**SBA Loan Default Prediction**

**Objective**:

The analysis aimed to clean, transform, and prepare the SBA loan dataset for predictive modeling. The goal was to identify key factors influencing loan default risks and develop robust classification models to assist in the decision-making process for granting or denying loans. This process also incorporated cost-sensitive metrics to optimize profitability.

**Procedure and Significance:**

Step 1: Data Loading and Initial Inspection

-Process:

- Imported libraries (`pandas`, `numpy`) for data manipulation.

- Loaded the dataset (`SBAnational.csv`) into a DataFrame.

- Explored data using `.info()` and `.head()` to identify column types, missing values, and dataset structure.

-Outputs:

- Dataset with 899,164 rows and 27 columns, including fields like `ApprovalDate`, `MIS\_Status`, and `DisbursementGross`.

- Observed inconsistencies such as dates stored as strings and numerical data stored as objects.

- Identified missing values across multiple columns.

- Significance:

- Provided a foundational understanding of the dataset.

- Highlighted preprocessing needs such as type conversion and handling missing data.

Step 2: Handling Date Variables

- Process:

- Converted date columns (`ApprovalDate`, `DisbursementDate`, `ChgOffDate`) to datetime format for consistency.

- Dropped these columns after confirming they were not directly relevant for prediction.

- Outputs:

- Uniform date formats ready for time-based analyses if needed.

- Significance:

- Simplified data by removing irrelevant columns.

- Ensured temporal features could be extracted if required for future analyses.

Step 3: Encoding Categorical Variables

- Process:

- Applied label encoding for variables such as `State`, `MIS\_Status`, `LowDoc`, and `RevLineCr`.

- Mapped specific categories:

- `MIS\_Status`: Converted 'CHGOFF' to 1 (default) and 'P I F' to 0 (paid in full).

- `FranchiseCode`: Transformed to binary (franchise vs. non-franchise).

- Outputs:

- Encoded categorical variables into numeric formats for modeling.

- Significance:

- Facilitated the use of categorical data in machine learning algorithms.

- Retained critical information necessary for predicting loan outcomes.

Step 4: Cleaning Financial Data

- Process:

- Removed non-numeric characters (e.g., `$`, `,`) from financial fields like `DisbursementGross`, `BalanceGross`, and `ChgOffPrinGr`.

- Converted these fields to floating-point numbers.

- Outputs:

- Cleaned financial metrics representing loan amounts and balances.

- Significance:

- Enhanced data accuracy for analysis and modeling.

- Addressed issues caused by mixed data types in critical fields.

Step 5: Standardization of Numerical Features

- Process:

- Standardized numerical features (e.g., `DisbursementGross`, `Term`) using `StandardScaler`.

- Outputs:

- Features standardized to a mean of 0 and variance of 1.

- Significance:

- Improved compatibility with machine learning models sensitive to feature scaling, such as Gradient Boosting and Neural Networks.

Step 6: Correlation Analysis

- Process:

- Computed correlations between numerical features and the target variable (`MIS\_Status`).

- Outputs:

- Identified strong predictors:

- `ChgOffPrinGr` (Charge-Off Principal): Strong positive correlation with loan default.

- `Term`: Longer terms associated with higher default risk.

- Significance:

- Highlighted impactful predictors, guiding feature selection for modeling.

Step 7: Statistical Tests

- Process:

- Conducted:

- T-tests: Compared means of numerical features across default categories.

- Chi-square tests: Assessed associations between categorical variables (`LowDoc`, `RevLineCr`) and loan outcomes.

- Outputs:

- Significant predictors included `LowDoc` and `RevLineCr`, indicating strong associations with loan status.

- Significance:

- Provided statistical validation for the chosen features.

- Ensured meaningful inputs for predictive modeling.

Step 8: Final Dataset Preparation

- Process:

- Selected key features: `ChgOffPrinGr`, `Term`, `LowDoc`, `RevLineCr`, `UrbanRural`.

- Saved the cleaned dataset as `selected\_sba\_loans\_data.csv`.

- Outputs:

- Compact dataset optimized for modeling.

- Significance:

- Reduced dimensionality improved computational efficiency.

- Retained only impactful features for prediction.

**Model Development and Evaluation:**

Step 9: Model Implementation

- Developed classification models:

- kNN

- Decision Tree, Bagging, Random Forest, Boosting

- Logistic Regression (Lasso, Ridge, ElasticNet)

- Neural Networks

- Discriminant Analysis (LDA, QDA)

Step 10: Cost-Sensitive Evaluation

- Incorporated cost-sensitive metrics:

- Profit for "Paid in Full" loans: +5% of disbursed amount.

- Loss for "Default" loans: -5× disbursed amount.

Step 11: Validation Metrics

- Calculated accuracy, precision, recall, F1-score, and ROC-AUC.

- Visualized performance using gains and lift charts.

**Results and Conclusions:**

Key Findings:

1. Model Performance:

- Gradient Boosting and Random Forest achieved the highest accuracy (~99.4%).

- Logistic Regression retained interpretability but showed lower accuracy.

2. Cost-Sensitive Insights:

- Gradient Boosting provided the highest net profit by minimizing false negatives.

3. Optimal Threshold:

- A threshold of 0.25 maximized net profit while maintaining reasonable sensitivity and specificity.

Business Implications:

1. Best Model:

- Gradient Boosting emerged as the most effective model.

2. Key Predictors:

- Significant features included `ChgOffPrinGr`, `Term`, `LowDoc`, and `RevLineCr`.

Business Recommendations:

* Incorporate the Gradient Boosting model into the loan approval pipeline.
* Use the optimized threshold to classify loans as "high risk" or "low risk."
* Regularly retrain the model with updated data to ensure reliability.