Multi-class Classification of Abalone Ring-age using Tree and Ensemble Learning

Dylan Wu
School of Mathematics & Statistics
University of New South Wales
Sydney, Australia
z5019314@ad.unsw.edu.au

***Abstract*—The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope – a boring and time-consuming task. Motivated by the potential of automating this arduous task, we develop a framework to classify abalone according to their ring-age in four distinct categories. We investigate if Classification and Regression Trees (CART) can provide a means to predict the number of rings on an abalone. We use decision trees and random forests on the dataset constructed by UCI Machine Learning Repository. Our results suggests that machine learning can provide a general basis for predicting the number of rings but requires improvements in classification accuracy to add value to the domain.**

Keywords—Decision trees, random forest, abalone, ensemble learning, CART, pruning, supervised learning, predictive modelling, machine learning

I. Introduction

Decision trees are one of the most popular classification algorithms to understand and interpret, with use cases that have ranged from automated telephone systems and emergency room triage to financial planning [1]. Its main strength lies in its ability to predict categorical labels or continuous values for the decision-making process under supervised learning. These models are referred to as Classification and Regression Trees (CART). Under supervised learning, the training data is fed to the algorithm including the desired solutions, called labels [2]. The work of key figures such as Wei-Yin Loh [3], [4], and John Ross Quinlan [5], [6], amongst many other academics, in conjunction with the availability and affordability of software, has propelled the popularity of these techniques in the wider scientific community. An extension of these models includes random forests, which represent one of the most powerful machine learning techniques available. It is comprised of ensemble learning which exploits the collective wisdom of many decision-makers as opposed to a single decision-maker [2]. Some key foundational papers that have contributed to this evolving field of study include those from Tin Kam Ho [7], [8], Leo Breiman [9], and Yali Amit and Donald Gemon [10].

The main challenges experienced by decision tree algorithms involved the model's sensitivity to noisy data which led to overfitting; the vulnerability to class imbalances creating a biased solution; and its inconsistency when dealing with small changes in variance, meaning it was less effective with continuous variables [11]. These challenges slightly differed with those experienced by random forests, because whilst random forests improved predictive performance and handled larger datasets more effectively [9], they produced output that was very difficult to interpret. Hence, from our experimentation, it was found that decision tree algorithms may not always be an optimal solution and its suitability is dependent on factors like the resources available.

At present, there is a vast array of analyses around abalone [12]-[14] but limited work evaluating the effectiveness of decision trees versus random forests in the classification of abalone ring-age. Determining the age of an abalone is a long and cumbersome process, and the construction of a machine learning model could accelerate this process and add value to the domain.

In this paper, we investigate the multi-class classification of abalone ring-age using tree and ensemble learning to better understand how the physical measurements of abalone can determine its ring-age. We employ classification models such as CART and bagging of trees via random forests whilst manipulating hyperparameters such as tree depth, to obtain the best performance. We use the abalone data set courtesy of the UCI Machine Learning Repository [15], which provides all the attribute information required for the classification problem, as the training dataset. We analyse how effective the CART machine learning algorithms are in classifying the abalone into four distinct age groups.

The rest of this paper is organised as follows. In section II, we present a framework that uses decision trees to predict the age of abalone from physical measurements. We compare this performance to the random forest model. Section III displays a visualisation of the trees and prediction results. Section IV provides a discussion with focus on the implications of the results. Section V concludes the paper with directions for future research.

II. Methodology

A. Abalone Data Extraction and Processing

We extract the raw Abalone dataset [16] that features 8 physical characteristics of 4,177 abalone which were observed in 1994. We check for any null or missing values and find that the original data has already removed entries with missing predicted values. We consider whether the dataset requires normalisation, however, the specification document [15] reveals the ranges of continuous values have already been scaled for use with an ANN by dividing by two hundred. We assign four distinct class labels according to the number of rings (which gives the age in years) shown in Table I. They are Class 1: '0-7 years', Class 2: '8-10 years', Class 3: '11-15 years', Class 4: 'Greater than 15 years'. After applying class labels, we one-hot encode the only categorical variable in 'Sex', by converting it into a dummy variable.

B. Framework

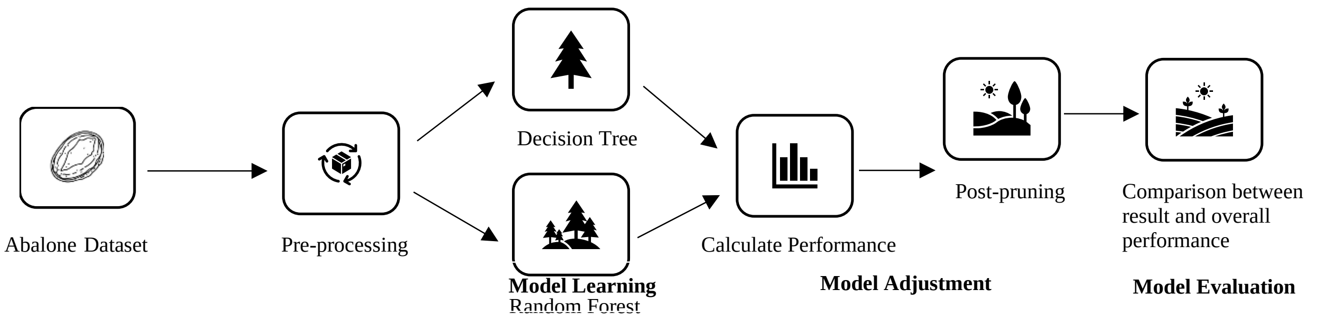
As highlighted in the introduction, we use decision trees and random forests as the machine learning methods for the classification problem. We execute both models with a data input composed of the one-hot encoded ‘Sex’ variable alongside all other continuous physical measurements excluding the predictor, rings. We choose the target as the class labels and create a feature frame containing the one-hot encoded ‘Sex’ variable, the ‘Length’, the ‘Diameter’, the ‘Height’, the ‘Whole weight’, the ‘Shucked weight’, the ‘Viscera weight’, and the ‘Shell weight’. We use the Scikit-learn Python module [17] for the machine learning algorithms, the Matplotlib [18] and Seaborn [19] libraries for data visualization, and the Pandas [20], [21] Python package for data processing.

TABLE I. TEN EXAMPLES OF ABALONE, DIVIDED INTO FOUR DISTINCT CLASSES.

Sex	Length	Diameter	Height	Whole weight	Shucked weight	Viscera weight	Shell weight	Rings	Class
M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	7	0-7 years
F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9	8-10 years
M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10	8-10 years
I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	7	0-7 years
I	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120	8	8-10 years
F	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330	20	Greater than 15 years
F	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260	16	Greater than 15 years
M	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165	9	8-10 years
F	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320	19	Greater than 15 years
F	0.525	0.380	0.140	0.6065	0.1940	0.1475	0.210	14	11-15 years

Figure 1 presents the framework for multi-class classification. At first, we compile and process the abalone data by the techniques described previously. We then train the decision trees and random forest models using the labelled abalone dataset to predict which class the abalone falls into based on its physical measurements. Afterwards, we analyse the performance of each model and change the hyperparameters of the models via pre-pruning and post-pruning to achieve the best performance.

Fig. 1. Framework for multi-class classification of abalone ring-age.



C. Decision Tree

We use the CART algorithm which produces only binary trees. The algorithm learns via recursive partitioning that mimics human-level decision making. It works by repeatedly finding the best predictor variable to split the data into two subsets. We select the splits according to the subsets that minimize Gini Impurity, and the predictor variable becomes the decision node [22]. We end the process once there are no remaining attributes or until our stop criteria is reached such as meeting the minimum number of observations for splitting. With the decision tree we can also estimate the probability that an instance belongs to a particular class [2].

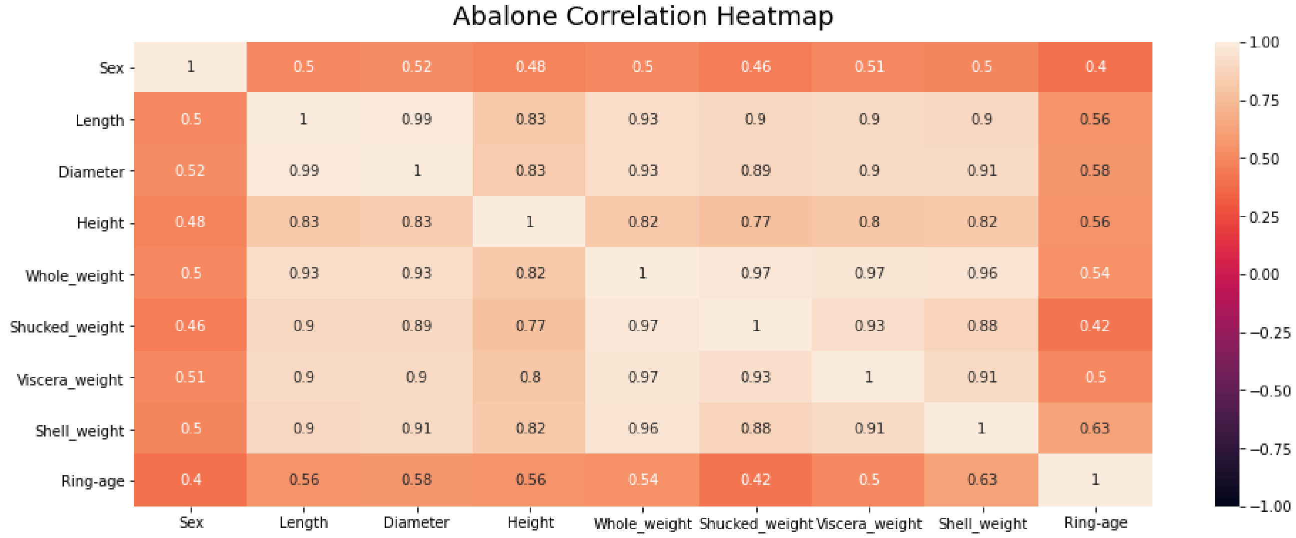
D. Bagging via Random Forests

In the case of the random forest model, the overall approach builds on the classification algorithm of the decision tree model with the addition of new hyperparameters. The model applies decision tree classifiers to various sub-samples of the abalone dataset and uses averaging to improve the predictive accuracy and control for over-fitting [17]. We select the subsets by sampling the training data uniformly and with replacement, a concept known as bagging.

E. Model Training

We conduct all experiments with a training and test split of 60/40% respectively. We execute ten experiments for each model with different random states for replication, and we select the Gini impurity criterion to measure the quality of a split in both models. We complete hyperparameter tuning with GridSearchCV [17] and adjust the tree depth of both models according to the results. We use one hundred trees in the random forest model, the default value provided by Scikit-learn.

Fig. 2. Correlation heatmap for the abalone dataset using Seaborn.



III. RESULTS

A. Data Analysis

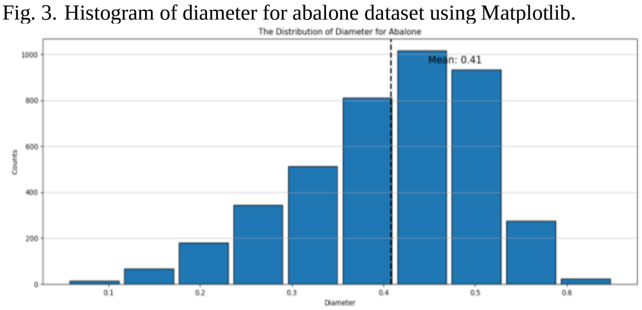
We immediately notice in Table II that there is a class imbalance in the dataset with 1,891 of the 4,176 observations lying in the age range of 8-10 years, which is 45.3% of the dataset.

To compensate for the class imbalance, we use the stratify parameter from the train_test_split function so that the proportion of the classes in the train and test sets will be the same as the proportion of the examples provided to the parameter [17].

TABLE II. CLASS DISTRIBUTION OF ABALONE DATASET.

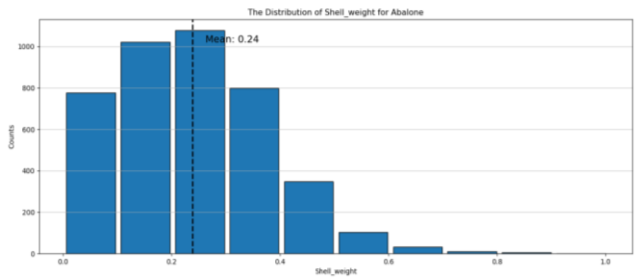
Class Label	Count	Percentage
Class 1: 0-7 years	839	20.1%
Class 2: 8-10 years	1,891	45.3%
Class 3: 11-15 years	1,185	28.4%
Class 4: Greater than 15 years	261	6.2%
Total:	4,176	100%

Fig. 3. Histogram of diameter for abalone dataset using Matplotlib.



In Figure 5, we observe that there are no major differences in the physical measurements of abalone between genders. We see that when the sex is infant, the count increases for the 0-7 years class. In Figure 7 and Figure 8, we observe the trends are consistent amongst classes, whereby the larger the ring-age the greater the physical measurements of the abalone will be.

Fig. 4. Histogram of shell weight for abalone dataset using Matplotlib.



We construct a heatmap shown in Figure 2 to identify which features exhibit the strongest linear relationships with rings. We observe the two most correlated features with ring-age are diameter and shell weight. Each feature generated a correlation coefficient of 0.58 and 0.63 respectively. In Figure 3 and Figure 4, we investigate the distribution of both features and find that the shape of the distribution for diameter and shell weight are the opposite of each other, and they are skewed to the left and the right respectively. We believe both attributes could be prime candidates for the attribute selection measures described in the methodology, and could be the decision nodes which split the records.

Fig. 5. Histogram of diameter according to sex and class.

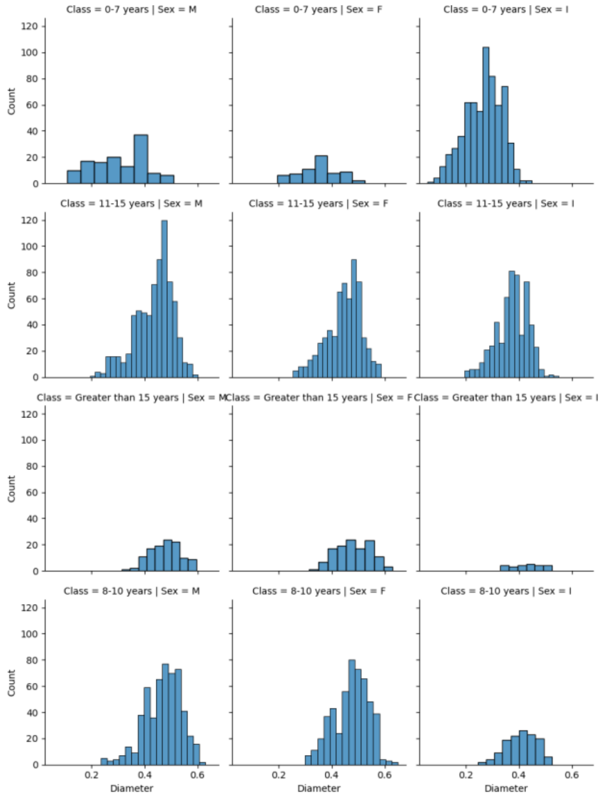


Fig. 6. Histogram of shell weight according to sex and class.

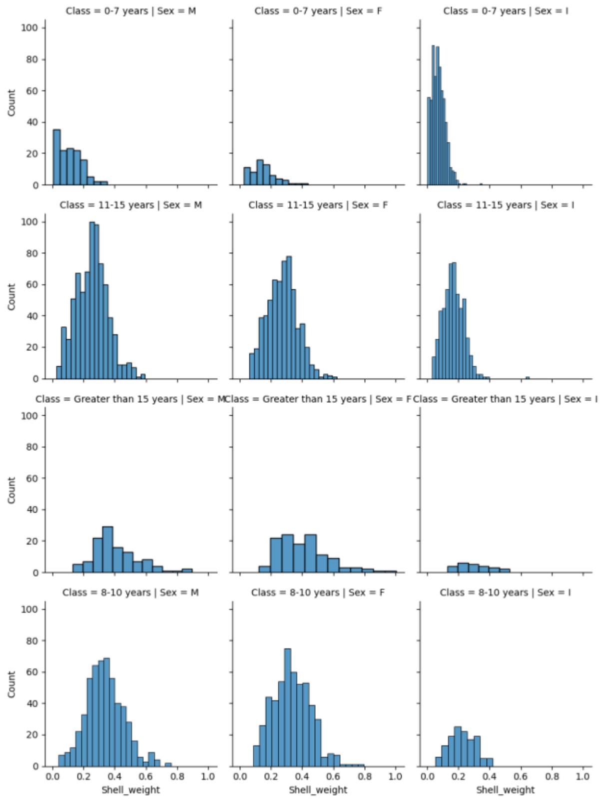


Fig. 7. Histogram of diameter according to class.

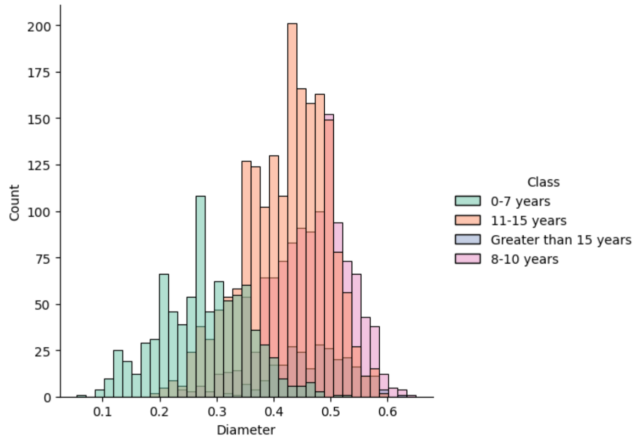
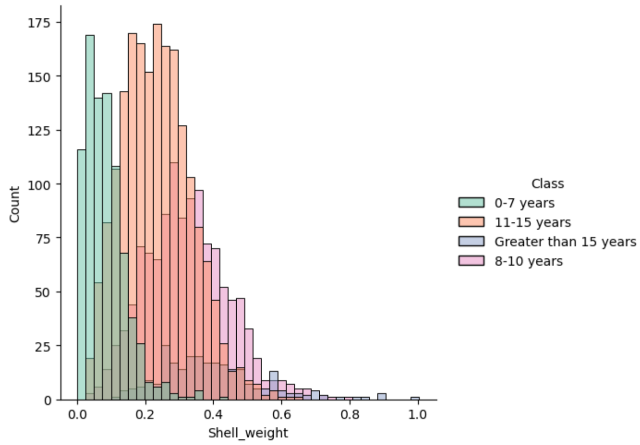


Fig. 8. Histogram of shell weight according to class.



B. Modelling and Predictions

In this section, we present the results of our proposed methodology. Table III presents the performance accuracy of the decision tree and random forest models after ten runs with a tree depth of four. Figure 9 and Figure 10 show the confusion matrix of the experiments with the best performance out of the ten runs for both models. Coincidentally, we see that it occurs at seventh run for each classification model. We observe minimal differences in performance accuracy between the decision tree and the random forest. We notice that for both models, the performance accuracy has minimal variation across each experiment and consistently hovers around the mean accuracy of 0.61 with a standard deviation of less than 0.01.

TABLE III. ACCURACY SCORES OF DEICISON TREE AND RANDOM FOREST MODELS.

Performance Accuracy Score		
Experiment Number	***Decision Tree***	***Random Forest***
1	0.6271	0.6116
2	0.6122	0.6248
3	0.6092	0.6110

Performance Accuracy Score		
Experiment Number	***Decision Tree***	***Random Forest***
4	0.6080	0.5966
5	0.6080	0.6056
6	0.6056	0.6092
7	**0.6230**	**0.6266**
8	0.6175	0.6128
9	0.6056	0.6086
10	0.6110	0.6128
Mean Acc.	**0.6127**	**0.6120**
Std. Acc.	**0.0071**	**0.0082**

We examine the confusion matrices to determine how the predicted and the true labels match up on the test set of 1,671 observations. We discover that the classification models are adept at classifying abalone in the age range of 0-7 years. In this class, both models exhibit the highest true positive to

Fig. 9. Confusion matrix of decision tree model at 7th run.

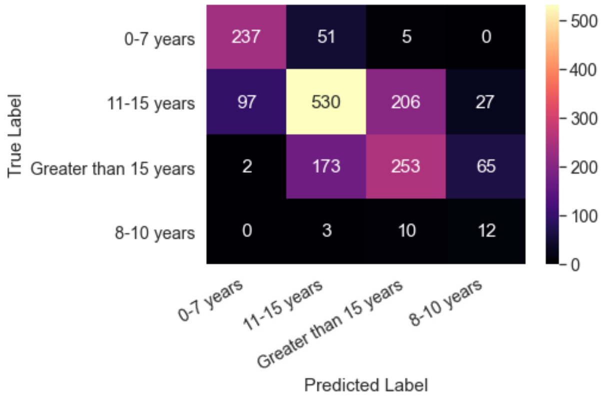
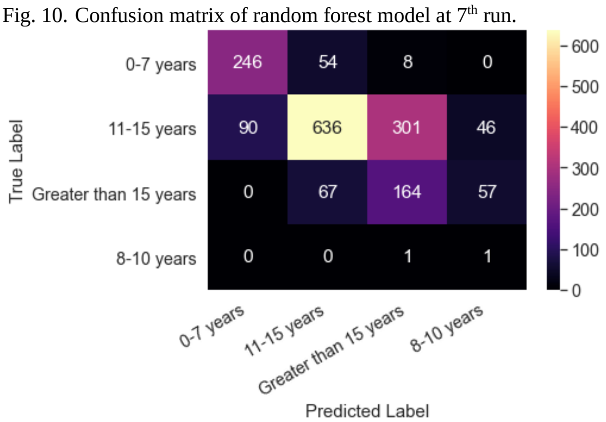


Fig. 10. Confusion matrix of random forest model at 7th run.



false positive plus false negative ratio. However, while the highest number of correct classifications is found in the age range of 11-15 years, it also has the most incorrect classifications.

In Table IV and V, we produce a classification report to confirm our findings. We use the precision score to reflect on how many of the class labels we predicted were in the correct class, and the recall score to determine how many of the values in each class were given the correct class label. The f1-score provides a weighted average of both measures so we do not overestimate the performance of the model in cases where one parameter may be drastically higher than another.

TABLE IV. CLASSIFICATION REPORT OF DECISION TREE MODEL ON 7TH RUN.

	Precision	***Recall***	***f1-score***	***Support***
0-7 years	0.71	0.81	0.75	293
8-10 years	0.53	0.51	0.52	493
11-15 years	0.70	0.62	0.66	860
Greater than 15 years	0.12	0.48	0.19	25
Accuracy			0.63	1,671
Macro avg.	0.51	0.60	0.48	1,671
Weighted avg.	0.64	0.62	0.66	1,671

TABLE V. CLASSIFICATION REPORT OF RANDOM FOREST MODEL ON 7TH RUN.

	Precision	***Recall***	***f1-score***	***Support***
0-7 years	0.73	0.80	0.76	308
8-10 years	0.35	0.57	0.43	288
11-15 years	0.84	0.59	0.70	1073
Greater than 15 years	0.01	0.50	0.02	2
Accuracy			0.63	1,671
Macro avg.	0.48	0.62	0.48	1,671
Weighted avg.	0.73	0.63	0.66	1,671

We can see that both models performed strongly when classifying ring-ages from 0-7 years but very poorly with older abalone aged greater than 15 years. We also find that the decision tree experiences mediocre performance when classifying abalone in the 8-10 years and 11-15 years age brackets at a f1-score of 0.52 and 0.66, respectively.

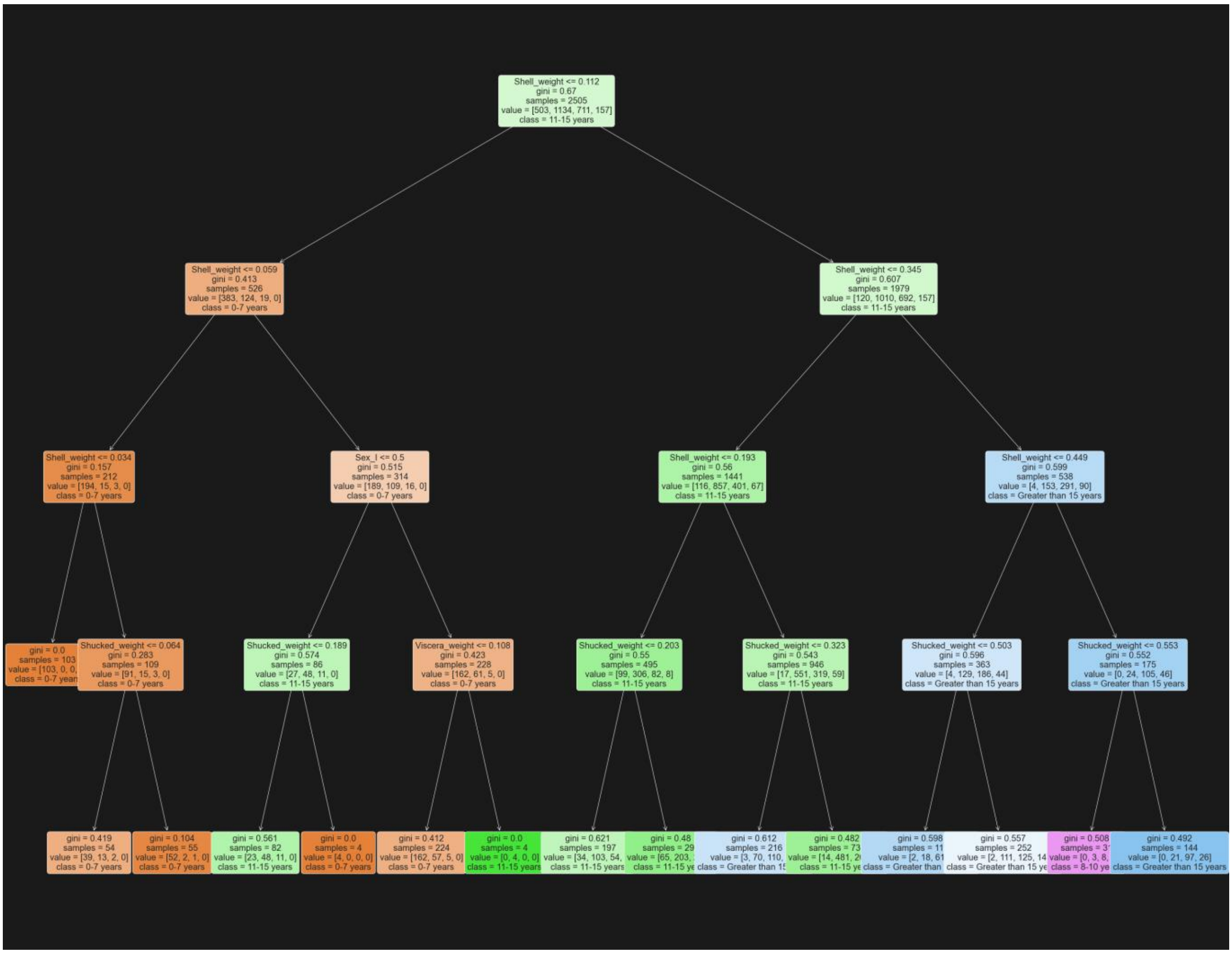
In Table VI and Figure 11, we look more closely at the features and visualize the decision rules of the decision tree to figure out how the model is calculating the importance of each of the features.

TABLE VI. FEATURE IMPORTANCE OF DECISION TREE MODEL ON 7TH RUN.

Feature	***Importance***
Shell weight	0.841
Shucked weight	0.107
Sex (Infant)	0.037
Viscera weight	0.009
Length	0.005
Diameter	0.000
Height	0.000
Whole weight	0.000
Sex (Male)	0.000

From Table VI, the values corresponding to each feature represent the amount the Gini index decreases when we choose that feature as the splitting node. Our goal is to minimize Gini impurity. We deduct that the shell weight contributes the largest decrease to the Gini index when chosen as the feature for splitting at 0.841.

Fig. 11. Visualisation of tree diagram for the decision tree model at 7th run.



As we can see in the tree diagram in Figure 11, we choose shell weight as the root node as it represents the best attribute selected for classification. As we progress further down the tree, we use shell weight to sift through the larger subsets and afterwards shucked weight is frequently used as the decision nodes to help categorize the remaining data subsets.

In Table VII, the random forest model displays a different order of feature importance as the decision tree model, and places far less emphasis on the importance of shell weight with a score of 0.343. In the random forest model, we consider diameter to be a more important feature in comparison to the decision tree model and determine that an abalone being an infant is far less important as a feature for classification.

TABLE VII. FEATURE IMPORTANCE OF RANDOM FOREST MODEL ON 7TH RUN.

Feature	***Importance***
Shell weight	0.343
Diameter	0.179
Whole weight	0.126
Height	0.103
Length	0.099
Shucked weight	0.054
Viscera weight	0.053
Sex (Infant)	0.038
Sex (Male)	0.006

In Figure 12 and 13, we compare the classifiers by measuring the area under the Receiver Operating Characteristics (ROC) curve. The ROC curve plots the true positive rate against the false positive rate [2]. The larger the area under the curve, the better the model is at classifying the abalone.

Fig. 12. ROC-AUC plot of decision tree model.

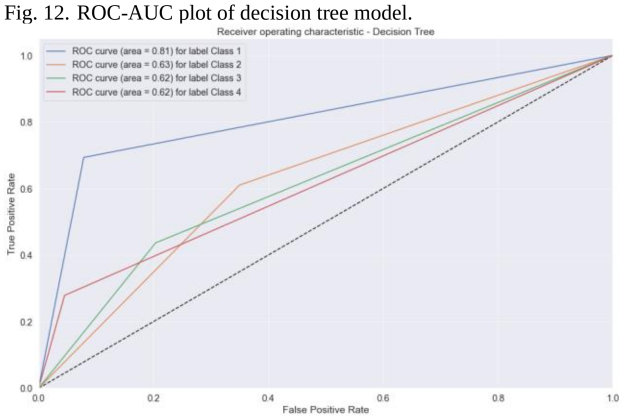
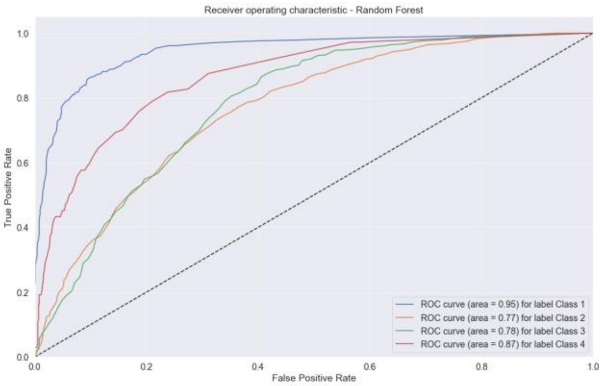


Fig. 13. ROC-AUC plot of random forest model.



We initially observe that the random forest is the superior classification model. However, because there are fewer positives compared to negatives in the dataset, the performance is exaggerated. The classification report portrays a more realistic assessment of both models and demonstrates that the classifier has room for improvement.

We now utilize the GridSearchCV function for hyperparameter tuning to see whether we can extract better performance out of our classification models. We pass predefined values for hyperparameters to the function and evaluate each combination and choose the set that provides the best performance. In Table VIII, we note that the best parameter set produces a mean recall score of 0.525. It involves a maximum tree depth of five and specifies that the minimum number of samples required to split an internal node should be four.

TABLE VIII. RECALL SCORES FOR DECISION TREE MODEL BASED ON DIFFERENT HYPERPARAMETERS.

Max depth	*Min samples split*	*Mean test score*	*Std. test score*
1	2	0.411	0.017
1	4	0.411	0.017
1	6	0.411	0.017
1	8	0.411	0.017
1	10	0.411	0.017
2	2	0.475	0.039
2	4	0.475	0.039
2	6	0.475	0.039
2	8	0.475	0.039
2	10	0.475	0.039
3	2	0.473	0.028
3	4	0.473	0.028
3	6	0.473	0.028
3	8	0.473	0.028
3	10	0.473	0.028
4	2	0.509	0.061
4	4	0.509	0.061
4	6	0.509	0.061
4	8	0.509	0.061
4	10	0.509	0.061
5	2	0.525	0.041
5	**4**	**0.525**	**0.041**
5	6	0.524	0.042
5	8	0.524	0.042
5	10	0.524	0.042

We also investigate whether we can improve performance further by post-pruning the tree. We employ cost complexity pruning as an option to control the size of the decision tree and recursively find the nodes with the “weakest link” [17]. The alpha value will help us determine which nodes we need to prune in the tree. We plot the effective alphas of the training set and its corresponding impurities in Figure 14. In Figure 15, we illustrate how the number of nodes and the tree depth decrease as the alpha increases. In Figure 16, we can see that setting an alpha of 0.048 maximizes the test accuracy.

Fig. 14. Visualizing the total impurity of leaves versus the effective alphas of pruned tree for the decision tree model.

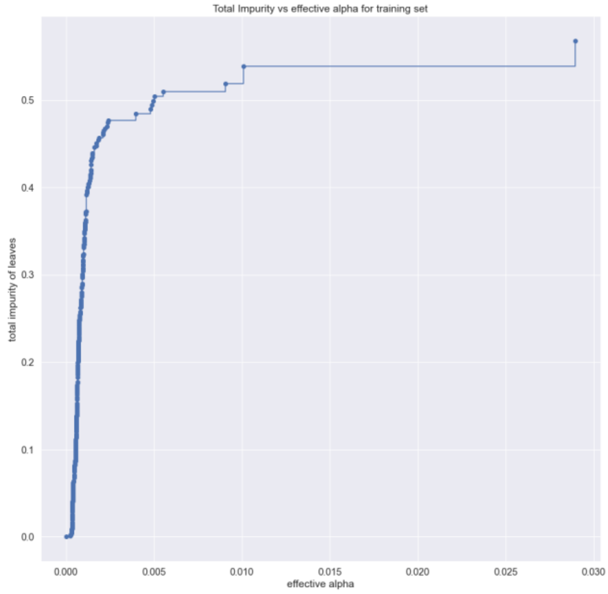
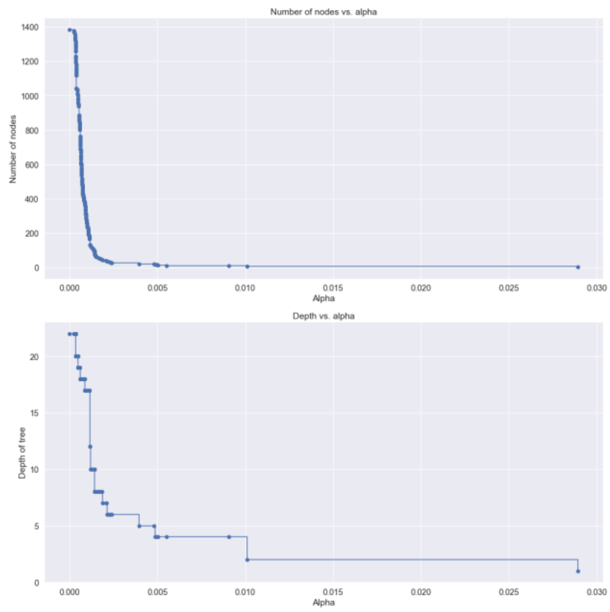


Fig. 15. Visualizing how the number of nodes and the tree depth are affected by an increase in alpha.



We apply the recommended changes to the pruned model and find that the classification rate has increased to over 63% for the best performing decision tree model (refer to Table IX). This is better than the accuracy in the previously best performing decision tree model by 0.0096%.

Fig. 16. Visualizing how the effective alphas impact the accuracy score for the decision tree model.

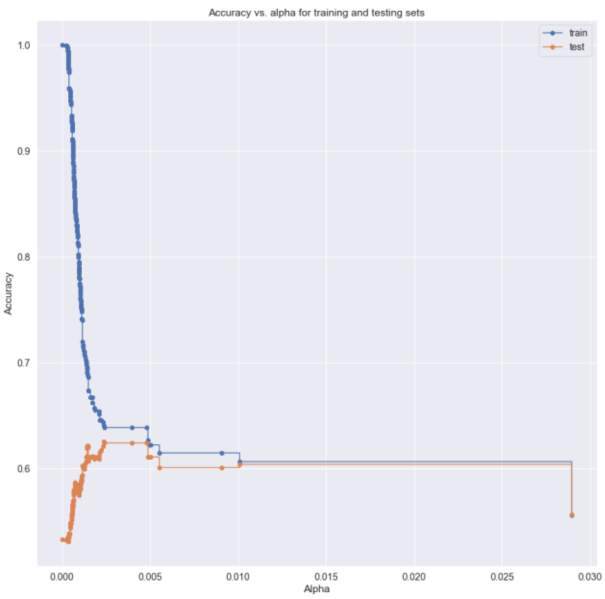


TABLE IX. ACCURACY SCORES OF PRUNED DECISION TREE MODEL.

Performance Accuracy Score	
Experiment Number	***Pruned Decision Tree***
1	0.6284
2	**0.6326**
3	0.6092
4	0.6170
5	0.6200
6	0.5907
7	0.6278
8	0.6182
9	0.6002
10	0.6242
Mean Acc.	**0.6168**
Std. Acc.	**0.0126**

In Table X, we identify that after pruning, the decision tree model has only recognized three features as important in distinguishing the ages of abalone in shell weight, shucked weight, and the infant sex. In Table XI, all the f1-score for class labels 2 and 4 experienced a large increase and we notice there are less false positives and negatives in the confusion matrix (Figure 17).

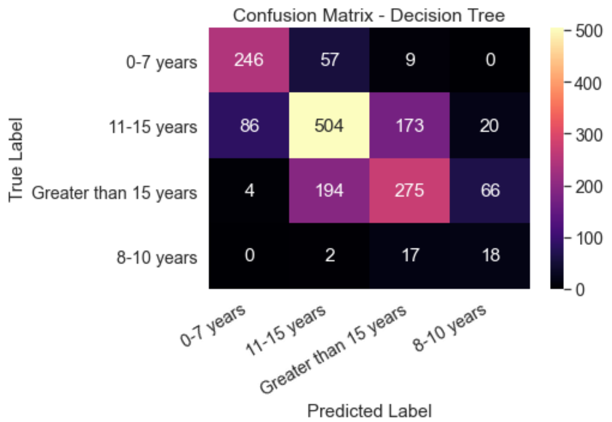
TABLE X. FEATURE IMPORTANCE OF PRUNED DECISION TREE MODEL ON 2ND RUN.

Feature	***Importance***
Shell weight	0.830
Shucked weight	0.115
Sex (Infant)	0.056
Viscera weight	0.000
Length	0.000
Diameter	0.000
Height	0.000
Whole weight	0.000
Sex (Male)	0.000

TABLE XI. CLASSIFICATION REPORT OF PRUNED DECISION TREE MODEL ON 2ND RUN.

	Precision	***Recall***	***f1-score***	***Support***
0-7 years	0.73	0.79	0.76	312
8-10 years	0.58	0.51	0.54	539
11-15 years	0.67	0.64	0.65	783
Greater than 15 years	0.17	0.49	0.26	37
Accuracy			0.62	1,671
Macro avg.	0.54	0.61	0.55	1,671
Weighted avg.	0.64	0.63	0.63	1,671

Fig. 17. Confusion matrix of pruned decision tree model on 2nd run.



As reflected in the visualization of the decision tree in Figure 18, we were able to reduce the complexity of the model whilst maintaining performance, thereby reducing overfitting, as well as making it easier to explain.

IV. DISCUSSION

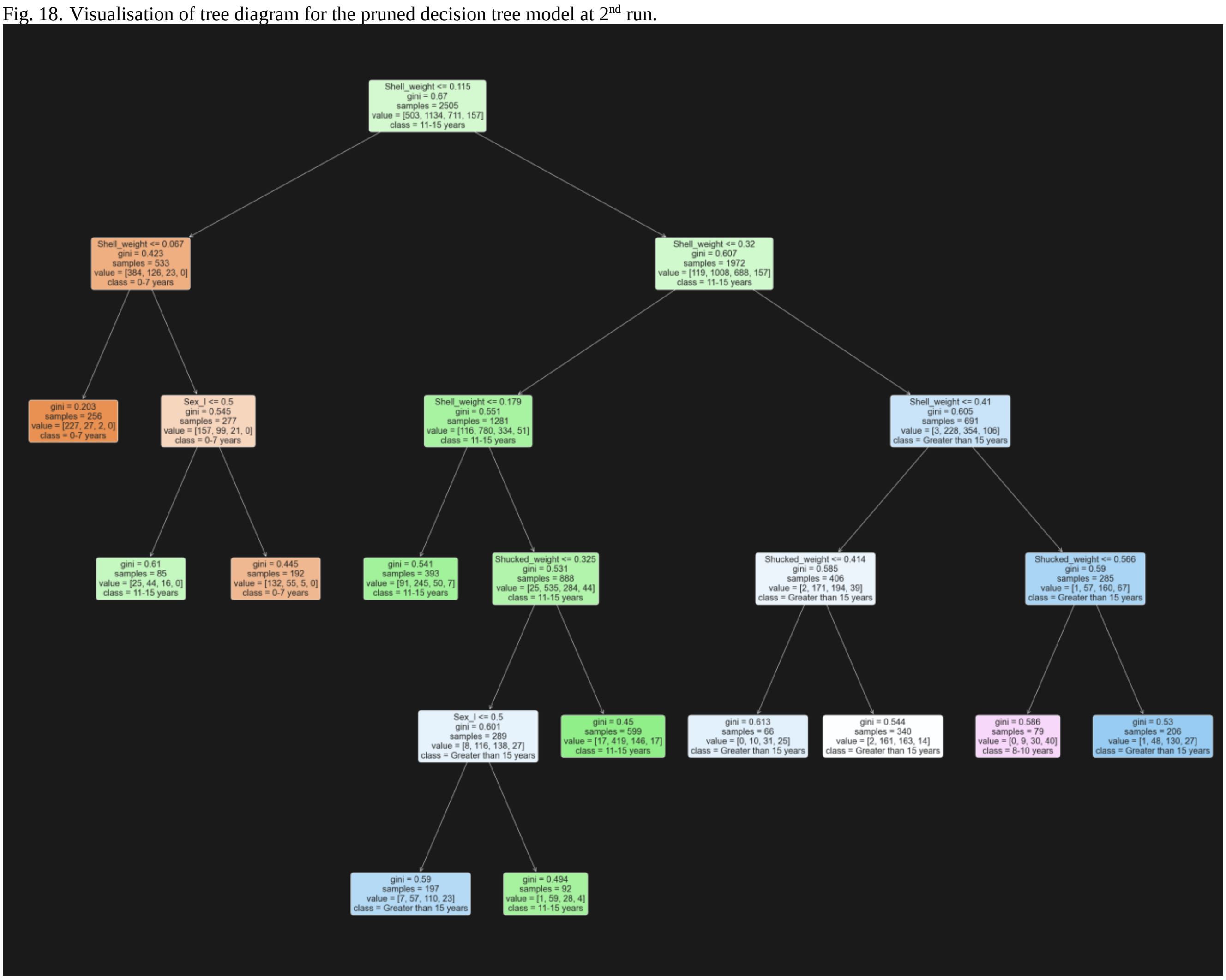
The primary purpose of our study was to understand whether machine learning models could assist in accurately classifying the ring-age of abalone based on physical measurements. We manipulated different hyperparameters to improve the performance accuracy of the models and experienced a mean accuracy increase of under 0.01%. We recognised that despite the lack of improved performance accuracy the pruned model is less complex and more easily interpretable when compared to the starting model. Additionally, when comparing the decision tree model against the random forest model, we favour the decision tree because it can explain its decision rules through a visualisation of the tree whilst the random forest model cannot. This would be important in the context of abalone classification because we need to understand how we know what age it is. Whilst the performance of the model may not be suitable for production, it does perform better than a 50/50 guess. We make recommendations in the conclusion on what future enhancements we could incorporate in the model.

Limitations to our methodology we identified included an imbalanced dataset. To address this issue Synthetic Minority Oversampling Technique (SMOTE) could be a solution to correct the balance. During the pre-processing stage, we could have applied a min-max scaler to the dataset for further normalisation. Additionally, we could have experimented with different loss criterions in the classification models such as entropy or log loss for the attribute selection measure. Confining the scope to two different classification models meant that we overlooked potentially better performing models like gradient boosting.

V. CONCLUSION

Our study highlighted that machine learning models have the potential to assist in biodiversity and that hyperparameters can be finetuned to extract the maximum value out of a classification model. Given the mediocre performance of the classification models, future work should expand to include regression models. During data analysis, there were features demonstrating a moderately positive linear relationship with rings, inferring that linear regression could outperform the tree-based model. To enhance the performance of the classification models, we suggest experimenting with boosting techniques such as AdaBoost, Gradient Boosting, and XGBoost that would convert the set of weak learners into

Fig. 18. Visualisation of tree diagram for the pruned decision tree model at 2nd run.



a strong learner. Furthermore, the performance of our machine learning models could improve by extending our list of classifiers to include the Gaussian Process or K-nearest Neighbors.

REFERENCES

[1] Software Testing Help. "Decision Tree Algorithm Examples in Data Mining." https://www.softwaretestinghelp.com/decision-tree-algorithm-examples-data-mining/#:~:text=Some%20of%20the%20decision%20tree,5%2C%20and%20CART.&text=%231)%20Learning%20Step%3A%20The,attribute%20i.e.%20%E2%80%9Cloan%20decision%E2%80%9D. (accessed 15 June 2022).

[2] A. Gron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc., 2017.

[3] W.-Y. Loh, "Fifty Years of Classification and Regression Trees," *International Statistical Review*, vol. 82, no. 3, p. 20, 2014.

[4] W.-Y. Loh, "Classification and regression trees," *WIREs Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14-23, 2011, doi: https://doi.org/10.1002/widm.8.

[5] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81-106, 1986, doi: 10.1007/BF00116251.

[6] J. R. Quinlan, "Simplifying decision trees," *International Journal of Man-Machine Studies*, vol. 27, no. 3, pp. 221-234, 1987, doi: https://doi.org/10.1016/S0020-7373(87)80053-6.

[7] H. Tin Kam, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 14-16 Aug. 1995 1995, vol. 1, pp. 278-282 vol.1, doi: 10.1109/ICDAR.1995.598994.

[8] H. Tin Kam, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832-844, 1998, doi: 10.1109/34.709601.

[9] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001/10/01 2001, doi: 10.1023/A:1010933404324.

[10] Y. Amit and D. Geman, "Shape Quantization And Recognition With Randomized Trees," *Neural Computation*, vol. 9, pp. 1545-1588, 10/01 1997, doi: 10.1162/neco.1997.9.7.1545.

[11] A. Naviani. "Decision Tree Classification in Python Tutorial." DataCamp. https://www.datacamp.com/tutorial/decision-tree-classification-python#pros (accessed 16 June 2022).

[12] M. Mullin and R. Sukthankar, "Complete Cross-Validation for Nearest Neighbor Classifiers," 2002.

[13] İ. Uysal and H. A. Güvenir, "Instance-Based Regression by Partitioning Feature Projections," *Applied Intelligence,* vol. 21, no. 1, pp. 57-79, 2004, doi: 10.1023/B:APIN.0000027767.87895.b2.

[14] P. Tan and D. Dowe, "MML Inference of Decision Graphs with Multi-way Joins and Dynamic Attributes," pp. 269-281, 2003.

[15] D. Dua and C. Graff, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, http://archive.ics.uci.edu/ml (accessed 10 June 2022).

[16] W. Nash, T. L. Sellers, S. R. Talbot, A. J. Cawthorn, and W. B. Ford, "7he Population Biology of Abalone (_Haliotis_ species) in Tasmania. I. Blacklip Abalone (_H. rubra_) from the North Coast and Islands of Bass Strait," *Sea Fisheries Division, Technical Report No*, vol. 48, 1994.

[17] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.

[18] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007, doi: 10.1109/MCSE.2007.55.

[19] M. L. Waskom, "seaborn: statistical data visualization," *Journal of Open Source Software*, vol. 6, no. 60, pp 3021, 2021.

[20] The pandas development team, "pandas-dev/pandas: Pandas," Zenodo, 2020, doi: 10.5281/zenodo.3509134.

[21] W. McKinney, "Data Structures for Statistical Computing in Python," *Proceedings of the 9th Python in Science Conference*, pp. 56-61, 2010, doi: 10.25080/Majora-92bf1922-00a.

[22] R. Aruna, "Construction of Decision Tree : Attribute Selection Measures," 2013.