Predicting Abalone Age from Physical Measurements Using Artificial Neural Network

Edward Loa
University of New South Wales
Sydney, Australia

***Abstract*—Determining abalone's age is important for farming and marketing purposes because of abalone's high value. The traditional method of determining abalone's age is time-consuming, which raises the need to develop a simpler—alternative method. Past studies have looked into methods such as logistic regression analysis and decision trees machine learning with their pros and cons. This study aims to provide an alternative machine learning method by utilizing the multiclass artificial neural network. After tuning some of the parameters (number of neurons in the hidden layer, number of hidden layers, learning rate, gradient descent algorithm), the most optimal model was found. The results show that the accuracy of this model is not great in predicting the multiclass abalone age groups but shows promise in binary classification. The issue seems to stem from the imbalanced data in each age group and we recommend future studies to address this issue using resampling techniques.**

***Keywords*—Abalone, Artificial Neural Network**

I. Introduction

For centuries, abalone has been known as a highly valued commodity [1]. Abalone's price is further driven up as many fisheries were closed to conserve the species from excessive harvesting, and as licensed abalone farming is considered a costly venture [1]. Because of this, it is important for farmers to be careful in treating their abalones. Age is one important factor in determining how the abalones should be treated (e.g., different diets based on age, different tank/cage placement [1], harvesting and seeding age [2]). Furthermore, the economical value of abalone could be based on their age as well [3], which is why determining abalone age accurately is crucial for both marketing and farming purposes. Traditionally, the abalone shell rings are counted to determine their age. However, this process is time-consuming as it requires multiple tasks involving the cutting of the shell, staining the shell in the laboratory to increase the rings visibility, and counting the rings under the microscope [4]. Hence, to minimize cost, an easier--alternative method of predicting abalone age needs to be developed.

Past studies have attempted to find simpler methods to predict abalone age based on its physical measurements. For instance, in one paper, logistic regression and decision tree methods were used [5]. Regression model is an algebraic equation that can predict the output based on weighted input values, which is built from statistical principles like correlation and least-square error [6]. Although logistic regression model is relatively easy to understand, the model assumes linear relationships between variables and the model is most reliable with a small number of input variables [6]. As there are plenty of physical variables that can be used to predict age, and as it is possible that the relationship between variables is not linear, it is necessary to consider other predictive options.

The second method that the author [5] used is decision trees machine learning. In this method, the input data--abalone's physical measurements--are trained to continuously split based on assigned parameters until they reach the final decisions/outcomes [7]. While the decision tree does not impose many restrictions such as linearity assumption, the decision tree model may cater for the training input data too much that it leads to an overfitting problem (i.e., the model works best only on the training data, but is not generalisable to new/different data) [8]. As these past methods have their own limitations, this study aims to contribute to this field by incorporating an alternative method to predict abalone age.

One method that can handle nonlinear and complex data is the artificial neural network (ANN). Mimicking the human neural system, ANN aims to create a computational system to provide solutions. The first part of this process is forward propagation where the weighted sum of the input information (akin to input from dendrites) enters the hidden layer of neurons (similar to nucleus). Each neuron processes these inputs using activation functions to calculate/predict an output value that will be sent to the next layer of neurons until it reaches the last layer [9]. The second process is backpropagation, where the algorithm gradient descent is then used on the last output layer to minimise the error between the predicted and the actual values (cost function), which will then be applied to update the weight of the previous layer. This process is then repeated until it reaches the original input layer [10]. Forward and backpropagation will then cycle until it reaches a set number of cycles (epoch) or the error meets a certain value.

Building an optimal ANN requires the testing of some features. For instance, more number of hidden layers may cause overfitting whereas low number of hidden layers may generate high bias [11]. Past studies have attempted to determine the optimal number of neurons in each layer and the number of hidden layers itself with inconclusive results and viewed the solution as a case-to-case basis [12][13]. The gradient descent algorithm, on the other hand, depends on the parameter 'learning rate'. Learning rate is the 'stepping size' that is taken on each iteration to reach the minimum cost function. A low learning rate means it can be more precise but requires a higher number of iterations which reduces the efficiency of the model to reach the minimum, and conversely, a high learning rate has a risk to overpass the minimum [14]. Hence it is crucial to test the learning rate as well. Alternatively, it is possible to test the model without knowing the learning rate by using another algorithm known as Adam which can adapt the learning rate during the training [15].

Taken together, this study aims to develop a simpler method to predict abalone age based on its physical measurements by utilizing ANN. With ANN, this paper will complement past studies that have attempted to predict

abalone age using regression and decision trees machine learning. To optimize the model, the ANN will be tested with different number of neurons per hidden layer, different number of hidden layers, different learning rate, and different gradient descent algorithm.

The next section of this paper will describe the data used in this project, the input and output of this task as well as the workflow of the project. The quantitative results will then be presented and discussed in section 3. Finally, section 4 will conclude this paper by summarising the results, discussing the implication of this study, as well as providing direction for future research.

TABLE 1. DATASET DESCRIPTION

Column Name	Data Type	Meas.	Description
Gender	nominal		M, F, and I (infant)
Length	continuous	mm	Longest shell measurement
Diameter	continuous	mm	perpendicular to length
Height	continuous	mm	with meat in shell
Whole weight	continuous	grams	whole abalone
Shucked weight	continuous	grams	weight of meat
Viscera weight	continuous	grams	gut weight (after bleeding)
Shell weight	continuous	grams	after being dried
Rings	integer		+1.5 gives the age in years

II. METHODOLOGY

Python 3.10.4 [16] and Keras 2.9.0 package[17] were used to build the ANN model and to analyse the data. The dataset was sourced from The University of California Irvine which contains physical measurements of abalones in the North Coast Islands of the Bass Strait in Tasmania [4]. The dataset is described in Table 1 above. The input data are the physical variables of the abalone. The output is the abalone ring age. The age data were split into 4 age groups shown in Table 2, and the goal is to successfully predict the age group using the model. The workflow process can be seen in Figure 1 below.

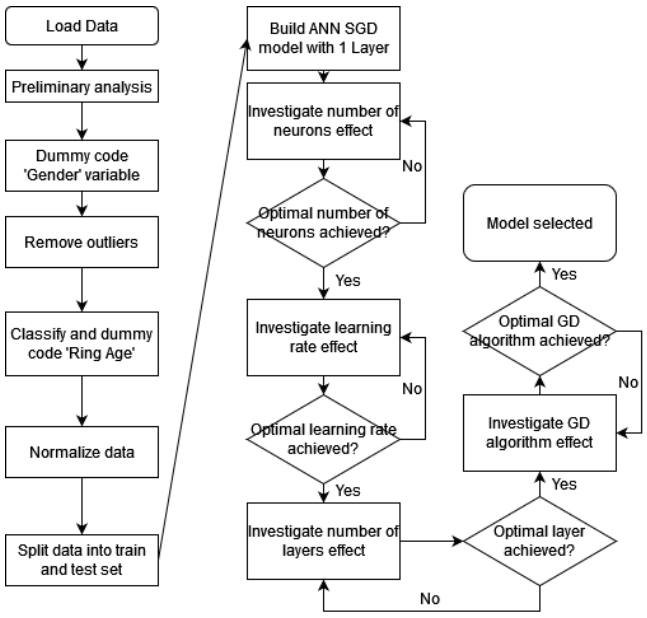

Fig. 1. Workflow chart

After preprocessing and splitting the data into training and test set with 60:40 ratio, the initial model was built using the training dataset with 5 neurons in one hidden layer, 0.1 learning rate, 200 epochs, 10 batch size, SGD algorithm optimiser, and repeated 10 times. The topology of this initial model is 7 neurons in the input layer (the physical variables of the abalone) to 5 neurons in one hidden layer to 4 neurons in the output layer (4 classes of abalone ring age). The activation function of the input and hidden layer/s is Rectified Linear Unit (ReLu) [18], and the activation function of the output layer is softmax [19] as this is a multi-classification study.

Next, we tuned the parameters of this initial model and evaluated them using the mean predictive-accuracy score. The model with the highest accuracy score is considered to have the best optimal parameter. Firstly, the number of neurons in one hidden layer were tested (5 vs 10 vs 15 vs 20). The model then incorporated the optimal number of neurons from here on. Next, the learning rates were tested (0.1 vs 0.3 vs 0.5 vs 0.7 vs 0.9). Again the best learning rate was selected and incorporated for the subsequent tests. The third test compared the effect of the number of hidden layer/s (1 vs 2). Using the optimal number of hidden layer/s, the last test compared the effect of optimizer (Adam vs SGD algorithm).

After finding the best model, a confusion matrix was produced along with the ROC/AUC binary classification curves. The normalized confusion matrix represents how the model fares in correctly predicting the true value. Similarly, ROC curves represent the probability of predicting the correct outcome, while AUC is the area below the ROC curve (how capable that model is to differentiate the classes) [20]. Hence, a higher area under the curve is better to predict the outcome. However, the AUROC curves can only represent binary predictions of each class individually (i.e., class 1 vs not class 1).

III. Results and Discussion

A. *Preliminary Analysis and Data Cleaning*

There were no null values detected in the dataset. Gender data were coded into numerical values (0: Female, 0.5: Infant, 1: Male). Several outliers (z-score > 3) were removed to smooth the data analysis, reducing the data from 4177 to 4027 rows. Ring Age data were classified into four classes and then turned into one-hot encoding format. The distribution of the Ring Age is described in Table 2. The class distribution seems to be unbalanced especially on class 4. Looking at the histograms of the input variables presented in Figures 2-8, it appears that most of the input data are skewed and hence the input data were normalized/scaled.

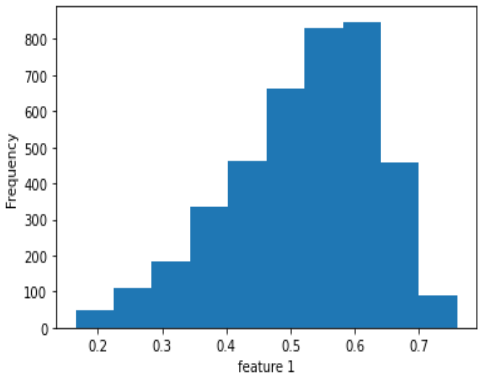

Fig. 2. Distribution of Length Data

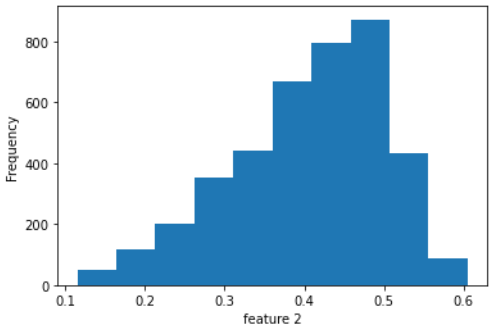

Fig. 3. Distribution of Diameter

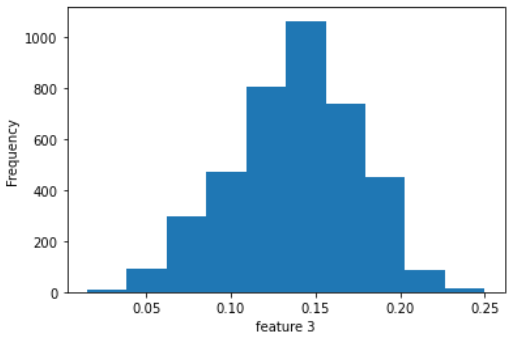

Fig. 4. Distribution of Height

TABLE 2. RING AGE DATA DISTRIBUTION

Class 1: 0-7 years	821
Class 2: 8-10 years	1877
Class 3: 11-15 years	1143
Class 4: >15 years	186

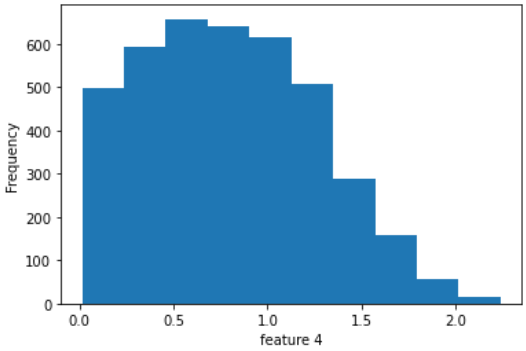

Fig. 5. Distribution of Whole Weight

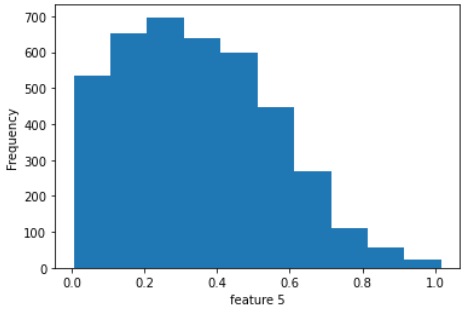

Fig. 6. Distribution of Shucked Weight

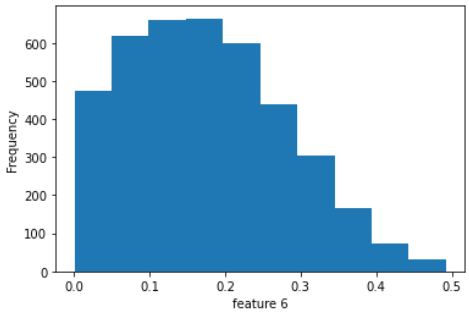

Fig. 7. Distribution of Viscera Weight

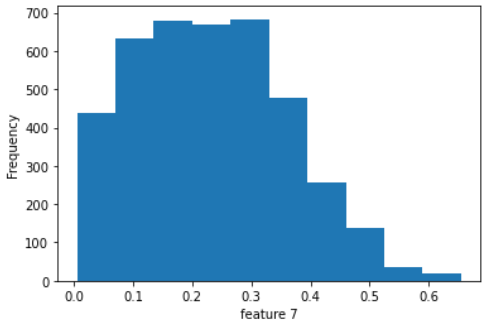

Fig. 8. Distribution of Shell Weight

B. *Finding the Most Accurate Model*

The results of the first investigation (SGD algorithm, 5-20 neurons, 1 hidden layer, 0.1 learning rate, 10 experiments) are shown in Table 3. These results have shown that the mean scores are similar across all cases, but the model with 15 neurons provides the best mean accuracy score.

TABLE 3. ACCURACY SCORES USING DIFFERENT NUMBER OF NEURONS

Number of Neurons		Mean Accuracy Score	SD of Mean Accuracy Score	Confidence Interval	
5	Test	0.6001	0.0447	0.5701	0.6301
	Train	0.6286	0.0514	0.5941	0.6631
10	Test	0.6138	0.0045	0.6107	0.6168
	Train	0.6465	0.0038	0.6440	0.6490
15	Test	**0.6150**	0.0032	0.6129	0.6172
	Train	0.6478	0.0022	0.6463	0.6493
20	Test	0.6142	0.0043	0.6113	0.6171
	Train	0.6475	0.0046	0.6444	0.6506

The scores of the second investigation (SGD algorithm, 15 neurons, 1 hidden layer, 0.1-0.9 learning rate, 10 experiments) are shown in Table 4. The mean scores, again, are similar across the learning rates but the best accuracy score is obtained with 0.7 learning rate.

TABLE 4. ACCURACY SCORES USING DIFFERENT LEARNING RATES

Learning Rate		Mean Accuracy Score	SD of Mean Accuracy Score	Confidence Interval	
0.1	Test	0.6136	0.0044	0.6106	0.6166
	Train	0.6482	0.0042	0.6454	0.6510
0.3	Test	0.6151	0.0036	0.6126	0.6175
	Train	0.6495	0.0034	0.6472	0.6518
0.5	Test	0.6161	0.0026	0.6143	0.6178
	Train	0.6498	0.0017	0.6487	0.6509
0.7	Test	**0.6164**	0.0036	0.6140	0.6188
	Train	0.6481	0.0037	0.6456	0.6506
0.9	Test	0.6142	0.0037	0.6117	0.6166
	Train	0.6478	0.0037	0.6453	0.6503

Table 5. shows the results of the third investigation (SGD algorithm, 15 neurons, 1 vs 2 hidden layers, 0.7 learning rate, 10 experiments). The mean accuracy score for the model with 1 hidden layer appears to be better than the model with 2 hidden layers.

TABLE 5. ACCURACY SCORES USING DIFFERENT LAYERS

Number of Layer		Mean Accuracy Score	SD of Mean Accuracy Score	Confidence Interval	
1	Test	**0.6164**	0.0036	0.6140	0.6188
	Train	0.6481	0.0037	0.6456	0.6506
2	Test	0.6099	0.0109	0.6022	0.6178
	Train	0.6347	0.0123	0.6259	0.6435

The results of the last investigation (SGD vs Adam algorithm, 15 neurons, 1 hidden layer, 0.7 learning rate, 10 experiments) are shown in Table 6. The model using Adam algorithm appears to be more accurate, which means this is the best model in this study. Note that the best accuracy score only hovers around 0.62, which is rather poor. Additionally, all the models seems to not overfit as the train mean accuracy scores are not that different from the test mean accuracy scores.

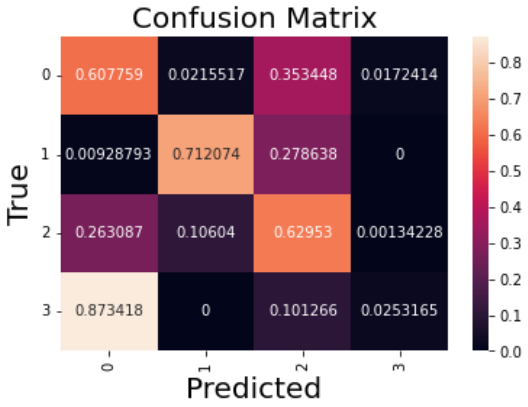

Fig. 9. Confusion Matrix

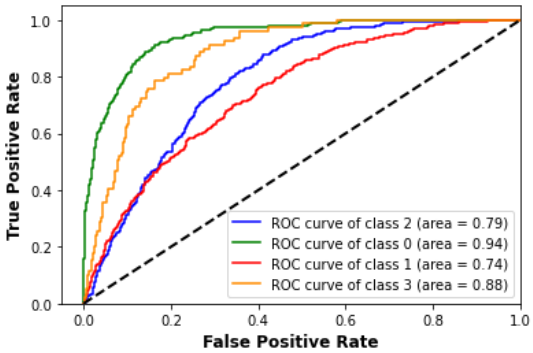

Fig. 10. The Binary Class AUROC Curves

The normalized confusion matrix and binary class AUROC curves from the best model experiment run are presented in Figures 9 and 10. Note that 0 here represents class 1, whereas 1 represents class 2, and so on. Based on the confusion matrix, the accuracy of this model is overall poor in predicting classes 1 and 3, decent in predicting class 2, and incapable of predicting class 4. On the contrary, the AUROC curves show that the model is good in predicting class 4 (binary prediction) and especially good in predicting class 1. This could be because the number of class distribution is uneven (very small numbers of class 4, whereas class 2 and 3 have many samples) which makes the model more biased towards predicting class 2 and class 3. This bias then dilutes the confusion matrix and reduces the overall accuracy score of the model as it is incapable of correctly predicting class 4 in a multiclassification setting.

TABLE 6. ACCURACY SCORES USING DIFFERENT GD ALGORITHM

Algorithm		Mean Accuracy Score	SD of Mean Accuracy Score	Confidence Interval	
SGD	Test	0.6164	0.0036	0.6140	0.6188
	Train	0.6481	0.0037	0.6456	0.6506
Adam	Test	**0.6188**	0.0021	0.6172	0.6204
	Train	0.6576	0.0025	0.6558	0.6594

IV. Conclusion

To conclude, it appears that the best ANN model that can predict the classes of the abalone age group is the model that uses Adam algorithm with 15 neurons in 1 hidden layer. However, the model predictive power is poor, possibly because the data is imbalanced (very small samples on class 4). This is especially true as the binary predictive power of the model on class 4 is shown to be high in the AUROC curves. Because of the overall low multiclass predictive accuracy, it is still inconclusive to claim that ANN is a good method for multiclass prediction of abalone age based on its physical measurements, and that future studies are needed to extend this study's results.

A possible solution that can be incorporated in future studies is by altering the age range in each class. Alternatively, resampling techniques can be used such as over-sampling the minority class (i.e., class 4 and class 1) until they are represented equally. One example of an over-sampling technique is the Synthetic Minority Over-Sampling Technique (SMOTE) [21]. This method creates synthetic samples for the minority class by linking the k-nearest neighbours of the minority class. This technique, however, has the risk to overfit the model.

References

[1] Agricultural Marketing Resource Center, "Abalone", Apr. 2022. Accessed: Jun. 6, 2022. [Online] Available: https://www.agmrc.org/commodities-products/aquaculture/aquaculture-non-fish-species/abalone

[2] Greenfish, "Abalone (perlemoen) | live box | cultivated | x6", Accessed: Jun. 6, 2022. [Online] Available: https://greenfish.co.za/products/live-box-abalone-perleomoen

[3] L. McDonald, "Jewels of the deep – Abalone", Mindfood, Aug. 20, 2020. Accessed: Jun. 6, 2022. [Online] Available: https://www.mindfood.com/article/jewels-of-the-deep/

[4] D. Dua, and C. Graff, "Abalone data set", UCI Machine Learning Repository, 2019. Accessed: Jun. 6, 2022. [Online] Available: https://archive.ics.uci.edu/ml/datasets/abalone

[5] K. Mehta, "Intro to machine learning via the abalone age prediction problem", Jul. 2020. Accessed: Jun. 6, 2022. [Online] Available: https://kunjmehta10.medium.com/intro-to-machine-learning-via-the-abalone-age-prediction-problem-4e290a8b2ed3

[6] M. H. Vidyashri, "Advantages and disadvantages of regression model", VTUPulse. Accessed: Jun. 6, 2022. [Online] Available: https://www.vtupulse.com/machine-learning/advantages-and-disadvantages-of-regression-model/

[7] J. R. Quinlan, "Induction of decision trees", *Machine Learning*, vol. I, pp. 81-106, Kluwer Academic Publishers, Boston, 1986. Accessed: Jun. 6, 2022. [Online] Available: https://link.springer.com/content/pdf/10.1007/BF00116251.pdf

[8] X. Ying, "An overview of overfitting and its solutions", *Journal of Physics: Conference Series*, vol. 1168(2), 2019. Accessed: Jun. 6, 2022. [Online] Available: https://iopscience.iop.org/article/10.1088/1742-6596/1168/2/022022/meta

[9] P. Skalski, "Deep dive into math behind deep networks", Towards Data Science, 2019. Accessed: Jun. 6, 2022. [Online] Available: https://towardsdatascience.com/https-medium-com-piotr-skalski92-deep-dive-into-deep-networks-math-17660bc376ba

[10] SAS, "Neural networks – What they are & why they matter", Accessed: Jun. 6, 2022. [Online] Available: https://www.sas.com/en_us/insights/analytics/neural-networks.html

[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting", *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014. Accessed: Jun. 6, 2022. [Online] Available: https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf

[12] M. Adil, R. Ullah, S. Noor, and N. Gohar, "Effect of number of neurons and layers in an artificial neural network for generalized concrete mix design", *Neural Computing and Applications*, vol. 34, pp. 8355-8363, 2022. Accessed: Jun. 6, 2022. [Online] Available: https://link.springer.com/article/10.1007/s00521-020-05305-8#citeas

[13] A. Gad, "Deduce the number of layers and neurons for ANN", Datacamp, Sept. 2018. Accessed: Jun. 6, 2022. [Online] Available: https://www.datacamp.com/tutorial/layers-neurons-artificial-neural-networks

[14] IBM Cloud Education, "Gradient descent", Oct. 2020. Accessed: Jun. 6, 2022. [Online] Available: https://www.ibm.com/cloud/learn/gradient-descent

[15] D. P. Kingma, and J. L. Ba, "Adam: a method for stochastic optimization", *ICLR*, 2015. Accessed: Jun. 6, 2022. [Online] Available: https://arxiv.org/pdf/1412.6980.pdf

[16] Python, "Download Python 3.10.4", Accessed: Jun. 6, 2022. [Online] Available: https://www.python.org/downloads/

[17] Keras, "Keras 2.9.0". Accessed: Jun. 6, 2022. [Online] Available: https://anaconda.org/conda-forge/keras

[18] P. Ramachandran, B. Zoph, and Q. V. Le. "Searching for activation functions". Google Brain, Oct. 2017. Accessed: Jun. 6, 2022. [Online] Available: https://arxiv.org/pdf/1710.05941.pdf

[19] Bala Priya C. "Softmax activation function: everything you need to know", Pinecone, Accessed: Jun. 6, 2022. [Online] Available: https://www.pinecone.io/learn/softmax-activation/

[20] V. Cortez, "Understanding the ROC curve in three visual steps", Towards Data Science, 2021. Accessed: Jun. 6, 2022. [Online] Available: https://edstem.org/au/courses/8454/lessons/20312/slides/144649

[21] N. V. Chawla, K. W. Bowyer, L.O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique", *Journal of Artificial Intelligence Research*, vol. 16, pp 321-357, 2002. Accessed: Jun. 6, 2022. [Online] Available: https://arxiv.org/pdf/1106.1813.pdf