Approach for Credit Card Default Challenge

Yeeshukant Singh | 200002082

Problem Description

Based on the data given your job is to predict whether an account has a risk of default or not, by analyzing various details of the account holder. A label of 1 signifies that the account has a definite risk of default, whereas a label of 0 denotes low/no risk accounts.

Approach

- Collecting Data
- Checking for Null/NaN values and getting overview of data
- Prepare training data and label data
- Encoding categorical features and scaling for faster processing.
- Dropping non useful features based on Mutual info. score
- Splitting data for Training and testing
- Training and Evaluation
- Preparing Final Submission Testing data to be same as training data and finally final submission.

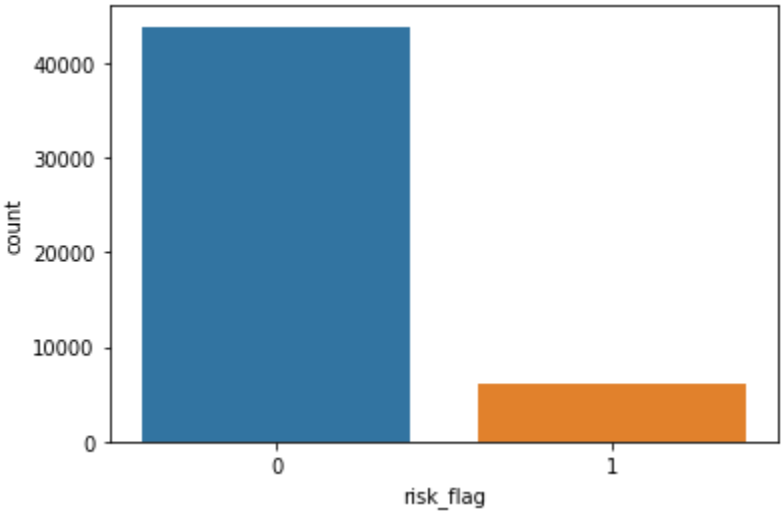
Workflow

- Created dataframes of training dataset and testing dataset.
    - Training dataframe (50000,12) (excluding label column)
    - Testing dataframe (20000,12)
    - Features
        - Id (integer)
        - Income (integer)
        - Age (integer)
        - Experience (integer)
        - Married (string)
        - House ownership (string)

        - Car ownership (string)
        - Profession (string)
        - City (string)
        - State (string)
        - Current job years (integer)
        - Current house years (integer)
    - Label
        - Risk flag (0/1)
- Checked for any null values, but there was none.
- Percentage of 0 label = 87.716

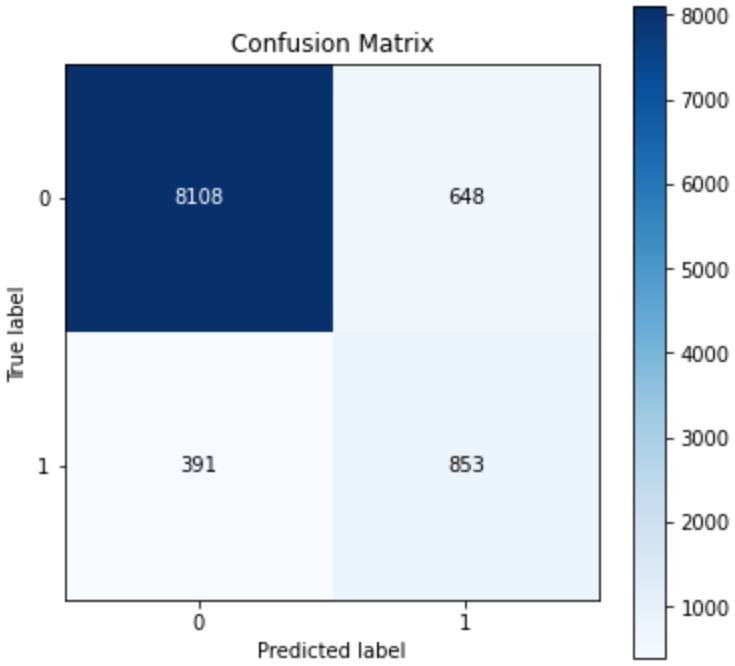
  1 label = 12.284

  Hinting to imbalanced data.
- Following is the count plot for risk flags

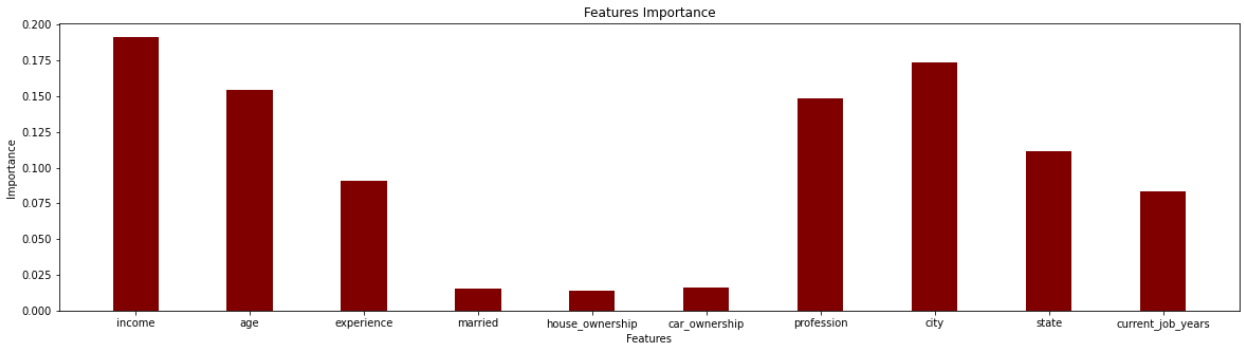


- Categorical columns to be encoded :
    - ['married','house_ownership','car_ownership','profession','city','state']
- Encoding method used : Applying column-wise Hash function
    - Reason
        - No new columns formed
        - Easy unknown value handling
- Using Standard Scaler scaling features for faster processing.
- Finding Mutual information scores to figure out less useful features.

    - Current house years was one with very low mutual information score and less variance, so this feature could be dropped without any loss.
- Final features :
    - Income (integer)
    - Age (integer)
    - Experience (integer)
    - Married (integer)
    - House ownership (integer)
    - Car ownership (integer)
    - Profession (integer)
    - City (integer)
    - State (integer)
    - Current job years (integer)
- Splitting the data for training and testing in 80:20 ratio using train test split
    - Training set (40000,10)
    - Testing set (10000,10)
- Best results were obtained from Random forest Classifier
    - Important parameter changes :
        - N_estimators : 150
        - Class Weight : “Balanced” (owing to unbalanced training set)
- Training model
- Performance estimation using ROC_AUC score and precision scores using test set
- Following is the confusion matrix obtained from the model on the test set.



- Following is the feature importance graph as estimated by the model.



- Final model score (on testing part of whole dataset):
  - Precision : 90.60
  - Recall : 89.61
  - ROC AUC : 92.45
- Preparing Testing data for submission using same operations as used for final training set which are:
  - Dropping column “Id” and “Current house years”
  - Encoding categorical features using Hash function
  - Scaling data column-wise

- Making predictions and converting to csv file for final submission.

==========