

Customer Churn Prediction Report

Github link - https://github.com/Begge10850/Telecom_customer_churn_rate

1. Business Problem

In today's competitive telecom market, customer retention is crucial. The company faces the challenge of losing customers, known as churn, which directly affects profitability. The objective of this project was to build a model to predict customer churn, helping the company identify high-risk customers and take action to retain them. By predicting churn, we can mitigate losses and improve customer satisfaction through targeted interventions.

2. Approach

To address this business problem, I set out to build a machine learning model capable of predicting whether a customer will churn or not. The model was trained using historical customer data, which included various factors like account length, service usage, and customer service interactions. The goal was to develop a tool that would allow the company to proactively engage with customers who are likely to churn.

3. Data Overview

The dataset used for this analysis contained customer information, such as:

State: Where the customer is located.

Account length: How long they've been a customer.

International plan: Whether the customer has an international plan.

Voice mail plan: Whether they have a voicemail plan.

Service Usage: Total minutes used during day, evening, and night periods.

Customer Service Interactions: The number of times the customer called support.

Churn: The target variable (True/False) indicating whether a customer has left the service.

4. Data Exploration

Before jumping into the modeling process, I spent time exploring the data to better understand its structure and identify any issues. I started by looking at the distribution of key features to see if there were any obvious trends or outliers that might affect the model.

Data Integrity: I checked for any missing or inconsistent data.

Feature Relationships: I explored how features like customer service calls or international plan might be related to churn. For example, customers with more customer service calls appeared more likely to churn, suggesting dissatisfaction.

5. Data Preparation

Next, I moved on to cleaning and preparing the data for model training. This step was crucial to ensure the models had quality data to learn from:

Encoding Categorical Variables: Features like State, International plan, and Voice mail plan were categorical, so I encoded them into numerical values.

Handling Missing Values: After confirming there were no major missing values, I moved forward without needing complex imputation strategies.

Feature Scaling: For models like Logistic Regression, I standardized the numerical features to make sure they were on the same scale, which helps with model convergence and improves performance.

6. Feature Engineering

One of the key steps in this process was feature engineering, where I created new, potentially useful features from the existing ones:

Total Minutes: I created a feature that aggregated the total day, evening, and night minutes into a single value. This provided a more holistic view of a customer's usage.

Customer Interaction: Since customer service calls seemed like an important predictor of churn, I gave it special attention as a potential driver in customer dissatisfaction.

7. Model Selection

With the data prepared, I trained several models to predict customer churn. These models included:

Logistic Regression: A straightforward, interpretable model for binary classification.

Random Forest: An ensemble method that combines many decision trees to improve predictions.

XGBoost: A high-performance boosting algorithm that's particularly effective for structured data.

Each model was chosen for its ability to handle classification tasks well, and I decided to compare their performance to find the best option.

8. Hyperparameter Tuning

To improve the accuracy of each model, I employed Grid Search to fine-tune their hyperparameters:

For Logistic Regression, I focused on adjusting the regularization parameter.

For Random Forest, I tuned the number of trees, depth of each tree, and the minimum number of samples required for a split.

For XGBoost, I optimized parameters like learning rate, number of estimators, and tree depth.

This tuning process helped ensure the models were as accurate and efficient as possible.

9. Model Testing

After training the models, I tested them on unseen data (the test set) to evaluate their real-world performance. I predicted whether each customer would churn and compared the predictions to the actual results. The goal was to determine how well each model performed in practice, not just on the training data.

10. Evaluation Metrics

I used several metrics to evaluate the performance of the models:

Accuracy: The percentage of correct predictions out of total predictions.

Precision: How many of the churn predictions were actually correct.

Recall: The model's ability to correctly identify all the churn cases.

F1-Score: A balance between precision and recall, especially important in cases of class imbalance.

Each model's performance was compared across these metrics to select the most effective one.

11. Feature Importance

In the end, I also wanted to understand which features were the most important in predicting churn. I used:

Random Forest and XGBoost to extract feature importance scores, which ranked features like customer service calls and total minutes as the top predictors.

In Logistic Regression, I examined the coefficients to determine how strongly each feature impacted the model's predictions.

This analysis provided insights into which factors are driving customer churn, offering actionable data for the company.

12. Conclusion

In this project, I built and evaluated three machine learning models to predict customer churn. By cleaning and preprocessing the data, engineering meaningful features, and fine-tuning the models, I was able to create an effective churn prediction tool. Among the models, I found that XGBoost provided the best balance between accuracy and interpretability, making it the most suitable choice for this task. Finally, I identified key features such as customer service interactions and total usage that are most strongly associated with churn, providing actionable insights for customer retention strategies.