Customer Churn Prediction

Data Preprocessing:
1. The dataset was loaded from a CSV file located in Google Drive using Google Colab.

2. Initial data exploration was performed to understand the dataset's structure, including checking for missing values and exploring the distribution of the 'Churn' variable.

3. Descriptive statistics were generated to gain insights into the numeric features.

Feature Engineering:
4. Several columns that were considered not informative or redundant were dropped, including 'Name,' 'Gender,' and 'CustomerID.'

5. Categorical variables were identified based on a threshold of unique values and encoded using one-hot encoding. Only the 'Location' column was one-hot encoded.

6. Numeric variables were scaled using Min-Max scaling, specifically 'Subscription_Length_Months' and 'Monthly_Bill.'

Handling Class Imbalance:

7. To address class imbalance, the minority class ('Churn' == 1) was upsampled to match the number of samples in the majority class ('Churn' == 0) using random sampling.

Model Building and Evaluation:

8. The dataset was split into training and testing sets with an 80-20 split.

9. Two classification models were implemented and evaluated:
   - Ridge Classifier
   - Random Forest Classifier with specified hyperparameters.

Model Improvement with GridSearchCV:

10. GridSearchCV was used for hyperparameter tuning of the Random Forest Classifier.

11. A parameter grid was defined for 'n_estimators' and 'max_depth,' and cross-validation was performed to find the best hyperparameters.

12. The best hyperparameters were identified, and the model was retrained with these hyperparameters.

Results:

- The Ridge Classifier achieved a certain level of accuracy on both the training and testing datasets.

- The Random Forest Classifier, initially trained with specified hyperparameters, also achieved accuracy scores on both datasets.

- GridSearchCV was used to improve the Random Forest Classifier's hyperparameters, and the best hyperparameters were identified.

Conclusion:

- The provided code outlines a comprehensive approach to customer churn prediction, including data preprocessing, feature engineering, handling class imbalance, and model selection.

- The final model's performance should be assessed further using additional evaluation metrics (e.g., precision, recall, F1-score) and potentially fine-tuning based on specific business goals.

- It's important to note that the provided report summarises the code's approach based on the information provided, and additional insights and details may be obtained by further analysis and interpretation of the results.