Comparison of Decision Trees and Random Forest Classification Algorithms applied to Red Wine Quality Dataset

Mussa Yousef

Motivation

The motivation for this report originated from Computer Visions, where recent studies [2] have shown Random Forest to be the primary tool used in image classification and person identification [3]. Therefore, exploring Random Forest in this experiment will enhance knowledge of the fundamentals behind this algorithm and see how it can be applied to a classification problem. The comparison will be made to the decision tree, as the Random Forest is an ensemble of Decision Trees. This project will deliver a comprehensive understanding of the two algorithms through the machine learning process.

Description of the problem

- Prediction on the Quality of Red Wine based on Physicochemical tests and features
- Compare and contrast the results from Decision Tree and Random Forest
- Evaluate and analyse hyperparameter tuning for chosen machine learning algorithms

Initial Analysis

- Dataset: "Wine Quality Data set"- From UCI ML
- Original dataset has 11 predictors all of which are numeric features
- Response variable 'Quality' scored between 3-8 however when pre-processing data, conversion of response mapped using wine rating (3-4) = 0=bad, (5-6) =1=Average and (7-8) =2=Good hence using 0,1 and 2 when classifying.
- The dataset has 1599 instances with no missing data.
- The dataset is very imbalanced with Bad=4% Average=82% and Good=14%.
- Correlation heatmap of features, Histogram illustrating their importance on each label and bar chart distribution of classification labels of the quality of wine (Fig 1(Across))
- Correlation heatmap shows the highest correlated feature to be Alcohol; however it is noticeable to see there isn't a 'high' correlation in relation to any of the features.
- I will be considering this imbalanced nature of the data in the evaluation of my models, potentially giving more weighting to Recall/F-Score when finding my Hyper-parameters

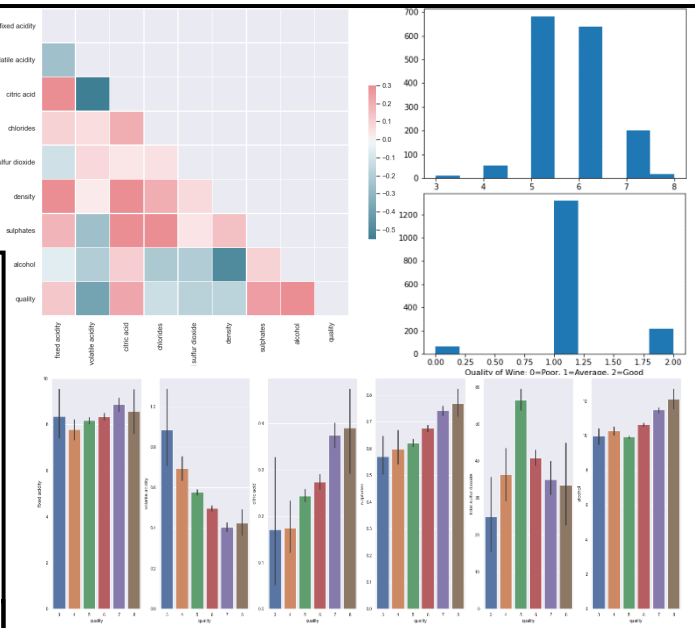

Fig.1: Heat map and Histogram depicting correlation in regard to classification labels and bar chart illustrating label distribution

Two Machine Learning Models: Pros and Cons

Decision Trees (DT): Supervised ML algorithm which in this case used for a classification problem. The DT follows a set of if-else conditions to visualise data and classify according to the condition.

Pros:

- The foundation of DT is essentially built on the human decision-making process hence it is simple to understand, quick to implement and interpret.
- The DL is robust and does not make any assumptions in regard to the shape of the data which will be used modelling, considering the imbalanced nature of the dataset in this experiment this is significant.
- As the process is based on if-else conditions feature selections happens automatically, unimportant features would not necessarily have an influence on the outcome furthermore correlated features will not affect the value of the results.

Cons:

- Decision Trees usually overfits to the Data which is used to train the model
- A small change in the data usually has an exponential change in the shape of the DT causing volatility.

Random Forest (RF): Used both for Classification and Regression algorithm which uses an ensemble of Decision Trees. Introduced in 2001, Breiman[1], RF uses bagging and the random choosing of features when building Decision-Trees, and creates a forest of trees where each concluding node is calculated, and the majority poll is used for final prediction.

Pros:

- Runs efficiently on large data sets as the number of trees-built balances data sets when as class is less frequent than others which will be interesting to see in this test.
- Robust when dealing with outliers
- Less variance than single decision trees

Cons:

- Naturally less interpretable than individual Decision Trees
- Biased when dealing with labels which are categorical.
- Potential computational cost: training large RF's may take a lot of time and memory.

Hypothesis Statement

1) Using all the features may potentially hinder the accuracy [4] for Decision Trees as previous results show 58.7% accuracy using all features.

2) Considering other classification experiment results RF demonstrated mildly better results for Accuracy, F Score and Recall. On Average Random Forrest performed 5% better in calculating these index's [6]

3) Expectation is that the Decision Tree algorithm will be more computationally efficient to Train compared to Random Forest [5]

Description of choice of training and evaluation of methodology

- Train 70% and Test 30% of data which has 1599 instances.
- Different approaches to training each model; with DT we used 10 fold cross validation for training and analysis whereas we used the Out of bag error when training and evaluating the random forest.
- Initially Creating a baseline model assessing how the models perform on the data then evaluating the hyper-parameters in optimising the best models for both DT and RF respectively.
- Comparison of the optimisation process by using both Bayesian optimisation and Grid search.
- Minimisation of error in weighing the success of the models, however as the data is greatly imbalance, we will scrutinise Recall and F1 Score as misclassification of labels which have low instances should be kept to a minimal

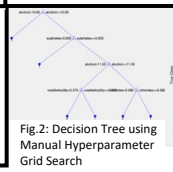

Fig.2: Decision Tree using Manual Hyperparameter Grid Search

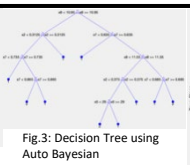

Fig.3: Decision Tree using Auto Bayesian Hyperparameter search

Choice of parameters and Results

Random Forest

- Train/Test set at:70%/30%
- Utilisation of Out-of-bag to Validate training set

Hyperparameters

- Tree bagger was used this creates an ensemble of Decision Trees and aggregates to subsample the training data for training therefore Out of Bag error was used as the performance indicator when evaluating the training model.

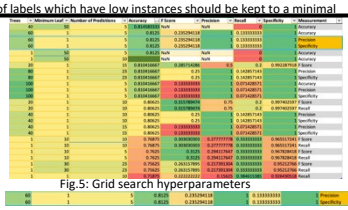

Fig.5: Grid search hyperparameters

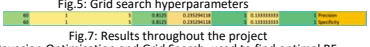

Fig.7: Results throughout the project

- Two techniques used for hyperparameter search: Bayesian Optimisation and Grid Search, used to find optimal RF for: Minimum Leaf Size, Number of Predictors to sample and number of trees
- Examining the changes to only using the important features in grid searching and Bayesian optimisation

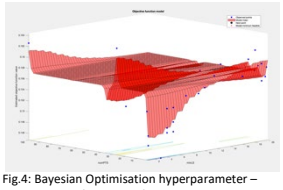

Fig.4: Bayesian Optimisation hyperparameter – optimisation of number of predictors and minimum leaf

Random Forest	Vs	Decision Tree
84.90%	Training Model Result	93.35%
Out-of-Bag Error	Validation: Method?	Cross Validation
84.30%	Validation Result	78.73
	Testing method	
80.00%	Bayesian Optimisation Model	83.21%
80.63%	Manual Gridsearch Model	82.34%

Fig.8: Accuracy results of both algorithms highlighted through each phase. Note: Both Bayesian Optimisation Model and Manual Grid search Model are the test results using hyperparameters in each respective method

Main experimental Results

- Bayesian optimisation: very robust, no flexibility in hyperparameter.Fig.4 visualises the rigid results as observed points are way off the generalised mean shape.
- Bayesian best hyperparameters: Trees=160, Minimum Leaf=19 and Predictors=8.
- Both Bayesian model and manual grid search produced a similar confusion matrix, where the imbalanced data produced skewed predictions causing a deficiency in accuracy.
- Manual grid search process gave flexibility in finding best model [Fig 5] ; we focused on choosing our hyperparameter which gave the maximum precision.
- Best Hyperparameters for grid search: Trees=60, Minimum Leaf=1 and Predictors = 5.
- Very slow to train, the higher the number of trees used the higher the computation time.

Decision Tree

- Test/Train set at 70%/30%
- Cross validation using 10-fold

Hyperparameters

- Using Loop search to optimise: Minimum Leaf Size, Maximum Split and Minimum Parent Size
- Also optimising hyperparameters using Bayesian Optimisation which automatically find parameters
- Comparing the two trees, confusion matrix and accuracy

Main Experimental results

- Parameters Found through Manual grid search using Cross-Validation maximum number of splits= 11, Minimum Leaf Size= 24, Minimum Parents Size 83
- We find that the model doesn't mispredicts the Poor quality at 100% both for manual and auto Hyperparameter searches considering the Poor quality only accounts for 3.125% of the test data even when 100% is misclassified accuracy would still be around 96% hence recall and F1 score is at 0 which is unacceptable.
- The Feature of 'Alcohol' is the root node used to split the data this is confirmed when visualising the feature importance
- The manual grid DT showed greater pruning hence although having 1% less accuracy compared to the Bayesian Optimised

Analysis and Critical Evaluation

- We found using all features in testing the models enabled better performance for RF however DT performed slightly inferior, this is seen when comparing the Bayesian Optimisation and Manual Grid search. We used the analysis of features importance when evaluating the Bayesian model yet used all features when evaluating Manual Grid search to examine the impact of the parameter 'Number of Predictors'.
- Considering the previous investigation using both Algorithms [6] we can confirm RF did provide a better F score and Recall however in this case the DT did produce a better Accuracy score. Nevertheless, we must not overlook the nature of the data. The imbalanced nature allows the robust approach that DT provides to predict at high accuracy. Therefore, considering previous classification experiments [6] saw 5% better performance from RF, in this case, we didn't align in results however this hypothesis still stands as the data used here is far too, imbalanced to object this hypothesis.
- The Decision Tree was far more superior in computational time compared to Random forest, considering the discrepancy in results this would be a huge factor in choosing DT over RF.

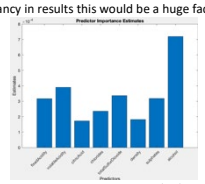

Fig.9: Feature importance (DT)

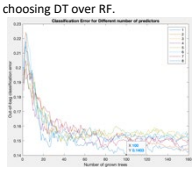

Fig.9: No.Predictors vs Number of trees grown (RF)

- When examining features, we found great flexibility using RF, with fig.9 showing that considering all 8 features when training the model delivers lowest out of bag error, at 100 trees which confirms results from our grid search fig.7 from the DT perspective a rigid approach evaluating the weight given to each feature at each node showed Alcohol had more than twice the influence on the outcome of our labels.

Conclusion:

- Although DT outperformed RF for this experiment for accuracy, we found DT to provide minimal flexibility. Given the imbalanced nature a robust approach may have been the most ideal. DT most certainly tailored to use for this as to maximise accuracy however if we scrutinise the precision and recall we find them to be far better results for RF. Hence, I will consider RF as the optimal algorithm going forward.
- We can therefore consider all original hypothesis statements to be confirmed and reinforced.

Lessons learned and future work

- The imbalanced nature of the data was a major downfall in this experiment perhaps may have been avoided by altering the division of labels [7] when classifying by using two labels.
- Advanced pre-processing analysis would have been done using PCA as previous experiments have shown an increase in accuracy and F1 score when applying PCA to features to use to analyse the algorithms performance on the same data.[2]
- Considering the nature of the data, misclassification may not have a huge impact however considering if this dataset was of a serious nature i.e., breast cancer or heart disease misclassification would have a huge cost. Therefore, in the future, we would balance the labels in order to maximise the reliability of the performance.

[1] Breiman, L., Random Forests, Machine Learning 45(1), 5-32, 2001.
[2] Ali, J., Khan, R., Ahmad, N. and Maqsood, I., 2012. Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, 9(5), p.272.
[3] Lepetit, V., Fua, P.: Keypoint recognition using randomized trees. IEEE Trans. Pattern Anal. Mach. Intell. 28(9), 1465– 1479 (2006)
[4] Apostolof, N., Zisserman, A.: Who are you? - real-time person identification. In: BMVC (2007).
[5] Er, Y. and ATASOY, A. (2016) "The Classification of White Wine and Red Wine According to Their Physicochemical Qualities", *International Journal of Intelligent Systems and Applications in Engineering*, pp. 23-26. doi: 10.18201/ijisae.265954.
[6] S. Lee, J. Park and K. Kang, "Assessing wine quality using a decision tree," 2015 IEEE International Symposium on Systems Engineering (ISSE), Rome, 2015, pp. 176-178, doi: 10.1109/SysEng.2015.7302752.
[7] S. Aich, A. A. Al-Absi, K. L. Hui, J. T. Lee and M. Sain, "A classification approach with different feature sets to predict the quality of different types of wine using machine learning techniques," 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon-si Gangwon-do, Korea (South), 2018, pp. 139-143, doi: 10.23919/ICACT.2018.8323674.
[8] A. H. P, M. K. M V and S. H A, "DRAP: Decision Tree and Random Forest Based Classification Model to Predict Diabetes," 2019 1st International Conference on Advances in Information Technology (ICAIT), Chikmagalur, India, 2019, pp. 271-276, doi: 10.1109/ICAIT47043.2019.8987277.
[9] https://www.kaggle.com/madhurisivalenka/basic-machine-learning-with-red-wine-quality-data