**Default Prediction: Exploring Traditional and Non-Traditional methods**

KAUSTAV NANDY

This project presents a comprehensive solution that employs two distinct approaches: **the traditional Logistic Regression mode**l and an **alternative approach utilizing the Random Forest algorithm**. The goal is accurate prediction of defaults, and studying the relative efficacy of non-traditional methods for this task. This document outlines the key steps and rationale behind each approach, highlighting the preprocessing, modeling, and evaluation techniques employed.

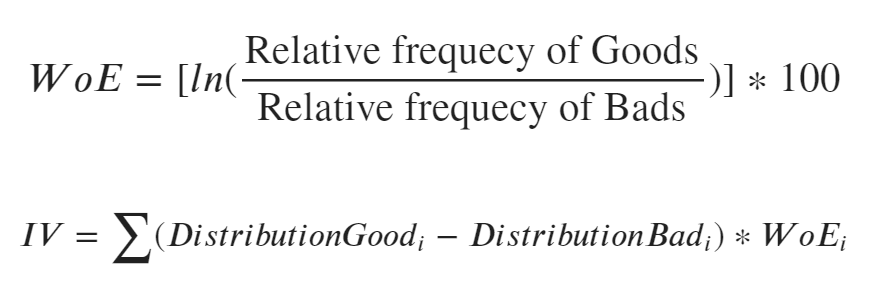
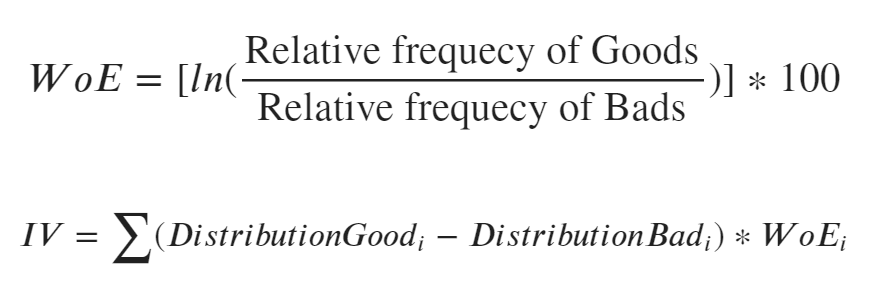
**Approach 1: Logistic Regression with WoE and IV Method**

This is the traditional and most commonly used model used for default prediction in the industry. The reasons for this are:

* Logistic Regression model offers the right blend of accuracy and interpretability, which is very important for explaining results to regulators.
* It is a simple model that performs really well as long as its assumptions are met and hence, is relatively less prone to overfitting.

**Preprocessing and Feature Selection**

To prepare the data for Logistic Regression, the Weight of Evidence (WoE) and Information Value (IV) methodology was used. This process helps establish monotonicity in the data, enhances model interpretability and mitigates the influence of outliers. The mathematical formula of WoE and IV is as follows (‘Bads’ mean defaults and ‘Goods’ mean non-defaults):

****

* The initial step involved the selection of features based on their Information Value (IV) scores, enabling the model to focus on the most predictive attributes.
* Continuous variables underwent discretization into bins, with their original values replaced by corresponding WoE values. The bins were then tweaked and merged until a monotonic relationship is established with the target variable.

**Modeling and Evaluation**

Utilizing the preprocessed dataset (done using scikit-learn pipelines), a Logistic Regression model was constructed. Evaluation encompassed a range of performance metrics:

**ROC-AUC and Gini Coefficient**: An indicator of the model's ability to distinguish between positive and negative classes.

**F2-Score**: A metric accounting for the asymmetric nature of default prediction, placing greater emphasis on minimizing false negatives, which have a higher cost in case of default prediction.

**ROC and Precision-Recall Curves**: Visualizations illustrating the trade-off between true positive rate and false positive rate, as well as precision and recall.

**Approach 2: Random Forest with Permutation Importances**

A lot of features in the data were observed to have non-linear relationships with the target. Hence, a non-linear model was suitable for the data. Random Forest was chosen because it fits this criteria without the risk of overfitting and also offers some level of interpretability.

**Preprocessing and Feature Selection**

The Random Forest approach involved utilizing permutation importance to identify the most influential features. This method gauges the reduction in model performance when a specific feature's values are randomly shuffled, indicating its impact on predictive accuracy.

* A new feature called 'random\_Col' with random values was introduced in the feature set. This feature is expected to have a very low importance score and acted as a benchmark feature.
* Any feature with a mean feature importance score less than the maximum score for the random feature was deemed to have an insignificant effect on the target and was dropped.

This idea is borrowed from the Boruta technique and applied here in a simpler manner.

**Modeling and Evaluation**

Leveraging the selected features, a Random Forest model was constructed. Evaluation procedures mirrored those of the Logistic Regression approach, encompassing ROC-AUC, F2-Score, Gini Coefficient, and the visualization of ROC and PR Curves.

**Results**

For the Logistic Regression model, we observed:

* AUCROC OF **0.73**
* Gini of **0.47**
* F2 score of **0.68**

The optimal decision threshold for the model was **0.47**.

All of these scores are acceptable, but could possibly be improved if the data is explored further in order to perform more feature engineering and find other ways to improve the model.

For the Random Forest model, we observed:

* AUCROC OF **0.77**
* Gini of **0.54**
* F2 score of **0.71**

The optimal decision threshold for the model was **0.57**.

These numbers indicate decent performance of the Random Forest model. **It performed better than the traditional Logistic Regression model!** Hence our alternative approach was successful.

**Possible Improvements**

The above results were arrived at under the constraints of limited time and data (both in terms of sample size and feature count). Upon sufficient availability of both, the following steps could have improved the models:

* Given more time for EDA, more **feature engineering** could be performed so that the models could be fed information-rich forms of the features instead of the raw features. For example, most of the statistics in the feature set are recorded at the last 30 days, 60 days, 180 days, lifetime etc. levels. We could have attempted to see whether the percentage change of these statistics from the last recorded time interval gives us any information and engineered new relevant features accordingly.
* Upon more EDA, we could have studied the **feature interactions**. This would have particularly benefited the Logistic Regression model which cannot detect such interactions by default. However, explicit feature engineering to capture any such interaction would have significantly boosted the model. Although Random Forest model can detect feature importances by itself, such feature engineering would have helped our Random Forest model to some extent as well.
* We see that many features have non-linear relationship with the target. Weight-of-Evidence method deals with this to a certain extent but it also loses a lot of information in the data. An alternative way to deal with this would be the use of **polynomial features, basis expansion** and **splines**, all of which can model non-linearity. This would specifically help the Logistic Regression model.
* We had to reject a lot of the features due to multicollinearity. However, they might have contained some exclusive information. We could have performed **Principal Component Analysis (PCA)** on the features (for example, among the mobile-device-related features) in order to preserve any exclusive information in each feature.