Predicting Customer Churn: Empowering Syriatel to Optimize Retention and Loyalty Utilizing Data-Driven Insights

Syriatel has been one of Syria's leading telecommunications companies since its establishment in 2000, primarily offering mobile network services. The company places a strong emphasis on customer satisfaction and social responsibility. Syriatel provides a wide range of telecom services, including mobile voice, data services, mobile internet, and value-added services.

Syriatel has established a robust network infrastructure with 24 points of service that cover the entire Syrian territory. The company operates four call centers in key cities—Damascus, Aleppo, Lattakia, and Tartous—serving over 23,000 customer queries daily. Additionally, Syriatel maintains 199 international roaming partnerships across 116 countries, ensuring global connectivity for its users.

With a focus on providing the best mobile communications experience, Syriatel aims to empower its customers, enhance employee satisfaction, and achieve sustained value creation for its shareholders. Despite challenges posed by the ongoing conflict, the company continues to play a vital role in Syria’s telecommunications landscape.

3. Objectives

a. Churn Prediction Insights:

- Provide notable insights from prediction the model to help SyriaTel identify customer groups that have the highest likelihood of churning.
- Recognize critical factors contributing to churn to inform retention strategies.

b. Model Evaluation and Comparison:

- Assess the models using performance metrics such as accuracy, precision, recall, F1-score, ROC AUC, and confusion matrix.
- Compare the models to determine which one provides the most reliable churn predictions.

c. Recommendations for Customer Retention:

- Using insights from the model, recommend strategies to minimize churns, such as targeted marketing, exclusive offers, or enhanced customer support.
- Propose a strategy for ongoing churn prediction and retention efforts to help SyriaTel continuously improve customer loyalty.

4. Metric For Success

The success will be determined by the model's accuracy in predicting churn, its impact on customer retention, and its influence on Syriatel's revenue and customer satisfaction.

a. Model Accuracy Metrics

Accuracy: The percentage of times the model correctly predicts whether a customer is likely to churn or not.

Precision: The proportion of customers predicted to churn who do churn.

Recall: How many of the actual churners were correctly identified by the model.

F1-Score: A combined measure of precision and recall, ensuring both are balanced and perform well.

AUC-ROC: The model's ability to differentiate between customers who will churn and those who won't.

5. Data Understanding

The data I chose to use for this analysis is extracted from the following Kaggle.com link : https://www.kaggle.com/datasets/becksddf/churn-in-telecoms-dataset . It contains information about customers of a telecom company, Syriatel, and their usage of various services. The objective is to predict customer churn Each row in the dataset corresponds to an individual customer, while the columns provide specific details about their behavior, preferences, and account information. I imported the libraries first then loaded the data in preparation for review, encoding and analysis ended up checking the shape of the data and I could see it has 3333 rows and 21 columns.

column meanings:

1. state: The state where the customer is located (e.g., KS, OH, NJ).
2. Account length: The number of months the customer has been with the telecom company.
3. Area code: The area code of the customer's phone number (e.g., 415, 408, 510).
4. Phone number: The customer's phone number.
5. International plan: Whether the customer subscribes to an international calling plan (Yes/No).
6. Voice mail plan: Whether the customer subscribes to a voicemail plan (Yes/No).
7. Number vmail messages: The number of voicemail messages the customer has received.
8. Total day minutes: The total number of minutes the customer has spent on calls during the day.
9. Total day calls: The total number of calls made by the customer during the day.
10. Total day charge: The total charge for the calls made by the customer during the day.

11. Total eve minutes: The total number of minutes the customer has spent on calls during the evening.
12. Total eve calls: The total number of calls made by the customer during the evening.
13. Total eve charge: The total charge for the calls made by the customer during the evening.
14. Total night minutes: The total number of minutes the customer has spent on calls during the night.
15. Total night calls: The total number of calls made by the customer during the night.
16. Total night charge: The total charge for the calls made by the customer during the night.
17. Total intl minutes: The total number of minutes the customer has spent on international calls.
18. Total intl calls: The total number of international calls made by the customer.
19. Total intl charge: The total charge for the international calls made by the customer.
20. Customer service calls: The number of times the customer has called customer service.
21. Churn: The target variable indicating whether the customer has churned (1 = Churned, 0 = Not Churned).

Learning about the data is important because it helps decide which parts of it to use and how to prepare it for making predictions. After going through the dataset I went straight ahead to its preparation since I had identified some main columns and rows that will make it easy for me to achieve my goal.

DATA PREPARATION & ANALYSIS

Before creating a prediction model with the data, it's essential to prepare it properly. This involves several steps to ensure that the data is organized, clear, and suitable for analysis. Here's a straightforward breakdown of what I did:

1. **Renaming multiple columns in a list for clarity**: I renamed columns like 'number vmail messages' to 'number voice mail messages', 'total eve minutes' to 'total evening minutes', 'total eve calls' to 'total evening calls' etc. This was to make it easier for someone to understand the columns better than before with the shortened forms.
2. **Checked for Missing Values:** There was No missing values on this dataset.
3. **Checked for duplicates**: I checked for duplicates and there were none.
4. **Dropped the columns, we don't need:** I dropped 'phone number ' and 'area code' since phone number is unique to each person so we cannot use the information for prediction. As 'state' is more detailed we don't need the 'area code' details as well.
5. **Added Adding new columns**: This was for trend analysis in the original data. I created a new feature in the original data for Total charges (sum of day, evening, and night

charges), Total calls (sum of day, evening, and night calls), Total minutes (sum of day, evening, and night minutes) and 'monthly charge' by summing up the relevant charge columns

6.Converted Categorical Data: checked the unique values in the following categorical predictors 'international plan' and "voice mail plan," which contain categorical values (Yes/No). It needed to be converted into a numerical format so that the machine learning algorithms could process them. For example, we can map "Yes" to 1 and "No" to 0.

7. Created a new copy of data df and performed encoding: In this scenario, I added encoding to the new copy so that we have a data set with added columns plus encoding' I Initialized the OneHotEncoder with drop='first' to create redundant columns, and the result was stored in a new DataFrame. I created a new DataFrame with only the categorical columns ('area code', 'international plan', and 'voice mail plan') and then converted the categories into numerical values.

8. Created a final new data set: I dropped 'area code', 'international plan' and 'voice mail plan', dropped Drop the 'total day minutes', 'total day calls', 'total day charge', 'total evening minutes', 'total evening calls', 'total evening charge', 'total night minutes', 'total night calls', 'total night charge', 'total international minutes', 'total international calls', 'total international charge' columns since they were irrelevant for the analysis, This is because I had created new columns with its sums.

9. Final data: Checked the final data for modeling set with new columns and encoding applied. After checking its new shape, it now contained 3333 rows and 12 columns which makes my work easier and definitely does not affect the results.

By completing these steps, I had a prepared dataset that was ready for modeling. These preparations are essential to ensure that the model can learn effectively from the data and provide accurate predictions of customer churn.

6.Exploratory Data Analysis

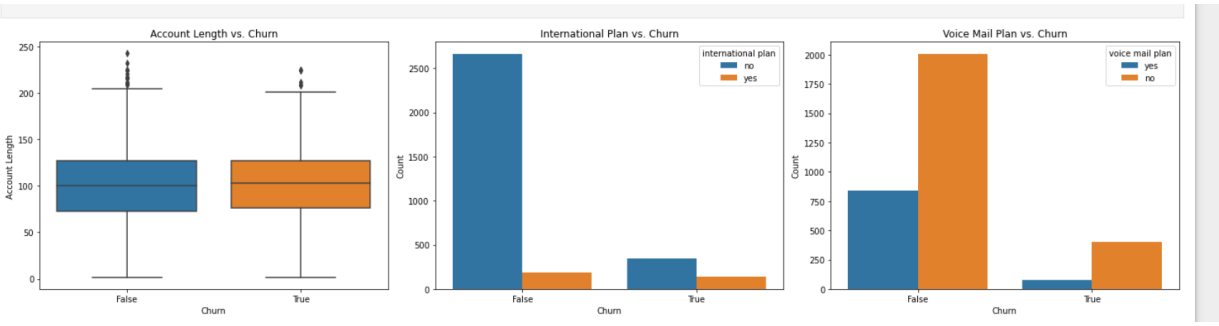
Exploring patterns in 'data_df': Here, we examine the relationship between churn and various continuous variables. The variables include:

- account length
- international plan
- voice mail plan
- number voice mail messages
- customer service calls

- Region
- total minutes
- total calls
- total charges
- monthly_charge

Step1

I created multiple plots to explore how different variables relate to customer churn. Each plot compares a specific variable with the churn status, helping us identify patterns or trends.



Plot 1: Account Length vs. Churn:

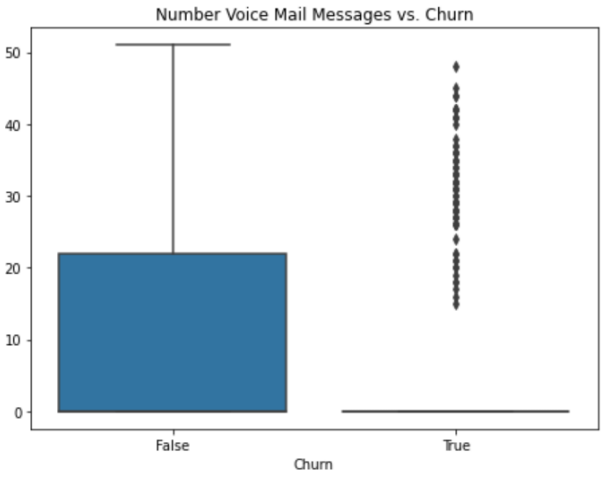
A box plot shows the distribution of "account length" for customers who churned and those who didn't. The x-axis is "churn" (0 for no churn, 1 for churn). The y-axis is "account length" (how long a customer has been with the company). This helps us see if account length affects churn.

Plot 2: International Plan vs. Churn:

A count plot shows how many customers with and without an international plan churned or stayed. The hue (color difference) represents whether the customer has an international plan or not. This plot helps us see if the international plan impacts churn.

Plot 3: Voice Mail Plan vs. Churn:

This is Similar to Plot 2 but It shows how having or not having a voice mail plan relates to churn

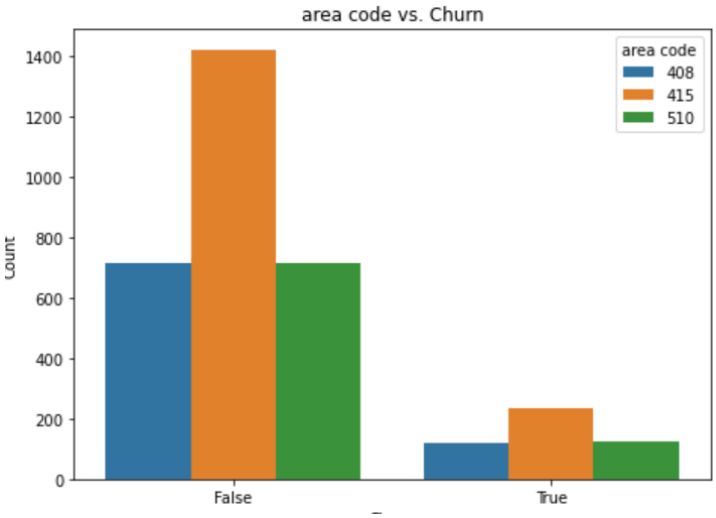


Plot 4: Number of Voice Mail Messages vs. Churn:

A box plot to compare how the number of voice mail messages differs between customers who churned and those who didn't.

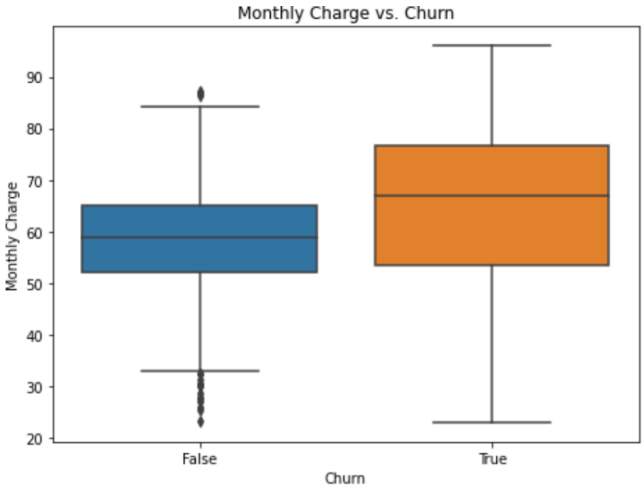
Plot 5: Customer Service Calls vs. Churn:

Another box plot showing the distribution of customer service calls for churned and non-churned customers. It helps us see if more customer service calls are linked to churn.



Plot 6: Area Code vs. Churn:

A count plot shows the churn distribution across different area codes. This can help identify if customers in certain regions are more likely to churn.

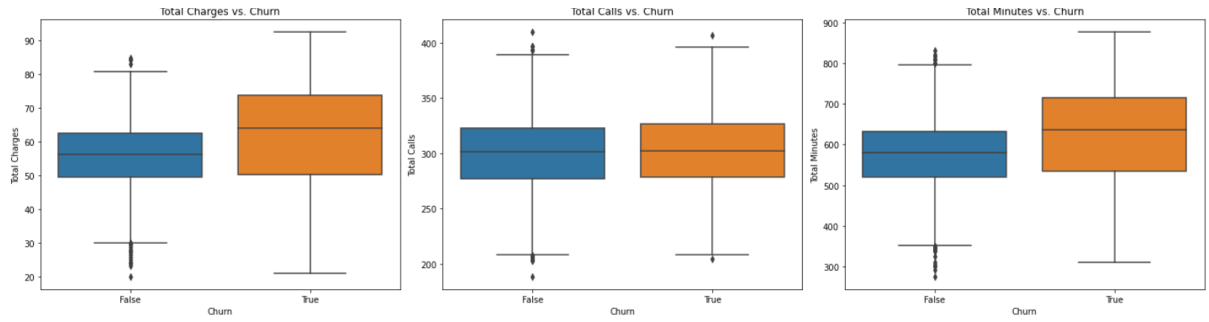


Plot 7: Monthly Charge vs. Churn:

A box plot compares the monthly charges of churned vs. non-churned customers. It reveals if higher or lower charges are related to churn

Step 2.

I also created three box plots to show how the variables Total Charges, Total Calls, and Total Minutes are related to whether a customer churned or not. A box plot is great for comparing distributions and spotting differences between groups.



Plot 1: Total Charges vs. Churn:

This box plot shows the distribution of Total Charges for churned and non-churned customers. The **x-axis** represents Churn (0 means no churn, 1 means churn). The **y-axis** represents Total Charges(the total amount a customer has been charged). It helps to see if customers who churned had higher or lower charges compared to those who didn't.

Plot 2: Total Calls vs. Churn:

This box plot compares the number of Total Call between churned and non-churned customers. The x-axis is again Churn and the y-axis is Total Cells. This plot helps identify if customers who made more or fewer calls are more likely to churn.

Plot 3: Total Minutes vs. Churn:

This box plot looks at Total Minutes (the total minutes a customer spent on calls). It compares how the minutes differ between customers who churned and those who stayed.

Step 3.

Next, I used the data set --'final2_data_df'

- As the 'data_df' dataset was used for checking the relationship that is there between the predictor variables and target variables.
- The results seen in the above visuals to check skewness, remove outliers and check correlation on the final2_data_df

As the above box plots have shown several outliers in the box plots we need to remove them. I used a function to remove outliers using IQR method.

I did not remove the outliers because they comprise over 30% of the data, meaning a significant portion would be lost.

- Effect of Removing Outliers:

When outliers are removed, the size of the dataset is significantly reduced—from 3,333 rows to just 653 rows. This can be problematic, as it may lead to the loss of valuable information, particularly if the outliers represent genuine data points reflecting real-world variations or edge cases.

- Impact on Model Performance:

Outliers can disproportionately influence the performance of certain models (e.g., linear regression, decision trees). Consequently, removing them might improve model performance. However, if the outliers are important and indicative of rare but significant patterns, their removal could diminish the predictive power of the model.

- Overfitting Risk:

Eliminating outliers may result in a model that performs well on the remaining data but fails to generalize to new or unseen data, especially if outliers occur in real-world scenarios.

Next, I checked how skewed the distribution of certain numerical columns is in the dataset. First I Imported the Skew Function**,** listed numerical columns to check for skewness then looped through each column and finally printed. What I found was;

- Nearly symmetric variables (account length, monthly charge):

  These variables don't need any transformations, as they have very small skewness values close to 0. They are already in a form suitable for modeling.

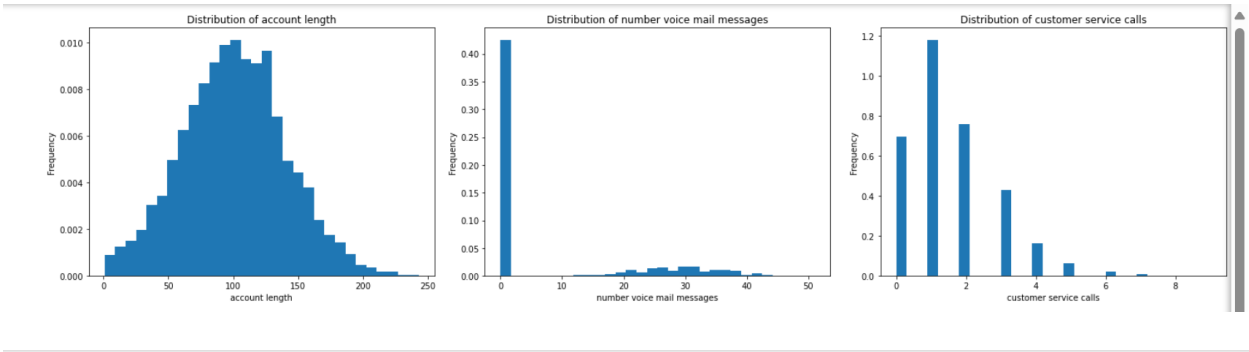
- Right-skewed variables (number voice mail messages, customer service calls):

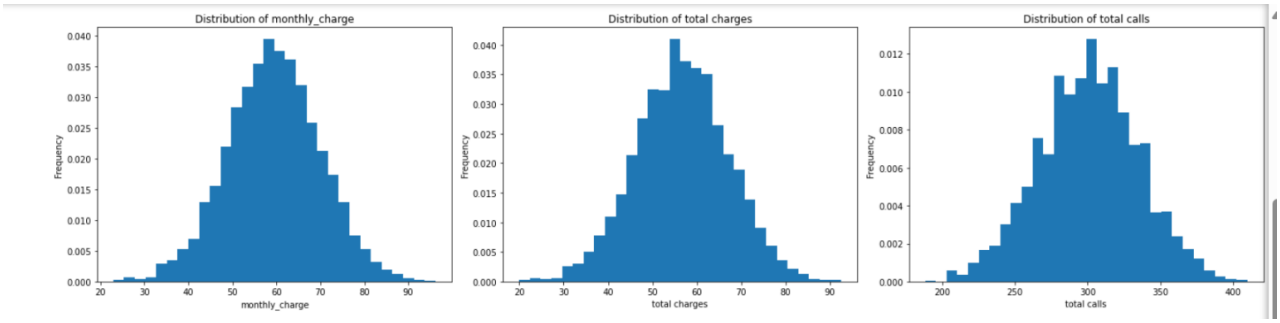
  These variables have moderate to high positive skewness, particularly number voice mail messages. You should consider applying transformations (like the log transformation) to reduce the skewness, especially if using algorithms that assume normality or are sensitive to extreme outliers (e.g., linear regression, logistic regression).

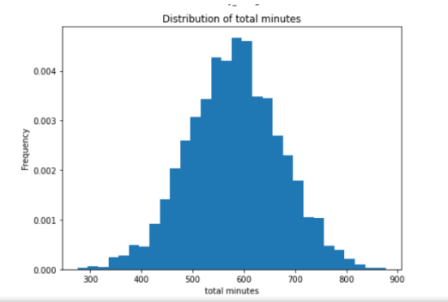
- For models like decision trees, random forests, or gradient boosting (which are less sensitive to skewed distributions), transformations may not be necessary.

Step 4.

Next step I created multiple histograms to visualize the distribution of several numerical variables. It helps to see the spread of the data and if it's skewed, normally distributed, or has any other patterns.

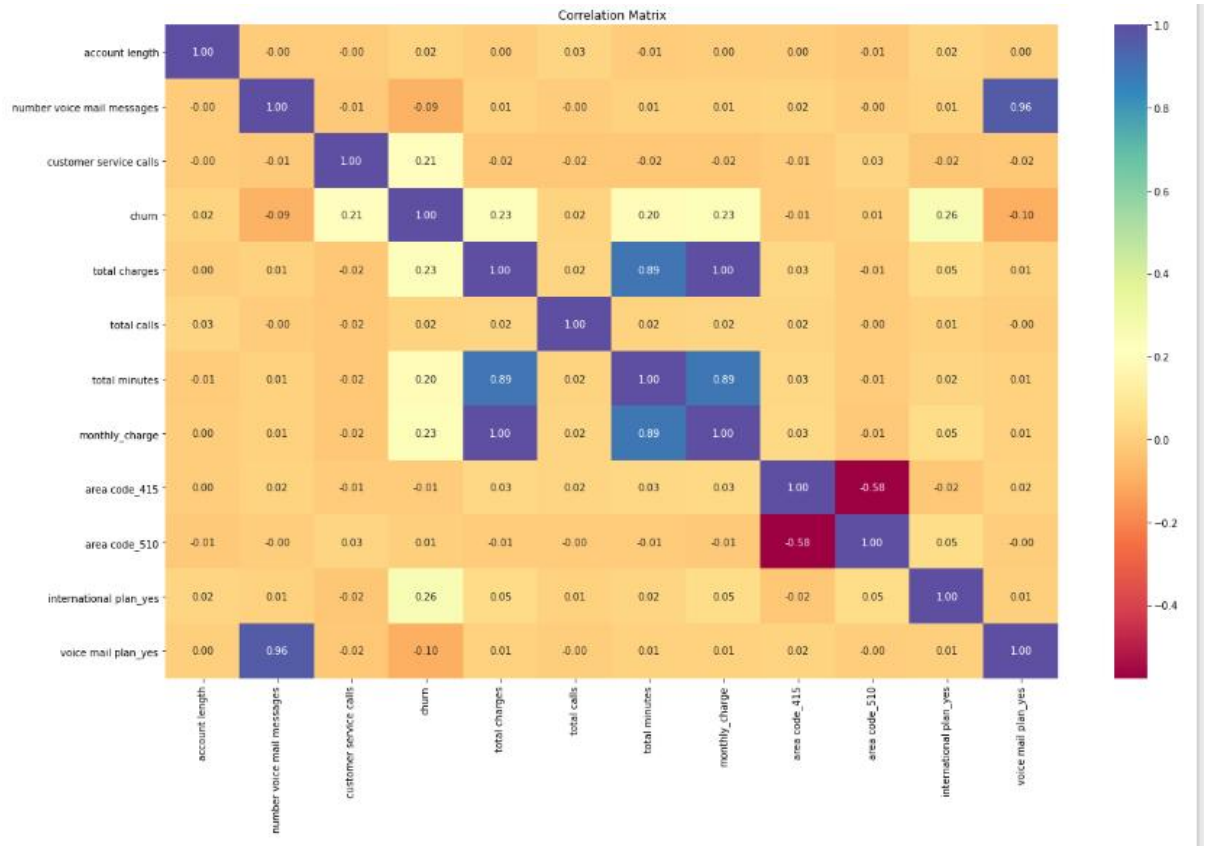






Step 5.

Created a heatmap to check for Correlation.



From the results above we see that columns such as total charge and minutes charge are highly correlated. Similarly voice mail plan_yes and number voice mail messages are highly correlated

7. Modelling

In this stage we will do the following:

- Check for multicollinearity using the VIF approach for feature selection purposes
- Apply log transformation where necessary
- Apply scaling to standardize the error for modeling
- check for and fix class imbalance
- Logistic regression
- Random forest

Step 1; checked for multicollinearity in the dataset and used a method called VIF to detect multicollinearity.

VIF Interpretation: Constant (const):

VIF = 140.14: The constant term (intercept) has a very high VIF. This is expected, as the constant term is perfectly correlated with itself. This value doesn't indicate a problem, since we're usually not concerned with the VIF for the constant. account length:

VIF = 1.00: A VIF of 1 means there is no correlation between account length and the other predictor variables. It is a perfectly independent variable in terms of collinearity. number voice mail messages:

VIF = 1.00: Similar to account length, number voice mail messages has a very low VIF, indicating that it is not highly correlated with the other predictors in the model. It is also considered independent. customer service calls:

VIF = 1.00: Again, this variable has a very low VIF, indicating that it does not suffer from multicollinearity with other features. monthly charge:

VIF = 194.27: This is a very high VIF, indicating that monthly_charge is highly correlated with one or more other predictors. High VIF values like this suggest that monthly_charge could be redundant and may need to be removed or combined with other features to reduce multicollinearity. This might be a signal that monthly_charge is too similar to total charges (as they are related in some way). total charges:

VIF = 198.86: This also has a very high VIF, suggesting high multicollinearity with other variables, possibly with monthly_charge (since both are related to the amount the customer is paying). A high VIF here means the effect of total charges on the target variable (churn) might be difficult to isolate due to this correlation. total calls:

VIF = 1.00: Similar to the other features with VIF = 1, total calls is not highly correlated with the other variables, indicating that it is a relatively independent predictor. total minutes:

VIF = 4.88: This is a moderate VIF, indicating some correlation with other predictors, but not severe. This may still be acceptable depending on the context of the model and the threshold for

multicollinearity. Key Takeaways: High VIFs (monthly_charge and total charges): These two variables have very high VIF values (194.27 and 198.86), suggesting that they are highly correlated with one or more other variables. In practice, you might consider removing one of these variables or combining them (e.g., through feature engineering or transformation) to reduce multicollinearity.

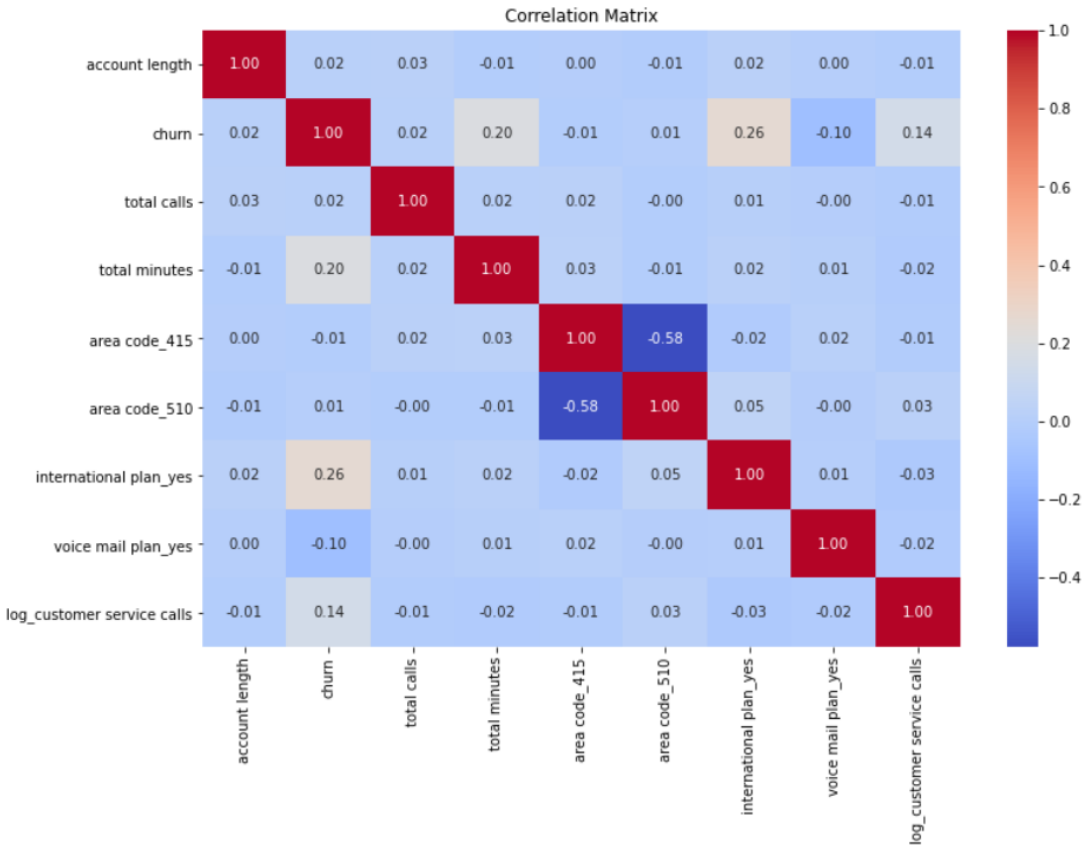
Low VIFs (1.00): Most of your other features have VIF values around 1, indicating no significant multicollinearity. These features are good candidates to keep in the model without significant risk of redundancy.

Step 2; Log Transformation.

This were the columns after. 'account length', 'number voice mail messages', 'customer service c alls', 'churn', 'total charges', 'total calls', 'total minutes', 'monthly_charge', 'area code_415', 'area co de_510', 'international plan_yes', 'voice mail plan_yes 'log_number voice mail messages', 'log_c ustomer service calls'], dtype='object'

- 'As one can see the column Voice mail messages seems to be the least impacted by the lo g transformation
  this same data also shows high multicolinearity with the customer service calls. Addition ally, recall that monthly charges and total charges seem to share a high correlation, and b oth also have a high IVF. Due to this, we will thus have to drop one of these columns

Step 3; Correlation Heatmap



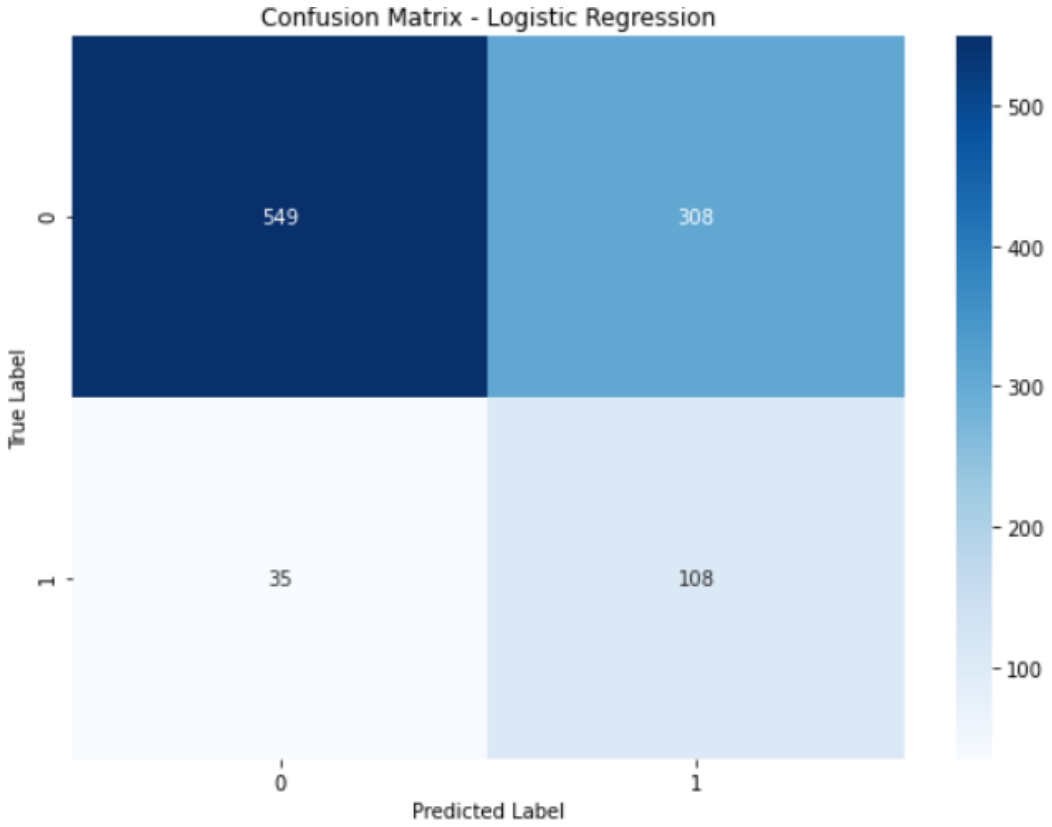
Perform Logistic Regression and Random Forest

This is to help predict whether a customer will churn or not. I Balanced the data using SMOTE to avoid biases and standardized the features to make them comparable. Trained two models (Logistic Regression and Random Forest) to predict churn. Evaluated both models and checked how well they perfor. Finally, I Identified which features are the most important in predicting churn using Random Forest.

Step 4;

Preparing the data by splitting it into training and testing sets. Balancing the data using SMOTE to fix class imbalance, standardizing the features to make the model learn better. Evaluating the best model with metrics like accuracy, ROC AUC, and classification report and finally Visualizing the model's performance with a confusion matrix.

This helps me find the best model to predict whether customers will churn and evaluate how well it performs.



The model is good at identifying no-churn customers (high precision), but it struggles to correctly predict churned customers (low precision, although good recall).

The overall performance is acceptable, but the model could be improved, especially in how it identifies customers who will churn.

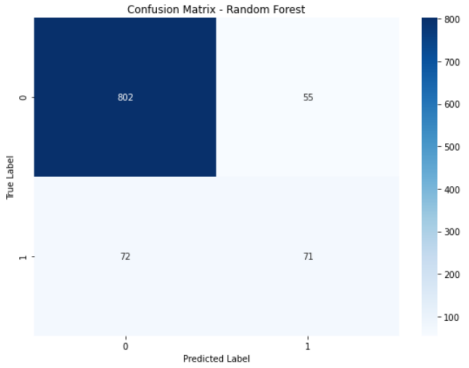
Conclusion:

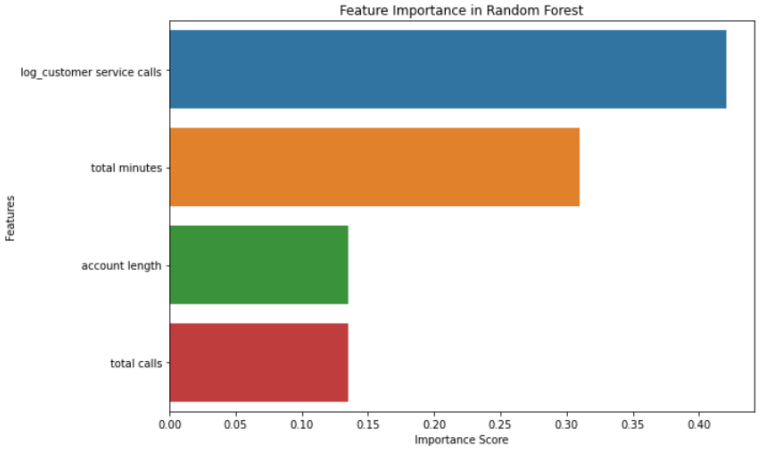
Accuracy (65.7%) is decent, but it's important to consider other metrics like precision, recall, and F1-score. In this case, the model is not great at predicting churn, especially because the precision for churn is low.

ROC AUC is decent (0.736), suggesting the model is still better than random guessing.

Step 5; Random Forest Hyperparameter Tuning;

Testing the model with different configurations, and evaluating how well it predicts customer churn. It helps build a more accurate model by optimizing hyperparameters, evaluating its performance, and understanding its behavior.





The best cross-validation score of 0.97 means the model performed well during testing with different subsets of the data. The model achieved an accuracy of 87.3%**,** which means it correctly predicted whether the customer churned in 87.3% of cases. The ROC AUC score of 0.80 indicates that the model is good at distinguishing between customers who churn and those who don't. The classification report shows that the model is better at predicting customers who don't churn (label 0) than customers who churn (label 1), with a higher precision and recall for non-churned customers.

Finally, the feature importance section shows which features the model found most important in predicting churn. For example, "log_customer service calls" had the highest importance, meaning that how many times a customer called customer service was a key factor in predicting churn. Other important features include "total minutes" and "account length."

Finally, I performed **cross-validation** to evaluate how well the models (Logistic Regression and Random Forest) perform on different subsets of the training data. I used **ROC AUC** as the

scoring metric, which helps measure how well the model distinguishes between classes in this case, customers who churn and those who don't.

Conclusion

In conclusion, the Random Forest model outperformed Logistic Regression in predicting customer churn. The accuracy of the Random Forest on the test data was 87.3%, while Logistic Regression achieved only 65.7%. This highlights the superior predictive capability of Random Forest.

Moreover, the Precision, Recall, and F1 scores were generally higher for Random Forest, particularly when it came to predicting customers who churn. Additionally, Random Forest excelled at distinguishing between customers who churn and those who don't, as indicated by its higher ROC AUC score of 0.799, compared to the 0.736 score of Logistic Regression.

Within the Random Forest model, the most significant feature for predicting churn was the number of customer service calls, which accounted for 42% importance. This was followed by total minutes, which contributed 31%. Other features, such as account length and total calls, were deemed less important.

Lastly, Random Forest demonstrated a stronger ability to generalize and perform well on new, unseen data, achieving a mean cross-validation score of 97.06%. In contrast, Logistic Regression had a mean cross-validation score of only 68.32%.

Recommendation

- Random Forest is the most effective model for predicting churn, as it performs well with our data. To enhance customer retention, we should focus on key factors such as customer service calls and total minutes used. Reducing the number of service calls could contribute to lowering churn rates.
- Although we utilized SMOTE to balance the data, we can further improve churn predictions by adjusting class weights or exploring different models.
- For straightforward insights, we can use Logistic Regression. Although it is less accurate, its simplicity makes it easier to interpret, making it suitable for quick and easy results

Next steps

- **Test other algorithms**: Consider testing various machine learning algorithms, such as Gradient Boosting, to determine if they outperform the Random Forest model.
- **Create New Features:** Develop new features or modify existing ones to extract more information from the data. For example: We can Combine two or more features to create

interaction terms that capture the relationships between existing variables. If possible, also incorporate additional customer information, such as demographics e.g. Age to enhance the model.

- **Improve Model Settings:** Adjust the settings of different models through hyperparameter tuning to identify the optimal version of each model. Additionally, fine-tune the settings of the Random Forest model to achieve better results.
- **Implement the Best Model:** Once we have identified the best-performing model, integrate it into a real business application to automatically predict customer churn.
- **Continuously Enhance the Model:** Collect new data over time and regularly retrain the model to ensure its accuracy remains high as customer behavior evolves.