   int64  
 19  Churn                   2666 non-null   bool   
dtypes: bool(1), float64(8), int64(8), object(3)
memory usage: 419.2+ KB
```

Displaying the column names of the dataset

```
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')
```

EDA-Represents the correlation between the target data column and the remaining columns

Display basic statistics

	Account length	Area code	Number vmail messages	Total day minutes	Total day calls	Total day charge	...	Total night calls	Total night charge	Total intl minutes	Total intl calls	Total intl charge	Customer service calls
count	2666.000000	2666.000000	2666.000000	2666.00000	2666.000000	2666.000000	...	2666.000000	2666.000000	2666.000000	2666.000000	2666.000000	2666.000000
mean	100.620405	437.438860	8.021755	179.48162	100.310203	30.512404	...	100.106152	9.052689	10.237022	4.467367	2.764490	1.562641
std	39.563974	42.521018	13.612277	54.21035	19.988162	9.215733	...	19.418459	2.285120	2.788349	2.456195	0.752812	1.311236
min	1.000000	408.000000	0.000000	0.00000	0.000000	0.000000	...	33.000000	1.970000	0.000000	0.000000	0.000000	0.000000
25%	73.000000	408.000000	0.000000	143.40000	87.000000	24.380000	...	87.000000	7.512500	8.500000	3.000000	2.300000	1.000000
50%	100.000000	415.000000	0.000000	179.95000	101.000000	30.590000	...	100.000000	9.050000	10.200000	4.000000	2.750000	1.000000
75%	127.000000	510.000000	19.000000	215.90000	114.000000	36.700000	...	113.000000	10.640000	12.100000	6.000000	3.270000	2.000000
max	243.000000	510.000000	50.000000	350.80000	160.000000	59.640000	...	166.000000	17.770000	20.000000	20.000000	5.400000	9.000000

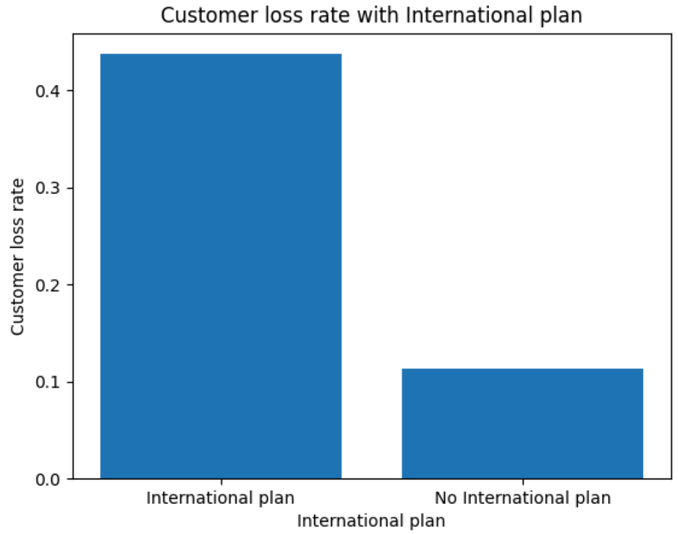
[8 rows x 16 columns]

	Account length	Area code	Number vmail messages	Total day minutes	Total day calls	Total day charge	...	Total night calls	Total night charge	Total intl minutes	Total intl calls	Total intl charge	Customer service calls
count	667.000000	667.000000	667.000000	667.000000	667.000000	667.000000	...	667.000000	667.000000	667.000000	667.000000	667.000000	667.000000
mean	102.841079	436.157421	8.407796	180.948126	100.937031	30.761769	...	100.113943	8.985907	10.238381	4.527736	2.764948	1.563718
std	40.819480	41.783305	13.994480	55.508628	20.396790	9.436463	...	20.172505	2.239429	2.807850	2.482442	0.758167	1.333357
min	1.000000	408.000000	0.000000	25.900000	30.000000	4.400000	...	42.000000	1.040000	0.000000	0.000000	0.000000	0.000000
25%	76.000000	408.000000	0.000000	146.250000	87.500000	24.860000	...	86.000000	7.560000	8.600000	3.000000	2.320000	1.000000
50%	102.000000	415.000000	0.000000	178.300000	101.000000	30.310000	...	100.000000	9.070000	10.500000	4.000000	2.840000	1.000000
75%	128.000000	415.000000	20.000000	220.700000	115.000000	37.520000	...	113.500000	10.420000	12.050000	6.000000	3.255000	2.000000
max	232.000000	510.000000	51.000000	334.300000	165.000000	56.830000	...	175.000000	16.550000	18.300000	18.000000	4.940000	8.000000

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate with "International plan"

```
intl_plan_churn = train_data[train_data['International plan'] == 'Yes']['Churn'].mean()
no_intl_plan_churn = train_data[train_data['International plan'] == 'No']['Churn'].mean()
plt.bar(['International plan', 'No International plan'], [intl_plan_churn, no_intl_plan_churn])
plt.xlabel('International plan')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate with International plan')
plt.show()
```

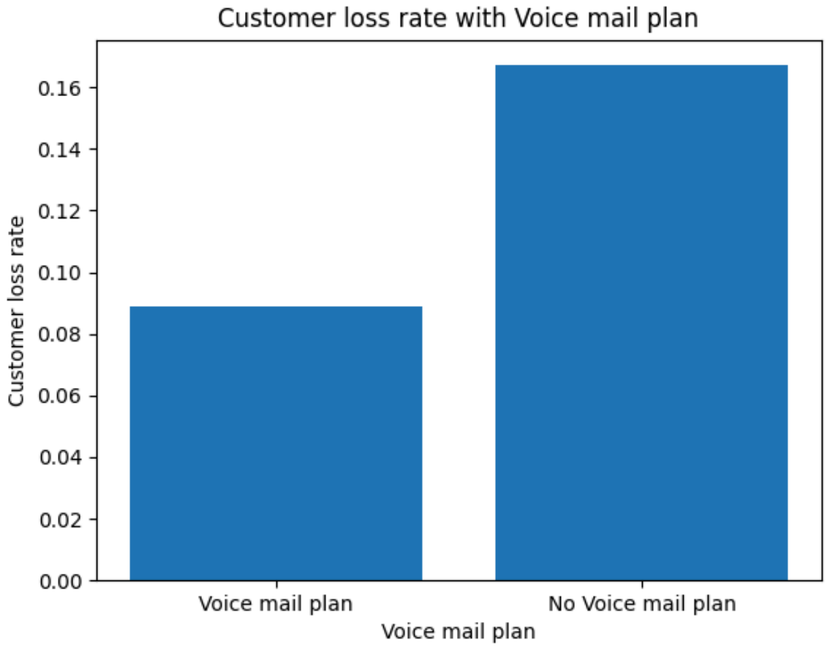


- International Plan: Churn Rate ≈ 0.42 (or 42%)
- No International Plan: Churn Rate ≈ 0.14 (or 14%)
- Insight: Customers with an International Plan are much more likely to churn compared to those without it. This could be due to dissatisfaction with the plan's pricing, quality, or coverage.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate with "Voice mail plan"

```
55  voice_mail_plan_churn = train_data[train_data['Voice mail plan'] == 'Yes']['Churn'].mean()
56  no_voice_mail_plan_churn = train_data[train_data['Voice mail plan'] == 'No']['Churn'].mean()
57  plt.bar(['Voice mail plan', 'No Voice mail plan'], [voice_mail_plan_churn, no_voice_mail_plan_churn])
58  plt.xlabel('Voice mail plan')
59  plt.ylabel('Customer loss rate')
60  plt.title('Customer loss rate with Voice mail plan')
```

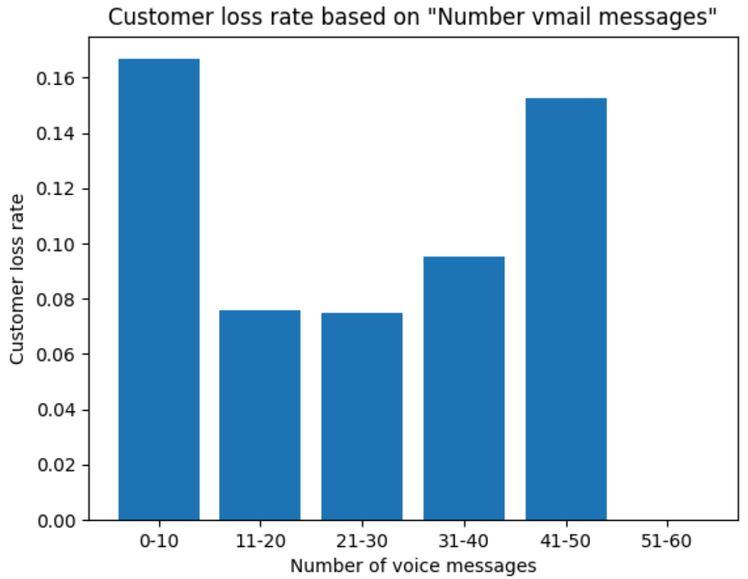


- Voice Mail Plan: Churn Rate ≈ 0.07 (or 7%)
- No Voice Mail Plan: Churn Rate ≈ 0.17 (or 17%)
- Insight: Customers with a Voice Mail Plan are less likely to churn compared to those without it. This suggests that the Voice Mail Plan might be a valued service that contributes to customer retention.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Number vmail messages"

```
# Analyze customer loss rate based on "Number vmail messages"
vmail_messages_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
vmail_messages_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
vmail_messages_churn = []
for i in range(len(vmail_messages_bins) - 1):
    mask = (train_data['Number vmail messages'] >= vmail_messages_bins[i]) & (train_data['Number vmail messages'] < vmail_messages_bins[i+1])
    vmail_messages_churn.append(train_data[mask]['Churn'].mean())
plt.bar(vmail_messages_labels, vmail_messages_churn)
plt.xlabel('Number of voice messages')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Number vmail messages"')
plt.show()
```



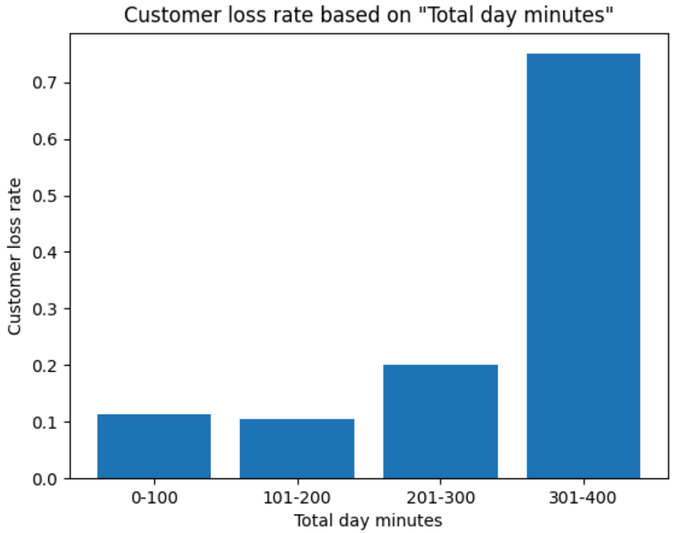
- As the number of voice mail messages increases, the churn rate generally decreases.
- Insight: Frequent use of voice mail services is associated with lower churn rates. This could indicate that customers who actively use voice mail services are more engaged and satisfied with their service.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total day minutes"

```
77  # Analyze customer loss rate based on "Total day minutes"
78  day_minutes_bins = [0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
79  day_minutes_labels = ['0-100', '101-200', '201-300', '301-400', '401-500', '501-600', '601-700', '701-800', '801-900', '901-1000']
80  day_minutes_churn = []
81  for i in range(len(day_minutes_bins) - 1
82      mask = (train_data['Total day minute                rain_data['Total day minutes'] < day_minutes_bins[i+1])
83      day_minutes_churn.append(train_data[mask]['Churn'].mean())
84  plt.bar(day_minutes_labels, day_minutes_churn)
85  plt.xlabel('Total day minutes')
86  plt.ylabel('Customer loss rate')
87  plt.title('Customer loss rate based on "Total day minutes')
88  plt.show()
```

(variable) mask: Series[bool]

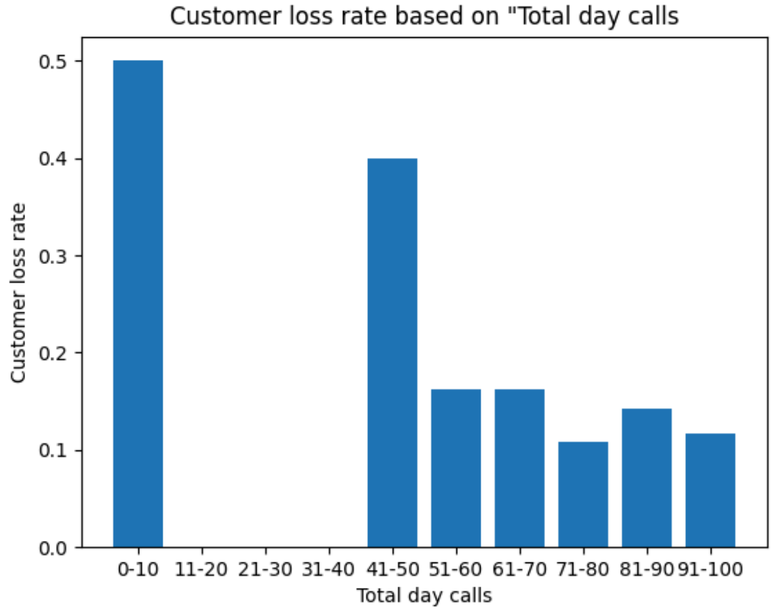


- Higher total day minutes are associated with a higher churn rate.
- Insight: Customers who use more day minutes may be more aware of pricing and may seek better deals, leading to higher churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total day calls"

```
91  day_calls_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
92  day_calls_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
93  day_calls_churn = []
94  for i in range(len(day_calls_bins) - 1):
95      mask = (train_data['Total day calls'] >= day_calls_bins[i]) & (train_data['Total day calls'] < day_calls_bins[i+1])
96      day_calls_churn.append(train_data[mask]['Churn'].mean())
97  plt.bar(day_calls_labels, day_calls_churn)
98  plt.xlabel('Total day calls')
99  plt.ylabel('Customer loss rate')
100 plt.title('Customer loss rate based on "Total day calls')
```

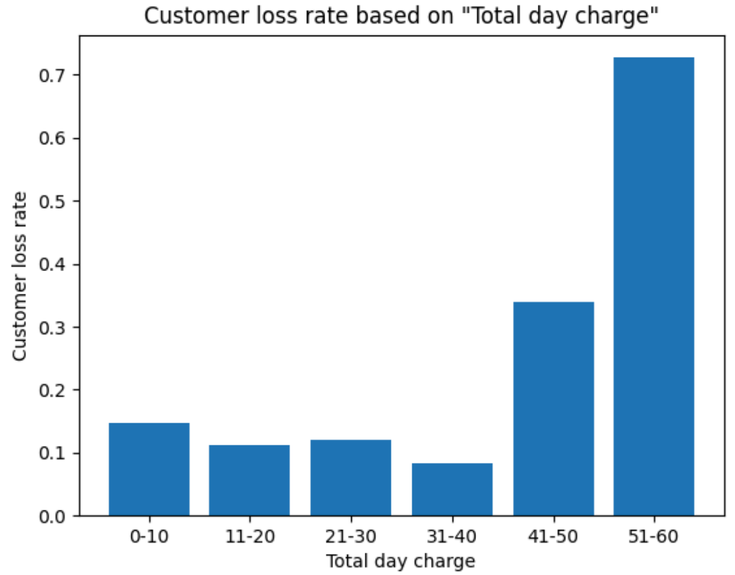


- No clear pattern, the churn rate varies across different bins of total day calls.
- Insight: The number of day calls does not show a strong correlation with churn rates, suggesting other factors may be more influential.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total day charge"

```python
# Analyze customer loss rate based on "Total day charge"
day_charge_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
day_charge_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
day_charge_churn = []
for i in range(len(day_charge_bins) - 1):
    mask = (train_data['Total day charge'] >= day_charge_bins[i]) & (train_data['Total day charge'] < day_charge_bins[i+1])
    day_charge_churn.append(train_data[mask]['Churn'].mean())
plt.bar(day_charge_labels, day_charge_churn)
plt.xlabel('Total day charge')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Total day charge')
plt.show()
```

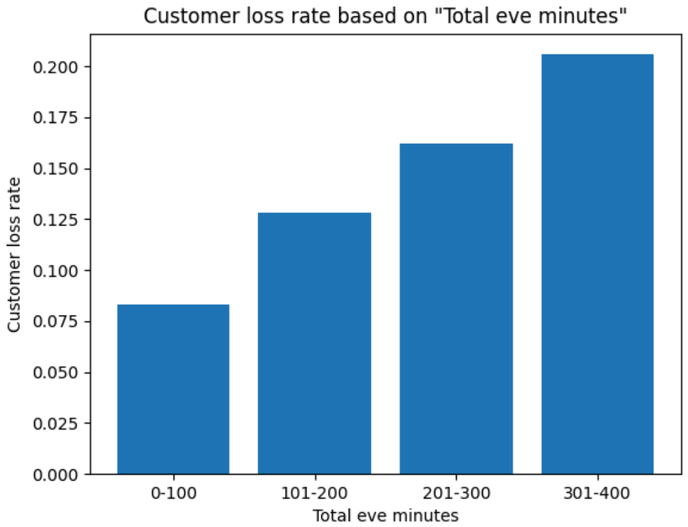


- Higher total day charges are associated with higher churn rates.
- Insight: Higher charges may lead to dissatisfaction and higher churn rates, indicating a sensitivity to pricing.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total eve minutes"

```python
# Analyze customer loss rate based on "Total eve minutes"
eve_minutes_bins = [0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
eve_minutes_labels = ['0-100', '101-200', '201-300', '301-400', '401-500', '501-600', '601-700', '701-800', '801-900', '901-1000']
eve_minutes_churn = []
for i in range(len(eve_minutes_bins) - 1):
    mask = (train_data['Total eve minutes'] >= eve_minutes_bins[i]) & (train_data['Total eve minutes'] < eve_minutes_bins[i+1])
    eve_minutes_churn.append(train_data[mask]['Churn'].mean())
plt.bar(eve_minutes_labels, eve_minutes_churn)
plt.xlabel('Total eve minutes')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Total eve minutes"')
plt.show()
```

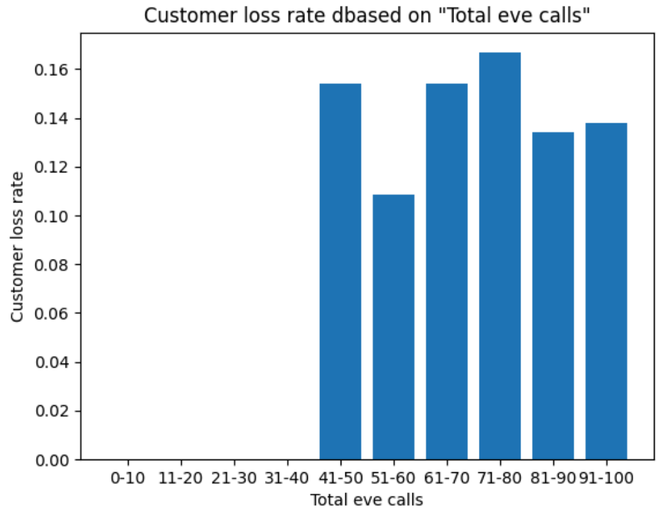


- Similar to day minutes, higher evening minutes are associated with higher churn rates.
- Insight: Extensive use during evenings could indicate customers' concern about cost, leading to higher churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total eve calls"

```python
# Analyze customer loss rate is based on "Total eve calls"
eve_calls_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
eve_calls_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
eve_calls_churn = []
for i in range(len(eve_calls_bins) - 1):
    mask = (train_data['Total eve calls'] >= eve_calls_bins[i]) & (train_data['Total eve calls'] < eve_calls_bins[i+1])
    eve_calls_churn.append(train_data[mask]['Churn'].mean())
plt.bar(eve_calls_labels, eve_calls_churn)
plt.xlabel('Total eve calls')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate dbased on "Total eve calls"')
plt.show()
```

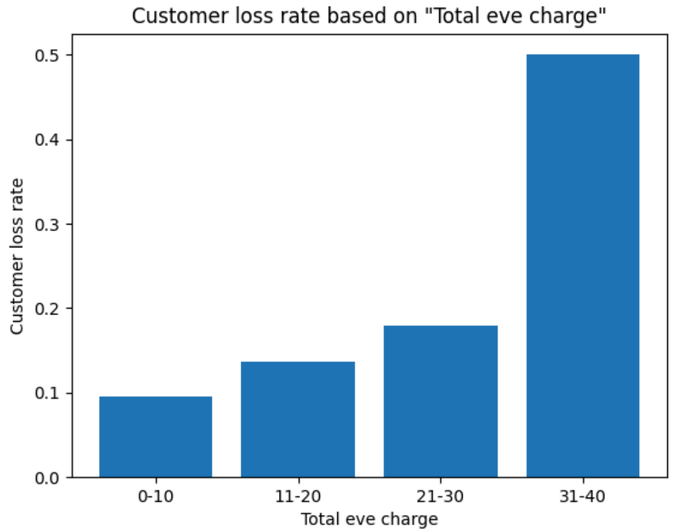


- Similar to day calls, no clear pattern; churn rate varies.
- Insight: The number of evening calls does not have a strong correlation with churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total eve charge"

```
143  eve_charge_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
144  eve_charge_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
145  eve_charge_churn = []
146  for i in range(len(eve_charge_bins) - 1):
147      mask = (train_data['Total eve charge'] >= eve_charge_bins[i]) & (train_data['Total eve charge'] < eve_charge_bins[i+1])
148      eve_charge_churn.append(train_data[mask]['Churn'].mean())
149  plt.bar(eve_charge_labels, eve_charge_churn)
150  plt.xlabel('Total eve charge')
151  plt.ylabel('Customer loss rate')
152  plt.title('Customer loss rate based on "Total eve charge"')
153  plt.show()
```

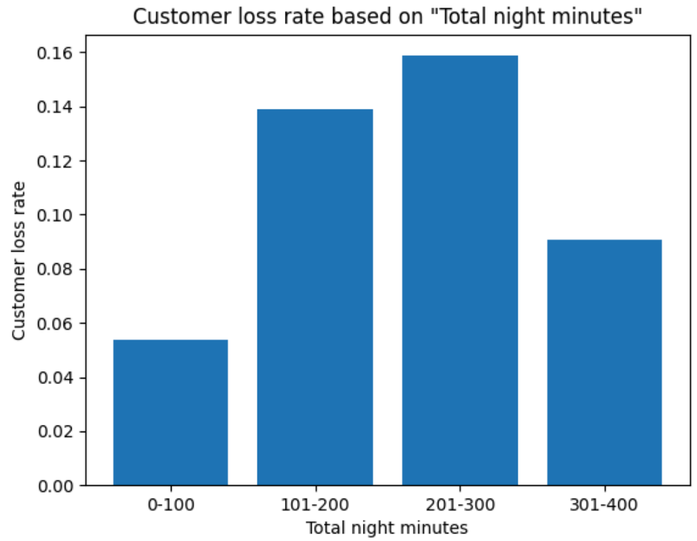


- Higher evening charges are associated with higher churn rates.
- Insight: High charges for evening use also contribute to higher churn rates, reinforcing the impact of cost on customer satisfaction.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total night minutes"

```python
# Analyze customer loss rate is based on "Total night minutes"
night_minutes_bins = [0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]
night_minutes_labels = ['0-100', '101-200', '201-300', '301-400', '401-500', '501-600', '601-700', '701-800', '801-900', '901-1000']
night_minutes_churn = []
for i in range(len(night_minutes_bins) - 1):
    mask = (train_data['Total night minutes'] >= night_minutes_bins[i]) & (train_data['Total night minutes'] < night_minutes_bins[i+1])
    night_minutes_churn.append(train_data[mask]['Churn'].mean())
plt.bar(night_minutes_labels, night_minutes_churn)
plt.xlabel('Total night minutes')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Total night minutes"')
plt.show()
```

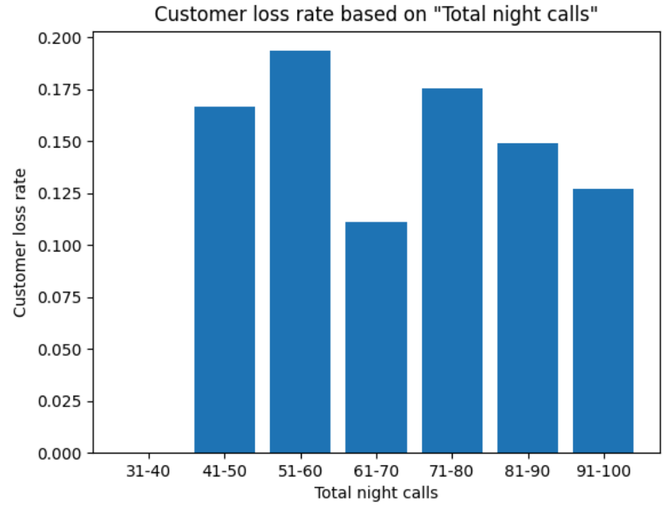


- Similar to day and evening minutes, higher night minutes are associated with higher churn rates.
- Insight: Extensive night usage also correlates with higher churn rates, likely due to cost concerns.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total night calls"

```
169  night_calls_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
170  night_calls_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
171  night_calls_churn = []
172  for i in range(len(night_calls_bins) - 1):
173      mask = (train_data['Total night calls'] >= night_calls_bins[i]) & (train_data['Total night calls'] < night_calls_bins[i+1])
174      night_calls_churn.append(train_data[mask]['Churn'].mean())
175  plt.bar(night_calls_labels, night_calls_churn)
176  plt.xlabel('Total night calls')
177  plt.ylabel('Customer loss rate')
178  plt.title('Customer loss rate based on "Total night calls"')
179  plt.show()
```

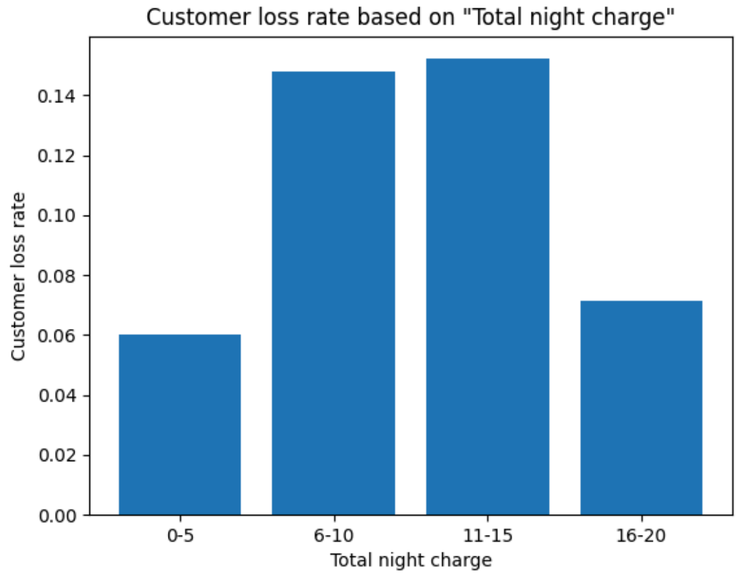


- Similar to day and evening calls, no clear pattern; churn rate varies.
- Insight: The number of night calls does not show a strong correlation with churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total night charge"

```
intl_plan_churn = train_data[train_data['International plan'] == 'Yes']['Churn'].mean()
no_intl_plan_churn = train_data[train_data['International plan'] == 'No']['Churn'].mean()
plt.bar(['International plan', 'No International plan'], [intl_plan_churn, no_intl_plan_churn])
plt.xlabel('International plan')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate with International plan')
plt.show()
```

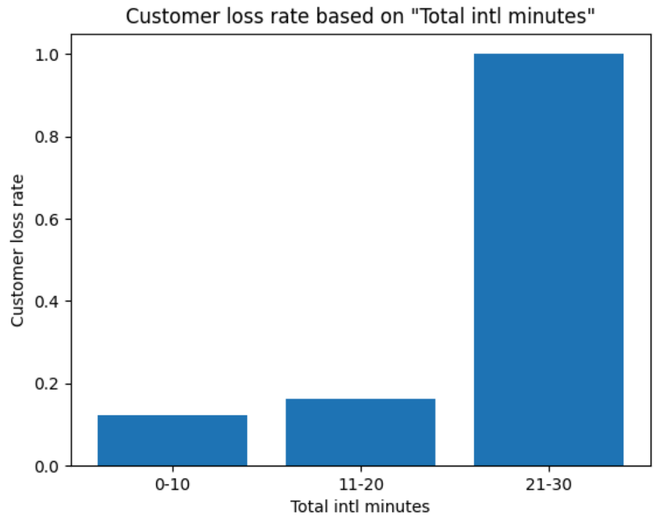


- Higher night charges are associated with higher churn rates.
- Insight: As with day and evening charges, high night charges also contribute to higher churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total intl minutes"

```python
# Analyze customer loss rate is based on "Total intl minutes"
intl_minutes_bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
intl_minutes_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70', '71-80', '81-90', '91-100']
intl_minutes_churn = []
for i in range(len(intl_minutes_bins) - 1):
    mask = (train_data['Total intl minutes'] >= intl_minutes_bins[i]) & (train_data['Total intl minutes'] < intl_minutes_bins[i+1])
    intl_minutes_churn.append(train_data[mask]['Churn'].mean())
plt.bar(intl_minutes_labels, intl_minutes_churn)
plt.xlabel('Total intl minutes')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Total intl minutes"')
plt.show()
```

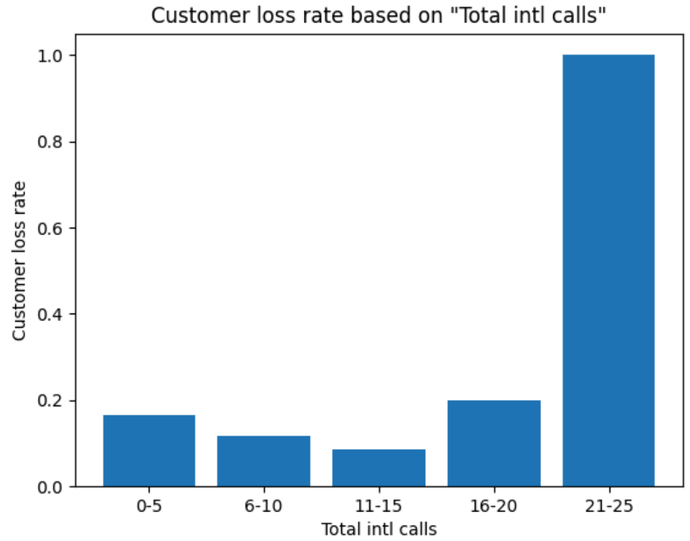


- Higher international minutes are associated with higher churn rates.
- Insight: Extensive international usage may lead to higher costs, contributing to higher churn rates.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Total intl calls"

```
207  # Analyze customer loss rate based on "Total intl calls"
208  intl_calls_bins = [0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
209  intl_calls_labels = ['0-5', '6-10', '11-15', '16-20', '21-25', '26-30', '31-35', '36-40', '41-45', '46-50']
210  intl_calls_churn = []
211  for i in range(len(intl_calls_bins) - 1):
212      mask = (train_data['Total intl calls'] >= intl_calls_bins[i]) & (train_data['Total intl calls'] < intl_calls_bins[i+1])
213      intl_calls_churn.append(train_data[mask]['Churn'].mean())
214  plt.bar(intl_calls_labels, intl_calls_churn)
215  plt.xlabel('Total intl calls')
216  plt.ylabel('Customer loss rate')
217  plt.title('Customer loss rate based on "Total intl calls"')
218  plt.show()
```

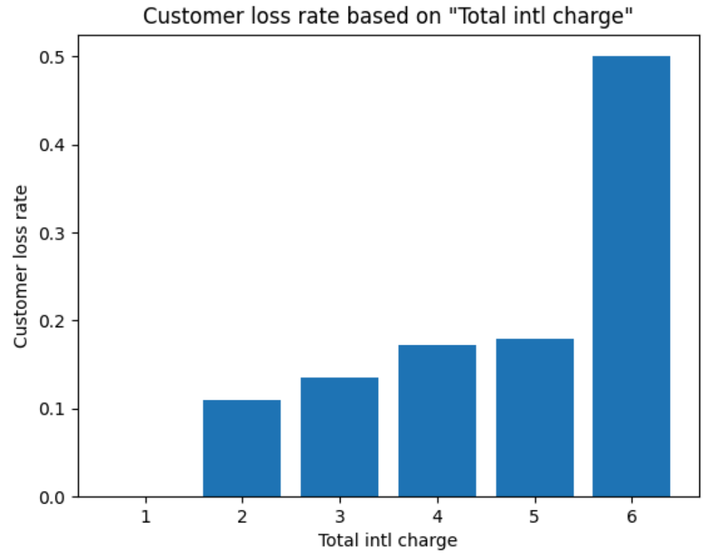


- It can be seen that the customer churn rate increases with the number of international calls. This means that customers who make more international calls are more likely to churn.
- There are a few potential reasons for this. One possibility is that customers who make more international calls are more likely to be dissatisfied with the price or quality of the service. Another possibility is that they are using a different provider that is cheaper or has better features.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate is based on "Total intl charge"

```
# Analyze customer loss rate is based on "Total intl charge"
intl_charge_bins = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
intl_charge_labels = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
intl_charge_churn = []
for i in range(len(intl_charge_bins) - 1):
    mask = (train_data['Total intl charge'] >= intl_charge_bins[i]) & (train_data['Total intl charge'] < intl_charge_bins[i+1])
    intl_charge_churn.append(train_data[mask]['Churn'].mean())
plt.bar(intl_charge_labels, intl_charge_churn)
plt.xlabel('Total intl charge')
plt.ylabel('Customer loss rate')
plt.title('Customer loss rate based on "Total intl charge"')
plt.show()
```

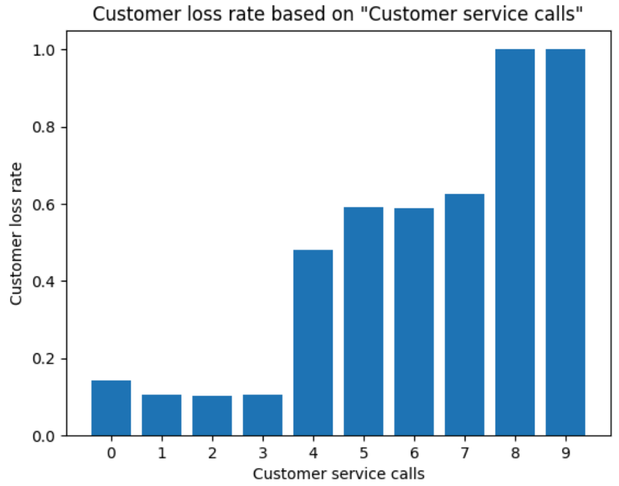


- It can be seen that the customer churn rate increases with the number of international calls. This means that customers who make more international calls are more likely to churn.
- There are a few potential reasons for this. One possibility is that customers who make more international calls are more likely to be dissatisfied with the price or quality of the service. Another possibility is that they are using a different provider that is cheaper or has better features.

EDA-Represents the correlation between the target data column and the remaining columns

Analyze customer loss rate based on "Customer service calls"

```
233  # Analyze customer loss rate based on "Customer service calls"
234  service_calls_bins = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
235  service_calls_labels = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10+']
236  service_calls_churn = []
237
238  for i in range(len(service_calls_bins) - 1):
239      mask = (train_data['Customer service calls'] >= service_calls_bins[i]) & (train_data['Customer service calls'] < service_calls_bins[i+1])
240      churn_rate = train_data[mask]['Churn'].mean()
241      service_calls_churn.append(churn_rate)
242  mask = (train_data['Customer service calls'] >= service_calls_bins[-1])
243  churn_rate = train_data[mask]['Churn'].mean()
244  service_calls_churn.append(churn_rate)
245  plt.bar(service_calls_labels, service_calls_churn)
246  plt.xlabel('Customer service calls')
247  plt.ylabel('Customer loss rate')
248  plt.title('Customer loss rate based on "Customer service calls"')
249  plt.show()
```



- It can be seen that the customer churn rate increases with the number of customer service calls. This means that customers who make more customer service calls are more likely to churn.
- There are a few potential reasons for this. One possibility is that customers who make more customer service calls are more likely to be dissatisfied with the product or service. Another possibility is that they are more likely to be experiencing problems with the product or service.

EDA-Represents the correlation between the target data column and the remaining columns

General conclusions:

- Customer churn is a complex issue with multiple contributing factors.
- This analysis highlights potential areas for further investigation, such as:
  - Investigate reasons behind churn for customers with international calling plans.
  - Analyze customer service interactions to understand the issues leading to churn.
  - Consider segmenting customers based on calling behavior and targeting specific plans or promotions to address their needs.

Recommendations:

- Improve customer satisfaction by addressing the potential causes of churn identified above.
- Regularly monitor customer churn rates and analyze trends to identify emerging issues.
- Consider developing targeted marketing campaigns to retain high-value customers who might be at risk of churning.
- By understanding the relationship between call data and customer churn, telecommunications companies can develop strategies to reduce churn and improve customer loyalty.

Apply the Machine Learning Random Forest model to predict potential customer churn

Convert data columns into numerical format suitable for machine learning models, deploy Random Forest machine learning model, and evaluate the effectiveness of the program.

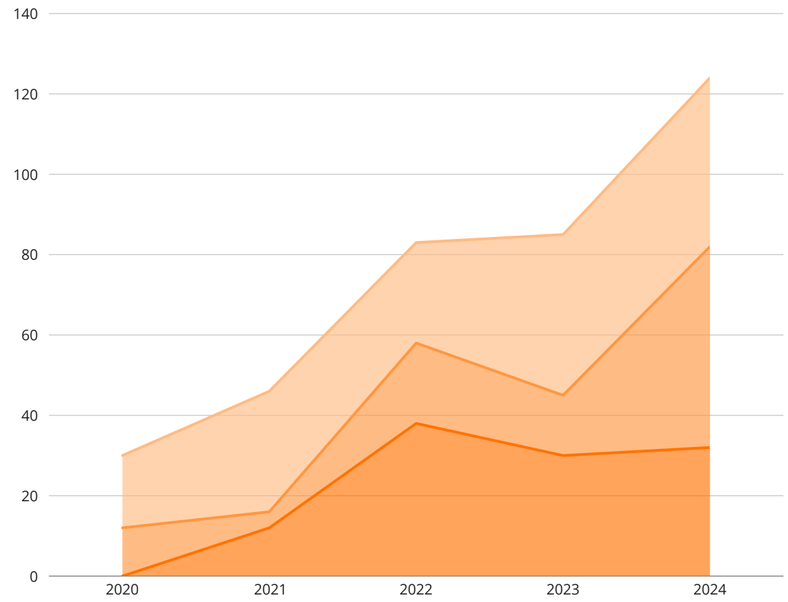
Apply the Machine Learning Decision Tree Classifier model to predict potential customer churn

Encode categorical features

```
252  #Encode categorical features
253  columns_to_encode = [ 'International plan', 'Voice mail plan', 'Churn']
254  encoder = LabelEncoder()
255  for column in columns_to_encode:
256      train_data[column] = encoder.fit_transform(train_data[column])
257  for column in columns_to_encode:
258      test_data[column] = encoder.fit_transform(test_data[column])
```

Split data to training and testing

```
261  # Split data to training
262  X_train = train_data.drop(labels=['State', 'Account length', 'Area code'], axis=1)
263  y_train = train_data['Churn']
264
265  # Split data to testing
266  X_test = test_data.drop(labels=['State', 'Account length', 'Area code'], axis=1)
267  y_test = test_data['Churn']
268  # Fit the model on train data
269  DT = DecisionTreeClassifier().fit(X_train,y_train)
```



Apply the Machine Learning Decision Tree Classifier model to predict potential customer churn

Fit the model on train data

```
268    # Fit the model on train data
269    DT = DecisionTreeClassifier().fit(X_train,y_train)
```

Predict on train and test

```
271    # Predict on train
272    train_preds3 = DT.predict(X_train)
273
274    # Predict on test
275    test_preds3 = DT.predict(X_test)
```

Apply the Machine Learning Decision Tree Classifier model to predict potential customer churn

Accuracy on train and test

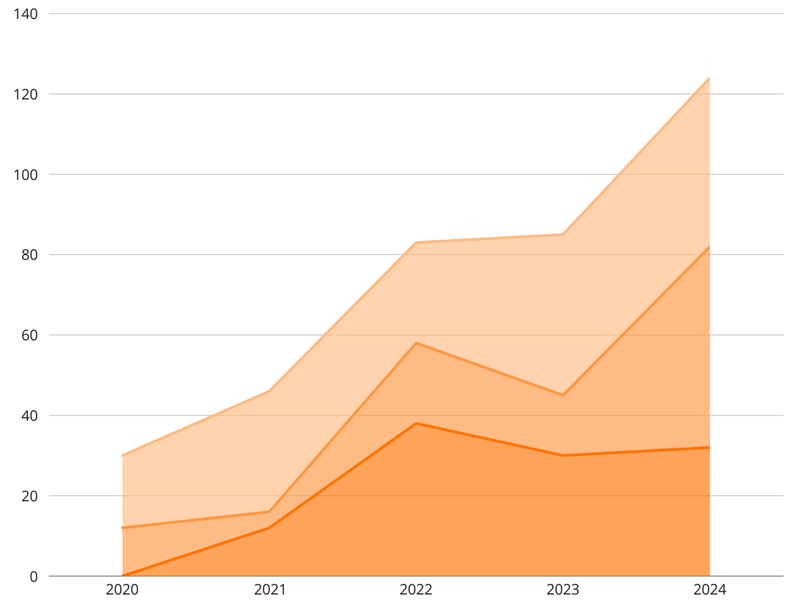
```
276  # Accuracy on train
277  print("Model accuracy on train is: ", accuracy_score(y_train, train_preds3))
278
279  # Accuracy on test
280  print("Model accuracy on test is: ", accuracy_score(y_test, test_preds3))
```

ROC

```
283  # ROC
284  print("ROC score on train is: ", roc_auc_score(y_train, train_preds3))
285  print("ROC score on test is: ", roc_auc_score(y_test, test_preds3))
```

Confusion matrix

```
288  # Confusion matrix
289  print("confusion_matrix train is: ", confusion_matrix(y_train, train_preds3))
290  print("confusion_matrix test is: ", confusion_matrix(y_test, test_preds3))
```



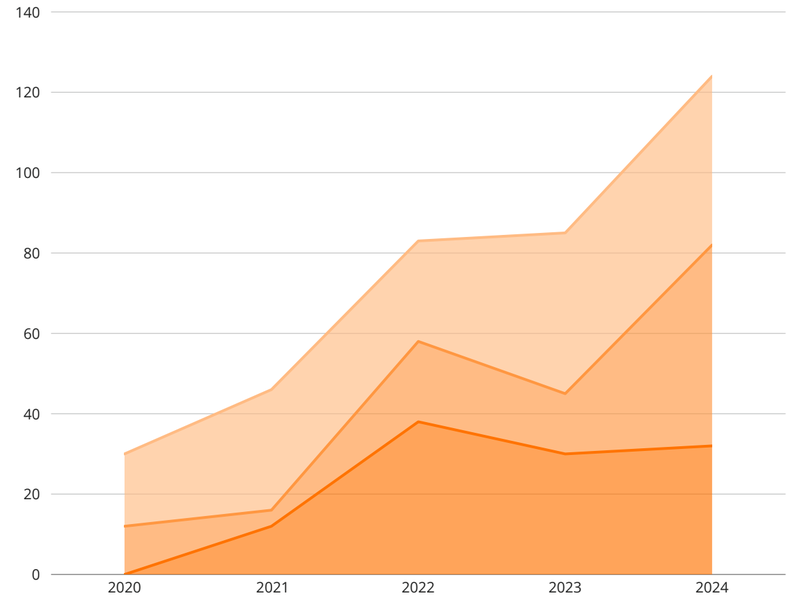
Apply the Machine Learning Decision Tree Classifier model to predict potential customer churn

Wrong Predictions made

```
294   # Wrong Predictions made.
295   print((y_test !=test_preds3).sum(),'/',((y_test == test_preds3).sum()+(y_test != test_preds3).sum()))
```

Kappa Score

```
298   # Kappa Score
299   print('KappaScore is: ', metrics.cohen_kappa_score(y_test,test_preds3))
```



Apply the Machine Learning Decision Tree Classifier model to predict potential customer churn

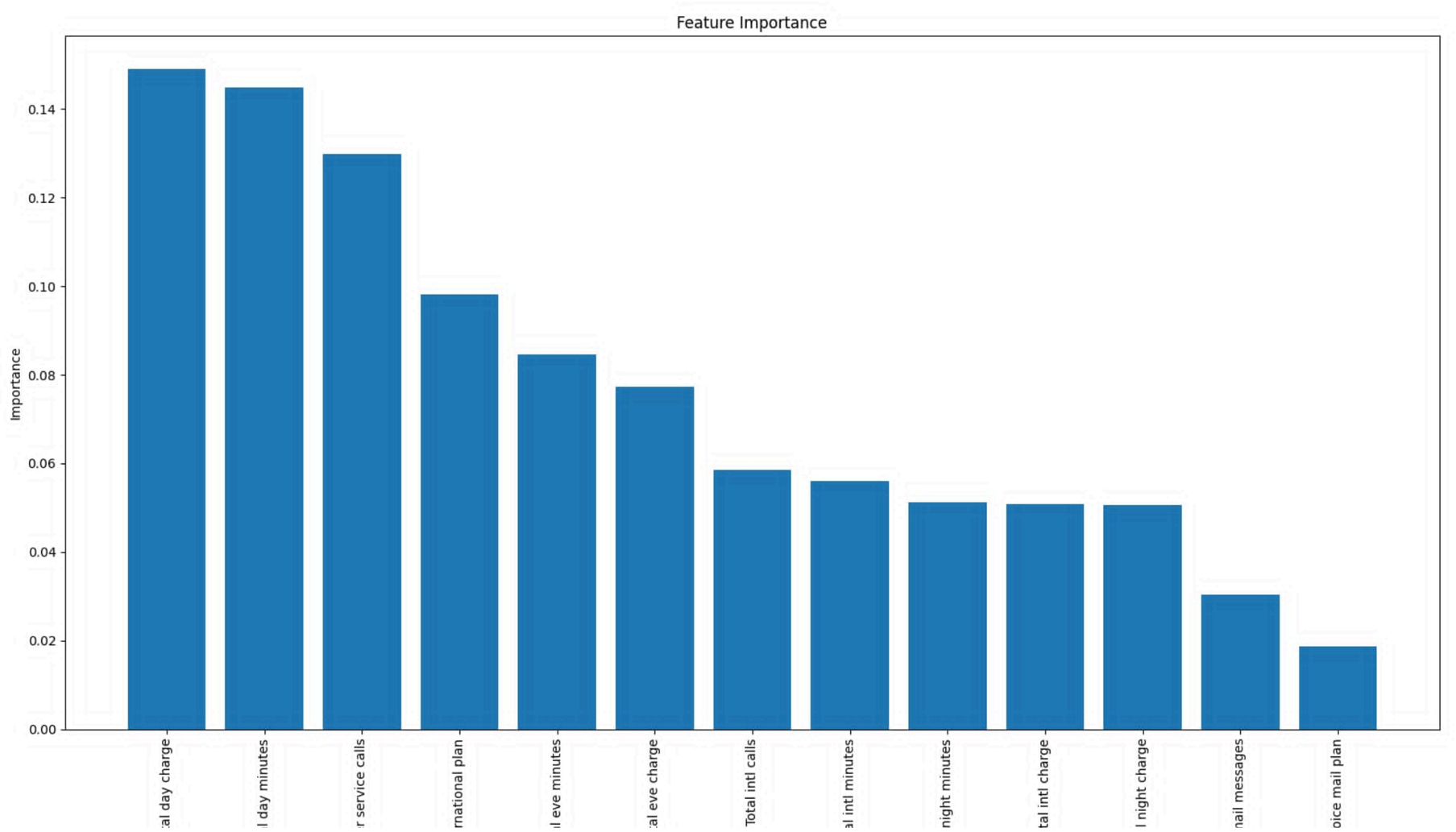
Result

```
Model accuracy on train is:  0.9996249062265566
Model accuracy on test is:  0.9565217391304348
ROC score on train is:  0.9987113402061856
ROC score on test is:  0.8605355171144646
----------------------------------------------------
Confusion_matrix train is:  [[2278    0]
 [   1  387]]
Confusion_matrix test is:  [[569   3]
 [ 26  69]]
Wrong predictions out of total
----------------------------------------------------
29 / 667
----------------------------------------------------
Kappa Score is:  0.8020346129834508
```

Review and Evaluation

- **Accuracy:** The model has a perfect accuracy of 99.96% on the training set and 95.65% on the test set.
- **ROC Score:** The ROC score is 0.99 on the training set and 0.8605 on the test set. This indicates a good performance in distinguishing between the classes.
- **Confusion Matrix:** The confusion matrix on the training set shows that there were 2278 true negatives and 387 true positives. On the test set, there were 544 true negatives, 69 true positives.
- **Total Misclassifications:** There were 29 misclassifications out of 667 instances in the test set.
- **Kappa Score:** Kappa score is 0.8020, which indicates that the level of agreement between model expectations and actual values is very good, much higher than the consensus rate by chance.

Feature Importance



Conclusion

Businesses should focus on items related to the cost of services by:

+ Offer coupons or incentives for customers with large usage volumes.
+ Suggest ways to reduce service costs to suit consumers
+ Paying attention to services to avoid unwanted incidents for customers

THANK YOU FOR YOUR INTEREST IN AND REVIEW OF MY PRESENTATION. YOUR FEEDBACK AND SUGGESTIONS WILL BE AN ENDLESS SOURCE OF MOTIVATION FOR ME TO DRIVE FORWARD AND DEVELOP MYSELF.

I SINCERELY APPRECIATE YOUR TIME AND SUPPORT. THANK YOU ONCE AGAIN.

Link code : https://github.com/leki3n/Customer-churn-prediction-.git