Classification Projects on Machine Learning for Beginners - 2
Project Overview

Overview

In machine learning, Classification is one of the most widely used techniques with various applications. For sentiment analysis, spam detection, risk assessment, churn prediction, and medical diagnosis classification have served as a very simple yet powerful method. In this project, we aim to give you hands-on experience and theoretical explanations of various ensemble techniques. You can find the first project of this series here.

Aim

Understanding various Ensemble techniques and implementing them to predict license status for the given business.

Data Description

The dataset used is a licensed dataset. It contains information about 86K different businesses over various features. The target variable is the status of the license, which has five different categories.

Tech Stack

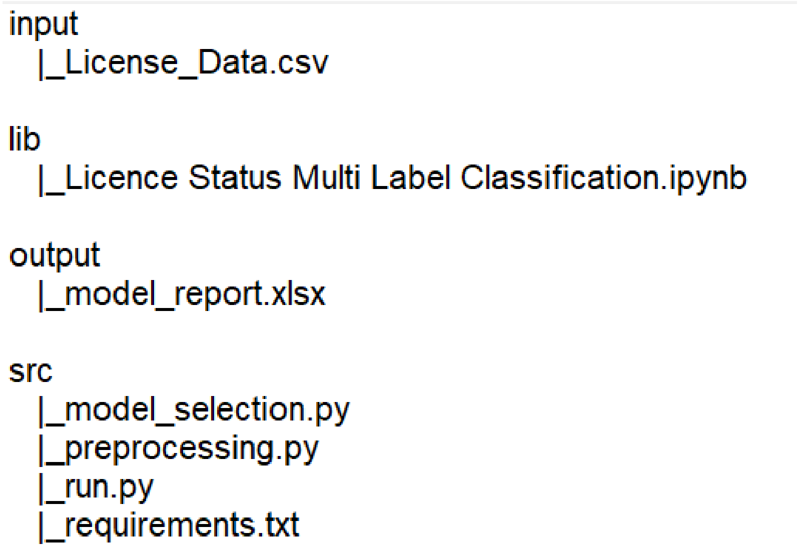
- ➔ Language: Python
- ➔ Libraries: pandas, scikit_learn, category_encoders, numpy, os, seaborn, matplotlib, hyperopt, xgboost

Approach

1. Data Description
2. Exploratory Data Analysis
3. Data Cleaning
    a. Missing Value imputation
    b. Outlier Detection
4. Data Imbalance

5. Data Encoding
6. Model Building
    a. Random Forest
    b. AdaBoost
    c. XGBoost
7. Feature importance
8. Hyperparameter tuning
    a. Random search optimization
    b. Grid search optimization
    c. Bayesian optimization

Modular code overview



```
input
  |_License_Data.csv

lib
  |_Licence Status Multi Label Classification.ipynb

output
  |_model_report.xlsx

src
  |_model_selection.py
  |_preprocessing.py
  |_run.py
  |_requirements.txt
```

Once you unzip the modular_code.zip file, you can find the following folders within it.

1. input
2. src
3. output
4. lib

1. The input folder contains the data that we have for analysis. In our case, it contains Licence_Data.csv.

2. The src folder is the heart of the project. This folder contains all the modularized code for all the above steps in a modularized manner.

The model_selection.py and preprocessing.py files contain all the functions in a modularized manner, which are appropriately named. These python functions are then called inside the run.py file.

The requirements.txt file has all the required libraries with respective versions. Kindly install the file by using the command pip **install** -r **requirements**.**txt**

3. The output folder contains an excel file for classification metrics scores of each model.

4. The lib folder is a reference folder. It contains the original ipython notebook that we saw in the videos.

Project Takeaways

1. What is Ensembling?
2. What is Bagging?
3. Understanding Random Forest model
4. Building Random Forest model
5. What are problems with bagging and how to overcome them?
6. What is Boosting?
7. Fundamentals of AdaBoost
8. Building AdaBoost model
9. XGBoost algorithm
10. Building XGBoost model
11. Understanding XGBoost hyperparameter Gamma
12. Understanding XGBoost hyperparameter Lambda
13. What is hyperparameter tuning?
14. GridSearch optimization
15. RandomSearch optimization
16. Bayesian optimization
17. Hyperparameter tuning for RandomForest model
18. Hyperparameter tuning for XGBoost model using hyperopt
19. Feature importance