**3.0.0.0. Phase 3: Modeling and Error Analysis**

In phase 2 EDA was performed and data preparation was done. In this phase the data prepared in phase 2 shall be used for training machine learning models. Various machine learning models will be trained and results will be analyzed to find the best model. Best model will be further tuned to improve the result. Lastly, error analysis will be performed.

The ipynb file corresponding to this phase is divided into sections. This phase 3 documentation has to be read along with the ipynb file for correlation of different terminologies and outputs. The description given here follows the same section wise approach as the ipynb file. All the relevant files can be accessed through the following link: https://drive.google.com/drive/folders/1evFZRwFWh4zkR9CiT46llB9PlaXFLfLA?usp=sharing

**3.1.0.0.** Common commands: The following actions are performed in this section:

3.1.1.0. Google Drive is mounted for accessing data files.

3.1.2.0. Relevant packages are installed.

3.1.3.0. Relevant libraries are imported. It may happen that some packages and libraries are not used. However, they appear because they were used in the process of development of the final ipynb file.

3.1.4.0. One custom function for dataframe optimisation is defined. Data sets created in the previous phase are imported and their shapes are printed.

3.1.5.0. Data sets (both train and test) are quite unbalanced. So, upsampling is performed on both data sets with all features and dataset with selected features. This will be used for model training and to conclude any advantage is drawn from upsampling or not.

**3.2.0.0.** Model comparison:

3.2.1.0. Model comparison is performed using a package named Pycaret. Pycaret is an easy to use package giving options to create a pipeline for data preprocessing, train & compare model and deploy model. There are other features too such as PCA, hyperparameter tuning etc.

3.2.2.0. Three models namely logistic regression, random forest and light GBM are compared for four different conditions:

- Model comparison with selected features and without upsampling
- Model comparison with selected features and with upsampling
- Model comparison with all features and without upsampling
- Model comparison with all features and with upsampling.

Thus we are training 12 models.

3.2.3.0. Column names for the datasets are modified to suit the acceptance of pycaret. As preprocessed train and test data are already available, the same data sets are used in the setup of data.

3.2.4.0. Based on the results of comparison, the following are the observations:

- Upsampling does not give good results. Hence, this strategy will be dropped henceforth.
- Light GBM gives best results with feature selected data without upsampling. This model will be further tuned for best results.
- Feature selected data shall be used for model tuning and further training.

**3.3.0.0.** Tuning the best model:

3.3.1.0. Based on the observations of the previous section, lightgbm is tuned for feature selected data without upsampling. Tuning is performed with two conditions - Tuning while optimizing for accuracy, and Tuning while optimizing for AUC. Both conditions give the same accuracy and AUC for test data.

3.3.2.0. As the results for 2 conditions are the same, the model with AUC optimized is selected for further tasks. We refer to this model as best_model_auc.

3.3.3.0. It is observed that the accuracy of this tuned model is best among all the models trained so far. However, the AUC is slightly lower than the untuned lightgbm for data with features selected and without upsampling (saved in section 2.1 of phase 3 .ipynb file as best_model_feature_selected). A final comparison is done between tuned and untuned lightgbm models based upon the confusion matrix in the next section.

**3.4.0.0.** Comparison of confusion matrix and final selection of model:

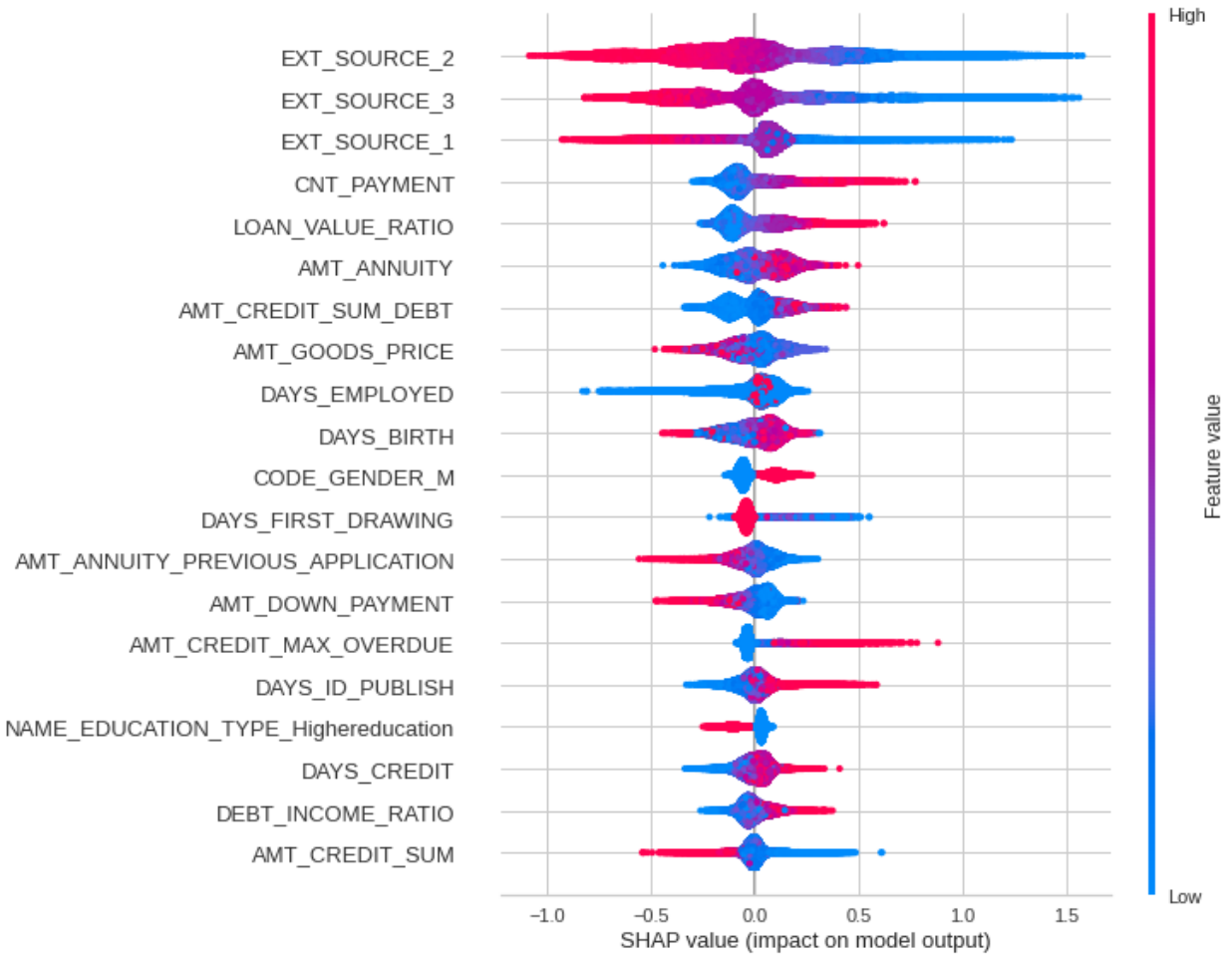
3.4.1.0. In this section confusion matrices are drawn for results obtained on test data using two models - untuned lightgbm and tuned lightgbm.

3.4.2.0. It is observed that the number of applicants who should not be given loan but are predicted as eligible for loan (i.e., the count of the 3rd quadrant) is lower in the tuned model. Hence, tuned ligbm is selected as the best model based on which error analysis will be performed in the next section. Here after this tuned lightgbm model will be named **best_model**.

**3.5.0.0.** Error Analysis

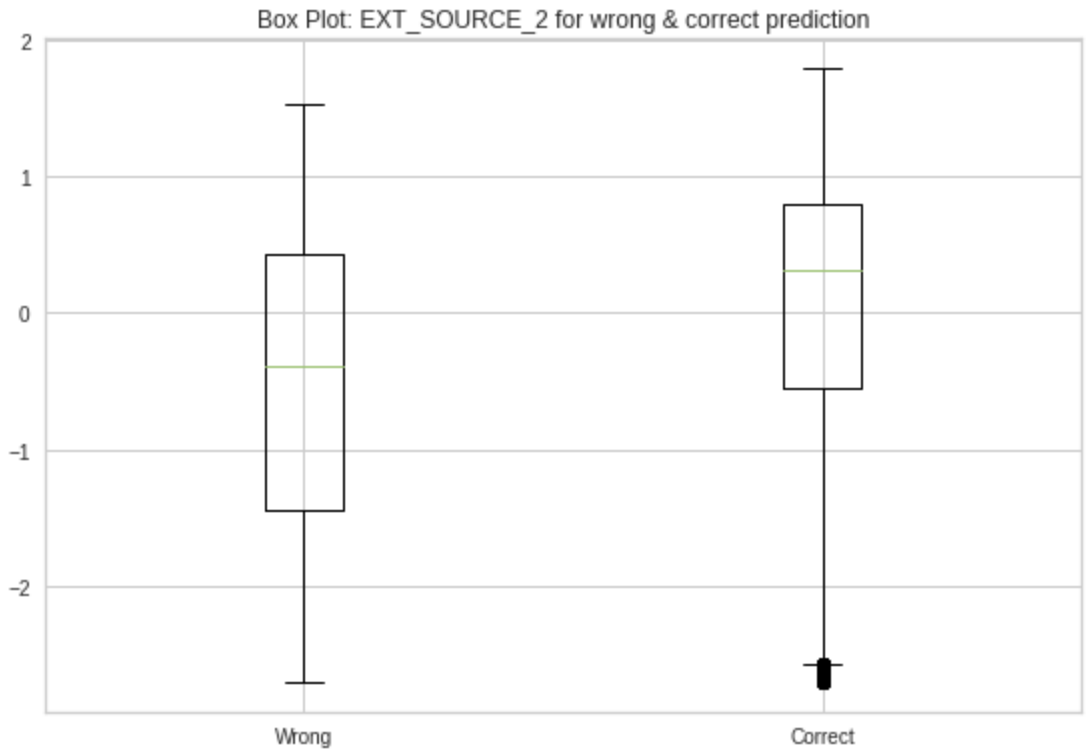
3.5.1.0. At the time of execution of this section of code, the saved tuned lightgbm model was throwing an error. Hence, lightgbm was again trained with tuned parameter values. Thus the model trained earlier and the one being trained currently will have the same weights and same results. Predictions are made on test data and the predicted values are saved for further use.

3.5.2.0. SHAP (from pycaret model interpretation) is used for getting the top 20 features. Top 3 features are EXT_SOUCE_2, EXT_SORCE_3 and EXT_SOURCE_1. SHAP interpretation from the ipynb file is reproduced here.

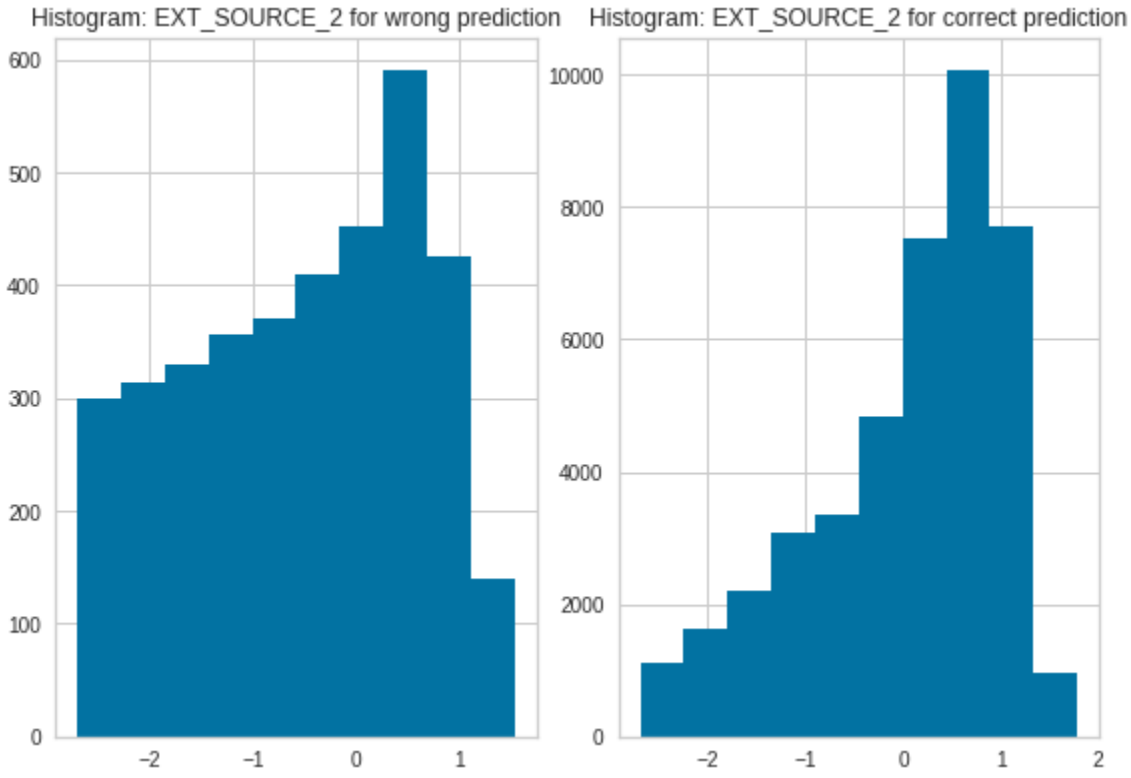



AUC is also plotted for best_model just for visualization. Lime is used for getting top 3 features for 10 wrongly predicted data points. Lime is also used for getting top 3 features for 10 correctly predicted data points. Thus it is observed that EXT_SOUCE_2, EXT_SORCE_3 and EXT_SOURCE_1 are the top 3 features for both wrongly predicted and correctly predicted data points.

Box plots and histograms are made for top 3 features for wrongly and correctly predicted points. Sample box plots and histograms from the ipynb file are reproduced here.




Sample Box plot




Sample Histogram

3.5.3.0. Observations:

- Top 3 features influencing wrongly and correctly predicted points are the same. They are EXT_SOUCE_2, EXT_SORCE_3 and EXT_SOURCE_1.
- Spread of data points for these 3 features for wrongly predicted data points is lower than correctly predicted data points.
- Statistically data is found in the same scale (Standard scaling was performed on data during phase 2) as observed in the box plot.
- From the histograms, it is observed that outliers are more pronounced for correctly predicted data points.
- It can be concluded that features are not having an impact on prediction as top 3 features for wrongly and correctly predicted data points are the same.
- Plots for the top features don't give any conclusive interpretation for predictions.
- It calls for introduction of more features through advanced feature engineering. Advanced feature engineering will be performed in the next section.