Comparison of Decision Tree and Random Forest on Predicting Raisin Classes

Patrick Zhang

Motivation and Problem Description

- ❖ Main objective is to utilize two classification models, Decision Tree and Random Forest, to predict the raisin class and comparing the results between them.
- ❖ Taking the results of the two methods and comparing them to those of the methods utilized in a study by Cinar, Koklu, and Tasdemir.

Initial Analysis of Dataset

- ❖ Dataset: Raisin Dataset from UCI Machine Learning Repository
- ❖ Consists of 900 rows and 8 columns, 450 are from both varieties (Kecimen and Beisini).
- ❖ No missing values.
- ❖ Table 1 displays all that stats of the features, including mean, min, max, and standard deviation.
- ❖ The boxplots (Figure 1) show the distribution of all the features and the outliers present for each of them.
- ❖ The correlation matrix (Figure 2) shows the correlation between the features and the raisin class. The MajorAxisLength and Perimeter have a strong influence on the raisin class.
- ❖ The raisin class column was converted to numerical values to make processing easier. 0 represents Kecimen while 1 represents Beisini.

	Area	MajorAxisLength	MinorAxisLength	Eccentricity	ConvexArea	Extent	Perimeter
count	900.000000	900.000000	900.000000	900.000000	900.000000	900.000000	900.000000
mean	87804.127778	430.929950	254.488133	0.781542	91186.090000	0.699508	1165.906636
std	39002.111390	116.035121	49.988902	0.090318	40769.290132	0.053468	273.764315
min	25387.000000	225.629541	143.710872	0.348730	26139.000000	0.379856	619.074000
25%	59348.000000	345.442898	219.111126	0.741766	61513.250000	0.670869	966.410750
50%	78902.000000	407.803951	247.848409	0.798846	81651.000000	0.707367	1119.509000
75%	105028.250000	494.187014	279.888575	0.842571	108375.750000	0.734991	1308.389750
max	235047.000000	997.291941	492.275279	0.962124	278217.000000	0.835455	2697.753000

Table 1. Stats of predictors

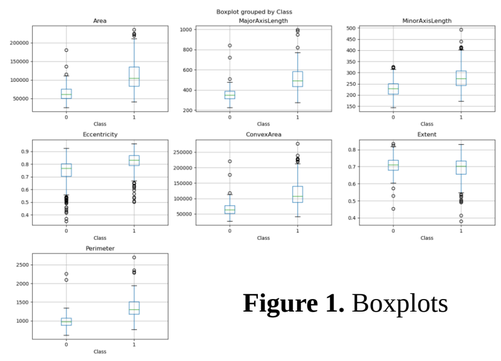

Figure 1. Boxplots

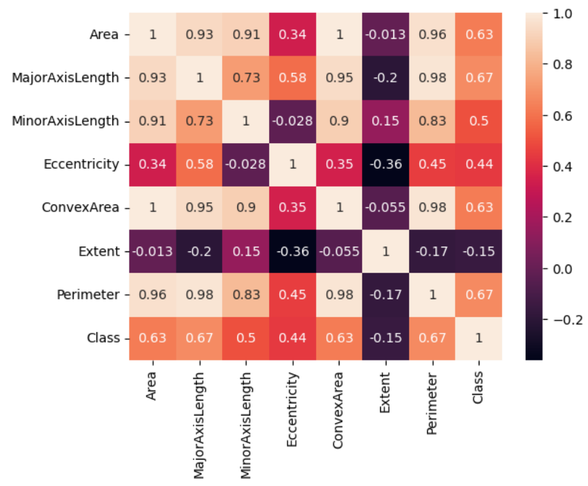

Figure 2. Correlation matrix

Decision Tree

- ❖ The Decision Tree algorithm is a supervised machine learning method where the data is split according to specific perimeters.
- ❖ The decision nodes split the data and the leaves are the outcomes.
- ❖ Proven to be useful in data mining (Mollazade, Omid and Arefi, 2012).

Pros

- ❖ Robust when dealing with outliers in training data (R, 2015).
- ❖ Requires less effort when preprocessing data.

Cons

- ❖ Prone to overfitting (R, 2015).
- ❖ Small changes can cause instability because of large changes to the tree structure.

Random Forest

- ❖ The Random Forest algorithm is a supervised machine learning method that is widely used in classification and regression problems.
- ❖ A group of trees make up an ensemble and the outcome, which is the model's prediction, is the class with the most trees.

Pros

- ❖ They don't overfit due to the Law of Large Numbers (Breiman and Schapire, 2001).
- ❖ Measures the features' importance regarding the training dataset (R, 2015).

Cons

- ❖ Biased when dealing with data containing categorical attributes (R, 2015).
- ❖ Slow in training due to the large number of trees being generated.

Hypothesis

- ❖ For large sets of data, random forest can achieve better classification performance and produces accurate and precise results (Ali et al., 2012).
- ❖ Random Forest will take longer to train than Decision Tree.

Training Choice and Methodology

1) Perform a 70:30 split for train and test data, meaning 30% of the data will be test data.
2) Optimize hyperparameters to get the best ones for the final models.
3) Testing the models and acquiring metrics like accuracy, precision, f1 score, recall and specificity.
4) Compare the results with each other.

Parameter Choices and Experiment Results

Decision Tree

- ❖ Fitting the model using Bayesian optimization.
- ❖ The hyperparameters for this model are the minimum leaf size and maximum number of splits
- ❖ Choice of parameters: 165 for minimum leaf size and 14 for the maximum number of splits

Random Forest

- ❖ Fitting the model using grid search.
- ❖ The hyperparameters for this model are the number of trees and the number of predictors for each split.
- ❖ Choice of parameters: 64 trees and 5 predictors for each split.

	Decision Tree	Random Forest
Accuracy	0.8741	0.8704
Precision	0.9007	0.8865
Specificity	0.8862	0.8730
Recall	0.8639	0.8681
F1 Score	0.8819	0.8772
Avg. AUC	0.8728	0.9386
Training Time	0.0214	0.0979

Table 2. Classification Results

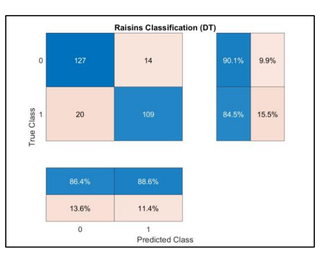

Figure 3. Decision Tree Confusion Matrix

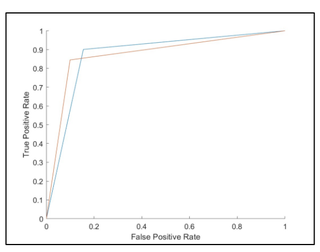

Figure 4. Decision Tree ROC Curve

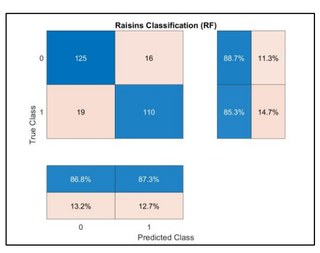

Figure 5. Random Forest Confusion Matrix

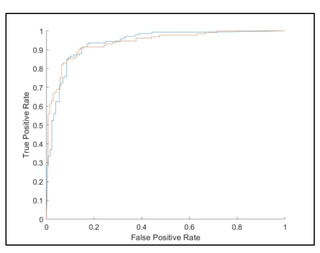

Figure 6. Random Forest ROC Curve

Analysis and Evaluation of Results

- ❖ As expected, the training time for decision tree was faster than random forest.
- ❖ In this experiment, the decision tree's accuracy was a bit higher than the random forest's accuracy. Typically, the accuracy for decision trees tends to be lower than that of random forests because of low computation.
- ❖ The decision tree outperforms the random forest in every metric except recall, meaning the decision tree performs wells in predicting true negatives and the random forest does better in predicting true positives.
- ❖ A 10-fold cross validation was used on the decision tree model and cross-validated classification error came to 0.154.
- ❖ The OOB (out of bag) error for the random forest was lower as it got closer to 64. The lowest error value came out to 0.146
- ❖ Taking both methods' accuracy values and comparing them to three algorithms utilized in a study by Cinar, Koklu and Tasdemir, the accuracies were 0.8522 for logistic regression (LR), 0.8633 for Multilayer Perceptron (MLP), and 0.8644 for support vector machine (SVM). Both decision tree and random forest in this experiment performed better than the three algorithms. However, in a study by Mollazade, Omid and Arefi, the MLP and SVM algorithms perform better than the decision tree algorithm.

Lessons Learned and Future Work

Lessons Learned

- ❖ Pay attention to any imbalances in classes as they can affect on the models' performance.
- ❖ Don't just rely on accuracy to determine a model's performance. Consider other metrics like precision, specificity, recall, and f1 score.

Future Work

- ❖ Perform other splits of the data, like 80:20 and 50:50, to see if it will impact the performance metrics.
- ❖ Find methods that can allow the results to be replicated without relying on the random seed generator
- ❖ Try training again with other algorithms and with the dataset free from outliers. With the outliers removed, the machine learning algorithms can work better.
- ❖ Utilize feature selection to see which features are irrelevant and if removed, can the models' predictive power be stronger?

References

Ali, J., Khan, R., Ahmad, N. and Maqsood, I. (2012). Random Forests and Decision Trees. *IJCSI*, 9(5).

Breiman, L. and Schapire, R. (2001). Random Forests. *Machine Learning*, 45, pp.5–32.

Cinar, I., Koklu, M. and Tasdemir, S. (2020). Classification of Raisin Grains Using Machine Vision and Artificial Intelligence Methods. *Gazi Journal of Engineering Sciences*, [online] 6(3). doi:10.30855/gmbd.2020.03.03.

Mollazade, K., Omid, M. and Arefi, A. (2012). Comparing data mining classifiers for grading raisins based on visual features. *Computers and Electronics in Agriculture*, 84, pp.124–131. doi:10.1016/j.compag.2012.03.004.

R, P.T. (2015). A Comparative Study on Decision Tree and Random Forest Using R Tool. *IJARCCE*, 4(1), pp.196–199. doi:10.17148/ijarcce.2015.4142.