Report: TVS Epic Analytics Case Study-1

By- Kunwarvir Singh

Problem Statement Overview

TVS has recently launched the services of ***Personal Loans (PL)*** for the customers who are in urgent need of finances for their personal requirements. However, there is a ***risk factor*** associated while offering a personal loan to an interested customer. So, the ***main objective*** of the given problem statement is to identify the segment of customers, who have a ***higher tendency to default***, if they are offered a Personal Loan. This would be useful in ***mitigating the risk*** associated with the riskiest set of customers and offer Personal Loan to only less risky customers.

Thus, ***using appropriate Analytics and Data Science techniques***, I have come up with ***Supervised Machine Learning*** model delivering acceptable accuracy for assessing the risk associated with customers who desire to obtain Personal Loans (PL) from TVS.

Documents and Deliverables attached along with the Documentation: -

1. Brief understanding of ***Case Study – 1*** with Data Dictionary (*PDF Format – Provided By TVS*).
2. Labelled Dataset for Training (*CSV File – Provided by TVS*)
3. **Code File - IPython Notebook (***IPYNB Format – Code file involved in training and evaluating the Machine Learning Model***)**

Coding Platform details: -

1. Language Used: *Python 3.8*
2. Platform: *Jupyter Notebook*

Approach: -

The main steps involved in assessing and evaluating the given Problem Statement (Case Study – 1) are described as follows: -

1. Importing the Dataset and required libraries for Data Pre-Processing.
2. Data Pre-Processing and Feature Selection
    - Encoding the Categorical features of Dataset
    - Dropping the unwanted features from the dataset (Ex Customer ID, Pin Code, Date of Birth etc.)
    - Dropping the features from the dataset having more than 80% data Missing.
    - Evaluating correlation between Dependent (V30) and all the independent variables form the dataset.
    - Dropping the features from the dataset having least significance based on correlation.
    - Updating the Training data with Significant Features only
3. Defining dependent and Independent variables
4. Splitting the dataset into Train Dataset and Test Dataset (75% Training Data, 25% Test Data)
5. Feature scaling of independent variables to avoid errors during training of machine learning model.
6. Fitting the Pre-Processed dataset into the following classification models and evaluating the metric score (Accuracy) of each model: -
    - Random Forest
    - Decision Tree
    - Gradient Boosting
    - Gaussian Naïve Bayes
    - Logistic Regression
7. Model with the *best metric score (Accuracy)* is finally used for assessing whether the customer who desire to obtain loan from TVS is risky (Bad/1) or not (Good/0).

Expected Benefit: -

- The best metric score achieved is 0.9787 i.e., ***accuracy of 97.87 % using Gradient Boosting Algorithm***. Thus, the model is ***highly accurate*** in identifying the risky and non-risky customers.
- The training model can be ***deployed on Real Time platforms*** to immediately identify the defaulters and prevent company for Approving Personal Loans to such customers easily.

Algorithms Used: -

1. **For feature selection** we used ***correlation*** as the criteria to identify significance of the features in training the machine learning model and involving only those that has high significance. Using all the features of dataset for training purpose may be time consuming and may produce less accuracy. ***The features that were finally involved for training purposes based on correlation are: -***

Features Selected	Correlation (w.r.t Dependent Variable V30)	Remarks
V30	1.000000	Target variable (1: Bad Customer / 0: Good Customer)
V26	0.142721	Number of times defaulted in last 6 months
V25	0.141551	Number of times defaulted in last 3 months
V27	0.139536	Number of times defaulted in last 12 months
V5	0.071633	Number of bounces with TVS Credit
V3	0.058623	Number of bounces in last 3 months Outside TVS Credit
V19	0.027094	Number of Live loans
V24	0.021416	Number of enquiries
Employment_Type_SELF	0.020701	Employment type of customer SELF: Self-employed
Gender_MALE	0.015864	Gender: Male
V2	0.012339	First EMI Bounce (0: No, 1: Yes) (existing loan)
Employment_Type_HOUSEWIFE	-0.010992	Employment type of customer HOUSEWIFE
V23	-0.011540	Number of closed loans
Gender_FEMALE	-0.015802	Gender: Female
Employment_Type_SAL	-0.016787	Employment type of customer SAL: Salaried,
V22	NaN	Number of new loans taken in last 3 months

2. **Classification algorithms:** The machine learning model was trained on ***5 Classification Algorithms*** and the model delivering best accuracy will be considered for deployment. The 5-classification algorithm used for training the machine learning model are as follows: -
   - *Random Forest Algorithm* – Advanced Ensemble Algorithm
   - *Decision Tree Algorithm* – Primitive Ensemble Algorithm
   - *Gradient Boosting* – Algorithm based on advanced ensemble techniques
   - *Gaussian Naïve Bayes (GNB)* – Probabilistic Classifier Algorithm based on Bayes Theorem
   - *Logistic Regression* – Most basic algorithm for Binary Classification (0/1).

Final Evaluation Metric: -

In order to evaluate all the above algorithms, pre processed data is trained over all the classification algorithm separately and Accuracy is calculated using Confusion Metrics.

Accuracy = 100*(True Positive +True Negative)/Total

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

The following Metric Scores were obtained for different classification algorithms:

```
RandomForest : 0.9773107556388462
DecisionTree : 0.9770430359413694
GradientBoosting : 0.9787162840505991
GNB : 0.9138946522990429
LogisticRegression : 0.9782477745800148
```

Examining the accuracy score of each classification algorithm, I found the GRADIENT BOOSTING algorithm delivers best results with least time taken to train the machine learning model. The accuracy obtained using GRADIENT BOOSTING algorithm is 97.87%

****