LABOR CONDITION APPLICATION DISCLOSURE

Analysis By: Anuj Verma

Depaul University

anuj.vermacom@gmail.com

# Project Goal

The goal for this project is to predict the case status of a Labor Condition Application (LCA) submitted by the employer to hire non-immigrant workers under the H-1B visa program. Employer can hire non-immigrant workers only after their LCA petition is approved. The approved LCA petition is then submitted as part of the Petition for a Non-immigrant Worker application for work authorizations for H-1B visa status.

Through this analysis, we want to uncover insights that can help employers understand the process of getting their LCA approved.

The Labor Condition Application (LCA) is a document that a perspective H-1B employer files with U.S. Department of Labor Employment and Training Administration (DOLETA) when it seeks to employ non-immigrant workers at a specific job occupation in an area of intended employment for not more than three years.

### Dataset Information

Source Link: <https://www.kaggle.com/trivedicharmi/h1b-disclosure-dataset>

Number of Attributes: 27

Number of Instances: 528134

#### Target Attribute

Our target attribute for this dataset is ‘CASE\_STATUS’. There are 4 categories for case status as given below:

1. Certified: Case approved by OFLC

2. CertifiedWithdrawn: Case withdrawn by employer post approval from OFLC

3. Withdrawn: Case withdrawn by employer

4. Denied: Case denied by OFLC.

# Methods Used

Below classification methods are used to achieve better results:

- Naive Bayes

- Linear Regression

- Decision Tree

- Random Forest Classifier

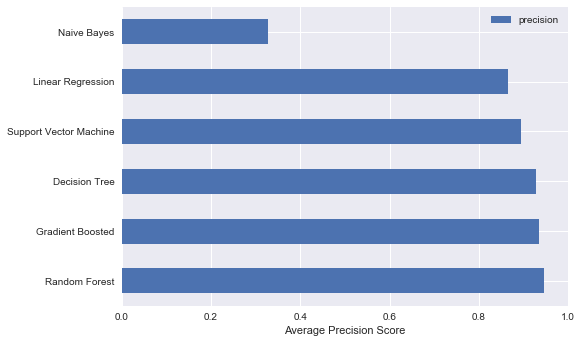
- Gradient Boosting Classifier

- Support Vector Machine

Please note that we are addressing a multiclass classification problem. Out of above listed models only SVM have built-in capability to handle multiclass classifications to some extent. Other than that, all other models works best for binary classification problem and will not produce good results for multiclass problem. We used advanced multiclass classification technique known as One-Vs-The-Rest Classification (also known as OneVsAll Classification) where only one classifier per class is fitted. For each classifier, the class is fitted against all the other classes. In addition to its computational efficiency (only n\_classes classifiers are needed), one advantage of this approach is its interpretability. Since each class is represented by one and only one classifier, it is possible to gain knowledge about the class by inspecting its corresponding classifier.

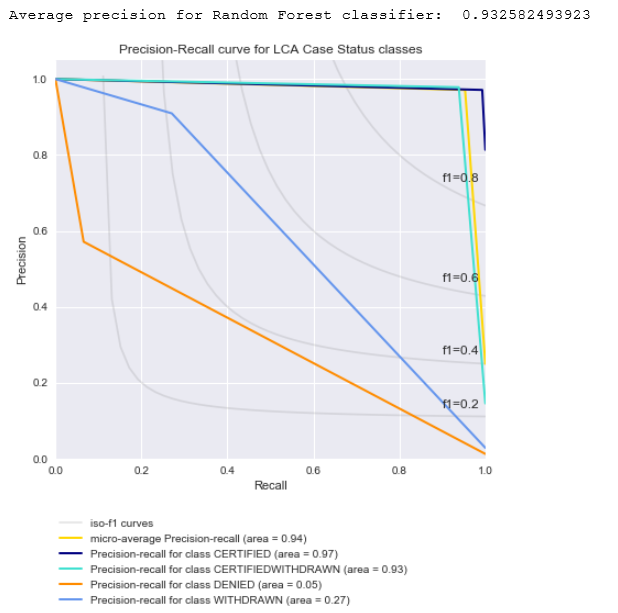
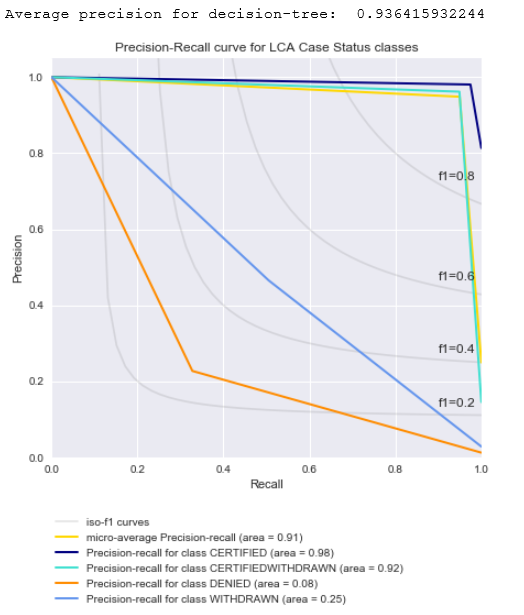
The Naïve Bayes model was run to determine the baseline performance and then other models were run to select the best model for hyper-parameter tuning.

### Model Performance

The models were compared using the average precision score from sklearn.metrics library. This method compares the classification problems using F1-score metric. Further, the ‘weighted-precision’ metric from F1-score was used to compare the precision of each class of target variable. This metric calculates the weighted average of F1-scores for each class of the target variable.

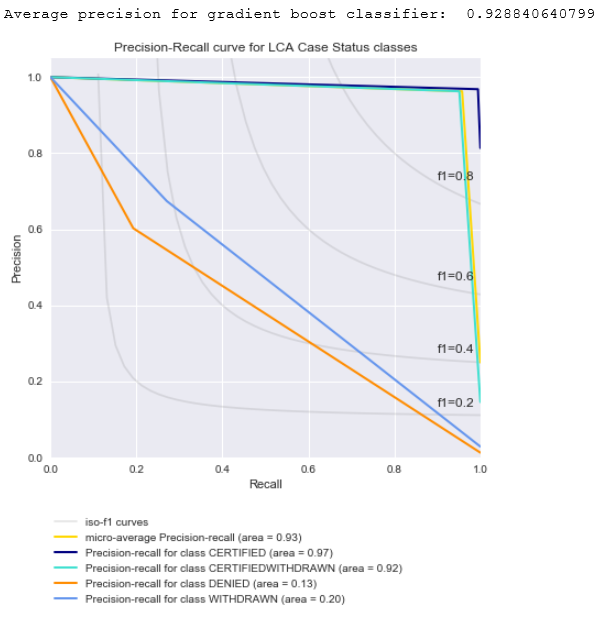
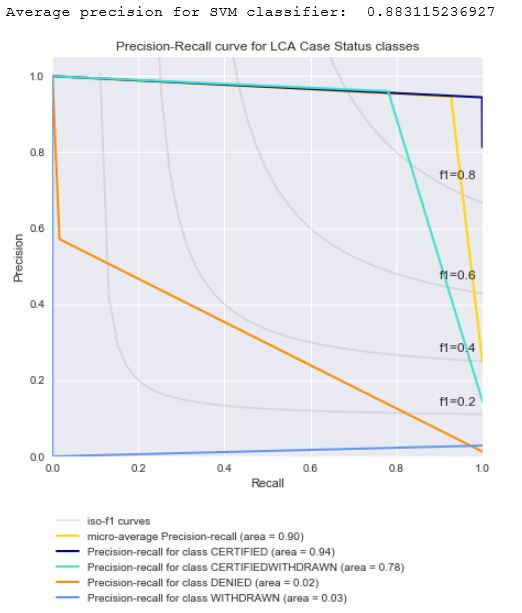
We can see from the figure that the Naïve Bayes performed worst of all the models and Random Forest, Decision Tree and Gradient Boosted classifiers performed similar with just 1% variance. The performance of the models can further be checked to classifying the different classes of target variable. These differences were plotted using precision-recall curve plot.

Random Forest Classifier Decision Tree Classifier

The above precision-recall curves, we can see that Gradient Boosted Classifier performed well in comparing all the classes of case\_status as compared to other models.

Gradient Boosted Classifier SVM Classifier

# Conclusion

In this analysis, our main goal was to be able to predict the case status of LCA applications that the employers submit to OFLC. We were dealing with the multiclass classification problem trying to predict the case status of four classes: [Certified, Certified-Withdrawn, Withdrawn, Denied]. We witnessed that our dataset was biased based on case status classes. We had more records for Certified, Certified-withdrawn classes and very few records for withdrawn and denied classes. We tried our best to preserve and correct the corrupt data as per our knowledge. As a result of this, our model accuracy came pretty high at 93-94%. These are good accuracy scores. However, our model was able to predict two most occurring classes really well and the other less frequent classes were not classified that well. We tried to improve our results by running the hyper-parameterization but that helped only little. At the end, we can conclude that our model can still be improved by trying even more rigorous hyper-parameterization. If this does not help then we can try adding more records of the less frequent classes to make our data unbiased.