FOML HACKATHAN

Roll No: Ai21mtech14003 and Ai21mtech14007

Overview:

The objective of the competition is to find the given driver is default or not with the given input. The dataset given had contained categorical columns majorly and it also had got missing values. Therefore, pre-processing of data had played a key role in building the model.

Building the model:

- In the dataset, by visual observation and practical knowledge it was found that few columns didn't help in building the model, so we dropped from the dataset.
- The categorical data were then converted into numerical using one hot encoding and label encoding.
- The null values were replaced with max occurring value of the respected columns.
- To build a model many algorithms were selected and finally shortlisted to two algorithms (XGBoost and Random forest)
- After comparing the model performance, it was decided to use XGBoost.
- XGBoost is a library for developing fast and high performance gradient boosting tree models.
- Execution Speed improves
- To fine tune the model, Hyperparameter tuning was done using RandomizedGridSearch along with Crossvalidation.

The result from randomized search:

```
'min_child_weight': 1,
'max_depth': 8,
'learning_rate': 0.15,
'gamma': 0.2,
'colsample_bytree': 0.5
```

The final model is:

```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=0.5,
              enable_categorical=False, gamma=0.2, gpu_id=-1,
              importance_type=None, interaction_constraints='',
              learning_rate=0.15, max_delta_step=0, max_depth=8,
              min_child_weight=1, missing=nan, monotone_constraints='()
              n_estimators=100, n_jobs=8, num_parallel_tree=1,
          objective='binary:logistic', predictor='auto', random_state=0
,             reg_alpha=0, reg_lambda=1, scale_pos_weight=1, subsample=1
              tree_method='exact', use_label_encoder=True,
              validate_parameters=1, verbosity=None)
```

The train accuracy: 0.871334239658186
The test accuracy on kaggle: 0.86705