# Credit Card Behavior Score - Documentation

## Title and Introduction

\*\*Title:\*\* Credit Card Behavior Score - Documentation

The purpose of this project is to develop a predictive model for credit card behavior scores, identifying customers who may default on their payments. This model aims to improve credit risk management by providing early warnings and actionable insights for better decision-making.

## Problem Statement

The goal is to create a model that predicts the likelihood of a customer defaulting (bad\_flag = 1) based on historical transaction and attribute data. The predictions will guide financial institutions in mitigating risks and optimizing credit strategies.

## Data Description

- \*\*Development Data (dev\_data):\*\* Contains 1216 columns, including the target variable (`bad\_flag`) and independent features such as `onus\_attribute\_1` and `transaction\_attribute\_1`. There are 96,806 records in this dataset.

- \*\*Validation Data (validation\_data):\*\* Includes 1215 columns, excluding the target variable, with 41,792 records.

## Approach

### Steps Followed:

1. \*\*Data Loading:\*\* Imported datasets for development and validation.

2. \*\*Exploratory Data Analysis (EDA):\*\* Identified missing values, feature distributions, and correlations.

3. \*\*Data Preprocessing:\*\*  
 - Handled missing values using the median imputation strategy.  
 - Standardized numerical features to ensure uniform scaling.

4. \*\*Feature Engineering:\*\* Reduced dimensionality using Principal Component Analysis (PCA).

5. \*\*Model Development:\*\* Evaluated three algorithms: GradientBoosting, XGBoost, and LightGBM.

6. \*\*Validation:\*\* Assessed models using metrics such as ROC-AUC and selected the best-performing model.

7. \*\*Prediction:\*\* Generated predictions on the validation dataset.

## Algorithm Description

\*\*Algorithm Used:\*\* Gradient Boosting Classifier

\*\*Reasons for Selection:\*\*

- Effective for tabular data.

- Robust performance with structured datasets.

- Interpretability and support for hyperparameter tuning.

## Steps and Implementation

1. \*\*Data Cleaning and Preprocessing:\*\*  
 - Handled missing values using `SimpleImputer` (median strategy).  
 - Standardized features with `StandardScaler`.  
 - Applied PCA for dimensionality reduction to 100 components.

2. \*\*Model Training and Validation:\*\*  
 - Split the data into training and test sets (80%-20%).  
 - Evaluated GradientBoosting, XGBoost, and LightGBM models.  
 - Selected Gradient Boosting as the best model with a test ROC-AUC score of 0.7638.

3. \*\*Hyperparameter Tuning:\*\*  
 - Performed grid search for optimal parameters.  
 - Best parameters: `n\_estimators=100`, `learning\_rate=0.1`, `max\_depth=3`.

4. \*\*Prediction on Validation Data:\*\*  
 - Used the final model to predict probabilities for the validation dataset.

## Key Insights

- \*\*EDA Observations:\*\*  
 - Approximately 26% of key features had missing values.  
 - High feature sparsity in some attributes.  
 - Imputation and scaling improved model performance.

- \*\*Model Observations:\*\*  
 - Gradient Boosting performed better than XGBoost and LightGBM.  
 - Top contributing features included key `transaction\_attribute` and `onus\_attribute` variables.

## Evaluation Metrics

- \*\*ROC-AUC Score:\*\* 0.7638 (test data), indicating strong predictive power.

- \*\*Other Metrics:\*\* Evaluated accuracy, precision, and recall during development but prioritized ROC-AUC due to imbalanced classes.

## Predicted Results

- Predictions for the validation dataset were generated and saved in `validation\_predictions.csv`.

- The file includes `account\_number` and `predicted\_probability` columns.

## Conclusion

- The Gradient Boosting model demonstrated robust performance in predicting default probabilities.

- Recommendations include further feature selection to enhance interpretability and retraining with updated data for improved accuracy.

## Submission Details

- \*\*Documentation File:\*\* Behavior\_Score\_Documentation.docx

- \*\*Prediction File:\*\* validation\_predictions.csv

Both files are included in the submission ZIP for final delivery.