Machine Learning Project Report: Customer Churn Prediction and Deployment

1. Introduction

This report presents the process and outcomes of a Machine Learning (ML) project aimed at predicting customer churn for a business. The project involves various stages including data processing, feature engineering, data splitting, model selection, and deployment. The primary objective of the project is to build a predictive model that can effectively identify potential churners among customers, thereby allowing the business to take proactive measures for retention.

2. Data Processing

The dataset used for this project had been pre-processed to address missing values, data imbalance, and skewness. No missing values were present in the dataset, ensuring the integrity of the data. Additionally, efforts were made to tackle data imbalance if any, as an imbalanced dataset can lead to biased results. Lastly, the absence of skewness indicates that the data was distributed relatively evenly across different features.

3. Feature Engineering

Feature engineering is crucial for enhancing the predictive capabilities of the model. In this project, categorical data was appropriately encoded to transform them into numerical representations. This was necessary since most ML algorithms require numerical input. Unnecessary columns that did not contribute significantly to the prediction task were dropped to simplify the model and improve efficiency.

4. Data Set Splitting

To evaluate the model's performance, the dataset was split into training and testing sets. An 80:20 split was chosen using the `train_test_split` function from the Scikit-Learn library. This ensured that the model was trained on a substantial portion of the data while retaining an independent subset for testing and validation.

5. Model Selection

For the churn prediction task, the XGBoost algorithm was chosen as the model of choice. XGBoost is known for its robustness and ability to handle complex relationships in data. During training, the model achieved a training accuracy of approximately 80%, indicating an excellent fit to the training data. However, to prevent overfitting and assess the model's generalization, cross-validation was performed. The cross-validation

accuracy was around 50%, suggesting that there might be room for further model improvement or parameter tuning.

6. Deployment using Streamlit

The final model was deployed using Streamlit, a user-friendly Python library for creating web applications for ML projects. The Streamlit app provides a simple interface for users to input relevant information, and the model uses this input to predict the likelihood of customer churn. The deployment enables non-technical users to interact with the model without requiring programming or data science. knowledge

7. Conclusion

In conclusion, this ML project successfully addressed the task of predicting customer churn. Data preprocessing and feature engineering ensured the quality of the dataset, while appropriate model selection led to the deployment of an XGBoost model. Although the model's cross-validation accuracy was moderate, its deployment through Streamlit allows the business to actively predict potential customer churn and take necessary actions to retain customers. Further model refinement and exploration of different algorithms could potentially enhance the predictive accuracy in future project iterations