Abalone Age Prediction Using Principal Component Analysis & Multiclass Classification

Asra Saeed [40094955]
Github link: https://github.com/asrask65/INSE-6220

Abstract - In this research paper we will be discussing about the most sought out advanced technique for unsupervised method in machine learning; Principal Component Analysis (PCA). PCA is one of the most widely used tool for the purpose of dimensionality reduction and it's featured to analyse data in machine learning for predictive models. For this report, I have used PCA to evaluate and analyse the age of Abalone. Due its aphrodisiac feature abalone is one of the world's most expensive seafood. In this research report, the analysis done on the dataset classifies the different types into 3 categories namely, Male, Female, and Infant. I have used different classifier to successfully predict the data among the 3 classes based on their age and other physical measurements with a

Keyword -- Machine Learning, Principal Component Analysis (PCA), Age Prediction, Model Training, Multiclass classification, KNN, Decision Tree, Logistics Regessuion

I. INTRODUCTION

Abalone is a marine gastropod mollusk. This large snail is mostly found in the cold waters of Australia, New Zealand, South Africa, Japan and west coast of North America [12]. Abalone is a marine snail, belonging to genus Haliotis and family Haliotide for the class Gastropoda; this is one of the world's finest and most expensive seafood for its extremely rich flavour and highly prized meat that is of culinary delicacy. It is because of such popularity; the acquiring of abalone led to overfishing and nearly brought this species to extinction due to which it has now been labelled illegal to sell widely.

II. DATASET DESCRIPTION

This dataset is acquired from the UCI Machine Learning Archive. The dataset was gathered by dissecting the shell through the cone, staining it, and counting the number of rings through microscope [18]

The dataset contains of 113 records, out of which 49 are classified as Female, 14 are classified as Infant and remaining 50 are classified as Male gender.

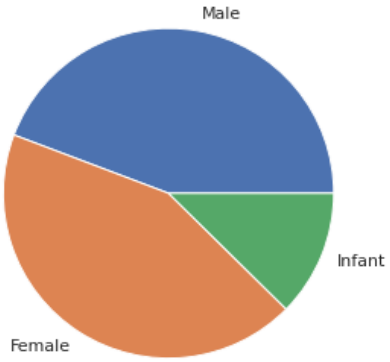

Figure 1: Pie chart

All of which are categorized based on the following physical measurement metric:

1. Length: Longest shell measurement (mm)
2. Diameter: Perpendicular to length (mm)
3. Height: Height along with meat (mm)
4. Whole weigh: whole abalone (mm)
5. Shucked weight: weight of meat (grams)
6. Rings: +1.5 gives the age in years

III. PCA METHODOLOGY

Generally, we work with 2-dimensional data to draw proceeding explanations by implying all machine algorithm methods and techniques on them for they are relatively easier to comprehend. However, datasets in real life are often complex based on their dimensionality and intricacy. Due to this it becomes

cumbersome for machine learning to perform well in higher dimensions. In some certain scenarios such limitations can result in poor accuracy, the variables come across being correlated on analysis, it becomes indecisive to implement strategic methods. For these situations, PCA comes into play, for taking control of such overwhelming datasets by reducing their dimensionality yet retaining the important components and variables in the dataset [1]. PCA tends to transform larger set of data into smaller ones with valuable information which are the new variables called Principal Components (PCs), this helps in visualizing and analyzing the data easily, making it comprehensible for the machine learning algorithms to process smoothly on them.[1][2][3].

STEP-1:
Standardization

The process of standardization means to scale the data in such a way to have all the variables and its corresponding values align within the same range. Without having to do this, we would risk having biased and inaccurate records that may affect the desired result. [1]

To start off we will center the data by subtracting the mean and dividing the result with the standard deviations for each acquired value.[1]

$$z = \frac{value - mean}{standard\ deviation}$$

Or for matrix, calculate the centered data matrix Y = HX by subtracting off-column means.[3]

STEP-2:
Covariance Matrix Computation

The main concept of this step is to evaluate the variables for the input data set to see how much they vary from the mean with respect to each other, to determine their relational strength and dependency. [1]
For computing the covariance matrix, where this centered data matrix M, is a $p \ x \ p$ entry recorded as follows:

$$S = \frac{1}{n-1} Y' Y$$

This computes the matrix to have symmetrical main diagonal.

STEP-3:
Eigenvectors & Eigenvalues

It is important to remember that every eigenvector has an eigenvalue and that this is equal to the number of the dimension in the dataset. These eigenvectors indicate the direction of the axes where there is most variance while the eigenvalue are simply the coefficients associated to these eigenvectors.[1][2]

So to compute the eigenvectors and eigenvalues of the matrix M using eigen-decomposition we use the following method:

$$S = A \wedge A' = \sum_{J=1}^{P} \lambda_J a_j a_j'$$

STEP-4:
Principal Component & Transformation

Now to rearrange the data with final principal components, representing the most significant information of the dataset.

Compute the transformed data matrix Z of size $n \ x \ p$ which is

Z = YA

$$Z = \left(z'_1, z'_2, \ldots, z'_i, \ldots, z'_p \right) = \begin{bmatrix} z_{11} & z_{12} & \cdots & \cdots & z_{1p} \\ z_{21} & z_{22} & \cdots & \cdots & z_{2p} \\ . & . & . & . & . \\ . & . & . & . & . \\ z_{n1} & z_{n2} & \cdots & \cdots & z_{np} \end{bmatrix}$$

This is done to obtain the feature vector to replace the original data axis with the newly constructed PCs.

IV. PCA IMPLEMENTATION & RESULTS

Due to the wide range of libraries available in MATLAB, the use of PCA in Python programming was much easier to achieve and the corresponding graphical and visual results were as following:

1- Box Plot

Box Plot illustrates the distribution of all attributes that are normalized. The central line denotes the mean value while the length of the extended lines denotes the remaining data for that attribute along with the points marked beyond the lines, which are declared as outliers of our dataset.

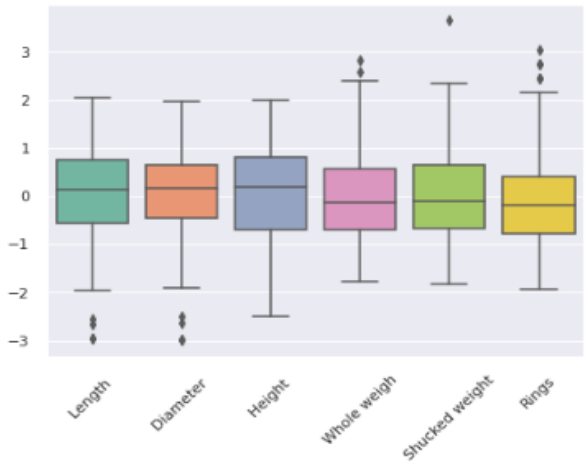

Fig 2: Box and whisker plot of centered featured

2- Strip plot

A strip plot complements to a boxplot where all the observations are shown with some representation of underlying distribution, it draws a scatter plot based on the category [15]

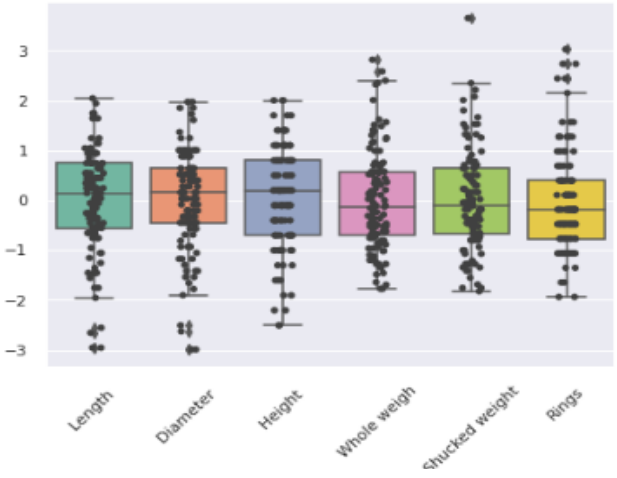

Fig 3: Strip Plot

3- Pair Plot

Pair plot demonstrates the relationship between all the numerical attributes and fields of the dataset individually and their relation to one another. The linear flow identifies how strongly associated the variable are.

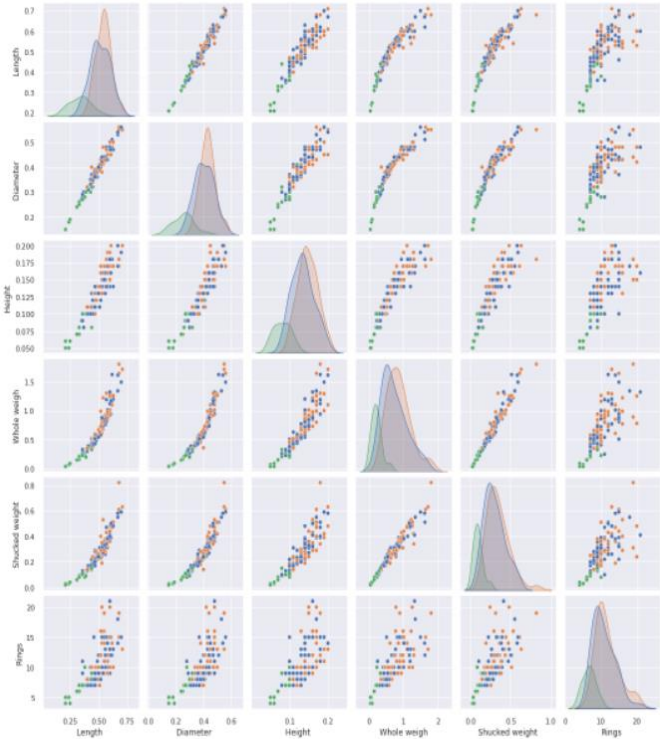

Fig 4: Pair Plot

4- Covariance Matrix

Covariance Matrix is a symmetric matrix that shows the distribution magnitude and direction of multivariate data in n-dimensional space.[13] With the help of these values we can extract information on how widely or closely spread the data is among 2- dimension. The figure below shows the relationship of each attribute corresponding to one another and itself.

Fig 5: Covariance Matrix

5- Eigenvector:

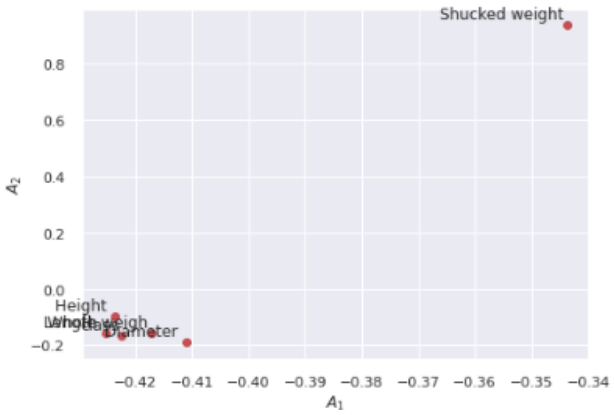

Fig 6: Eigenvectors

The implementation of PCA resulted in the reduction of the dataset from six features (p =6) to r features with r < 6. For the original dataset of $n \times p$, this is reduced down using the eigenvector matrix (A).

Here the eigenvector matrix [A]=

$$\begin{bmatrix} -0.4223 & -0.1672 & -0.2736 & -0.5339 & -0.1250 & -0.6466 \\ -0.4252 & -0.1580 & -0.2410 & -0.4149 & 0.1892 & 0.7266 \\ -0.4110 & -0.1904 & -0.5242 & 0.7056 & -0.1477 & 0.0145 \\ -0.4236 & -0.0963 & 0.4419 & 0.2047 & 0.7318 & -0.1959 \\ -0.4172 & -0.1560 & 0.6283 & 0.0535 & -0.6235 & 0.1232 \\ -0.3438 & 0.9365 & -0.0460 & 0.0081 & -0.0490 & 0.0046 \end{bmatrix}$$

The corresponding eigenvalues are:

$$\lambda = \begin{bmatrix} 5.3200 \\ 0.4327 \\ 0.1541 \\ 0.1032 \\ 0.0279 \\ 0.0154 \end{bmatrix}$$

The percentage of accumulated variance for j_{th} PC is calculated with the help of the following formula:

$$\ell_j = \frac{\lambda_j}{\sum_j^p \lambda_j} \times 100\%, \quad for\ j = 1, \ldots, p$$

The variance explained by the first 2 PCs are ℓ_1 = 87.8% and ℓ_2 = 7.14% which accumulates to a total of 97.57% of the variance for the original dataset

Following are the principal components:

Z1 = (-0.42234214)*X1 + (-0.4252851)*X2 + (-0.41104151)*X3 + (-0.42367101)*X4 + (-0.41724171)*X5 + (-0.34382671)*X6

Z2 = (-0.16721158) *X1+ (-0.15803489) *X2+ (-0.19044596) *X3 + (-0.09630782) *X4 + (-0.15603693) *X5 + (0.93657508) *X6

6- Scatter Plot

A scatter plot comprises of dots that represents value for two or more different numeric variables that of which helps in understanding the relationship and correlation between them. Below given is a scatter plot for out dataset that represents the class namely Male, Female and Infant for all 113 records.

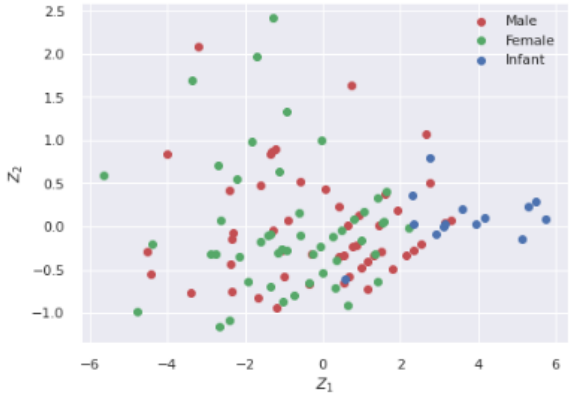

Fig 7: Scatter Plot

7- PC Coefficient

The PCA here demonstrates the reducing the dimension of the dataset. With the help of PC coefficient plot we can demonstrate here the impact of each variable on the PCs.

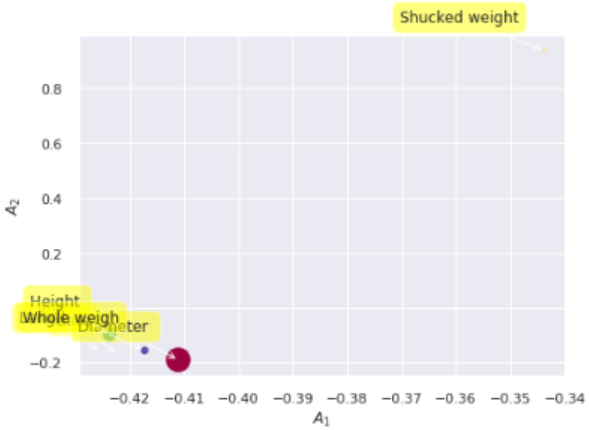

Fig 8: PC coefficient plot

8- Scree Plot

A scree plot is a graphical tool used to select the number of relevant components or factors that would be considered in the principal component analysis. The scree plot demonstrates the eigenvalues on y-axis and number of factors on x-axis, and the point where the slope is curve indicating "the elbow" denotes the number of factors generated by the analysis. [16][17]. The first two components accumulate to a 97.57% of the total variance therefore the matrix is reduced to r=2 as only PC1 and PC2 deemed to have satisfied the criteria.

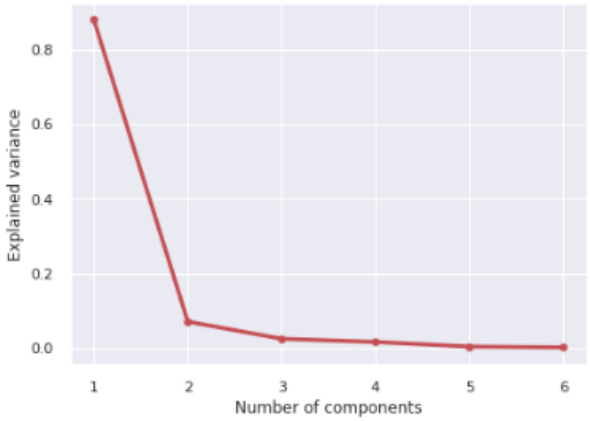

Fig 9: Scree Plot

9- Pareto Chart

With the help of Pareto chart, it can be inferred that the explained variance is represented by bars, achieving 2 PCs with 97.57% of the variance

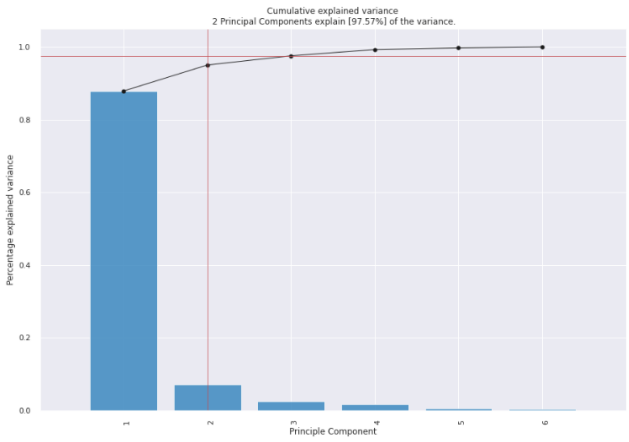

Fig 10: Pareto Chart

10- Bi Plot

This Biplot is a graphical display of information for both observations and the data variables. In this the biplot depicts both principal components coefficients and principal component scores for each record. As shown below, the variables here are associated with a vector of directional length indicating contribution of both principal component in the biplot.

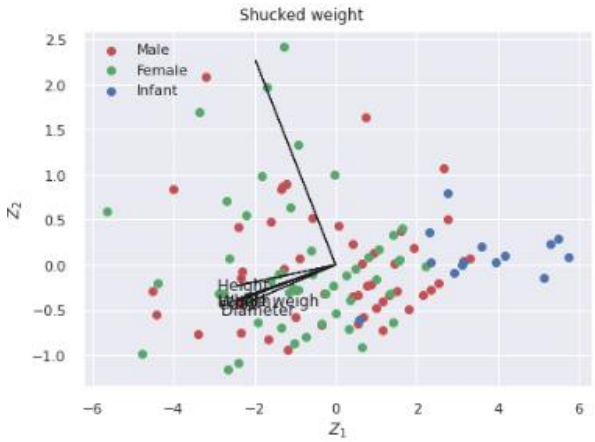

Fig 11: Bi Plot

V. CLASSIFICATION ALGORITHMS & RESULTS

The process of classification is termed for the grouping of objects into their respectively predetermined classes or categories. The classification algorithm is a supervised learning technique that identifies categories based on the observations in the training dataset. Classification in machine learning assigns a class label to classify the data into groups.[4] In this report we have used 4 classification algorithm with the help of Pycaret libraries where the dataset was split into 70 % - 30 % for training and testing module respectively. Following it was assessed with a "compare_model()" function that resulted Logistics Regression to be the best suited model for the dataset

Following figures show the performance of all the models and their accuracy level. Most widely used models such as Decision Tree Classifier,

Logistic Regression, KNN , Naïve Bayes and Random Forest were also calculated here to later tune each of them.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lr	Logistic Regression	0.6018	0.7211	0.6000	0.5895	0.5632	0.3324	0.3512	0.366
knn	K Neighbors Classifier	0.5607	0.6642	0.5944	0.5823	0.5458	0.2778	0.2961	0.114
lda	Linear Discriminant Analysis	0.5607	0.6759	0.5889	0.5343	0.5323	0.2821	0.2888	0.016
et	Extra Trees Classifier	0.5464	0.6357	0.5556	0.5587	0.5194	0.2524	0.2642	0.442
ridge	Ridge Classifier	0.5446	0.0000	0.5333	0.5174	0.5077	0.2492	0.2567	0.014
lightgbm	Light Gradient Boosting Machine	0.5357	0.6224	0.6056	0.5357	0.5041	0.2566	0.2932	0.071
svm	SVM - Linear Kernel	0.5179	0.0000	0.5333	0.4256	0.4540	0.1988	0.2431	0.061
rf	Random Forest Classifier	0.5054	0.6274	0.5667	0.4777	0.4744	0.1919	0.2021	0.465
dt	Decision Tree Classifier	0.4625	0.5550	0.5000	0.4361	0.4171	0.1291	0.1483	0.016
ada	Ada Boost Classifier	0.4625	0.5838	0.3778	0.3535	0.3892	0.0871	0.0932	0.099
gbc	Gradient Boosting Classifier	0.4482	0.5694	0.5000	0.4206	0.4147	0.0977	0.1001	0.205
qda	Quadratic Discriminant Analysis	0.4089	0.0000	0.3333	0.1692	0.2389	0.0000	0.0000	0.017
dummy	Dummy Classifier	0.4089	0.5000	0.3333	0.1692	0.2389	0.0000	0.0000	0.014

Fig 12: Compare Model

- ***Decision Tree***

  This is one of the predictive modelling approaches widely used in machine learning and data mining. It uses a decision tree model that can predict the value of a target variable based on the training which via the input variables. [6] Out of all the models when compared, decision tree was the one that stood out in terms of its accuracy performance for predicting the age of abalone based on its classification.

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.6250	0.7333	0.6667	0.4062	0.4881	0.4545	0.5455
1	0.4286	0.5833	0.5556	0.6190	0.4286	0.1765	0.2000
2	0.4286	0.5000	0.5556	0.3143	0.3571	0.0667	0.0778
3	0.4286	0.5000	0.3333	0.3857	0.3857	0.0000	0.0000
4	0.4286	0.5000	0.3333	0.3857	0.3857	0.0000	0.0000
5	0.7143	0.7917	0.7778	0.8214	0.6769	0.5625	0.6211
6	0.2857	0.3750	0.4444	0.2500	0.2653	-0.1667	-0.1725
7	0.2857	0.3750	0.2222	0.2500	0.2653	-0.2500	-0.2609
8	0.4286	0.5417	0.5556	0.2857	0.3401	0.1250	0.1380
9	0.5714	0.6500	0.5556	0.6429	0.5782	0.3226	0.3341
Mean	0.4625	0.5550	0.5000	0.4361	0.4171	0.1291	0.1483
SD	0.1304	0.1310	0.1591	0.1837	0.1250	0.2453	0.2716

Fig 13 : Pre-tuning of DT

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.6250	0.8021	0.6111	0.5938	0.5964	0.4286	0.4392
1	0.5714	0.5417	0.4444	0.4714	0.4929	0.2500	0.2858
2	0.4286	0.6607	0.5556	0.4286	0.4286	0.0667	0.0667
3	0.4286	0.3810	0.3333	0.2143	0.2857	0.0000	0.0000
4	0.5714	0.7560	0.6667	0.7143	0.5544	0.3438	0.3795
5	0.7143	0.5893	0.7778	0.8286	0.6786	0.5333	0.6228
6	0.5714	0.7321	0.6667	0.5714	0.5592	0.3000	0.3105
7	0.7143	0.7143	0.7778	0.8286	0.6786	0.5333	0.6228
8	0.5714	0.6964	0.6667	0.5714	0.5592	0.3000	0.3105
9	0.5714	0.7214	0.5000	0.4000	0.4643	0.3000	0.3558
Mean	0.5768	0.6595	0.6000	0.5622	0.5298	0.3056	0.3394
SD	0.0918	0.1181	0.1356	0.1843	0.1125	0.1655	0.1915

Fig 14 : Post-Tuning of DT

- ***Logistics Regression***

  Logistic Regression is a statistical analysis method that can predict a binary result such as True/False, Yes/No on account of the observations priorly provided of the data set [7]. This is to understand a relationship between the dependent variable of one or more independent variables by evaluating probabilities using logistics regression equation. This type of analysis helps in predicting the likelihood of an event to occur or a decision to make accordingly. For our dataset following is the accuracy for this model

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.8750	0.9750	0.8889	0.9167	0.8750	0.8140	0.8333
1	0.2857	0.3929	0.2222	0.1714	0.2143	-0.2500	-0.2858
2	0.5714	0.6786	0.6667	0.5714	0.5592	0.3000	0.3105
3	0.7143	0.8214	0.5556	0.6071	0.6531	0.5000	0.5217
4	0.7143	0.8929	0.5556	0.6857	0.6643	0.5000	0.5715
5	0.7143	0.9286	0.7778	0.8286	0.6786	0.5333	0.6228
6	0.5714	0.6429	0.6667	0.5714	0.5592	0.3000	0.3105
7	0.5714	0.4286	0.6667	0.5714	0.5592	0.3000	0.3105
8	0.7143	0.8571	0.7778	0.8286	0.6786	0.5333	0.6228
9	0.2857	0.5929	0.2222	0.1429	0.1905	-0.2069	-0.3062
Mean	0.6018	0.7211	0.6000	0.5895	0.5632	0.3324	0.3512
SD	0.1811	0.1958	0.2120	0.2462	0.2007	0.3165	0.3606

Fig 15: Pre-tuning LR

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.8750	0.9750	0.8889	0.9167	0.8750	0.8140	0.8333
1	0.2857	0.3929	0.2222	0.2143	0.2449	-0.1667	-0.1725
2	0.5714	0.7500	0.6667	0.5714	0.5592	0.3000	0.3105
3	0.5714	0.8571	0.4444	0.5000	0.5306	0.2500	0.2609
4	0.7143	0.8929	0.7778	0.7143	0.7143	0.5333	0.5333
5	0.8571	0.9643	0.8889	0.8929	0.8531	0.7667	0.7936
6	0.5714	0.7143	0.6667	0.5714	0.5592	0.3000	0.3105
7	0.5714	0.4286	0.6667	0.5714	0.5592	0.3000	0.3105
8	0.7143	0.8571	0.7778	0.8286	0.6786	0.5333	0.6228
9	0.4286	0.5571	0.3889	0.4571	0.4048	0.0667	0.0754
Mean	0.6161	0.7389	0.6389	0.6238	0.5979	0.3697	0.3878
SD	0.1723	0.2020	0.2097	0.2067	0.1825	0.2858	0.2975

Fig 16: Post-tuning LR

- ***K-Nearest Neighbor***

K-Nearest Neighbor – (KNN) is a supervised machine learning algorithm that is used to solve both classification and regression problems. The number of nearest neighbor is unknown so it has to be predicted or classified, which is denoted by symbol “k”. Implementing KNN model on our dataset generated the following results

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.7500	0.9646	0.7778	0.8229	0.7089	0.6279	0.6758
1	0.2857	0.3810	0.2222	0.2857	0.2857	-0.1667	-0.1667
2	0.5714	0.6250	0.6667	0.5714	0.5592	0.3000	0.3105
3	0.2857	0.4464	0.2222	0.2500	0.2653	-0.2500	-0.2609
4	0.8571	0.9107	0.8889	0.8929	0.8531	0.7667	0.7936
5	0.7143	0.8571	0.7778	0.7143	0.7143	0.5333	0.5333
6	0.5714	0.7143	0.6667	0.5714	0.5592	0.3000	0.3105
7	0.4286	0.5000	0.5556	0.4286	0.4286	0.0667	0.0667
8	0.7143	0.8929	0.7778	0.8286	0.6786	0.5333	0.6228
9	0.4286	0.3500	0.3889	0.4571	0.4048	0.0667	0.0754
Mean	0.5607	0.6642	0.5944	0.5823	0.5458	0.2778	0.2961
SD	0.1890	0.2231	0.2278	0.2169	0.1864	0.3253	0.3436

Fig 17: Pre-tuning KNN

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.5000	0.7167	0.5556	0.3542	0.4143	0.2558	0.2753
1	0.5714	0.6250	0.6667	0.5714	0.5592	0.3000	0.3105
2	0.4286	0.6250	0.5556	0.4286	0.4286	0.0667	0.0667
3	0.4286	0.8214	0.3333	0.2143	0.2857	0.0000	0.0000
4	0.7143	0.6250	0.7778	0.7143	0.7143	0.5333	0.5333
5	0.8571	0.9643	0.8889	0.8929	0.8531	0.7667	0.7936
6	0.7143	0.7321	0.7778	0.8286	0.6786	0.5333	0.6228
7	0.5714	0.7321	0.6667	0.5714	0.5592	0.3000	0.3105
8	0.5714	0.7500	0.6667	0.5714	0.5592	0.3000	0.3105
9	0.2857	0.5500	0.2778	0.3929	0.3129	-0.1290	-0.1336
Mean	0.5643	0.7142	0.6167	0.5540	0.5365	0.2927	0.3090
SD	0.1580	0.1126	0.1833	0.2034	0.1712	0.2562	0.2711

Fig 18: Post-tuning KNN

- ***Random Forest***

Random Forest classification is another supervised machine learning algorithm constructs many decision trees on different samples and based on most votes it classifies [10]. Random forest reduces the variance of the regression predictors compared to single tree while leaving the bias changed [11]

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.6250	0.7333	0.6667	0.3917	0.4812	0.4419	0.5353
1	0.5714	0.6964	0.6667	0.5714	0.5592	0.3000	0.3105
2	0.2857	0.5714	0.4444	0.2500	0.2653	-0.1667	-0.1725
3	0.1429	0.3750	0.1111	0.1071	0.1224	-0.5000	-0.5217
4	0.5714	0.5476	0.6667	0.7143	0.5544	0.3438	0.3795
5	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
6	0.4286	0.5714	0.5556	0.4286	0.4286	0.0667	0.0667
7	0.5714	0.5357	0.6667	0.5714	0.5592	0.3000	0.3105
8	0.5714	0.7143	0.6667	0.5714	0.5592	0.3000	0.3105
9	0.2857	0.5286	0.2222	0.1714	0.2143	-0.1667	-0.1976
Mean	0.5054	0.6274	0.5667	0.4777	0.4744	0.1919	0.2021
SD	0.2259	0.1603	0.2406	0.2545	0.2327	0.3905	0.4070

Fig 19: Pre-tuning RF

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	0.5000	0.7292	0.5556	0.3167	0.3875	0.2558	0.3099
1	0.4286	0.6786	0.5556	0.3143	0.3571	0.0667	0.0778
2	0.5714	0.6786	0.6667	0.3571	0.4286	0.3000	0.4743
3	0.4286	0.6786	0.3333	0.2143	0.2857	0.0000	0.0000
4	0.4286	0.6786	0.5556	0.2857	0.3401	0.1250	0.1380
5	0.7143	0.6786	0.7778	0.8286	0.6786	0.5333	0.6228
6	0.7143	0.6786	0.7778	0.8286	0.6786	0.5333	0.6228
7	0.5714	0.6786	0.6667	0.3571	0.4286	0.3000	0.4743
8	0.7143	0.6786	0.7778	0.8286	0.6786	0.5333	0.6228
9	0.5714	0.6143	0.6111	0.8286	0.5680	0.3824	0.4900
Mean	0.5643	0.6772	0.6278	0.5160	0.4831	0.3030	0.3833
SD	0.1127	0.0258	0.1316	0.2580	0.1457	0.1863	0.2249

Fig 20 : Post-tuning RF

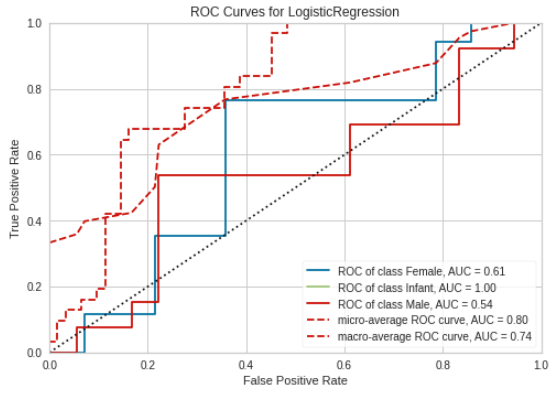

Fig 21 : SHAP Summary Plot

For the above figure the ROC class Female is better resulted compared to the male class.

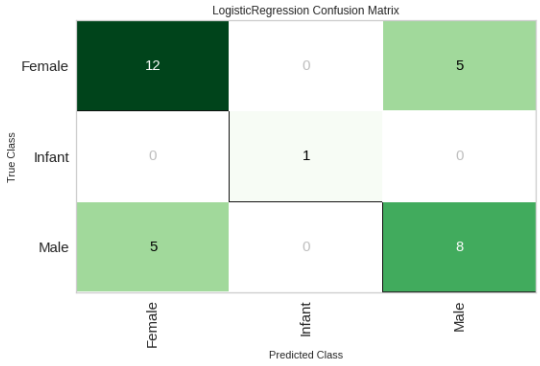

Fig 21 : SHAP Summary Plot

From the above figure it is concluded that 12 records were predicted correctly for female, while 5 were incorrectly recorded as male instead of female.

Also 8 were correctly recorded as male while 5 were incorrectly recorded as female instead of male.

VI. EXPLAINED AI WITH SHAP

Shapley values is a concept widely used in game theory that involves gains and costs to all active actors working in coalition [5]. Shapley values constitutes to the weighted average marginal contribution of a feature value across all coalitions [18]. This helps in determining a payoff for all players when each of them might have contributed comparatively to one another [5]. With the support of SHAP libraries in python inbuilt functions, it was easier to implement machine learning model, specifically for the random forest classifier that of which we have used in correspondence with the principal component for our dataset as follows

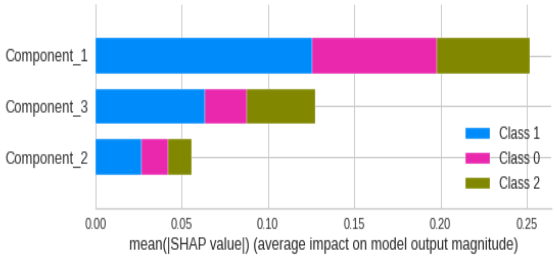

Fig 21 : SHAP Summary Plot

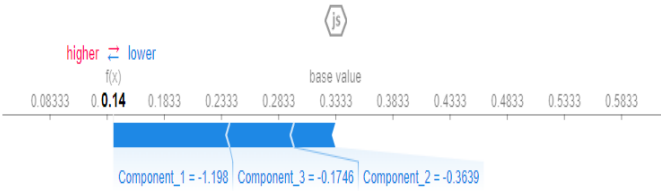

Fig 22 : Single Prediction

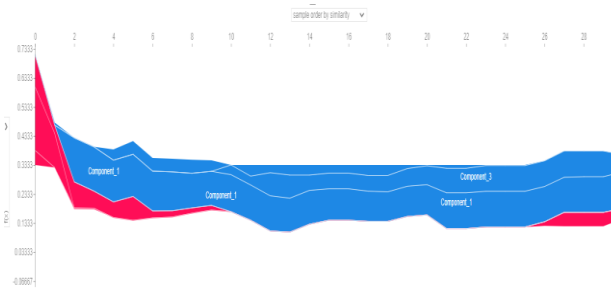

Fig 23 : Sample order by similarity

VII. CONCLUSION

In a nutshell, PCA along with four most widely known machine learning algorithms were used to predict the age of abalone. Upon applying PCA to the dataset, it yielded a variance of 97.575 for the first two PC components, eventually reducing the data set to 2 components for it satisfied the criteria. Proceeding this, models such as Decision Tree, Logistics Regression, KNN and Random Forest were implemented to observe their outcome both before and after tuning which eventually resulted for Logistic Regression to be the best model based on its performance.

VIII. REFERENCES

[1] https://medium.com/@cmukesh8688/importance-of-principal-component-analysis-e9184a47ffa8

[2]] https://builtin.com/data-science/step-step-explanation-principal-component-analysis

[3] Ben Hamza, A. (2021). Advanced Statistical Approaches to Quality. Concordia institute for information systems engineering.

[4] https://machinelearningmastery.com/types-of-classification-in-machine-learning/

[5] https://www.investopedia.com/terms/s/shapley-value.asp17-https://www.theanalysisfactor.com/factor-analysis-how-many-factors/

[6] https://en.wikipedia.org/wiki/Decision_tree_learning

[7]https://www.techtarget.com/searchbusinessanalytics/definition/logistic-regression

[8] https://www.ibm.com/topics/logistic-regression

[9]https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

[10] https://towardsdatascience.com/understanding-random-forest-58381e0602d2#:~:text=The%20random%20forest%20is%20a,that%20of%20any%20individual%20tree.

[11] https://math.unm.edu/~luyan/research/biasrf.pdf^

[12] https://www.britannica.com/animal/abalone

[13] https://towardsdatascience.com/5-things-you-should-know-about-covariance-26b12a0516f1

[14] https://www.geeksforgeeks.org/stripplot-using-seaborn-in-python/

[15] https://methods.sagepub.com/reference/the-sage-encyclopedia-of-educational-research-measurement-and-evaluation/i18507.xml#:~:text=A%20scree%20plot%20is%20a,analysis%20or%20a%20factor%20analysis.

[16] https://www.theanalysisfactor.com/factor-analysis-how-many-factors/

[17] https://christophm.github.io/interpretable-ml-book/shapley.html#general-idea

[18] https://archive.ics.uci.edu/ml/datasets/abalone