

Predicting An Abalone's Age Through Its Features

Melvin Tjandradjaja
Z537881

University of New South Wales
Singapore, Singapore
m.tjandradjaja@student.unsw.edu.au

Abstract—Forecasting an abalone's age by determining how many rings its shell has can be monotonous. Under this circumstance, a different, faster, and straightforward method was needed. The method proposed in this study was the utilization of a Multi-Layer Perceptron (MLP) Classifier model to ascertain an abalone's age by classifying it into multiple class labels. Several MLP Classifier models with different values for hidden node size, learning rate, hidden layer size and optimizer were developed for this study. Each model's performance was evaluated to decide which one of them was superior in forecasting an abalone's age. Once the best model had been selected, its ability to segregate the multi class age labels was evaluated as well. The final assessment indicated that the model did not have an optimal performance in segregating the age classes.

Keywords—MLP Classifier, number of nodes, number of hidden layers, learning rates, optimizers

I. INTRODUCTION

Abalone's meat has been considered to be a rare and expensive source of food for a long period of time [1]. Its age determines its market price. Unfortunately, ascertaining the age of an abalone is a costly tedious task that takes a considerable amount of time [2]. Given this situation, a faster and more manageable way to predict the age was called for.

The objective was to utilize an abalone's physical features to forecast its age. The data set utilized to start developing the alternate forecasting method was provided by S. Waugh [3]. The development of the age forecasting method began with grouping the abalone's age variable in the data set. The completion of the grouping task was succeeded by choosing the machine learning model suitable for the age prediction task.

Since the input variables and output variable, which was grouped into different classes, were provided, a supervised machine learning model was selected [4]. In this case, it is the multi-layer perceptron (MLP) model from the Scikit-Learn machine learning library. The MLP model was capable of separating the multi class age labels by using backpropagation learning algorithm to learn the training data set. Once the learning process was finished, the model was able to forecast class labels from the new data set [5].

Selecting the appropriate machine learning model was only the beginning of the development process for the automated age class label prediction methodology. It was imperative to prevent the model from either underfitting or overfitting. To be more precise, the model should not underfit as it will guarantee failure in terms of learning the training data set and generalizing new data. The model should also not overfit as it will learn the training data set too well and not being able to generalize unseen data. The objective here was to reach a compromise in order for the model to be able to generalize new data [6].

This project's main goal involved adjusting the model's learning rate, optimizer, number of hidden layers, number of hidden layers' nodes, randomization of training data, and randomization of weight values to reach the compromise. As soon as the optimal adjustments was determined, the model's abalone age forecasting accuracy was assessed to ascertain the model's eligibility for the prediction task.

II. METHODOLOGY

A. Data Set Exploration

This project utilized a data set consisting of 4177 instances and 8 predictors (TABLE I.).

TABLE I. PREDICTORS DESCRIPTION

Feature	Data Type	Description
Sex	Nominal	M, F, and I (Infant)
Length	Continuous	Longest shell measurement
Diameter	Continuous	Perpendicular to length
Height	Continuous	With meat in shell
Whole weight	Continuous	Whole abalone
Shucked weight	Continuous	Weight of meat
Viscera weight	Continuous	Gut weight (after bleeding)
Shell weight	Continuous	After being dried

The response variable (TABLE II.) is described as follows;

TABLE II. RESPONSE VARIABLE DESCRIPTION

Target	Data Type	Description
Rings	Discrete	Plus 1.5 gives the age in years

B. Data Set Processing

The data set processing phase was initiated with importing the data set into a data frame which in turn was investigated for missing values and ensured that all the stored data were numerical values.

The findings in TABLE III indicated that the data frame has no null values.

TABLE III. DATA FRAME DESCRIPTION

Column	Non-Null Count	Dtype
sex	4177 non-null	object
length	4177 non-null	float64
diameter	4177 non-null	float64
height	4177 non-null	float64
whole_weight	4177 non-null	float64
shucked_weight	4177 non-null	float64
viscera_weight	4177 non-null	float64
shell_weight	4177 non-null	float64
rings	4177 non-null	int64

No action was required in regard to missing value replacement.

MLP model can only process numerical values. In order to adhere to this condition, the non-numerical values stored in the abalone sex column needed to be changed into numerical values [7].

In this case, the abalone gender labels M, F, and I in the sex column were changed to numerical values (M: 1, F: 2, and I: 3).

Abalone instances with height 0 were populated with the average height.

The following abalone age class labels for the different age ranges were utilized for the age forecasting task;

- Class 1 ranging from 0 to 7 years old
- Class 2 ranging from 8 to 10 years old
- Class 3 ranging from 11 to 15 years old
- Class 4 older than 15 years old

A column containing class labels indicated above was appended to the data frame to fulfill the role of a response variable for the age prediction task.

MLP model can only deal with data of the same type. In order to satisfy this condition, columns storing integers were changed to floats since integer data types were the minority in this case.

C. Adjusting the Number of Nodes in the Hidden Layer

Steps taken to adjust the number of nodes in the hidden layer are as follows;

1. The number of nodes in the hidden layer considered for this experiment were 5, 10, 15, and 20

2. Splitting the test data predictors into 60 percent for training and 40 percent for testing
3. Splitting the test data response variable into 60 percent for training and 40 percent for testing
4. Applying randomization on the predictor and response variable split to give the model more chances to perform well in forecasting an abalone's age [8]
5. Specifying the chosen node size value for the model
6. Leaving the learning rate, optimizer, and activation function of the model to their respective default values
7. Applying randomization on the model's initial weights
8. Utilizing training predictors and response variable to train the model
9. Forecasting an abalone's age utilizing the test predictors
10. Executing step 2 through 9 10 times to simulate 10 experiments with each number of nodes considered in the first step
11. Ascertaining the optimal number of nodes in the hidden layer from the comparisons made on the models' accuracy mean, standard deviation, and 95 percent confidence interval

Multiple experiments conducted on step 10 was important as they can be the most potent way to ascertain the number of nodes that can handle the data set well [9].

D. Adjusting the Learning Rate

Steps taken to adjust the learning rate are as follows;

1. The learning rates considered for this experiment were 0.1, 0.01, 0.001, and 0.0001
2. Splitting the test data predictors into 60 percent for training and 40 percent for testing
3. Splitting the test data response variable into 60 percent for training and 40 percent for testing
4. Applying randomization on the predictor and response variable split to give the model more chances to perform well in forecasting an abalone's age [8]
5. Specifying the optimal number of nodes, selected after the completion of the node size experiment, for the model
6. Specifying the learning rate value for the model

7. Leaving the model's optimizer and activation function to their default values
8. Applying randomization on the model's initial weights
9. Utilizing training predictors and response variable to train the model
10. Forecasting an abalone's age utilizing the test predictors
11. Executing step 2 through 10 10 times to simulate 10 experiments with each of the learning rates considered in the first step
12. Ascertaining the optimal learning rate from the comparisons made on the models' accuracy mean, standard deviation, and 95 percent confidence interval

The learning rate can only be determined via trial and error which makes multiple experiments in step 11 important in ascertaining the optimal value for the learning rate [10].

E. Adjusting the Number of Hidden Layers

Steps taken to adjust the number of hidden layers are as follows;

1. The number of layers considered for this experiment were 1, 2, 3, and 4
2. Splitting the test data predictors into 60 percent for training and 40 percent for testing
3. Splitting the test data response variable into 60 percent for training and 40 percent for testing
4. Applying randomization on the predictor and response variable split to give the model more chances to perform well in forecasting an abalone's age [8]
5. Specifying the optimal number of nodes, selected after the completion of the node size experiment, for the model
6. Specifying the optimal learning rate, selected after the completion of the learning rate experiment, for the model
7. Specifying the number of hidden layers for the model
8. Leaving the model's optimizer and activation function to their default values
9. Applying randomization on the model's initial weights

10. Utilizing training predictors and response variable to train the model
11. Forecasting an abalone's age utilizing the test predictors
12. Executing step 2 through 11 10 times to simulate 10 experiments with each of the layer sizes considered in the first step
13. Ascertaining the optimal number of hidden layers from the comparisons made on the models' accuracy mean, standard deviation, and 95 percent confidence interval

Multiple experiments conducted on step 12 was important as they can be the most potent way to ascertain the optimal number of hidden layers that can handle the data set well [9].

F. Adjusting the Optimizer

Steps taken to adjust the optimizer are as follows;

1. Optimizers considered for this experiment were 'SGD' and 'ADAM'
2. Splitting the test data predictors into 60 percent for training and 40 percent for testing
3. Splitting the test data response variable into 60 percent for training and 40 percent for testing
4. Applying randomization on the predictor and response variable split to give the model more chances to perform well in forecasting an abalone's age [8]
5. Specifying the optimal number of nodes, selected after the completion of the node size experiment, for the model
6. Specifying the optimal learning rate, selected after the completion of the learning rate experiment, for the model
7. Specifying the optimal number of hidden layers, selected after the completion of the hidden layer experiment, for the model
8. Specifying the optimizer for the model
9. Leaving the activation function to a default value
10. Applying randomization on the model's initial weights
11. Utilizing training predictors and response variable to train the model
12. Forecasting an abalone's age utilizing the training predictors

13. Executing step 2 through 12 10 times to simulate 10 experiments with each of the optimizers considered in the first step
14. Ascertaining the optimal optimizer from the comparisons made on the models' accuracy mean, standard deviation, and 95 percent confidence interval
15. Repeating step 1 to 14 to forecast an abalone's age with the test predictors

Most of the time, ADAM is favored to ensure a good performance. However, Stochastic Gradient Descent (SGD) in combination with a good learning rate can outperform ADAM [11].

Multiple experiments conducted in step 13 with the test data set was important in selecting the optimal optimizer where the accuracy mean and standard deviation were used to analyze the model's performance and conclude whether SGD can outperform ADAM or not.

G. Evaluating the Model

The model with the optimal settings was evaluated via the following means;

1. Confusion matrix
2. Classification report
3. One vs. Rest (OvR) approach to get the age class labels' AUC scores and average them to acquire the AUC score for the model [12]

III. RESULTS

A. Analyzing the Data

Fig. 1 shows that the male abalones make up for the majority of the proportion in this observation.

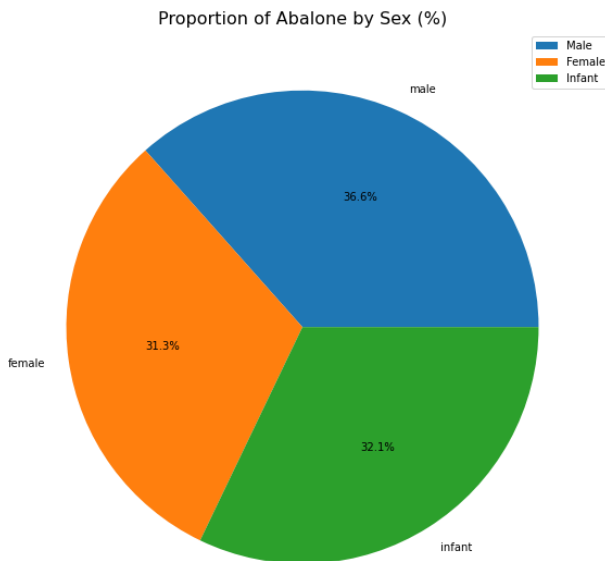


Fig. 1. Gender proportion based on the abalone data set

Fig. 2 shows that most of the abalones' lengths in this study were roughly in the range of 0.44 and 0.67 mm.

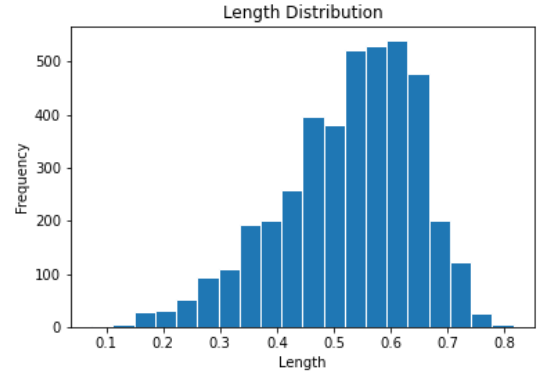


Fig. 2. Abalone length distributed based on the abalone data set

Fig. 3 shows that most of the abalones' diameters in this study were roughly in the range of 0.33 and 0.5 mm.

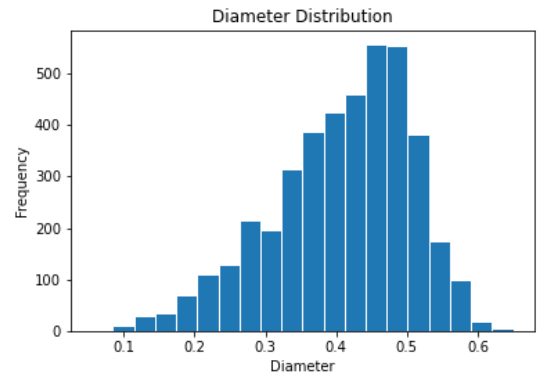


Fig. 3. Abalone diameter distributed based on the abalone data set

Fig. 4 shows that most of the abalones' heights in this study were roughly in the range of 0.1 and 0.18 mm.

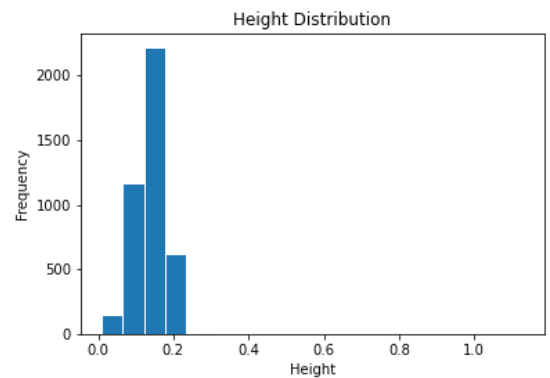


Fig. 4. Abalone height distributed based on the abalone data set

Fig. 5 shows that most of the abalones' whole weights in this study were roughly in the range of 0.48 gram and 1.2 grams.

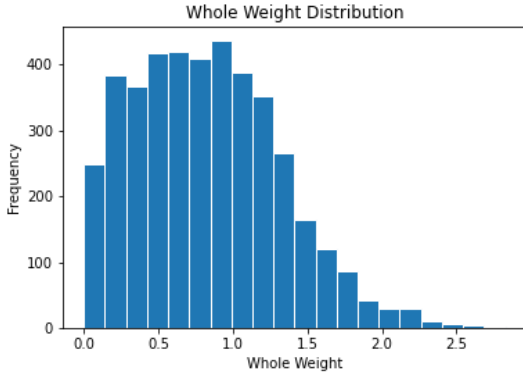


Fig. 5. Abalone whole weight distributed based on the abalone data set

Fig. 6 shows that most of the abalones' shucked weights in this study were roughly in the range of 0.08 and 0.47 grams.

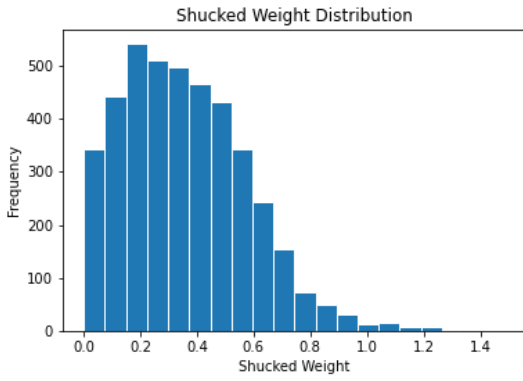


Fig. 6. Abalone shucked weight distributed based on the abalone data set

Fig. 7 shows that most of the abalones' viscera weights in this study were roughly in the range of 0.048 and 0.23 gram.

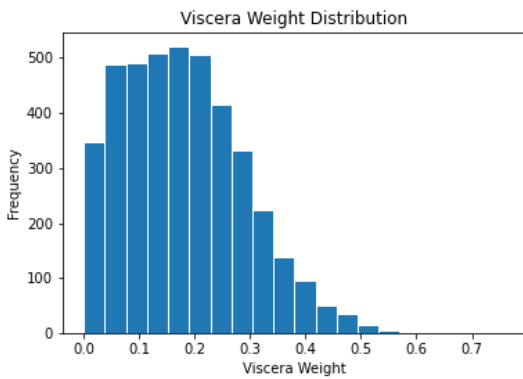


Fig. 7. Abalone viscera weight distributed based on the abalone data set

Fig. 8 shows that most of the abalones' shell weights in this study were roughly in the range of 0.1 and 0.3 gram.

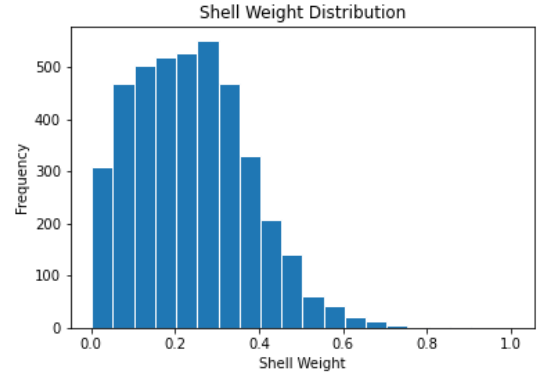


Fig. 8. Abalone shell weight distributed based on the abalone data set

Fig. 9 shows significant differences between class 2 proportion and the rest of the classes.

Under this circumstance, more class 1, 3, and 4 observations were needed for the MLP model to train on and predict.

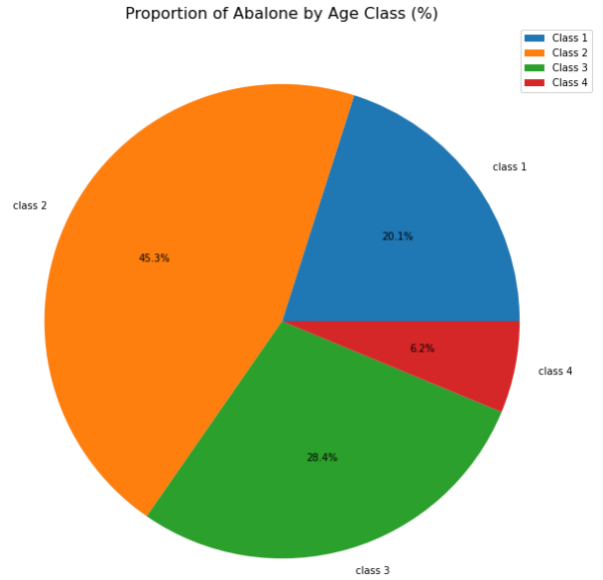


Fig. 9. Abalone age class proportion based on the abalone data set

B. Analyzing Forecasting Accuracy

a) *Results from the Hidden Layer Node Sizes Assessment:* The node sizes assessment was to determine how many nodes could give the MLP model its optimal performance. TABLE IV shows assessment results for 5, 10, 15, and 20 nodes.

TABLE IV. ASSESSMENT ON NUMBER OF NODES

	5 Node Model	10 Node Model	15 Node Model	20 Node Model
Mean of the Accuracy	0.594	0.609	0.607	0.618
Standard Deviation of the Accuracy	0.018	0.02	0.02	0.016
Lower Bound	0.57	0.586	0.584	0.595
Upper Bound	0.618	0.633	0.631	0.641

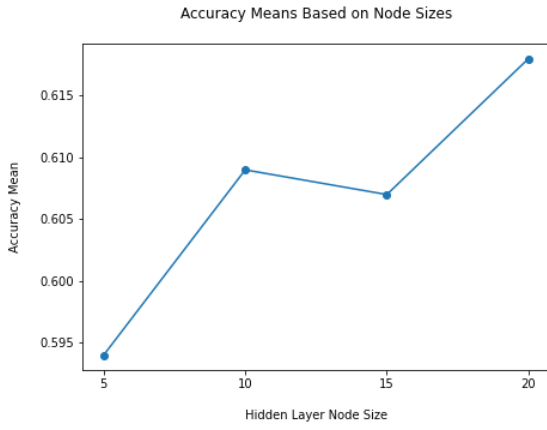


Fig. 10. Accuracy mean values obtained from the node sizes assessment

Fig. 10 shows that the forecasting accuracy was at the highest at 20 nodes.

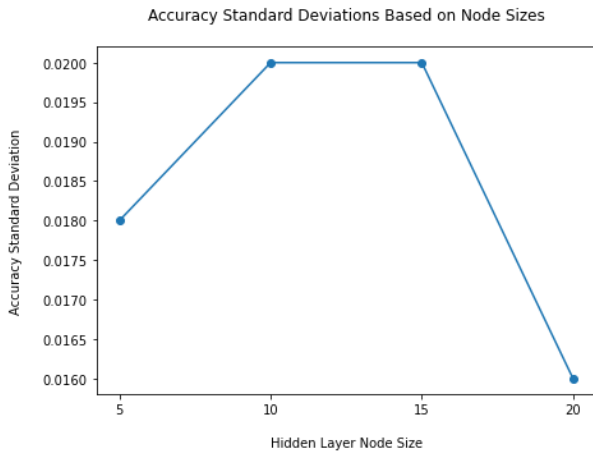


Fig. 11. Accuracy standard deviations values obtained from the node sizes assessment

Fig. 11 shows that the standard deviation was at the lowest at 20 nodes.

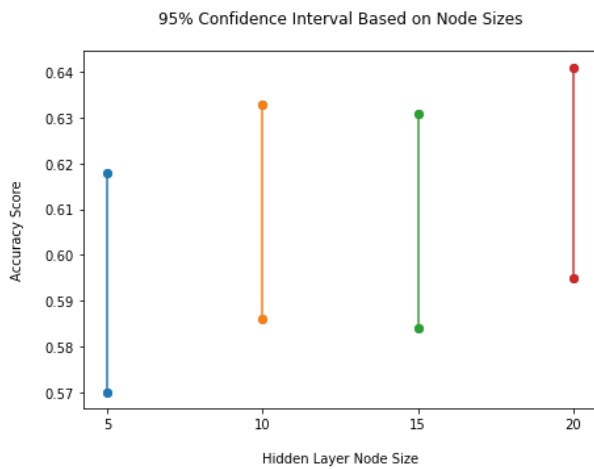


Fig. 12. 95% confidence interval obtained from the node sizes assessment

Fig. 12 shows that the accuracy range was at the highest at 20 nodes.

Fig. 10, Fig. 11, and Fig. 12 seem to indicate that the MLP model with 20 nodes had the highest prediction accuracy. Therefore, the MLP model with 20 hidden layer nodes was considered to have optimal performance.

This assessment had shown that the model with larger width capacity can learn the data better than the ones with smaller width capacities [15].

b) Results from the Learning Rates Assessment: The learning rates assessment was to determine which learning rate value gave the MLP model its optimal performance. TABLE V shows assessment results for learning rate value 0.1, 0.01, 0.001, and 0.0001.

TABLE V. ASSESSMENT ON LEARNING RATES

	Model with Learning Rate at 0.1	Model with Learning Rate at 0.01	Model with Learning Rate at 0.001	Model with Learning Rate at 0.0001
Mean of the Accuracy	0.629	0.635	0.618	0.572
Standard Deviation of the Accuracy	0.014	0.011	0.016	0.018
Lower Bound	0.606	0.612	0.595	0.548
Upper Bound	0.652	0.658	0.641	0.595

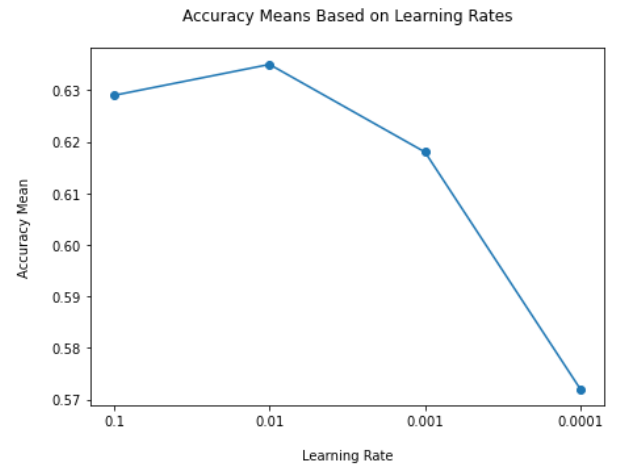


Fig. 13. Accuracy mean values obtained from the learning rates assessment

Fig. 13 shows that the forecasting accuracy was at the highest at learning rate 0.01.

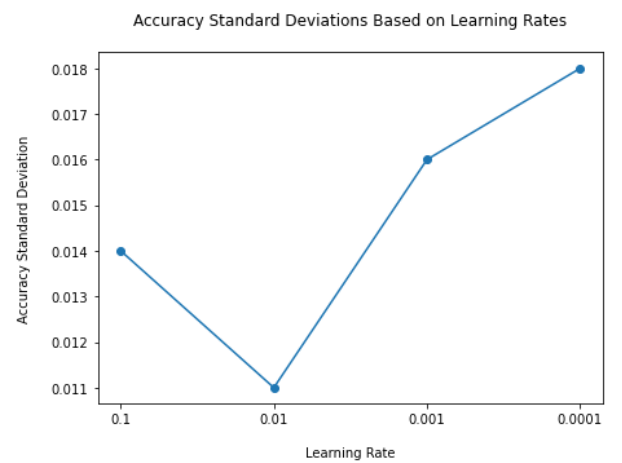


Fig. 14. Accuracy standard deviations obtained from the learning rates assessment

Fig. 14 shows that the standard deviation was at the lowest at learning rate 0.01.

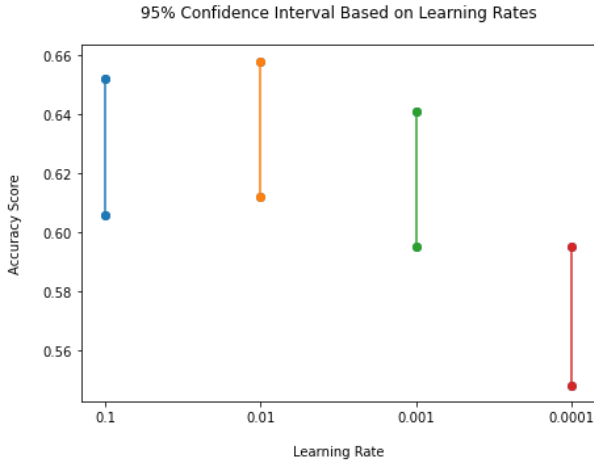


Fig. 15. 95% confidence interval obtained from learning rates assessment

Fig. 15 shows that the accuracy range was at the highest at learning rate 0.01.

Fig. 13, Fig. 14, and Fig. 15 seem to indicate that the MLP model with learning rate 0.01 and 20 nodes had the highest prediction accuracy. Therefore, the MLP model with learning rate 0.01 and 20 nodes was considered to have optimal performance.

Since the learning rate between 0.1 and 0.0001 were not too large or too small, the problem of exploding weights and non-converging learning rate were avoided [10].

c) Results from the Hidden Layer Sizes Assessment: The hidden layers assessment was to determine how many hidden layers could give the MLP model its optimal performance. TABLE VI shows assessment results for 1 layer, 2 layers, 3 layers, and 4 layers.

TABLE VI. ASSESSMENT ON HIDDEN LAYERS

	1 Layer Model	2 Layer Model	3 Layer Model	4 Layer Model
Mean of the Accuracy	0.635	0.635	0.631	0.634
Standard Deviation of the Accuracy	0.011	0.009	0.013	0.017
Lower Bound	0.612	0.612	0.608	0.61
Upper Bound	0.658	0.658	0.654	0.657

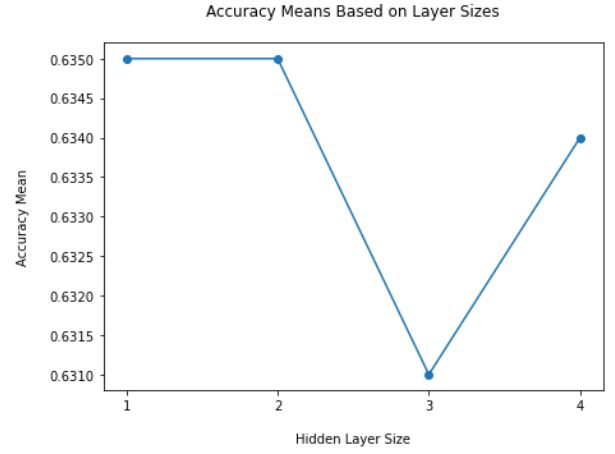


Fig. 16. Accuracy mean values obtained from the layer sizes assessment

Fig. 16 shows that the forecasting accuracy was at the highest at 1 layer and 2 layers.

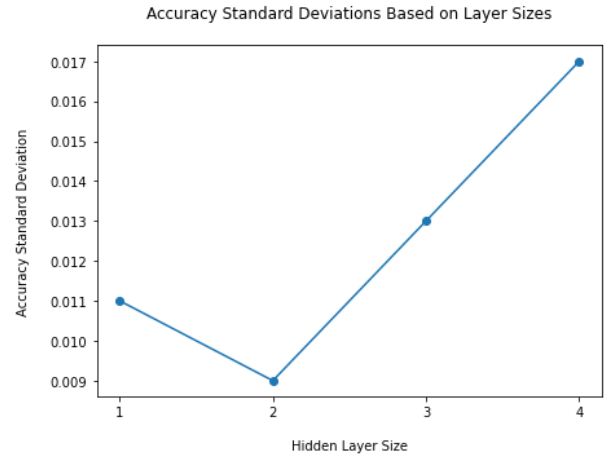


Fig. 17. Accuracy standard deviations obtained from the layer sizes assessment

Fig. 17 shows that the standard deviation was at the lowest at 2 layers.

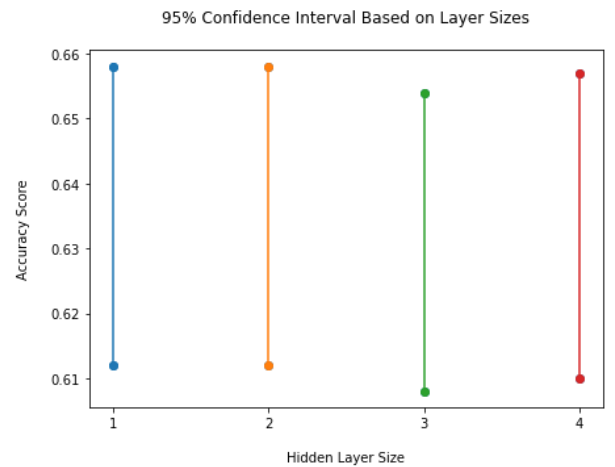


Fig. 18. 95% confidence interval obtained from the layer sizes assessment

Fig. 18 shows that the accuracy range was at the highest at 1 layer and 2 layers.

Fig. 16, Fig. 17, and Fig. 18 seem to indicate that the MLP model with 1 layer and 2 layers had the highest prediction accuracy. However, the model with 2 layers had the lowest standard deviation. Therefore, the MLP model with 2 layers, learning rate 0.01 and 20 nodes was considered to have optimal performance.

This assessment had shown that the MLP model with lower depth capacity accompanied with enough number of nodes had the ability to learn the data better the models with higher depth capacity [15].

d) Results from the Optimizers Assessment:

The optimizers assessment was to determine which optimizer gave the MLP model its optimal performance. TABLE VII shows assessment results for Stochastic Gradient Descent (SGD) with training data, Adam with training data, SGD with test data, and Adam with test data.

TABLE VII. ASSESSMENT ON OPTIMIZERS

	Training		Test	
	Model with SGD	Model with Adam	Model with SGD	Model with Adam
Mean of the Accuracy	0.643	0.648	0.635	0.644
Standard Deviation of the Accuracy	0.006	0.007	0.009	0.01
Lower Bound	0.62	0.626	0.612	0.621
Upper Bound	0.666	0.671	0.658	0.667

Since other assessments used test data for the predictions, this assessment only focused on the performance metrics based on the test data.

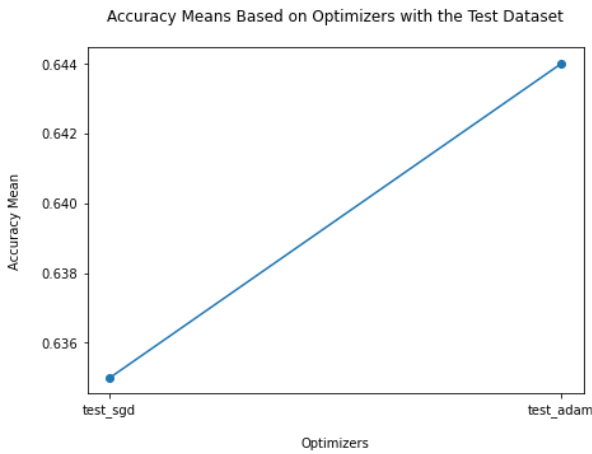


Fig. 19. Accuracy mean values obtained from the optimizers assessment

Fig. 19 shows that the forecasting accuracy for Adam optimizer was higher than SGD.

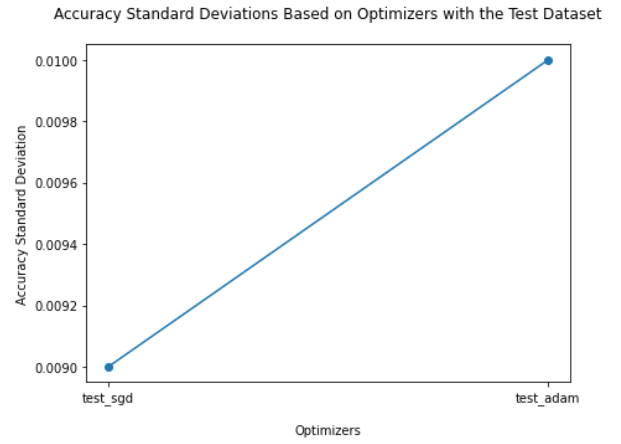


Fig. 20. Accuracy standard deviations obtained from the optimizers assessment

Fig. 20 shows that the standard deviation was at the lowest with SGD optimizer.

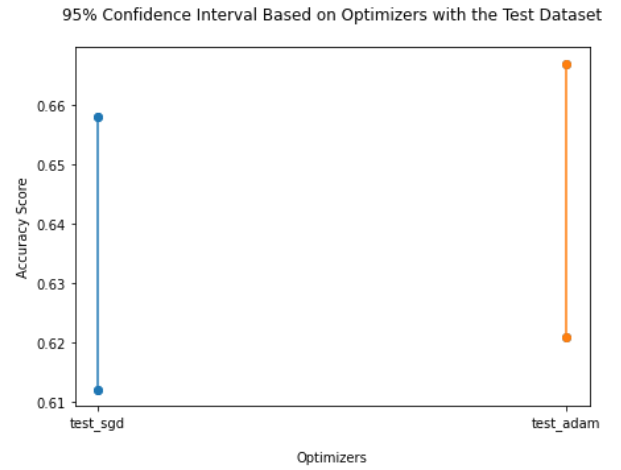


Fig. 21. 95% Confidence interval obtained from the optimizers assessment

Fig. 21 shows that the model with Adam optimizer had higher accuracy range.

Fig. 19, Fig. 20, and Fig. 21 seem to indicate that the MLP model with Adam optimizer had higher prediction accuracy. Therefore, the MLP model with Adam optimizer, 2 layers, learning rate 0.01 and 20 nodes was considered to have optimal performance.

This assessment had shown that a model with Adam optimizer performed better than one with SGD optimizer [11].

e) Results from the Optimal Model Assessment:

TABLE VIII shows that the optimal model, with 20 nodes, learning rate 0.01, 2 hidden layers, and Adam optimizer, was only 63 percent accurate in forecasting the abalone age class where the positive instances were identified [13].

TABLE VIII. CLASSIFICATION REPORT WITH TEST DATA SET

	precision	recall	f1-score	support
1	0.79	0.7	0.74	331
2	0.6	0.83	0.7	767
3	0.55	0.34	0.42	469
4	0.59	0.18	0.28	104
accuracy			0.63	1671
macro avg	0.63	0.51	0.53	1671
weighted avg	0.62	0.63	0.6	1671

Fig. 22 shows that only 1049 (approx. 63 percent) out of 1671 instances in the test data set were correctly classified.

Confusion Matrix for Class Prediction with Test Dataset

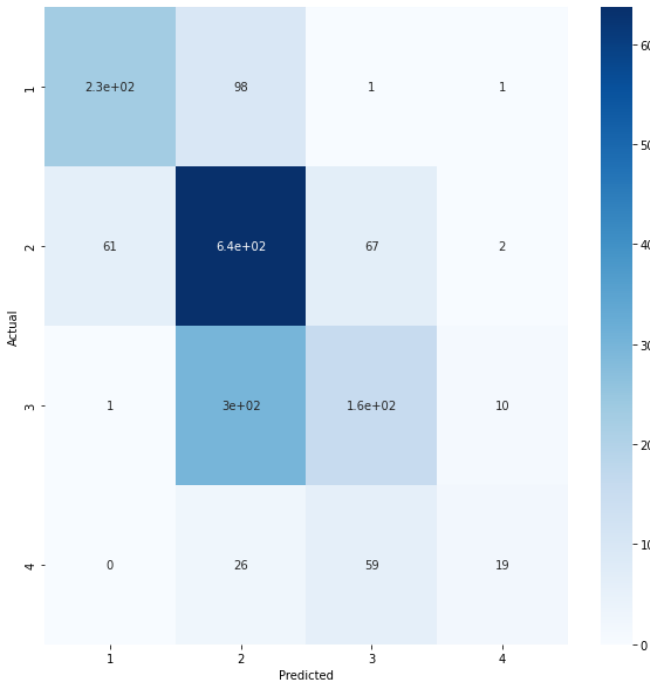


Fig. 22. Not all the abalone age class was correctly classified

Fig. 23 shows excellent classification between the binary classes whereas Fig. 24, Fig. 25, and Fig. 26 show poor classification[16].

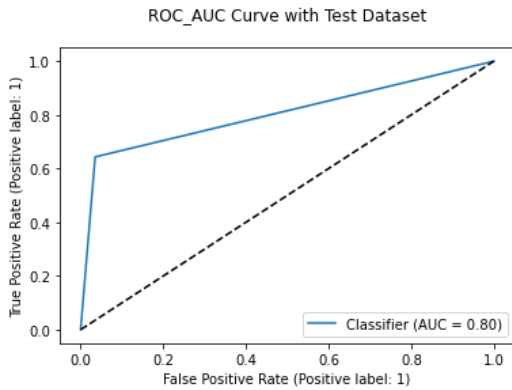


Fig. 23. AUC score 0.80 indicated excellent separation between age class 1 and the rest

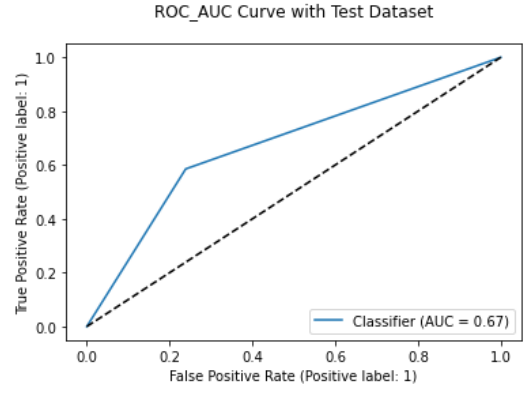


Fig. 24. AUC score 0.67 indicated poor separation between age class 2 and the rest

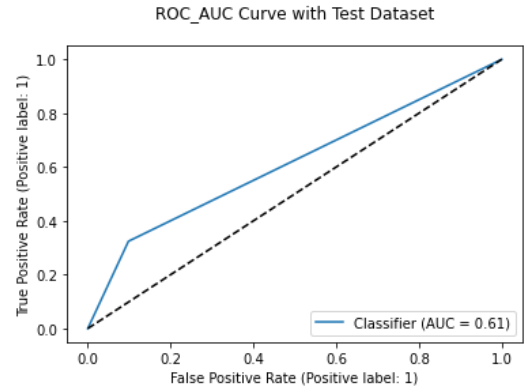


Fig. 25. AUC score 0.61 indicated poor separation between age class 3 and the rest

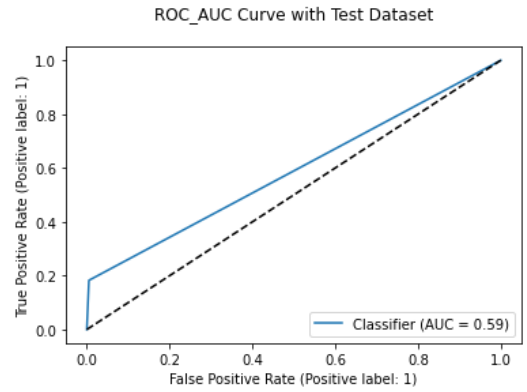


Fig. 26. AUC score 0.59 indicated poor separation between age class 4 and the rest

TABLE IX. displays each of the One vs. Rest (OvR) AUC score where the average OvR AUC score served as the MLP model's overall performance metric for binary classification [12].

TABLE IX. AUC SCORE SUMMARY

One versus Rest (OvR)	AUC
1 versus Rest	0.8
2 versus Rest	0.67
3 versus Rest	0.61
4 versus Rest	0.59
Average	0.6675

IV. CONCLUSION

This project involved 4177 abalone instances to assess a multi-layer perceptron (MLP) model's ability in forecasting an abalone's age from its physical attributes. The assessment was done by determining the optimal number of nodes, learning rate, number of hidden layers, and optimizer for the MLP model. Results obtained from the assessment showed that 20 nodes, learning rate 0.01, 2 layers, and Adam optimizer to be the optimal settings. In terms of multi-class classification, the MLP model with optimal hyper-parameters had a poor performance at 63 percent accuracy when predicting an abalone's age. In terms of binary classification, the model had a poor performance with an average AUC score 0.6675 when predicting an abalone's age using One vs. Rest (OvR) strategy.

The MLP model used for this project was assessed using a data set with an unbalanced abalone age class distribution. Re-evaluation on the MLP model can involve Synthetic Minority Oversampling Technique (SMOTE) to handle the disproportion issue during the data preparation phase [14]. Moreover, Principal Component Analysis (PCA) with a more balanced data set, DecisionTree model, RandomForest model, AdaBoost model, etc. can be considered for assessment on further improvements.

REFERENCES

- [1] "Abalone," en.wikipedia.org. <https://en.wikipedia.org/wiki/Abalone> (accessed: May. 29, 2023).
- [2] M. M. Hossain and M. N. M. Chowdhury, "Econometric ways to estimate the age and price of abalone." University of Nevada, Jan. 3, 2019. Accessed on: Jun. 2023. [Online]. Available: https://mpira.ub.uni-muenchen.de/91210/1/MPRA_paper_91210.pdf
- [3] S. Waugh, 1995, "Abalone Data Set", Department of Computer Science, University of Tasmania, Tasmania. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/Abalone>.
- [4] J. Brownlee, "Supervised and Unsupervised Machine Learning Algorithms", machinelearningmastery.com <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/> (accessed: May. 29, 2023).
- [5] "Neural network models (supervised)", scikit-learn.org. https://scikit-learn.org/stable/modules/neural_networks_supervised.html (accessed: May. 29, 2023).
- [6] J. Brownlee, "Overfitting and Underfitting With Machine Learning Algorithms", machinelearningmastery.com. <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/> (accessed: May. 29, 2023).
- [7] S. Garg, "How to Deal with Categorical Data for Machine Learning", kdnuggets.com. <https://www.kdnuggets.com/2021/05/deal-with-categorical-data-machine-learning.html> (accessed: May. 31, 2023).
- [8] R. Pramoditha, "Why do we set a random state in machine learning models?", towardsdatascience.com. <https://towardsdatascience.com/why-do-we-set-a-random-state-in-machine-learning-models-bb2dc68d8431#:~:text=The%20random%20state%20hyperparameter%20in,test%20sets%20across%20different%20executions> (accessed: May. 31, 2023).
- [9] J. Brownlee, "How to Configure the Number of Layers and Nodes in a Neural Network", machinelearningmastery.com. <https://machinelearningmastery.com/how-to-configure-the-number-of-layers-and-nodes-in-a-neural-network/> (accessed: Jun. 1, 2023).
- [10] J. Brownlee, "How to Configure the Learning Rate When Training Deep Learning Neural Networks", machinelearningmastery.com. <https://machinelearningmastery.com/learning-rate-for-deep-learning-neural-networks/> (accessed: Jun. 1, 2023).
- [11] D. Giordano, "7 tips to choose the best optimizer", towardsdatascience.com. <https://towardsdatascience.com/7-tips-to-choose-the-best-optimizer-47bb9c1219e> (accessed: Jun. 1, 2023).
- [12] T. Trevisan, "Multiclass classification evaluation with ROC Curves and ROC AUC", towardsdatascience.com. <https://towardsdatascience.com/multiclass-classification-evaluation-with-roc-curves-and-roc-auc-294fd4617e3a> (accessed: Jun. 2, 2023).
- [13] S. Kohli, "Understanding a Classification Report For Your Machine Learning Model", medium.com. <https://medium.com/@kohlishivam5522/understanding-a-classification-report-for-your-machine-learning-model-88815e2ce397> (accessed: Jun. 3, 2023).
- [14] J. Brownlee, "SMOTE for Imbalanced Classification with Python", machinelearningmastery.com. <https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/> (accessed: Jun. 4, 2023).
- [15] J. Brownlee, "How to Control Neural Network Model Capacity With Nodes and Layers", machinelearningmastery.com. <https://machinelearningmastery.com/how-to-control-neural-network-model-capacity-with-nodes-and-layers/> (accessed: Jun. 5, 2023).
- [16] Zach, "What is Considered a Good AUC Score?", statology.org. <https://www.statology.org/what-is-a-good-auc-score/> (accessed: Jun. 5, 2023).