MAT 422: Final Project

Michael Mykhaylov

November 26, 2023

1 Introduction

Abalones, marine mollusks of significant ecological and economic importance, pose a unique challenge to age determination through traditional manual methods. This research endeavors to improve the current age prediction process of abalones by harnessing the power of machine learning techniques. In a world where the intricate interplay between environmental factors and biological attributes shapes the growth of these creatures, predicting their age accurately becomes paramount for marine biologists and fisheries alike.

The dataset chosen for this study, originating from Warwick Nash in 1994, encapsulates a comprehensive set of features essential for predicting the age of abalones. By employing a diverse set of predictive models, including multiple linear regression, gradient-boosted decision tree regressors, and neural networks, we aim to navigate through the intricate relationships between physical measurements and age. This approach contributes to the academic understanding of machine learning models and holds practical implications for optimizing the labor-intensive task of manually counting growth rings in abalone shells.

Our methodology, echoing established practices in predictive modeling, will not only shed light on the capabilities and limitations of each model but also contribute novel insights to the burgeoning field of marine ecology. The ensuing sections of this paper will unfold systematically. We will delve into related work, drawing parallels between abalone age prediction and analogous predictive modeling tasks. The proposed methodology will lay the foundation for our exploration, providing a rationale for selecting each predictive model and the anticipated performance based on the dataset's unique characteristics. Through meticulous experiment setups and result discussions, we will bridge theory and practice, evaluating the real-world efficacy of our chosen models.

The forthcoming comparative analysis will serve as the crux of our findings, offering a nuanced understanding of the strengths and weaknesses inherent in each model. As we conclude our research, we will summarize the key takeaways and set the stage

for future investigations. This work not only aspires to elevate the predictive accuracy of abalone age determination but also prompts contemplation on potential enhancements, such as incorporating additional environmental variables.

2 Related Work

Numerous research endeavors have leveraged machine learning (ML) techniques for predictive tasks across diverse domains. This section provides an overview of prior studies in age prediction, explicitly focusing on the abalone species. These investigations lay the groundwork for our exploration into predicting the age of abalones using various ML models.

Guney et al. [2] contributed significantly to understanding abalone age prediction. Their work involved a comprehensive analysis of machine learning algorithms, including backpropagation feed-forward neural networks (BPFFNN), K-Nearest Neighbors (KNN), Naive Bayes, Decision Tree, Random Forest, Gauss Naive Bayes, and Support Vector Machine (SVM). The study recognized the importance of age determination for assessing the economic value of abalones and showcased the efficacy of machine learning models in this context. Our research builds upon this foundation by incorporating advanced ML models and expanding the repertoire of methods for abalone age prediction.

A notable study by Misman et al. [7] specifically applied artificial neural networks (ANN) to predict the age of abalones. The proposed regression-based ANN model, featuring three hidden layers, demonstrated the ability to predict abalone age efficiently. The economic implications of predicting abalone age, such as aiding farmers and sellers in determining market prices, were underscored. Our research aligns with this approach, incorporating regression-based neural networks, albeit with different architectures, to ascertain the age of abalones from physical measurements. We aim to contribute to the existing body of knowledge by exploring and comparing the performance of multiple ML models on the same abalone dataset.

Moreover, a study by Wang [12] highlighted the limitations of traditional machine learning methods in dealing with complex datasets for abalone age prediction. Applying the Cascade Correlation (Cascor) algorithm, a dynamic neural network approach, demonstrated improved efficiency and effectiveness in classification tasks. This research aligns with our exploration of advanced ML models, such as gradient-boosted decision tree regressors and neural networks, to address the intricacies of predicting abalone age from physical measurements. The study's emphasis on the power of artificial neural networks and dynamic algorithms like Cascor resonates with our research goals.

The existing literature showcases the diverse array of ML models employed for abalone age prediction. Researchers have explored various avenues, from traditional methods like decision trees and support vector machines to advanced techniques such as artificial neural networks and dynamic algorithms like Cascor. Our work aims

to contribute to this body of knowledge by rigorously comparing the performance of multiple predictive models on the well-established Abalone dataset, providing insights into their effectiveness and uncovering potential avenues for improvement.

3 Methodology

This section discusses the proposed work, including data retrieval, preprocessing, feature engineering, design of compared machine learning algorithms, and analysis of the results.

3.1 Dataset

This study's Abalone age prediction dataset is sourced from the UC Irvine Machine Learning repository [13]. The dataset was chosen for its relevance to the research objective of predicting the age of abalones based on physical measurements, aiming to provide a more efficient alternative to manual ring counting in their shells. The dataset comprises various features related to the physical characteristics of abalones. The data collection process involved acquiring the dataset, cleaning it, and labeling the target variable, the mollusk's age. The dataset consists of a total of 4178 instances and 9 attributes. As part of the preprocessing steps, we will elaborate on how the dataset was prepared for analysis, including any necessary data normalization or transformation.

3.2 Preprocessing

Kaggle rates the usability of the dataset as 10/10, which means that the dataset is fully documented, available, clean, and ready for use. Indeed, out of 4178 entries in the dataset, there are no missing values, duplicate entries, formatting errors, or varying precision between entries. Therefore, there was no need to use data cleaning techniques such as dropping incomplete rows, replacing the missing values with the median, binning, or manually wrangling string data.

One of the features of the dataset is the Sex of the measured Abalone, which is represented as a discrete value of either "M," "F," or "I." Since multiple linear regression is not fit for dealing with discrete values, we decided to use one-hot encoding to transform the "Sex" column into three columns, "M," "F," and "I," each of which contained a binary value, with 1 indicating a particular sex. This transformation allowed us to use this feature along with the rest in the multiple regression model.

The Abalone dataset has multiple features, each with its unique range of values. This non-uniformity has empirically been shown to negatively impact the convergence of machine learning models due to them assigning relative importance to features based on their values rather than predictive power. Therefore, a standardizing

methodology has been utilized to standardize the dataset's numerical (not categorical) features. The well-known approach to standardization is to convert the numerical value of a feature in a particular sample to its Z-score. This conversion is performed using the following equation:

$$Z = \frac{x - \mu}{s}$$

where μ is the mean of the particular feature and s is the standard deviation. Using this strategy, the values of each feature attain a mean of 0 and a variance of 1, which is a more suitable range for machine learning models.

Finally, the original dataset had the target variable column named "Rings" since that is the indicator of an Abalone's age. The column in the dataset has been renamed to "Age" for clarity.

3.3 Feature selection

The original dataset had 8 feature columns and 1 column for the target variable. After processing the discrete "Sex" column, that number was increased to 11. It was decided to use all of the feature variables in the machine learning for the results of predicting the age of the Abalones. Table 1 contains transformed data that was used to train the models:

	Mean	Std	Min	Max
Length	-5.834718e-16	1.000120	-3.739154	2.423480
Diameter	-3.027929e-16	1.000120	-3.556267	2.440025
Height	3.912493e-16	1.000120	-3.335953	23.683287
Whole weight	9.185853e-17	1.000120	-1.686092	4.072271
Shucked weight	-1.020650e-17	1.000120	-1.614731	5.085388
Viscera weight	2.704723e-16	1.000120	-1.643173	5.286500
Shell weight	2.976897e-16	1.000120	-1.705134	5.504642
M	3.658128e-01	0.481715	0.000000	1.000000
F	3.129040e-01	0.463731	0.000000	1.000000
I	3.212832e-01	0.467025	0.000000	1.000000
Age	9.93	3.224169	1	29

Table 1: Summary statistics for the dataset.

3.4 Machine Learning model selection

The study aims to create an Abalone age prediction model by employing three machine learning techniques: multiple linear regression, gradient-boosted decision trees,

and neural networks, ordered in order of increasing complexity. The study results would be representative of whether an additional complexity of the algorithm is reflected in its performance on this relatively simple dataset.

3.4.1 Multiple Linear Regression

An Ordinary Linear Regression algorithm was chosen to represent the Multiple Linear Regression class of machine learning models, mainly for its simplicity and interpretability. Moreover, regularized algorithms such as Lasso [5], Ridge [11], and Elastic-Net [14] would require an investigation of the effects of l_1 and l_2 regularization coefficients on the performance of the algorithm, which is outside the scope of this paper. In its simplest form, multiple linear regression minimizes the following quantity:

$$\min_{\mathrm{w}} ||Xw - y||_2^2$$

where w is an array of coefficients corresponding to each feature.

3.4.2 Gradient-boosted Decision Trees

The choice to utilize Ada-boosted Decision Trees as representatives of gradient-boosted decision tree regressors in this study stems from the intriguing exploration of Decision Trees (DTs) typically employed in classification tasks and the unique opportunity to evaluate their performance in regression. Decision Trees, renowned for their interpretability and flexibility, are traditionally associated with classification problems, making their application to regression an area of interest. The Ada-boost algorithm [1] was specifically chosen as it is a well-established gradient boosting technique, known for enhancing the predictive power of weak learners like Decision Trees. Ada-boost operates iteratively, assigning more weight to instances with the most significant error in each iteration, thereby sequentially improving the model's accuracy. This ensemble approach is expected to be advantageous in refining the predictive capabilities of Decision Trees for regression tasks, providing a robust representation of gradient-boosted decision tree regressors in our comparative analysis.

3.4.3 Neural Networks

The neural network (NN) selected for this study is a powerful tool for approximating complex functions, making it particularly well-suited for regression tasks. Leveraging the inherent capability of neural networks to approximate any function given a sufficient number of neurons, we tailored our architecture to the simplicity of the abalone age prediction dataset. The chosen neural network comprises two inner layers, each housing 64 neurons. This streamlined architecture was designed to strike a balance

between model complexity and dataset intricacy, ensuring efficient learning without unnecessary overfitting. The rectified linear unit (ReLU) activation function was employed, aligning with industry standards due to its ability to introduce nonlinearity to the model, enabling it to learn intricate patterns within the data. For training, we employed the Adam optimizer, which is currently considered state-of-the-art in optimization algorithms. The Adam optimizer's default settings were utilized, reflecting a commitment to transparency and reproducibility while leveraging the optimizer's adaptive learning rates and momentum-like features.

3.5 Performance Evaluation

The chosen evaluation metric for this study was the Root Mean Squared Error, a widely recognized measure that quantifies the square root of the mean of squared differences between the predicted and actual values. This metric provides a comprehensive insight into the accuracy of the models by emphasizing the significance of large prediction errors. Moreover, it is more intuitive since the error units are the same as the target variable, compared to squared units of MSE. Mathematically, RMSE has the following formula:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

where n is the number of samples in the dataset.

4 Experimental setups and results discussion

In this section, we discuss the experimental setups for each of the machine learning models and the results of the experiments.

4.1 Experimental setup

The experiments were performed on a Macbook Pro 15 2017 running macOS Ventura 13.6.1 with an Intel® Core™ i7-7820HQ CPU running at 2.9 GHz, 4 cores, 16 GB RAM, and a 512 GB SSD. The Python version utilized was 3.11.5. Various third-party libraries were utilized to facilitate the experimentation process:

- **Pandas** 2.1.1 [6] was used for data manipulation and analysis.
- **NumPy** 1.21.2 [3] was used for scientific computing.
- **Matplotlib** 3.4.3 [4] was used for data visualization.

- **SciKit-Learn** 1.3 [10] was used for machine learning, specifically multiple linear regression and gradient-boosted decision tree regressors.
- **PyTorch** 2.1 [9] was used for neural networks.

In evaluating the performance of the machine learning models, the dataset was systematically divided into training and testing sets, with a standard split ratio of 70:30. This division ensured that the models were trained on a substantial portion of the data while retaining a separate and unseen subset for rigorous evaluation. The performance assessment primarily focused on the test dataset, serving as a robust benchmark for model generalization. The process was performed using the SciKit-Learn train-test-split functionality.

4.1.1 Multiple Linear Regression

The multiple linear regression model was configured with default parameters and lacked any form of regularization. Sticking with default settings aimed to maintain simplicity and transparency in the modeling process. The dataset was not subjected to feature engineering to generate second or higher-order regression terms in this setup. Emphasis was placed on preserving linearity, as all relationships between the features were assumed to be linear. This decision aligns with the fundamental assumption of multiple linear regression, which assumes a linear relationship between the independent and dependent variables. Refraining from introducing higher-order terms or complex feature interactions enhances the model's interpretability, allowing for a clear understanding of how each feature contributes to the prediction of abalone age. This straightforward approach aligns with the foundational principles of multiple linear regression while providing a benchmark for comparison against more complex models in the subsequent analysis.

4.1.2 Gradient-boosted Decision Trees

The experimental setup for the gradient-boosted decision tree regressor predominantly employed default parameters, with careful considerations for effective model performance. The base regressor was selected as the `DecisionTreeRegressor` with `max_depth=10`. This decision was driven by the aim to allow the base regressors to adeptly capture nuanced patterns within smaller subsets of the Abalone dataset while maintaining simplicity to facilitate the viability and speed of the gradient boosting process. A `max_depth` of 10 strikes a balance, preventing overfitting while ensuring the individual decision trees can sufficiently contribute to the ensemble. The AdaBoost regressor, an integral part of the gradient-boosting ensemble, was configured with 50 base estimators. This choice represents a deliberate trade-off between model complexity and computational efficiency, acknowledging that an ensemble with 50 base estimators provides a robust representation of the data's underlying patterns

while maintaining a reasonable level of computational speed. The careful selection of these parameters in the gradient-boosted decision tree regressor setup aims to optimize predictive accuracy while managing computational resources effectively.

4.1.3 Neural Networks

For the neural network experimental setup, the architecture consists of an input layer with 10 neurons, each corresponding to a specific feature of the Abalone dataset. Two hidden layers follow, each comprising 64 neurons, striking a balance between model complexity and computational efficiency. Rectified Linear Unit (ReLU) activation functions are employed in the hidden layers, reflecting an industry-standard choice for introducing nonlinearity and enabling the network to capture complex relationships within the data. The output layer comprises a single neuron responsible for predicting the age of the Abalone. The Adam optimizer is employed, utilizing standard parameter values of 0.99 for the exponential decay rates of past gradient moments and 0.999 for the square of past gradient moments. A critical aspect of the experimental setup is the learning rate, a hyperparameter influencing the step size during optimization. To strike a balance between rapid convergence and model stability, we have chosen a learning rate of 0.001. The model was trained for 20 epochs and was evaluated on the test set after every epoch to track its performance trends.

4.2 Experimental result analysis

This subsection discusses the results of the experiments performed on the three machine learning models. The three models were constructed, trained, and evaluated. Afterward, the models were used to predict the test set, and their RMSE was calculated. Table 2 contains the experimentally obtained results:

Model	RMSE
Multiple Regression	2.187416
Gradient Boosting	2.195712
Neural Network	2.113594

Table 2: RMSE of the three models on the test set.

As can be seen, the neural network model performed the best, with the lowest RMSE of 2.113594. The multiple linear regression model performed slightly worse, with the RMSE of 2.187416. The gradient boosting model was the worst, with an RMSE of 2.195712.

The performance of the gradient-boosted regressors could be better, especially given the complexity of the model. While the exact cause of this lackluster performance is hard to determine, it might be due to the use of decision trees outside their

usual domain, classification tasks. It shall also be noted that the model was much slower to train than the multiple linear regression, taking about 5-10 seconds instead of 200ms for the latter. Overall, this result demonstrates that the complexity of the model is not a guarantee of great performance.

With that being said, the slowest and most complex model of the three, the Neural Network, has performed the best. Its loss graph on the train and test sets was fairly smooth, and the model relatively quickly converged to a close-to-optimal solution. Figure 1 shows the loss graph of the model:

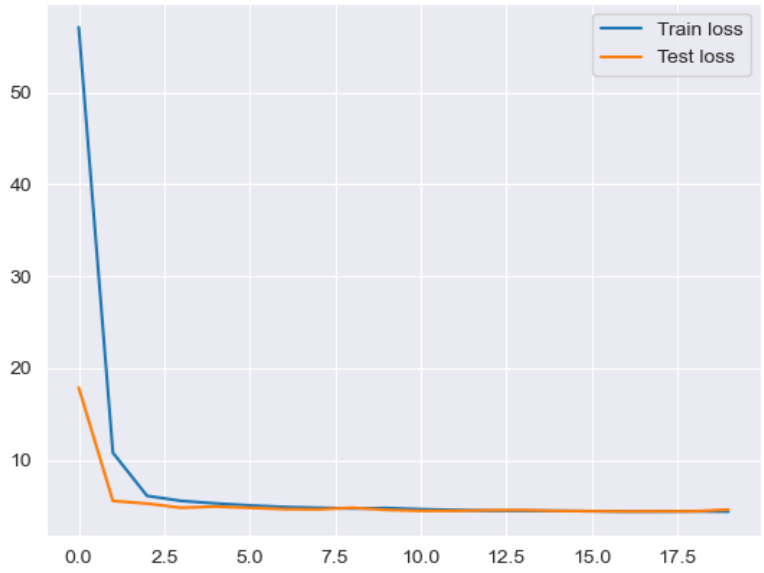

Figure 1: Loss graph of the neural network model.

It is worth noting, however, that at the time of stopping the training process, the training loss was still slowly trending downward, so there is merit in evaluating whether a more extended training process would yield an even better result.

5 Comparison

This section compares the results of our study with previous works in the field of abalone age prediction. Table 3 compares our results with those from other referenced studies. Note that "—" means that the researchers did not use the same approach as our study ([12] used a classification algorithm and, as such, did not provide the MSE metrics).

Study	Model	RMSE
[2]	BPFFNN	1.88
[7]	ANN	1.74
[12]	Cascor	—
This study	Neural Network	2.113594

Table 3: Comparison of the results of this study with previous works.

As can be seen, the results of our study are comparable to those of other researchers. The neural network model performed slightly worse than the BPFFNN and ANN models, but the difference is insignificant. The results of this study are also difficult to compare to the Cascor model since they operate within different problem spaces (regression vs classification). However, these results demonstrate that there is more complexity in the dataset that was evident and that more elaborate models can capture it and perform better when predicting the Abalone's age.

6 Conclusion

In this work, we explored the application of machine learning models to predict the age of abalones based on physical measurements. The proposed methodology involved a comprehensive analysis of multiple linear regression, gradient-boosted decision tree regressors, and neural networks. The experimental setups were designed to balance model complexity and computational efficiency, ensuring that the models were robust and viable for real-world applications. The ensuing comparative analysis provided a nuanced understanding of the strengths and weaknesses inherent in each model, offering insights into their effectiveness and uncovering potential avenues for improvement. The results of this study demonstrated that the neural network model performed the best, followed by the multiple linear regression model, and the gradient boosting model was the worst. The results of this study are comparable to those of other researchers, demonstrating that there is more complexity in the dataset that was evident and that more elaborate models can capture it and perform better when predicting the Abalone's age.

Several areas would be fruitful to investigate in future works. For instance, our multiple linear regression model used no regularization techniques. It is viable that by using regularization, a better performance can be achieved. Moreover, there is a potential for improving the performance of the gradient-boosted regressor, either by adjusting the number of estimators or replacing the base estimator with a different one. Finally, more advanced neural network architectures, such as CNN or RNN, could improve the already good performance of our neural network.

Acknowledgements

The author would like to thank Prof. Haiyan Wang for the opportunity to work on this project and for providing the necessary background knowledge to be successful in its completion.

Author contributions

The author confirms sole responsibility for the authorship and the accuracy of the content of this paper.

Ethical statement

The author confirms that this work is original and has not been published elsewhere, nor is it currently under consideration for publication elsewhere. This article contains no studies with human participants or animals performed by the author.

Data availability

The source dataset is available at [13]. The Jupyter notebook used in the experiments is available at [8]

References

[1] Harris Drucker. "Improving Regressors using Boosting Techniques". In: ().

[2] Seda Guney et al. "Abalone Age Prediction Using Machine Learning". In: *Pattern Recognition and Artificial Intelligence*. Ed. by Chawki Djeddi et al. Vol. 1543. Series Title: Communications in Computer and Information Science. Cham: Springer International Publishing, 2022, pp. 329–338. ISBN: 978-3-031-04111-2 978-3-031-04112-9. DOI: 10.1007/978-3-031-04112-9_25. URL: https://link.springer.com/10.1007/978-3-031-04112-9_25 (visited on 10/23/2023).

[3] Charles R. Harris et al. "Array programming with NumPy". In: *Nature* 585.7825 (Sept. 17, 2020), pp. 357–362. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-020-2649-2. URL: https://www.nature.com/articles/s41586-020-2649-2 (visited on 11/26/2023).

[4] John D. Hunter. "Matplotlib: A 2D Graphics Environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55. URL: http://ieeexplore.ieee.org/document/4160265/ (visited on 11/26/2023).

[5] Seung-Jean Kim et al. "An Interior-Point Method for Large-Scale -Regularized Least Squares". In: *IEEE Journal of Selected Topics in Signal Processing* 1.4 (Dec. 2007), pp. 606–617. ISSN: 1932-4553, 1941-0484. DOI: 10.1109/JSTSP.2007.910971. URL: http://ieeexplore.ieee.org/document/4407767/ (visited on 11/26/2023).

[6] Wes McKinney. "Data Structures for Statistical Computing in Python". In: Python in Science Conference. Austin, Texas, 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a. URL: https://conference.scipy.org/proceedings/scipy2010/mckinney.html (visited on 11/26/2023).

[7] Muhammad Faiz Misman et al. "Prediction of Abalone Age Using Regression-Based Neural Network". In: *2019 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS)*. 2019 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS). Sept. 2019, pp. 23–28. DOI: 10.1109/AiDAS47888.2019.8970983. URL: https://ieeexplore.ieee.org/abstract/document/8970983 (visited on 11/25/2023).

[8] Michael Mykhaylov. *General · mikemykhaylov/mat422Coursework*. GitHub. URL: https://github.com/mikemykhaylov/mat422Coursework (visited on 11/27/2023).

[9] Adam Paszke et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Dec. 3, 2019. DOI: 10.48550/arXiv.1912.01703. arXiv: 1912.01703[cs,stat]. URL: http://arxiv.org/abs/1912.01703 (visited on 11/26/2023).

[10] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12.85 (2011), pp. 2825–2830. ISSN: 1533-7928. URL: http://jmlr.org/papers/v12/pedregosa11a.html (visited on 11/26/2023).

[11] Ryan M Rifkin and Ross A Lippert. "Notes on Regularized Least-Squares". In: ().

[12] Zhengjie Wang. "Abalone Age Prediction Employing A Cascade Network Algorithm and Conditional Generative Adversarial Networks". In: ().

[13] Tracy Sellers Warwick Nash. *Abalone*. 1994. DOI: 10.24432/C55C7W. URL: https://archive.ics.uci.edu/dataset/1 (visited on 10/22/2023).

[14] Hui Zou, Trevor Hastie, and Robert Tibshirani. “On the ”degrees of freedom” of the lasso”. In: *The Annals of Statistics* 35.5 (Oct. 1, 2007). ISSN: 0090-5364. DOI: 10.1214/009053607000000127. arXiv: 0712.0881[math,stat]. URL: http://arxiv.org/abs/0712.0881 (visited on 11/26/2023).