Project Summary: Customer Churn Prediction
Sean W. Ryan

Introduction

Customer churn is a critical problem for businesses, representing the loss of customers who discontinue their services. Predicting churn allows companies to take proactive measures to retain customers, thus improving business performance. In this project, I use the "Telco Customer Churn" dataset from Kaggle to build models that predict whether a customer will churn or not.

Data Exploration and Cleaning

The dataset contains 7,043 entries with 21 columns, including customer demographics, account information, and service usage details. Initial data exploration revealed no missing values, but the TotalCharges column required conversion from an object to a numeric type. I dropped the customerID column as it doesn't contribute to the prediction task and performed one-hot encoding on categorical features. Additionally, I created new features: tenure groups and average monthly charges to enhance model performance.

Feature Selection

I employed Recursive Feature Elimination (RFE) with a RandomForestClassifier to select the top 10 most significant features:

- tenure
- MonthlyCharges
- TotalCharges
- avg_monthly_charges
- gender_Male
- InternetService_Fiber optic
- Contract_One year
- Contract_Two year
- PaperlessBilling_Yes
- PaymentMethod_Electronic check

Model Building

I started with a baseline Logistic Regression model, achieving an accuracy of 79%. To improve performance, I explored Random Forest and Gradient Boosting models. Hyperparameter tuning using Grid Search identified the best parameters for the Gradient Boosting model: learning_rate=0.1, max_depth=3, and n_estimators=200.

Model Improvement

Addressing class imbalance using SMOTE (Synthetic Minority Over-sampling Technique) improved the recall for the churn class. The tuned Gradient Boosting model with SMOTE achieved a recall of 0.65 for the churn class, indicating better identification of churned customers.

Stacking Classifier

To further enhance performance, I implemented a Stacking Classifier combining Random Forest and Gradient Boosting models with a Logistic Regression meta-model. Hyperparameter tuning for the Stacking Classifier identified the best C parameter for the Logistic Regression meta-model.

Results

The table below summarizes the performance of different models:

Model	Accuracy	Precision (Churn)	Recall (Churn)	F1-Score (Churn)
Logistic Regression	0.79	0.62	0.52	0.56
Random Forest	0.78	0.63	0.47	0.54
Gradient Boosting	0.79	0.64	0.48	0.55
Tuned Gradient Boosting	0.79	0.63	0.50	0.56
Tuned GB with SMOTE	0.77	0.56	0.65	0.60
Stacking Classifier	0.79	0.64	0.47	0.54

SHAP Analysis

Using SHAP (SHapley Additive exPlanations) to analyze feature importance, I discovered:

- Tenure: Longer tenure significantly reduces the likelihood of churn.
- InternetService_Fiber optic: Associated with a higher churn likelihood.
- Contract Length: Two-year contracts significantly reduce churn risk.
- PaymentMethod_Electronic check: Associated with higher churn likelihood.

Conclusion

The Gradient Boosting model with SMOTE provided the best recall for the churn class, balancing the need for accuracy and the ability to identify churned customers. The Stacking Classifier also showed promising results. Future work could explore further feature engineering, advanced ensemble methods, and more sophisticated hyperparameter tuning to enhance performance. This project demonstrates a comprehensive approach to solving a real-world problem using data science techniques.