INTRODUCTION TO MACHINE LEARNING

LEAF IDENTIFICATION

Poornima Devi Krishnasamy Karthikeyan – Data Science and Scientific Computing

Feb 23, 2023

OBJECTIVE:

The goal is to propose a method for leaf identification based on the provided leaf attributes from the given dataset. In this project, supervised machine learning techniques such as logistic regression, SVM and random forest classifiers are selected and implemented. They are further attempted to improvise their performance by various methods.

DATASET INFORMATION:

The dataset consists of 16 attributes with 340 rows and 16 columns. The columns include class, specimen number, eccentricity, aspect ratio, elongation, solidity, stochastic convexity, isoperimetric factor, maximal indentation depth, lobedness, average intensity, entropy, average contrast, smoothness, third moment, uniformity. Class is selected as the target variable. There are 36 unique classes within the class column, while there are 16 unique specimens within the specimen number column.

PERFORMANCE METRICS:

Evaluation metrics is necessary to quantify the performance of the solution model. For this project, precision, recall, F1 score, Macro averaged metrics, are taken into the account.

Precision: $Prec = TP\ /\ TP{+}FP$

Recall: $Rec = TP/\ P = TPR$

F1 Score: *F1 Score = 2* Precision* Recall / Precision + Recall*

Accuracy: $Acc = 1\ n \sum \mathbf{1}(\boldsymbol{y}\ (\boldsymbol{i}) = \hat{\boldsymbol{y}}\ (\boldsymbol{i}))$

Macro averaged: *macro-avg = (sum of metric for each class) / number of classes*

Weighted average: *weighted avg = (sum of (weight * value)) / sum of weights*

EXPLORATORY DATA ANALYSIS:

Data Cleaning: There are no missing values.

Missing values	Associated tasks	Attribute characteristics
Nil	classification (multiclass)	Integer, float

Data Visualization: Almost all the features in the dataset have skewed distribution which indicates outliers.

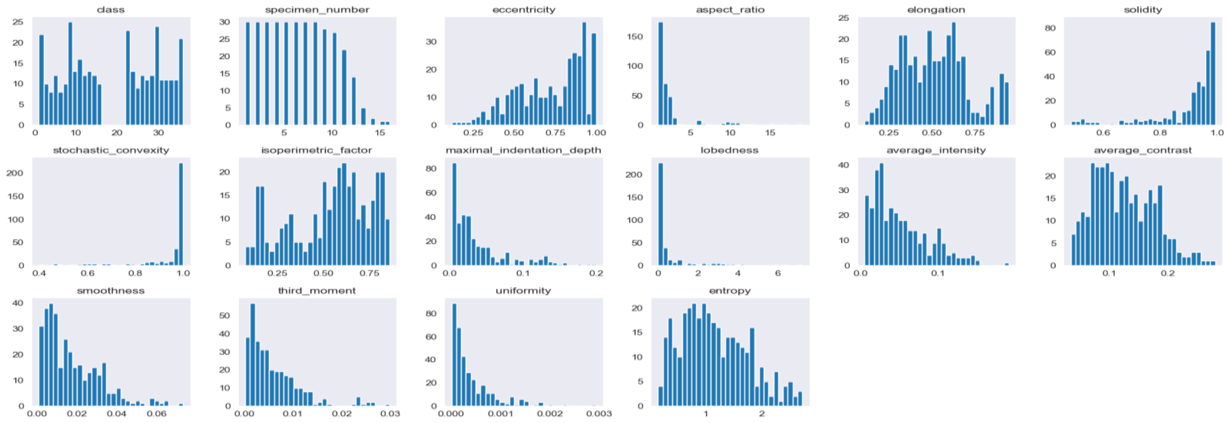


Figure 1 : Exploring data with distributions – Histogram

Skewness observation from the features: The eccentricity, isoperimetric factor, elongation, average intensity, average contrast, entropy is symmetrical in original dataset. The aspect ratio, lobedness, maximal indentation depth, smoothness, third moment, uniformity is skewed positive to the right highly whereas solidity, stochastic convexity is highly negatively skewed.

Outlier detection: Outliers can have a significant impact on statistical analyses and modelling, as they can distort the results and lead to inaccurate conclusions. Hence, they were treated further into the analysis.

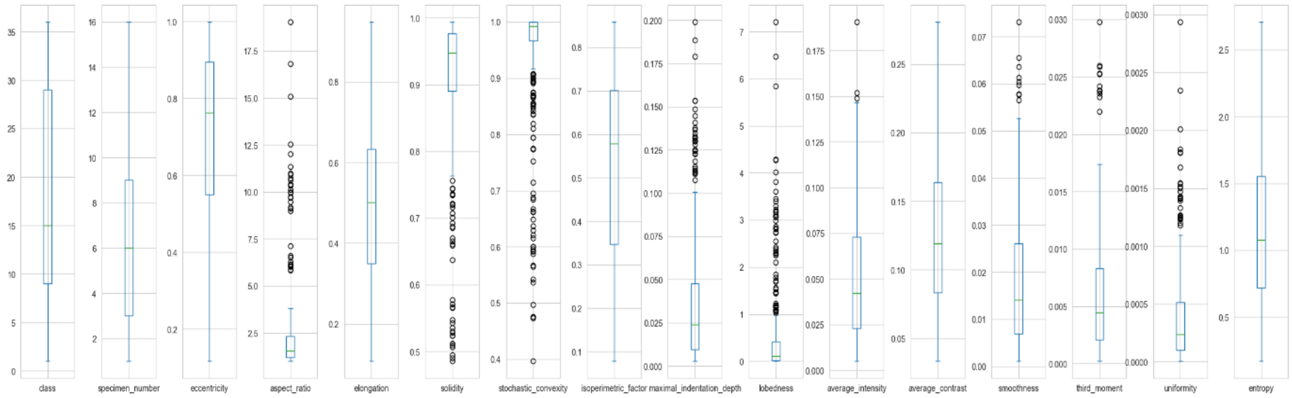


Figure 2 : Further, analysing the outliers using boxplot.

Quantile transformation:

:

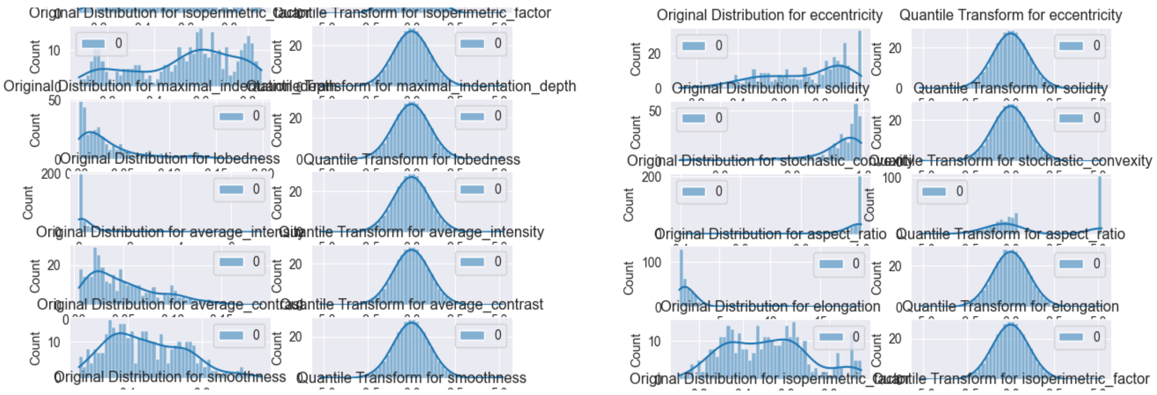
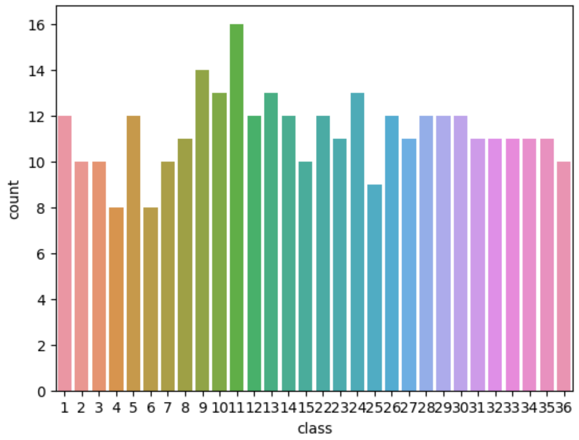
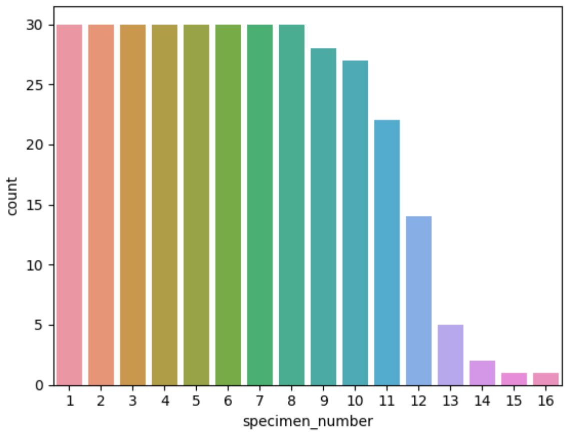


Figure 3 : Quantile transform of all the features.

Imbalanced dataset: The target variable is ‘class’ feature. When visualized the distribution of leaf in the class found to be unbalanced (it has more samples for some classes than others). This can make the classifier **biased** toward the one or two classes with lost of samples, while dwarfing others that have less (i.e. the classifier learns the classes with more samples better and remains weak on the smaller classes).





METHODOLODY:

The given dataset is very small. In machine learning, the size of the dataset plays an important role in determining the complexity of the model that can be used. When dealing with a small dataset, it is often better to use a simpler model rather than a complex one. The models are trained in a way to overcome the issue with imbalanced dataset.

Splitting the dataset into train and test: It is imported from the sklearn’s model_selection module to split our data into training and testing.

Logistic regression [Default]: When logistic regression is trained with default parameters, it performed not so great. In default, logistic regression in Scikit-learn assumes equal class weights, where the classes are given equal importance during model training. However, if one class is underrepresented in the training data, it may be beneficial to adjust the class weights to give the minority class more importance during training.

Improved Logistic regression with balanced class weights: Since, class weight have to be balanced. The class_weight parameter in Scikit-learn's LogisticRegression was set to "balanced" to automatically adjust the weights based on the inverse frequency of the classes in the training data. Alternatively. The results after this clearly shown improved performance. Thus, it helped to improve the model's performance on the minority class by ensuring that it is not overshadowed by the majority class during training.

Random forest classifier [Default]: random forests can be effective for handling imbalanced datasets because they combine the predictions of multiple decision trees. This can help to reduce the impact of any individual tree that may be biased towards the majority class. Overall, it performed well.

Improved Random forest classifier with hyperparameter tuning: Grid search was used to find the best values for hyperparameters such as the number of trees in the forest, the maximum depth of the trees, and the number of features to consider at each split. This is a popular technique for hyperparameter tuning in machine learning.

Support vector machine: I tried SVM with a linear kernel. The linear function is used to separate data points in a high-dimensional space. The performance was not bad but needed improvement.

Support vector machine- One vs rest classifier: In this model, a separate binary SVM classifier is trained for each class, with that class being the positive class and all other classes combined as the negative class. The class with the highest score (i.e., the SVM decision function output) is then predicted as the final output. I used this approach to handle multiclass classification.

CHALLENGES: In order to improve the performance of the model, I had to balance the dataset. Oversampling is one way to handle this issue. I tried to implement SMOTE as it can generate synthetic samples by interpolating between existing minority class samples, based on their nearest neighbors but it lead me to an error, "Expected n_neighbors <= n_samples, n_samples = 1, n_neighbors = 6". I found out that apparently when the number of minority class samples is very small (in this case, n_samples = 1), it may not be possible to find n_neighbors samples to use for interpolation thus the given error.

COMPARISON AND MODEL EVALUTION:

Model Name	Precision		Recall		F1 Score		Accuracy F1 score
	Macro Average	Weighted Average	Macro Average	Weighted Average	Macro Average	Weighted Average	
Logistic Regression (Default)	0.73	0.79	0.72	0.69	0.67	0.69	0.70
Improved logistic regression with balanced class weights	0.77	0.81	0.76	0.75	0.75	0.75	0.74
Random forest Classifier (Default)	0.77	0.82	0.76	0.75	0.72	0.74	0.75
Improved Random forest classifier with (Hyperparameter tuning)	0.75	0.79	0.75	0.73	0.71	0.72	0.73
Support Vector Machine	0.71	0.77	0.71	0.70	0.68	0.70	0.70
Support vector Machine one vs rest classifier	0.73	0.72	0.70	0.71	0.69	0.69	0.71

CONCLUSION:

Random forest classifier and Logistic regression with class weights balanced performed well comparatively. The main problem associated with imbalanced dataset is that it can cause bias in the model performance. Thus, it can predict the majority class accurately and may not have enough examples to learn the pattern in the minority class. For this reason, using accuracy metrics can be misleading. Hence, all other metrics compared in results are more appropriate.