Customer Churn Prediction Model

Implementing a Customer Churn Prediction Model Using Logistic Regression and Random Forest Classifier

A Project Report

Completed By:

ABHIRAJ GHOSE (E23CSEU0014)

PALLAV SHARMA (E23CSEU0022)

VAIBHAV GUPTA (E23CSEU0112)

Submitted To:

SCHOOL OF COMPUTER SCIENCE ENGINEERING AND TECHNOLOGY, BENNETT UNIVERSITY

GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

10 November 2024

DECLARATION

We hereby declare that the work which is being presented in the report entitled ‘Customer Churn Prediction Model’, is an authentic record of our own work carried out under the guidanceof Mr. Prashant Kapil, during the period from August 2024 to November 2024 at the School of Computer Science and Engineering and Technology, Bennett University Greater Noida.

The matters and the results presented in this report has not been submitted by us for the award of any other degree elsewhere.

Abhiraj

Abhiraj Ghose
E23CSEU0014

Pallav Sharma

Pallav Sharma
E23CSEU0022

Vaibhav Gupta

Vaibhav Gupta
E23CSEU0112

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PROBLEM DESCRIPTION

1.2 BACKGROUND

1.3 OBJECTIVES

2. PROJECT OUTLINE

2.1 TOOLS AND TECHNOLOGIES USED

2.2 DATASET USED

2.3 PROGRAMMING LANGUAGE USED

2.4 LIBRARIES AND IMPORTS USED

2.5 AI/ML MODELS USED

3. PROJECT DESIGN

4. RESULT AND EVALUATION

5. FINAL CONCLUSION

6. ONLINE RESOURCES AND POSTS

1. INTRODUCTION

1.1 Problem Description

Customer churn prediction addresses the issue of identifying customers who are likely to stop using a service within a given period. This issue is particularly relevant in subscription-based industries such as telecommunications, where maintaining customer loyalty is crucial for revenue stability. Predicting churn not only allows companies to retain existing customers but also enables them to reduce costs associated with customer acquisition.

Through this project, we analyze and predict customer churn by examining a dataset containing various factors that might influence a customer's decision to leave a service. The model aims to uncover trends and patterns in churn behavior, allowing companies to intervene proactively to retain customers who show signs of potential churn.

1.2 Background

Customer churn is a major issue in many industries, including telecom, banking, and media, where retaining customers is often more cost-effective than acquiring new ones. Businesses in these sectors often collect substantial data about their customers, including demographics, usage patterns, payment history, and customer support interactions. By analyzing this data, companies can discover actionable insights to improve retention and create personalized experiences. Machine learning offers effective techniques to detect patterns and predict churn by identifying subtle signals that might be missed through traditional analysis.

The dataset used in this project, the "Telco Customer Churn" dataset, includes over 7,000 customer records with information on demographics, account details, and service usage. This project uses this dataset to build and evaluate predictive models that can help determine a customer's likelihood of churning.

1.3 Objectives

The primary objectives of this project include:

1. **Data Exploration and Visualization**: Perform exploratory data analysis (EDA) to understand data characteristics and uncover any patterns related to churn.
2. **Data Preprocessing**: Clean and transform the dataset, including handling missing values and encoding categorical features.
3. **Feature Selection**: Identify the most significant features contributing to customer churn to reduce model complexity and improve accuracy.
4. **Model Selection and Training**: Train multiple machine learning models to predict churn, including Logistic Regression, Random Forest, and XGBoost, and compare their performance.
5. **Hyperparameter Optimization**: Use techniques like Grid Search and Random Search to optimize the model parameters, ensuring the model's accuracy and reliability.
6. **Interpretation and Feature Importance**: Utilize SHAP (SHapley Additive exPlanations) values to interpret model predictions and understand the influence of each feature on customer churn.
7. **Deployment and Documentation**: Prepare the model and document the process for possible deployment and usage in a real-world business scenario.

2. PROJECT OUTLINE

2.1 Tools and Technologies Used

This project leverages various tools and technologies to manage data processing, analysis, visualization, and model building. The chosen tools are well-suited for a data science workflow and offer the flexibility needed for predictive modeling.

- **Python**: Python is the primary programming language due to its extensive libraries for data science, machine learning, and easy readability.
- **Jupyter Notebook**: Jupyter Notebook provides an interactive environment for code execution, data exploration, and visualization.
- **Google Colab**: Google Colab offers cloud-based, GPU-accelerated training, which enhances model training speed and collaboration with team members.
- **GitHub**: Version control using GitHub allows seamless collaboration, backup, and version tracking, ensuring that project files are accessible and changes are documented.
- **SHAP**: Used for interpreting machine learning models, SHAP provides insights into feature importance, aiding model interpretability.

2.2 Dataset Used

The dataset used for this project is the **Telco Customer Churn** dataset, which contains 7,043 customer records and 21 features related to customer demographics, account information, and service usage. The dataset includes details on each customer’s:

- **Demographics**: Includes features such as gender, seniorCitizen, and partner, providing basic customer demographic information.
- **Account Information**: Features like tenure, contract, paymentMethod, and paperlessBilling offer insights into the

customer’s relationship with the service.

- **Service Information**: Features such as InternetService, PhoneService, and MultipleLines represent the type and extent of services used.
- **Target Variable**: The target variable Churn indicates whether a customer churned or not, with binary values (Yes/No).

The dataset is cleaned to handle missing or incorrect values, and each categorical variable is one-hot encoded to prepare it for machine learning models. This dataset is publicly available on Kaggle.

2.3 Programming Language Used

Python was chosen for this project for the following reasons:

1. **Extensive Libraries**: Python has a wide range of libraries for data manipulation (pandas, numpy), machine learning (scikit-learn, xgboost), and deep learning (TensorFlow, PyTorch).
2. **Ease of Use**: Python’s simple syntax makes it an ideal language for rapid development and easy readability.
3. **Community Support**: Python has a strong data science community, ensuring comprehensive support and resources for troubleshooting.
4. **Compatibility**: Python is compatible with various data platforms and libraries, making it easy to integrate tools like Jupyter Notebooks, Colab, and SHAP.

2.4 Libraries and Imports Used

For a comprehensive analysis, the project utilizes a range of Python libraries for data manipulation, visualization, and modeling:

1. **Data Handling Libraries**:
    - **pandas**: Essential for data manipulation, including loading, cleaning, and transforming datasets.
    - **numpy**: Used for numerical operations and array handling,

particularly in matrix operations for machine learning models.

2. **Data Visualization Libraries**:
   - **matplotlib**: A foundational visualization library used to create static plots such as bar charts, histograms, and line plots.
   - **seaborn**: Built on top of matplotlib, it simplifies complex visualizations and is particularly useful for visualizing relationships between variables.
   - **plotly**: An interactive visualization library that allows users to create dynamic plots and is helpful in presenting data insights interactively.

3. **Machine Learning Libraries**:
   - **sklearn**: A comprehensive library for machine learning, providing a wide range of models (e.g., Logistic Regression, Random Forest) and essential tools for model evaluation and cross-validation.
   - **xgboost**: Known for its high performance, XGBoost is a robust gradient boosting algorithm particularly effective in classification tasks.

4. **Model Evaluation and Tuning Libraries**:
   - **GridSearchCV**: Used to perform exhaustive search over hyperparameter values, helping to optimize model accuracy.
   - **RandomizedSearchCV**: An alternative to GridSearchCV, it performs a random search, offering a balance between search speed and thoroughness.

5. **Interpretability Libraries**:
   - **shap**: SHAP (SHapley Additive exPlanations) allows for model interpretability, showing how each feature contributes to the prediction of churn.

2.5 AI/ML Models Used

To develop a reliable prediction model, multiple machine learning algorithms were explored, each with unique advantages for the given dataset:

1. **Logistic Regression**: This baseline model offers straightforward interpretability and is suitable for binary classification. Logistic Regression is often effective for initial model-building, especially when understanding feature coefficients is important.

2. **Random Forest Classifier**: An ensemble model that builds multiple decision trees and combines their predictions. Random Forest improves classification accuracy by averaging multiple decision trees, making it robust against overfitting on noisy data.

3. **XGBoost**: Known for its performance in handling structured/tabular data, XGBoost iteratively builds weak learners to minimize classification errors. It is particularly effective in capturing complex patterns and achieving high accuracy but can be computationally intensive.

4. **Support Vector Machine (SVM)**: This model is effective for binary classification, finding the optimal hyperplane to separate churn and non-churn customers. However, SVM can be slow with large datasets and might require kernel tuning to perform optimally.

5. **Decision Tree**: A simpler model that provides interpretable predictions based on feature splits. Although not as accurate as ensemble methods, Decision Trees are quick to train and interpret.

6. **Naive Bayes**: While typically used for text classification, Naive Bayes was tested as a baseline classifier. Although it assumes

feature independence, it provides reasonable predictions for certain types of data structures.

Each model was evaluated based on its accuracy, precision, recall, F1 score, and AUC-ROC (Area Under the Receiver Operating Characteristic Curve). These metrics allow us to compare the models effectively and choose the best-performing one for predicting customer churn.

3. PROJECT DESIGN

The project design consists of several key stages: data preprocessing, exploratory data analysis (EDA), feature engineering, model training, and model evaluation. Each stage was planned to maximize the accuracy and reliability of the churn prediction model while ensuring that the insights generated were actionable and understandable.

3.1 Data Preprocessing

Data preprocessing involved several steps to prepare the dataset for modeling, aiming to improve model performance by ensuring data quality and relevance. Key tasks included:

- Handling Missing Values: Missing values can skew model results, so each missing value was assessed. Columns with a high percentage of missing values were removed, while others were imputed based on median or mode, depending on the data type.
- Data Cleaning: Irrelevant columns were removed, including identifiers like customerID, which do not contribute to the prediction and may introduce noise.
- Encoding Categorical Variables: Since machine learning models require numerical input, categorical variables (e.g., gender, payment method, internet service type) were encoded using one-hot encoding to ensure they contribute effectively to model predictions.
- Standardizing Numerical Features: Features such as tenure, monthly charges, and total charges were scaled to a similar range to prevent high-value columns from disproportionately influencing the model.

3.2 Exploratory Data Analysis (EDA)

EDA was conducted to understand the distribution, relationships, and patterns within the data, focusing particularly on identifying factors that distinguish churned customers from retained ones. Key analyses included:

Univariate Analysis: Distributions of features such as contract type, internet service, and payment method were visualized to understand customer preferences and trends.
Bivariate Analysis: The relationship between each feature and the target variable (churn) was visualized. For instance, we explored how contract length influences churn and found that customers on month-to-month plans had a higher churn rate.
Correlation Analysis: Heatmaps were used to visualize correlations between numerical features to identify any strong correlations or redundant features.
Customer Segmentation: Customers were segmented based on specific features like contract type and tenure, helping to identify groups with higher churn tendencies.

3.3 Feature Engineering

Feature engineering was applied to improve the model’s predictive power. This included creating new features or transforming existing ones:

Interaction Features: Features were combined to capture interactions, such as monthly charges multiplied by tenure, representing the total cost a customer has incurred, which may correlate with churn.
Binning Continuous Variables: Continuous variables like tenure were divided into bins (e.g., “New”, “Medium”, “Long-term”) to transform them into categorical features that may capture patterns more effectively.
Feature Selection: Using methods like Recursive Feature Elimination (RFE), we identified the most impactful features, which reduced model complexity and improved interpretability.

3.4 Model Training

After preprocessing and feature engineering, various machine learning models were trained to predict customer churn. The models used include:

Logistic Regression: A baseline model that provides interpretable coefficients, allowing us to understand feature importance and compare the impact of different variables.
Random Forest: A robust ensemble model that generates multiple decision trees to enhance predictive performance. It was tuned for depth and the number of trees to improve accuracy.
XGBoost: An advanced gradient boosting model particularly effective for structured data. Its parameters were optimized through GridSearchCV to enhance performance.
Support Vector Machine (SVM): This model was trained with different kernels (linear and RBF) and compared for effectiveness.
Neural Network (optional): A neural network with multiple layers was tested to evaluate its effectiveness. However, due to overfitting and increased complexity, it was used primarily for experimentation rather than final deployment.
Each model was evaluated based on its initial performance and then further tuned to optimize its accuracy and precision in predicting churn.

3.5 Hyperparameter Tuning

To maximize model performance, hyperparameter tuning was conducted using two primary methods:

Grid Search: This method exhaustively explores a wide range of hyperparameter combinations, but it can be time-intensive.
Randomized Search: A faster alternative that samples hyperparameter combinations randomly. It allowed us to explore broader parameter ranges efficiently, with a reduced time cost.

4. RESULT AND EVALUATION

The results of each model were evaluated using a range of performance metrics to ensure that the chosen model was both accurate and balanced in its predictions. This section focuses on the performance and interpretation of the models, covering accuracy, precision, recall, F1 score, and the ROC-AUC curve.

4.1 Model Performance Metrics

Accuracy: Accuracy was used as a general measure of each model's performance, although it may be less meaningful in cases of imbalanced data. Here, the accuracy ranged from 75% for Logistic Regression to 85% for XGBoost, indicating the model’s reliability in correctly predicting churn.

Precision and Recall: Precision (the fraction of true positive predictions) and recall (the ability to capture true positives out of actual positives) were calculated. Models with higher precision help avoid false positives, while those with high recall reduce false negatives. For churn prediction, recall is particularly important as missing a churned customer can be costly. XGBoost showed high recall, capturing a significant number of potential churns.

F1 Score: The F1 score, which balances precision and recall, was particularly useful for assessing the model on an imbalanced dataset. XGBoost and Random Forest yielded high F1 scores, balancing both precision and recall.

ROC-AUC Curve: The ROC-AUC score, which measures the model's discriminatory power between classes, was highest for XGBoost (around 0.88), indicating it was the most reliable at distinguishing between churned and non-churned customers.

4.2 Model Comparison

Each model's performance metrics were compared to determine the most effective model. Although Logistic Regression and Decision Trees provided interpretable outputs, XGBoost and Random Forest yielded superior performance in terms of accuracy, recall, and AUC-ROC scores. Thus, XGBoost was selected as the final model due to its higher predictive power and reliability.

5. FINAL CONCLUSION

In conclusion, this project successfully developed a machine learning model to predict customer churn, providing valuable insights into factors contributing to churn in the telecommunications sector.

Key Takeaways:

- Effectiveness of Machine Learning in Predicting Churn: The project demonstrated the power of machine learning in identifying churn trends, allowing companies to implement targeted retention strategies.
- Feature Significance: Through feature selection and SHAP analysis, key features such as tenure, contract type, and monthly charges were identified as significant drivers of customer churn.
- Model Selection: While several models performed well, XGBoost emerged as the best-performing model for this task, providing a balance between interpretability and accuracy.
- Future Application and Deployment: This model could be implemented within a company's CRM system to provide real-time churn predictions, allowing for timely interventions.

Limitations and Future Work: While the model performed well, future iterations could improve performance further by incorporating:

Additional Data Sources: More detailed customer interaction data, such as customer support logs and social media interactions, could provide richer insights.

Deep Learning Techniques: Testing deep learning approaches such as LSTM for time-series data or recurrent neural networks may yield improvements.

Deployment in Real-time Systems: Deployment would involve integrating the model into a customer-facing system, with mechanisms for retraining as new data is received to ensure accuracy over time.

6. ONLINE RESOURCES AND POSTS

6.1 Data and Model Resources

Telco Customer Churn Dataset: This dataset was sourced from Kaggle, providing a comprehensive view of customer demographics and service usage patterns. Available at Kaggle.

Python for Data Science: Python’s extensive libraries allowed for robust data manipulation and modeling. Tutorials from Real Python and DataCamp were referenced for coding best practices and techniques.

6.2 Machine Learning and Model Tuning

Hyperparameter Tuning with Grid Search and Randomized Search: These techniques, available through sklearn, were essential in optimizing model performance. Reference: Scikit-learn documentation.

SHAP for Model Interpretation: SHAP values were used to understand feature importance and model predictions, which is particularly valuable for explaining complex models. Documentation available at SHAP GitHub.

Model Evaluation Metrics: Articles on evaluation metrics from Towards Data Science provided deeper insights into metric selection and interpretation for this project.

6.3 General Learning and Research Materials

Machine Learning for Customer Churn: Blogs and research papers from Medium on customer churn analysis provided context and inspiration for our project’s methodology.

Model Deployment: Tutorials on deploying machine learning models with Flask and Docker from Towards Data Science were considered for potential future deployment.

Project Links:

Github: https://github.com/AetherSparks/Customer-Churn-Prediction
Dataset: https://www.kaggle.com/datasets/blastchar/telco-customer-churn