Customer Churn Prediction: Detailed Analysis and Report

Introduction

Customer churn, the phenomenon where customers leave a service or company, is a major concern for many businesses, especially in competitive industries like telecommunications. The loss of a customer represents not just the immediate revenue loss, but also the potential impact on future profits, as retaining existing customers is far more cost-effective than acquiring new ones.

In this project, we aimed to predict whether a customer would churn using machine learning models. Specifically, we used ensemble learning, a powerful technique that combines multiple models to improve prediction accuracy. The dataset we worked with was the Telco Customer Churn Dataset, which contains detailed information about telecom customers and whether they churned or stayed.

The objective was to build a model that can predict whether a customer would churn based on their demographic, account, and service usage data. We applied an ensemble method combining Random Forest and Gradient Boosting classifiers in a Voting Classifier model.

Dataset Overview

The Telco Customer Churn Dataset is a classic example used to explore customer churn prediction. It contains several features about telecom customers, including:

- **Demographic Information**: Features like gender, age, and geographical location.
- **Account Information**: Contract type, payment method, tenure (how long the customer has been with the company), etc.
- **Service Usage**: Number of services the customer uses, whether they have internet service, and monthly charges.

The dataset has 2113 records and 20 feature columns, with the target variable being **Churn** (1 for churned customers, 0 for retained customers).

Data Preprocessing and Preparation

To ensure that the model can effectively learn from the data, several preprocessing steps were necessary:

1. **Handling Missing Values**: We began by checking for any missing data. Fortunately, there were no missing values in this dataset, so no imputation was required.

2. **Categorical to Numerical Conversion**: The dataset contains several categorical features (e.g., gender, contract type). We used one-hot encoding to convert these into numerical values. This transformation is necessary because machine learning models only work with numerical data.

3. **Feature Scaling**: Features like monthly charges and tenure are on different scales, which can be problematic for certain models. To resolve this, we applied **StandardScaler** to ensure all features are on a comparable scale, making it easier for the model to learn from the data.

4. **Train-Test Split**: We split the data into a training set (70%) and a testing set (30%) to train the model and evaluate its performance on unseen data.

Model Building

We used ensemble learning, a technique that combines the predictions of multiple models to improve accuracy and robustness. Specifically, we chose two powerful models:

- **Random Forest Classifier**: This model creates multiple decision trees and combines their outputs to improve predictive accuracy and reduce overfitting.

- **Gradient Boosting Classifier**: Unlike Random Forest, which builds trees in parallel, Gradient Boosting builds trees sequentially, each one trying to correct the mistakes of the previous one. This method tends to perform well on difficult datasets.

Both models were combined into a Voting Classifier. The Voting Classifier takes the predictions from both models and predicts the class (Churn/No Churn) based on a majority vote.

5. Evaluation and Results

After training the ensemble model on the training set, we evaluated its performance on the testing set using several metrics. Here's how the model performed:

Model Accuracy:

- The ensemble model achieved an accuracy of 78.94%. This means that the model correctly predicted whether a customer would churn or not in about 79% of cases. While this is a respectable result, there are still areas for improvement.

Confusion Matrix:

The confusion matrix is a tool that allows us to see the performance of the model in greater detail. Here's the matrix:

[[1446 93]

[352 222]]

- **True Negatives (TN)**: 1446 customers were correctly predicted as "No Churn."
- **False Positives (FP)**: 93 customers were incorrectly predicted as "Churn" when they did not.
- **False Negatives (FN)**: 352 customers were incorrectly predicted as "No Churn" when they actually churned.
- **True Positives (TP)**: 222 customers were correctly predicted as "Churn."

Classification Report:

The classification report provides further insights into how well the model performed for each class (Churn and No Churn):

```
Classification Report:
              precision    recall  f1-score   support

       False       0.80      0.94      0.87      1539
        True       0.70      0.39      0.50       574

    accuracy                           0.79      2113
   macro avg       0.75      0.66      0.68      2113
weighted avg       0.78      0.79      0.77      2113
```

- **Precision for Churn** (0.70) indicates that when the model predicts a customer will churn, it is correct 70% of the time.
- **Recall for Churn** (0.39) is quite low, meaning the model fails to identify 61% of the customers who actually churned. This is a key area that needs improvement.
- **F1-Score** is a balance between precision and recall. For "Churn", the F1-Score of 0.50 reflects the model's struggle to correctly identify customers likely to churn.

Key Insights and Challenges

- **Class Imbalance**: A significant challenge with churn prediction models is class imbalance, where the number of customers who stay (No Churn) vastly outnumbers those who leave (Churn). This imbalance can lead the model to be biased toward predicting "No Churn." In our case, the model has a high recall for "No Churn" (0.94), but a low recall for "Churn" (0.39). This results in many false negatives for the "Churn" class, which is problematic for businesses trying to retain customers.

- **Model Bias**: Although the model is accurate overall, it struggles with predicting "Churn" cases. This bias toward predicting "No Churn" can lead to missed opportunities for intervention.

Recommendations for Improvement

To improve the performance of this model, particularly in identifying "Churn" customers, the following steps could be taken:

1. **Handle Class Imbalance**:
    - **SMOTE (Synthetic Minority Over-sampling Technique)**: This technique generates synthetic samples for the minority class ("Churn") to balance the class distribution.
    - **Adjusting Class Weights**: Modifying the class weights in the Random Forest and Gradient Boosting models would penalize misclassifying "Churn" customers, which could improve recall for the "Churn" class.
2. **Hyperparameter Tuning**:
    - **Grid Search or Random Search**: These techniques could be used to fine-tune the parameters of the Random Forest and Gradient Boosting classifiers, potentially improving model performance.
3. **Advanced Models**:
    - Exploring more advanced ensemble methods such as **XGBoost** or **LightGBM**, which are known to perform well on imbalanced datasets, might lead to better results.
4. **Feature Engineering**:
    - Further feature engineering could improve model performance. For example, combining existing features or introducing new features like customer tenure,

total charges, and usage patterns could help the model better distinguish between customers likely to churn and those who are not.

5. **Threshold Adjustment**:
    - Adjusting the decision threshold for predicting "Churn" could increase the model's sensitivity to this class, improving recall at the cost of a potential increase in false positives.

Conclusion

In conclusion, the Ensemble Learning approach combining Random Forest and Gradient Boosting performed well, achieving an accuracy of 78.94%. However, the model's performance on predicting customer churn could be improved, particularly in increasing recall for the "Churn" class. By addressing class imbalance, tuning hyperparameters, and exploring more advanced techniques, this model could be optimized to better identify customers who are likely to churn.

This project provides a solid foundation for customer churn prediction, and with further improvements, it could become a powerful tool for businesses to reduce customer loss and increase retention efforts.