Machine Learning Project

Felipe Quiterio

Problem:

The age of the Abalone must be taken by physical measures. Abalone is a snail so the age must be taken by cutting through its shell, staining it and counting the number of rings present in it through a microscope. Manual tasks like this usually takes a lot of time to finish, so other measurements easier to obtain can also be used to predict the age.

Motivation:

There is an issue of determining the age of the Abalone snail that is the complexity in which the age is calculated, by having to manually cut the shell and stain it to count the rings, and this task is very time consuming so by using features of the Abalone, a machine learning model can be created to predict the age by recognizing a pattern

Dataset Information:

The dataset contain 8 attributes as follows.

Sex, nominal in F, M or I (infant)

Length, continuous in millimeters

Diameter in millimeters

Height in millimeters

Whole weight in grams

Shucked wight in grams

Viscera weight in grams

Shell weight in grams

Rings in integer, + 1.5 give the ages in years

Feature Processing and Feature Engineering:

The first feature processing to be made is with the first column containing sex classification of the Abalone. To be able to process this data, a numerical encoding was performed. Using ordinal encoder, the 3 classifications, male (M), female (F) and infant (I), were encoded respectively as 2.0, 0.0 and 1.0.

Features were also required to be scaled as one of the algorithms used is K Nearest Neighbor. The scaling method used is Standard Scaler

To the second algorithm, the first feature had also to be numerically encoded same as the previous model and data had to be split between training and testing using shuffle split with 1000 splits and test size of 5. A high number of splits were chosen due to testing with accuracy. Randomness was added using random_state parameter.

To fit in the second algorithm no scaled data was required but was also tested with and concluded that had no difference in the result of the model performance.

Machine Learning Model Development:

The first algorithm chosen was K Nearest Neighbor, the model requires scaled data so Standard Scaling was applied.

Target column had also to be transformed as the label type was continuous and invalid to K Nearest Neighbor algorithm so was transformed to multiclass in order to fit.

To be able to choose the best value of k, a loop was used to test the accuracy of the model ranging from 1 to 20 neighbors.

To the second algorithm, it was chosen Linear Regression. After the model was trained, we can see that the R^2 score is negative, scoring -0.917 and mean absolute error and mean squared error also gave values that show that the model does not fit right the data.

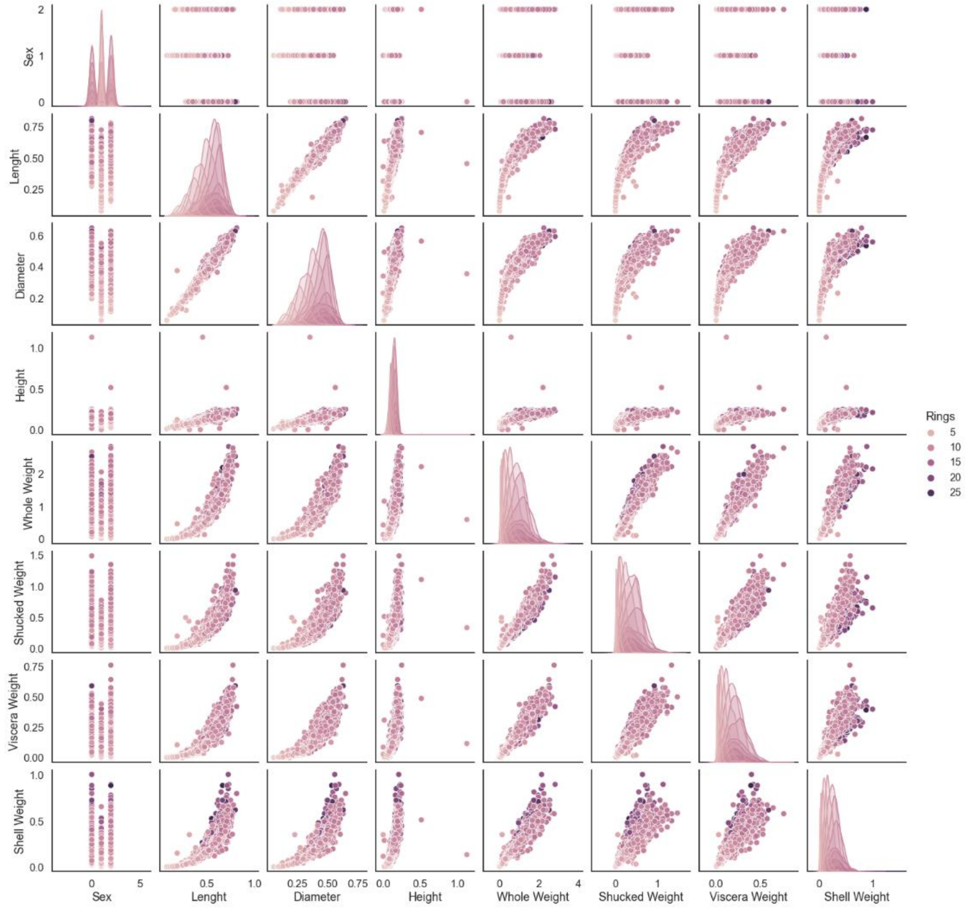
Prediction / Result:

The first model, using K nearest neighbor algorithm had an 83% accuracy with a best fit for k being 7. Values of k were tested between 1 and 20 having low accuracy change as k reaches 20, so 7 was chosen to reduce the time for compilation.

For the second model, Linear Regression, R^2 score is negative, scoring -0.917, mean absolute error is 4.434 [good] and mean squared error equals 5.181.

Evaluating the Results and metrics:

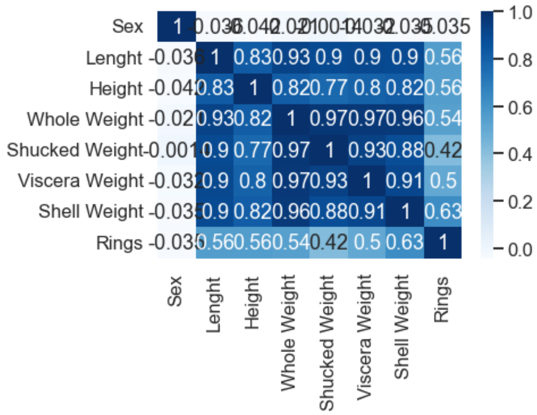
To evaluate the results, the pair plot needs to be looked at to see if any of the features has a direct relationship with the number of rings in the abalone shell, but as we look closely, we see that is no relationship between them.



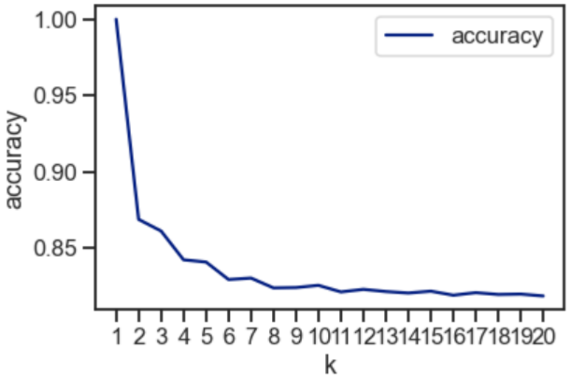
The first algorithm had a good accuracy score for 83% with the accuracy score stopping to vary a lot when k reaches 7.

The second algorithm, the R^2 value is negative showing that the model fits poorly the data, and probably a Support Vector Regression should be used instead

Also, a generated heatmap can be used to see this relation, and we see that the closest relationship between them is shell weight and ring count with a 0.63 correlation.



Now, evaluating the algorithm metrics we see how value of k affect the accuracy of the model and as soon k reaches 7 the accuracy does not change much.



Conclusion:

To conclude, the best algorithm to fit the model is K Nearest Neighbors due to the easiness to implement in the assigned data and an accuracy value in 83% when stabilizes. Linear Regression was not the best algorithm to use as we see that the data does not follow a liner pattern and a non-linear regression method should be used instead.

References:

https://scikit-learn.org/stable/supervised_learning.html#supervised-learning

https://pandas.pydata.org/docs/user_guide/index.html#user-guide

https://matplotlib.org/stable/users/index.html

https://seaborn.pydata.org/