Good Device / Bad Device

Tuhin Sharma

Outline

- Following notebooks are to be run sequentially:-
    - Data Cleaning (Data Cleaning.ipynb)
    - Model Selection (Model Selection.ipynb)
    - Train the Model (Training.ipynb)
    - Predict the Test data (Prediction.ipynb)

1. Data Cleaning

- Total number of Training records 3722.
- All the columns have standard deviation more than 0.001.
- Total 32 columns have None values including the decision column (‘Machine_State’)
    - 31 Independent columns have 56 None values and all happen to be for the same 56 records.
    - Decision column has 19 None values and those are for the other 19 rows
    - This makes a total of 75 rows having at least one None values.
- Those 19 rows containing None values for the Decision Column Field are dropped.
- The cleaned dataframe is stored as “./data/cleaned_training_data.csv" havinf 3703 records.

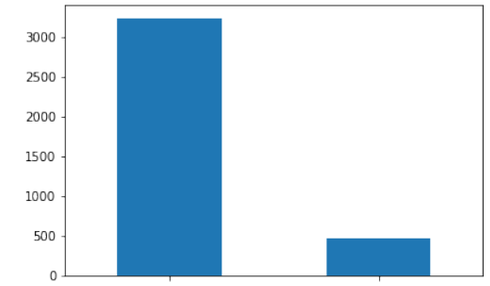
2. Model Selection

- Feature Selection
- Model defined
- Evaluation

Feature Selection

- Data is having missing data (columns having None values). So Transformer Imputer is used to impute missing data.

- Data is having class imbalance problem. so SMOTE technique is used for over-sampling. Number of records before SMOTE sampling 3703. Number of records after SMOTE sampling 6480.



- StandardScaler is used to scale the respective values of individual columns.

- LabelEncoder is used to encode the categorical data to numeric data for the decision column (‘Machine_State’)

- Outliers are removed from the data using LocalOutlierFactor with a neighbour count of 20. Number of records before removing outliers 6480. Number of records after removing outliers 5832.

- Dimension of the data is 219 which is quite high and it has collinearity problem. For making the decision stable PCA is used for dimensionality reduction and making every dimension orthogonal to each other. 60 attributes are chosen as for remaining attributes the contribution towards variation of the data is negligible (order of exp(-4)).

- RandomForest is used to select suitable attributes from these 60 attributes. And finally 20 attributes are chosen.

Model Defined

- Following models are defined for evaluation:-
    - Linear Discriminant Analysis
    - LogisticRegression
    - Quadratic Discriminant Analysis
    - Random Forest Classifier
    - Adaboost Classifier (Boosting)
    - Naive Bayes Classifier

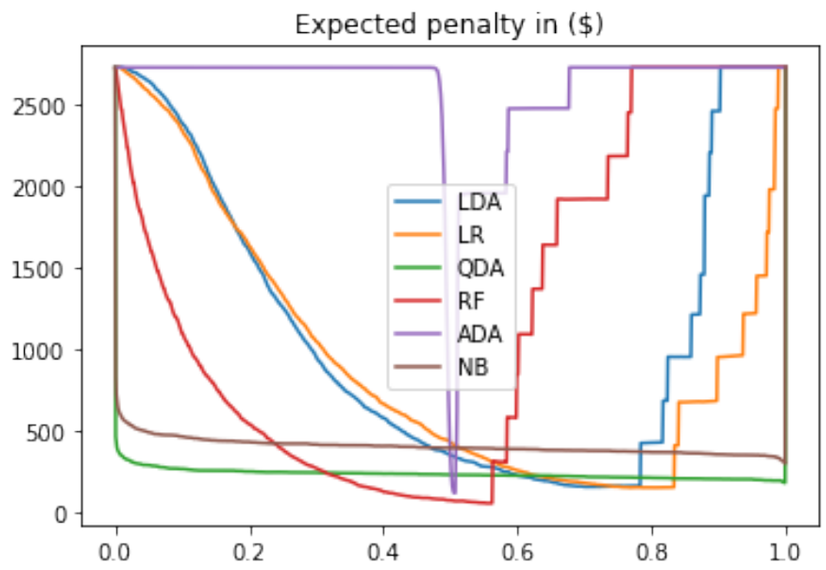
Evaluation

- A generic expected value framework is implemented which will give the expected cost in $ for individual models corresponding to given FP cost and FN cost. TP cost and TN cost does not play any role here so they are made default 0.
    - Expected value (EV) is calculated as follows,
        - **EV = PT*(TPR*tp_penalty+FNR*fn_penalty)+PN*(TNR*tn_penalty+ FPR*fp_penalty)**
            - PT = Probability of 1 in the actual decision.
            - PN = Probability of 0 in actual decision
            - TPR = True positive ratio
            - FNR = False negative ratio
            - TNR = True negative ratio
            - FPR = False positive ratio
            - tp_penalty = penalty for true positive and tn_penalty = penalty for true negative (0 by default)
            - fp_penalty = penalty for false positive and fn_penalty = penalty for false negative
- K-cross validation approach (k=10) is adopted for the evaluation purpose. The nature of K-cross validation output graph closely resembles the nature of test metrices (not by magnitude). So suitable threshold for minimum cost can be obtained very easily.
- For each of the model, for 1111 threshold values (n_threshold = 1111) the EV is averaged for k=10. And the threshold corresponding to the lowest EV is shown.

Case 1: Evaluation (contd.)

- Cost (Bad is passed as Good) = Cost(FP) = fp_panalty = $5000
- Cost (Good is detected as Bad) = Cost(FN) = fn_panalty = $500
- Observation:-
    - Random Forest outperforms other models by a huge mergin (average cost of $59.93 per observation) for a threshold value of 0.56.

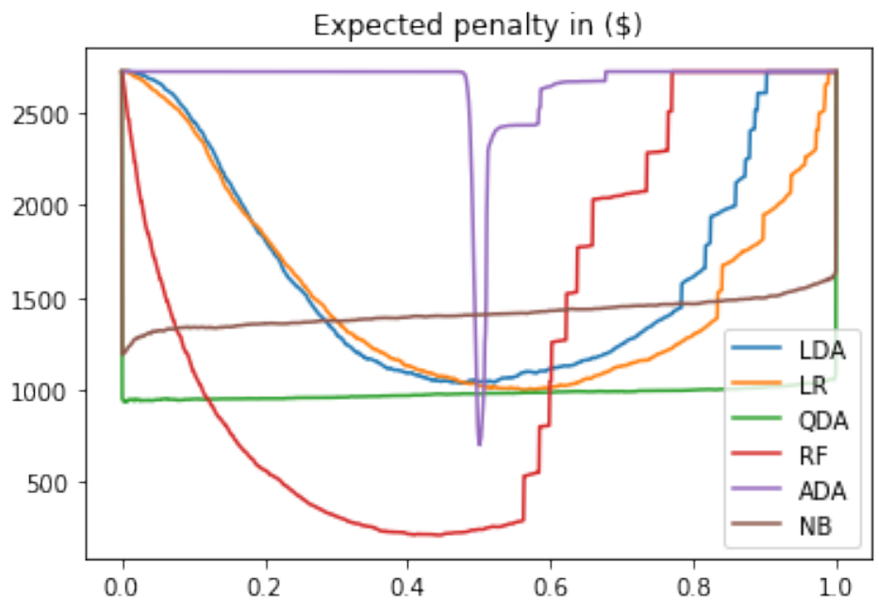
classifier	EV ($)	Threshold
Linear Discriminant Analysis	159.724735074	0.705405405405
Logistic Regression	154.666463028	0.809009009009
Quadratic Distriminant Analsyis	184.497697314	0.999099099099
Random Forest	59.9329753989	0.561261261261
AdaBoost	122.773532038	0.505405405405
Naive Bayes	301.438150567	0.999099099099



Case 2: Evaluation (contd.)

- Cost (Bad is passed as Good) = Cost(FP) = fp_panalty = $5000
- Cost (Good is detected as Bad) = Cost(FN) = fn_panalty = $5000
- Observation:-
    - Random Forest outperforms other models by a huge mergin (average cost of $212.62 per observation) for a threshold value of 0.44.

classifier	EV ($)	Threshold
Linear Discriminant Analysis	1036.52429568	0.475675675676
Logistic Regression	998.800195023	0.56036036036
Quadratic Distriminant Analsyis	936.191522357	0.0036036036036
Random Forest	212.622477032	0.444594594595
AdaBoost	701.326393947	0.500900900901
Naive Bayes	1190.81011067	0.0018018018018



3. Train the Model for Case 1

- A Random forest Model is trained on the whole training data after applying all the relevant transformers.

- The transformers (Imputer, LabelEncoder,StandardScaler, PCA, SFM) and the model itself (RandomForest Model) is stored for prediction.

4. Predict for Case 1

- All the stored transformers are loaded along with the random forest model.

- The transformers are applied to the Test data in the same order as of the training case.

- The Random forest model is applied to calculate the probability of getting selected as 1. A threshold of 0.56 is applied (as per slide no 8) to label the records as 1 and 0.

- The Label encoder transformer is applied to get back the original labels.

- The submission dataframe is thus created and saved as ‘./data/Submission.csv'.

THANK YOU