jgatharia /
Phase_3_Project

Code Issues Pull requests Actions Projects Wiki Security Insights Settin

0 stars 0 forks 1 watching 1 Branch 0 Tags Activity

Public repository

1 Branch 0 Tags Go to file t Go to file + Add file Code ···

jgatharia Final ipynb file run all command 1292559 · 3 minutes ago

| | | |
|---|---|---|
| Images | SyriaTel Logo | 2 days ago |
| .gitignore | Files Created | last week |
| README.md | Final review done | 2 days ago |
| Student.ipynb | Final ipynb file run all command | 3 minutes ago |
| SyriaTel Customer Churn Pre... | Presentation Done | 12 minutes ago |

README

# Phase 3 Project

## SyriaTel Customer Churn Prediction

### Project Overview

## 1. Business Problem

Customer retention is at the heart of every thriving Telecom company. Managing and reducing customer churn is essential for maintaining revenue, profitability, and market share. By focusing on churn reduction, telecom companies can enhance customer satisfaction, increase the lifetime value of their customers and secure a stronger position in the competitive market. SyriaTelecommunication is well aware of the common marketplace comment that "**it is cheaper to retain a converted customer than acquire a new client**". As a result, I have been tasked to build a classification model that will predict whether a customer will soon stop doing business with them.

The research at hand delves into machine learning algorithms and offers recommendations tailored to the telecommunications industry. In a competitive telecom sector where customers can effortlessly switch from one provider to another, telecom companies are understandably concerned about customer retention and devising strategies to retain their clientele. By preemptively identifying customers likely to switch providers through behavioral analysis, they can devise targeted offers and services based on historical records.

The core objective of this study is to predict churn in advance and pinpoint the primary factors that may influence customers to migrate to other telecom providers. The machine learning algorithms such as logistic regression and decision trees will be used to develop a robust churn prediction model. Model performance will be evaluated using metrics such as accuracy, precision, recall, and AUC-ROC to ensure the best possible outcomes. This will provide insight to the board members when making policies and procedures that will enable the business gear towards retaining the customers and continue being relevant in the marketplace.

See below questions the project aims to answer:

1. What is the churn current % rate.
2. What features/attributes do the customers who churn have.
3. What strategies can SyriaTel implement to increase customer retention.

## Business Objectives

Develop a predictive model that accurately identifies customers who are at risk of churning (leaving the service) and achieving an overall model accuracy of at least 85%, while maintaining a recall rate of at least 70% for the churn class.

## Data Mining Objective

Build a classification model that predicts whether a customer will churn.

## 2. Data Understanding

This project utilizes the SyriaTel dataset, which was downloaded from Kaggle. The dataset contains 3,333 records (rows) and 21 features (columns). The data is stored in the file named SyriaTel_Customer_Churn.csv. We also observed several findings from our EDA analysis that we will further discuss.

See below columns and what they represent:

- State: The geographical location of the customer.

- Account Length: How long the customer held their account.
- Area Code: Customer's phone number area code.
- Phone Number: Customer's mobile number.
- International Plan: A indicator of whether the customer has an international plan or not.
- Voice Mail Plan: An indicator whether the customer has a voice mail plan.
- Number Vmail Messages: How many voicemail messages the customer has.
- Total Day Minutes: Total minutes the customers spend on a call in the day.
- Total Day Calls: Total number of calls the customer made in a day.
- Total Day Charge: Total charge incrued for the day calls.
- Total Eve Minutes: Total minutes the customers spend on a call in the evening.
- Total Eve Calls: Total number of calls the customer made in a evening.
- Total Eve Charge: Total charge incrued for the evening calls
- Total Night Minutes: Total minutes the customers spend on a call in the night.
- Total Night Calls: Total number of calls the customer made in a night.
- Total Night Charge: Total charge incrued for the day night.
- Total Intl Minutes: Total minutes spent on an international call.
- Total Intl Calls: Total international calls made.
- Total Intl Charge: Total charge incured on the international plan.
- Customer Service Calls: How many calls the customer made for support to SyriaTel.
- Churn: Target variable indicating whether the customer has churned (True) or not churn (False) respectively.

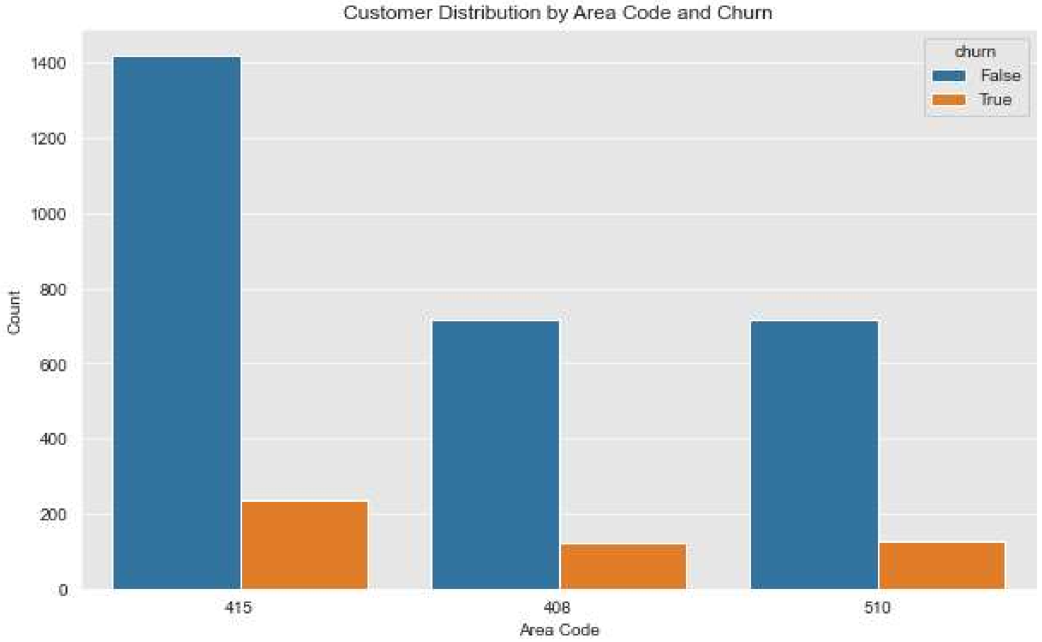
All the other features are potential contributing factors to churn which our project will focus on to eventually tell which features are more significant than the others.

## EDA Data Analysis Findings:

In this project we installed the python, pandas, numpy and scikit learn libraries.

#### Finding 1: Data Geographical Distribution

We identified that the data we have was collected from 3 geographical areas. Area code 415', '408' and '510'. The area code with the highest churn number is area code 415 followed by 510 and lastly 408. In terms of distribution most customers are in the area code 415 as well. See the visualization:

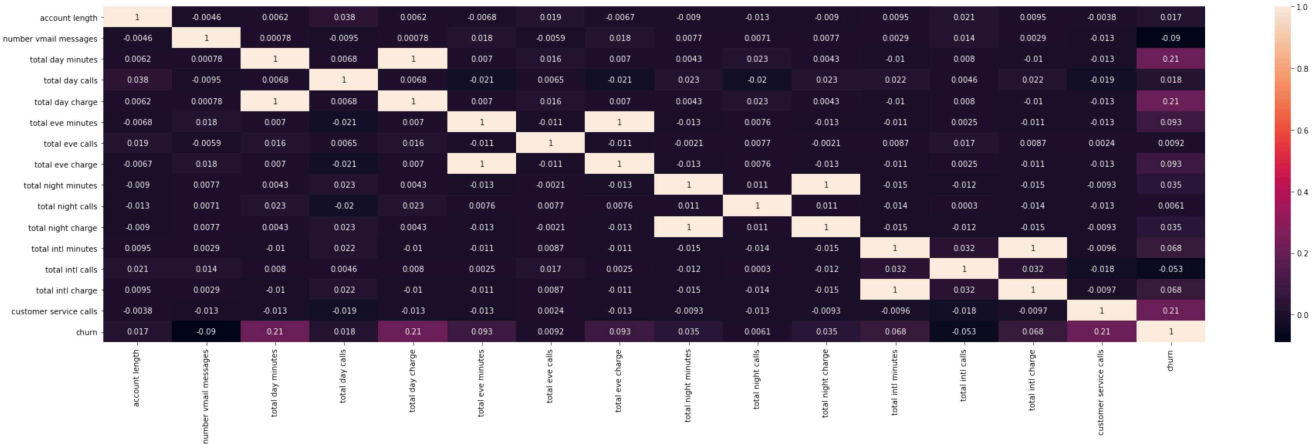


### Finding 2: Data Type Conversion

We identified that the 'area code' column, originally an integer, represents categorical labels rather than numerical values. To avoid misinterpretation in our predictive model, we converted this column to a string data type. This ensures the model treats 'area code' correctly as a categorical feature, preserving the integrity of our predictions.

### Finding 3: Multicollinearity

Our analysis revealed high correlations between several columns, indicating multicollinearity. For instance 'total day charge', 'total day minutes', 'total eve minutes', 'total eve charge', 'total night charge', 'total night minutes', 'total int minutes' and 'total int charge' have perfect multicollinearity. This can obscure the unique impact of each variable and potentially lead to overfitting, particularly in models like Logistic Regression that are sensitive to multicollinearity. To address this, we plan to implement techniques such as regularization, ensuring our models remain reliable and interpretable.



### Finding 4: Outliers

We observe the presence of a significant number of outliers in our dataset. Outliers have the potential to impact our modeling process. However, it is important to note that, in this case, these outliers are not anomalies that should be removed. Instead, they are a noteworthy aspect of our dataset that we should be aware of during our modeling process. These outliers may carry valuable information or insights that could be relevant to our analysis therefore it is essential to consider and account for them when developing our models and interpreting the results. Understanding the nature and impact of these outliers is a critical part of ensuring the robustness and accuracy of our data analysis.

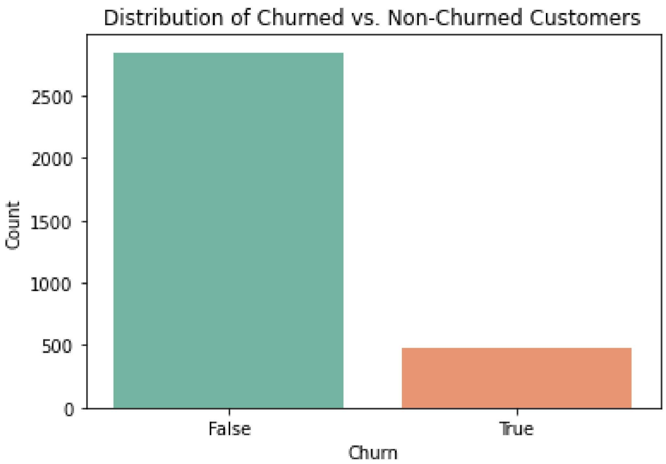
Adding regularization to our model can help reduce the impact of outliers by penalizing extreme parameter values making the model more generalizable and robust.

#### Finding 5: Class Imbalance

From the target variable above we saw that the churn class value count was 483 whereas the no churn count was 2850. We note a significant class imbalance here where the churn is the minority class and not churn is the majority class. This is common in churn datasets.

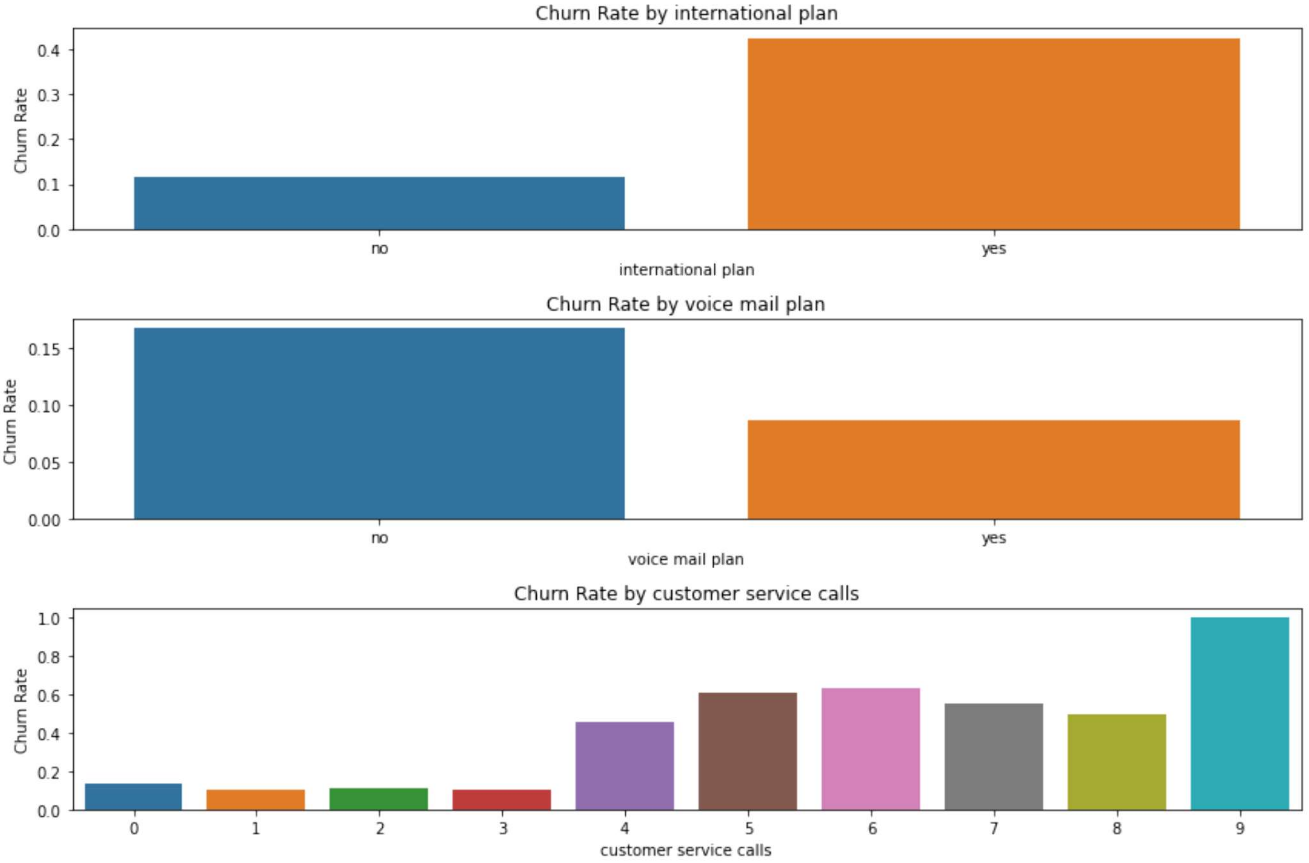
85.5% customer did not churn while 14.5% customers churned. The imbalance means that a model trained without addressing this issue will be biased toward predicting the majority class (customers not churning). This will lead to high accuracy but poor performance in identifying actual churners.

We will address class imbalance using a technique such as SMOTE before modeling to balance the 'churn' and 'not churn' classes. This should help improve overall model metrics like Precision, Recall, F1-score, and AUC-ROC. In imbalanced datasets a model may achieve high accuracy by being biased toward the majority class but this metric alone would be misleading and not truly reflective of the model's performance on the minority class which is what our model is meant to predict.



#### Finding 6: Features For Customers Likely To Churn

Most customers who churn do have the international plan, do not have the voice mail plan and have a record of calling the customer service line. The churn rate for the customers who have the international plan is the highest. The churn rate is also high when a customer calls for the 4th time. Customers who do not have the voice mail plan also tend to churn easily than the one's with the voice mail plan.



## Data Preprocessing

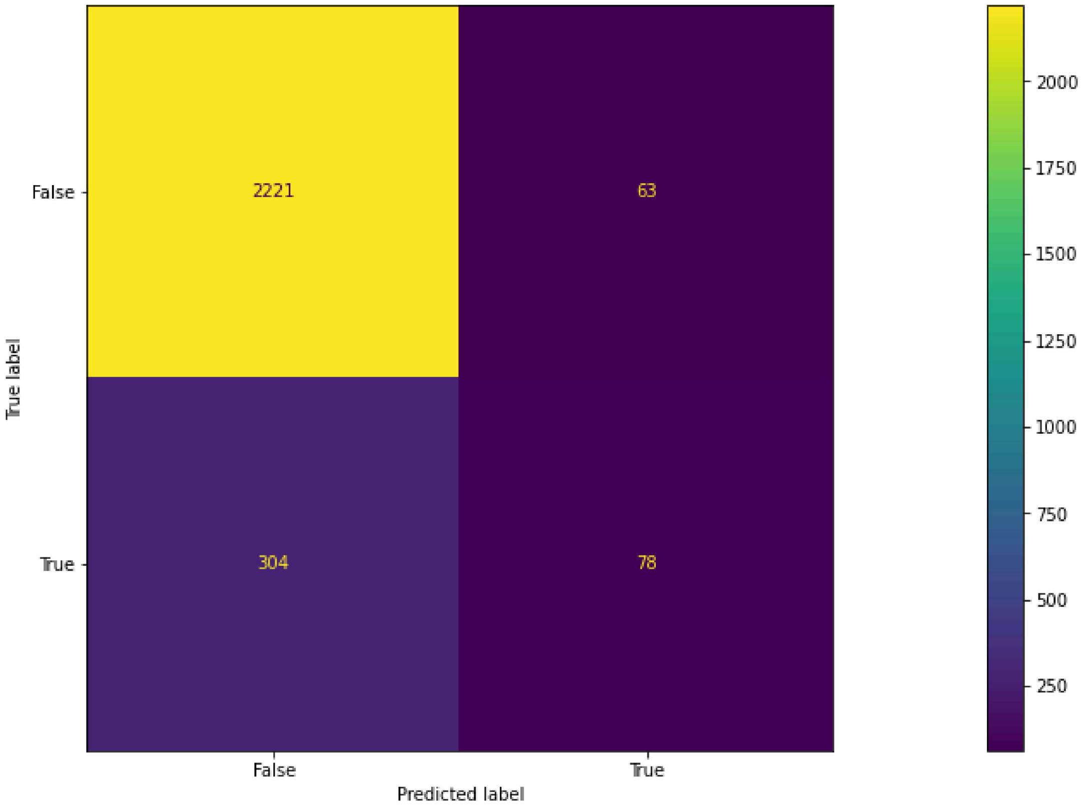
1. Data Splitting: We performed this step before preprocessing in order to prevent data leakage. This ensures that the test data remains completely unseen until the evaluation phase. Use the random state of 42 and a test size of 20%.
2. Handling Missing Data: We did not have missing data therefore there isnt much to handle here in this preprocessing step.
3. Encoding Categorical Variables: The 2 features 'international plan' and 'voice mail plan' and that are in the datatype object. We shall proceed to convert this variables to dummy ohe using the OneHotEncoder from sklearn.
4. Data Normalization using MinMaxScaler: MinMaxScaler with default parameters will create a maximum value of 1 and a minimum value of 0. This will work well with our binary one-hot encoded data.
5. Concatenating the Normalized and OheHotEncoded Train and Test Data: This is done to create a final dataframe with the ohehotencoded and normalized set for both the trained and tested sets.
6. Feature Selection Using Domain Knowledge: We proceeded to select our features using domain knowledge. From our dataset we eliminated the columns: area code, phone number and state as we did not find this columns relevant in our business problem.

# 3. Modeling

**Model Selection:**

Since this is a classification problem our first model will be **Logistic Regression** as it has a binary target variable then followed by **Decision Trees** as it is a powerful and flexible tool for classification problems, offering ease of interpretation, handling non-linear relationships, and providing automatic feature selection.

First we train our logistic regression baseline model with the imbalanced target variables. As we fit our model, we also generated the confusion matrix which gave us the below results:



- True Positive (TP): 78 customers who were predicted to churn actually did churn.
- False Negative (FN): 304 customers who were predicted not to churn actually did churn.
- False Positive (FP): 63 customers who were predicted to churn actually did not churn.
- True Negative (TN): 2221 customers who were predicted not to churn actually did not churn.

Our second logistic regression iterative model was done after class imbalance using the technique SMOTE. We also employed the regularization technique and used a lower C value. The random_state of 42 was maintained. Regularization helps prevent overfitting by penalizing large coefficients in the logistic regression model. A lower C value (where C is the inverse of the regularization strength) increases regularization, pushing coefficients toward zero.

The Decision tree baseline model was trained using the Decision Tree Classifier, criterion=entropy and the same random state was maintained to ensures that the model's behavior is reproducible and any changes in performance are due to the model's settings and not random variation.

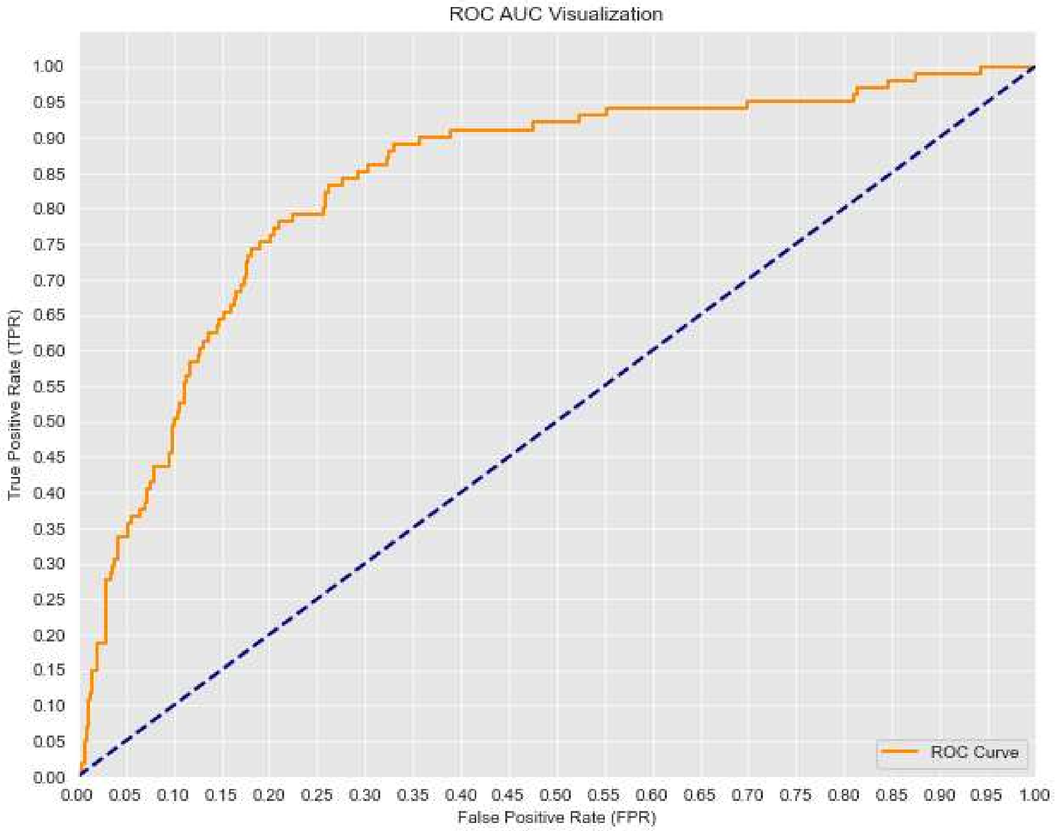
The second Decision tree iterative model had manually tuned step by step parameters that is max_features, max_depth, min_sample_splits, min_sample_leafs. We used the manually obtained optimum values of each feature to train our model.While decision trees are powerful and interpretable, they may not generalize well without proper tuning especially in the presence of class imbalance. To improve performance we considered getting the optimal values constraints on the tree.

Our last decision tree model was modified by grid search technique. Grid search systematically explores combinations of hyperparameters to find the set that yields the best performance based on cross-validation. We obtained the best parameters: criterion: gini, max_depth: 10, min_samples_leaf: 4, min_samples_split: 10. This was used to train our 3rd iterative model.

## 4 . Evaluation

The logistic regression baseline model struggles significantly with predicting customer churn especially due to the class imbalance. Although it achieves high overall accuracy this is largely due to correctly predicting non-churners which doesn't align with the business goal of accurately identifying churners. The low recall and F1 scores indicate that the model is not effectively capturing the customers at risk of leaving, which is critical for implementing successful retention strategies. Improving the model's performance on the minority class (churners) is essential for it to be truly valuable in a churn prediction context.

Logistic regression iterative model 2 is likely the better choice for predicting customer churn. Its higher recall ensures that more potential churners are identified, which is crucial in churn prevention strategies. While it has a lower precision the trade-off is justified by the significant gain in recall and F1 score making Model 2 more reliable for targeting retention efforts and ultimately reducing customer attrition. See below the ROC



AUC visualization:

We also cross validated the baseline and iterative model Performance: The baseline model achieved a cross-validation score of 0.861215. This score represents the model's ability to generalize to unseen data based on the training data it was given. The iterative model after some hyperparameter tuning and SMOTE achieved a slightly higher cross-validation score of 0.861590.

The decision tree baseline model performs very well for the "no churn" class. This is expected given the class imbalance. It achieves high precision, recall, and F1-score for this class, making it reliable for predicting customers who are likely to stay. It is weaker for the "churn" class with lower precision, recall, and F1-score. However, it still identifies a reasonable portion of churners which is crucial for proactive measures.

We got an even worse AUC when training the model using the identified feature points from the hyperparameter tuning and pruning. This is because we got the points one at a time. Considering this metrics we proceeded to use a more sophisticated technique called the **grid search**.

The third decision tree iterative model showed considerable improvements over the baseline, with higher accuracy, better precision and recall, improved F1-scores, a more balanced performance across classes, and a stronger ability to distinguish between churn and no churn, as evidenced by the higher ROC AUC score. These enhancements suggest a more effective model for predicting customer churn.

## Model of Choice

From the above models we have seen that the best performing model is the **Decision Tree Tuned by Grid Search technique**. The model has high accuracy and strong performance metrics for both churn and non-churn classes. The high recall for non-churn (False) indicates that the model effectively identifies customers who are likely to stay. The precision for churn (True) shows the model's effectiveness in identifying actual churners among the predicted churn cases. This balanced performance makes it a robust model for business applications where both accurate churn and non-churn predictions are crucial. The logistic regression iterative model has higher recall making it better at catching more churners, but the much lower precision means it also predicts churn for many who won't actually churn, potentially wasting resources.

In our case where the cost of missing a churner is significant but **precision** also matters for resource allocation, the decision tree model is the better choice. It is therefore likely more suitable for making business decisions related to customer retention and marketing strategies.

## Probable Limitations In Production

While our model might show high precision and recall for churners on the validation set, it could miss many churners or falsely classify non-churners in the real-world scenario,\ reducing its effectiveness in retaining customers. This would be due to overfitting.

Our recall for churners is low, the model might not effectively identify all potential churners, impacting the ability to proactively address churn and retain customers.

The model may not accurately predict churners if customer behavior has evolved since the model was last trained, potentially reducing its relevance and accuracy in the current context.

## Mitigation Strategies

Regular Monitoring and Updating: Continuously monitor the model’s performance in production. Implement regular updates and retraining to account for shifts in customer behavior and data distribution.

Validation on Real-World Data: Validate the model on recent real-world data before full-scale deployment. This helps ensure the model performs well under current conditions.

Combine with Other Models: Consider using ensemble methods or combining predictions from multiple models to improve overall prediction accuracy and robustness.

## 5. Conclusions

### Releases

No releases published
Create a new release

### Packages

No packages published
Publish your first package

### Languages

● **Jupyter Notebook** 100.0%