### Machine Learning Models

##### **FIRST DATASET PREPROCESSING:**

The first step after having an idea about the biological features for the DORIAN mice is to have a good look at the data shape and make a decision on what to take as a baseline. Since the main concept of our project is to figure out the ***health-span and Aging process***, it becomes essential to focus on the **mice's age** and the mother's diet while pregnant (keeping in mind the data is token for their children), focusing mainly on the **wild type**.

##### **SECOND FEATURES ENGINEERING:**

The next step will be by handling the features taking a good step to:

1. Find out features correlation and acceptance.
2. Find out the best features to have a good prediction accuracy.

The main idea is to take the **Age** as the class label so we have 3 different class labels holding the mouse age (3, 12, 18 months).

**the first run** will be by predicting the mouse age through the lab measurements, taking all 75 features (with continuous values )

I replaced each age of (3, 12, 18 months) to be (1, 2, 3)

* **1 = 3 months**
* **2 = 12 months**
* **3 = 18 months**

**It’s necessary to say that there is:**

* 3 months = 215
* 12 months = 173
* 18 months = 186

I added (**diet**) to the dataset as a feature.

Another replacement held for the diet feature replacing (cd, hfd) to be (0,1)

* **0= cd (controlled diet)**
* **1 = hfd (high fat diet)**

**Now I will split the dataset to be 2 (training and testing),** this step is important since we need to check the classifier accuracy in order to make sure whether the model can make a good prediction. Where the **test.csv** does not contain the class label (age), and the **train.csv** holds all features values plus the (age) column.

* **Train file = 474 mouse (with redundant values )**
* **Test file = 100 mouse (with redundant values)**

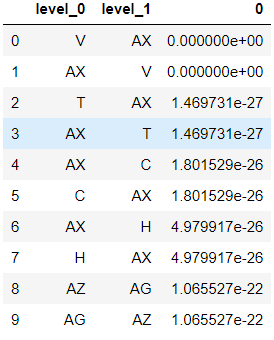
The main file consists of (575row x 76column)

#### Missing Data Treatment

The missing values filled using the mean\_value function, every cell filled with its column mean value.

#### Features Correlation

Here you can see the Correlation between the attributes for train dataset, to take advantage in running I gave an alternative name for the features where:

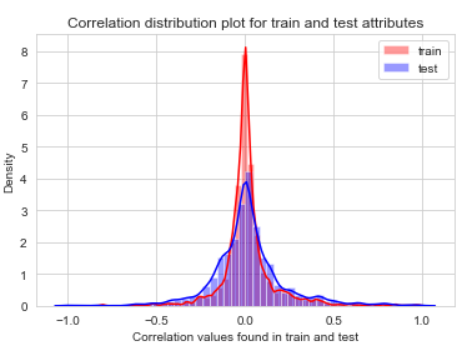


The abbreviation and explanation for the above table can be shown here:

|  |  |  |
| --- | --- | --- |
| **Symbole** | **Feature** | **Correlation** |
| C | VISIT NO NP pre-s | With order 5 with a high correlation with LEPTIN-plasma goes from 1.801529 and e-26, where e is a fixed float number. |
| H | visit light phase post-s | With order 7 with a high correlation with LEPTIN-plasma goes from 4.979917 and e-26. |
| T | liking post-s | With order 2 with a high correlation with LEPTIN-plasma goes from 1.469731 and e-27, where e is a fixed float number. |
| V | wanting post-s | Taking first place 1 with a high correlation with LEPTIN-plasma goes from 0.00000 and e-00. |
| AG | % AMB npk pre-s | With order 2 with a high correlation with ADIPON-plasma goes from 1.065527 and e-22, where e is a fixed float number. |
| AZ | ADIPON-plasma(post-prandial) | With order 8 with a high correlation with AMB npk pre-s goes from 1.065527 and e-22, where e is a fixed float number. |
| AX | LEPTIN-plasma | This feature has a high effect and correlated with most of the above values as the following:  wanting post-s between 0.000000 and e+00  liking post-s between 1.469731 and e-27  VISIT NO NP pre-s between 1.801529 and e-26  visit light phase post-s between 4.979917and e-26 |

This figure below shows the correlation between training and testing files as we can see the density for the training features is higher (the reason is upon large data capacity in the training file).

**Figure (21)**



#### 

#### **Machine learning (ML)**

ML spread to cover supervised learning, which is a data mining to derive the function from the data which is already labeled. Another branch of ML is the unsupervised learning, its main aim to categorize the dataset, which is unlabeled into many groups named clusters where each cluster holds members that share high similarity with members from other clusters. My main target in this comprehensive implementation of different ML methods is to give you a word of wisdom from what I tried to accomplish after applying several algorithms in the (DORIAN 3-12-18) datasets. Explicit explanations will describe the utilities supervised ML algorithms in detail. But so far I will show you the results and small explanation of those models.

##### **THIRD MODEL RUNNING**

I will use those Machine learning algorithms to implement the idea without any change to the features.

1. The main reason for using this type of algorithms, that we have 3 class labels, the mice age (1, 2, 3) that it’s capable of handling three class categories with ordering such as small, medium, and large.
2. The second reason is that we have numeric, continuous and dependent measurements.

ML algorithms will be used are the following:

1. Gradient Boosting with DTC.
2. Gradient boosting with grid search and Cross-Validation (CV).
3. Support Vector Machines (SVM)**-without Kernel** and Gamma parameters.
4. Support Vector Machines (SVM)**-with** **Linear** Kernel.
5. **Polynomial** Support Vectors Machine (PSVM).
6. Support Vector Machines (SVM)-with **RBF** Function.
7. Random forest (RF).
8. Naive Bayes (NB).
9. Bagging with Naive Bayes (BNB).
10. Gaussian Naive Bayes (GBN).
11. Multi-Layer Perceptron Classifier.
12. Logistic Regression (LR).
13. Adaptive Boosting (AB).
14. Decision Tree (default parameters).
15. Decision Tree 2.
16. Decision Tree 3.
17. Bagging Classifier Decision tree.
18. Gradient boosting with grid search cv.
19. Voting-Ensembling (VE).
20. Ordinal Logistic Regression (OLR).

#### **Deep Learning**

Neural Networks (NN), the first step of Deep Learning (DL). Please refer to the last section.

##### 

##### **FOURTH CLASSIFIERS OUTPUT**

Figure (1): Gradient Boosting (GB)

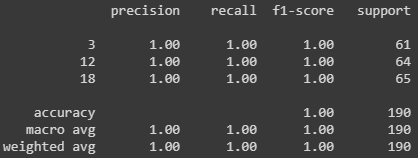


Figure (2): Gradient boosting with grid search cv

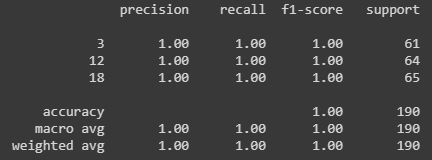


Figure (3): Support Vector Machines (SVM)

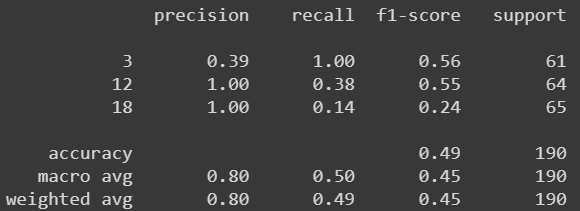


Figure (4): SVM with Linear kernel

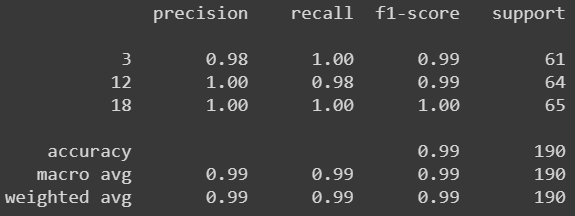


Figure (5): Polynomial Support Vectors Machine (PSVM)

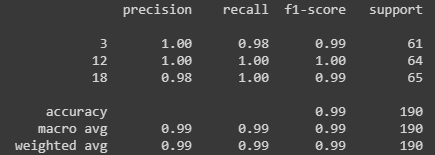


Figure (6): SVM with RBF kernel

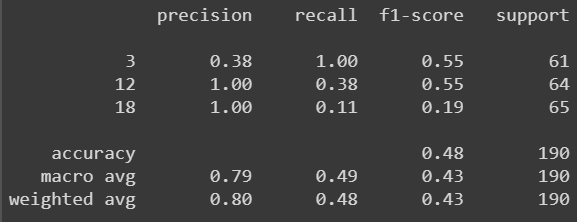


Figure (7): Random forest (RF)

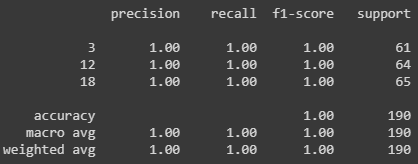


Figure (8): Naive Bayes (NB)

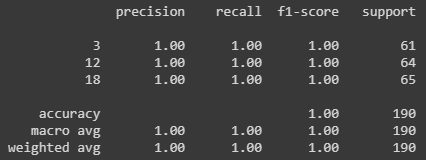


Figure (9): Bagging with Naive Bayes (BNB)

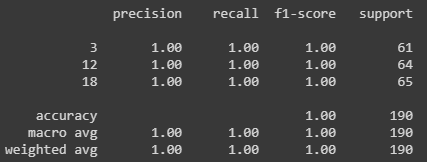


Figure (10): Gaussian Naive Bayes (GBN)

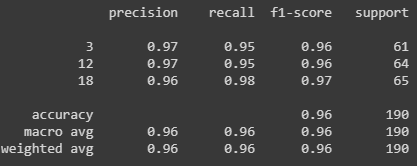


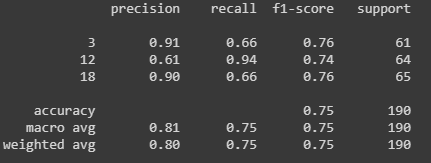
Figure (11): Multilayer Perceptron (MLP)

Figure (12): Logistic Regression (LR)

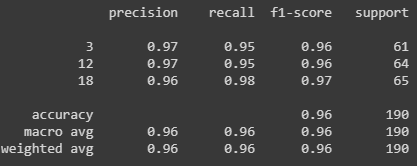


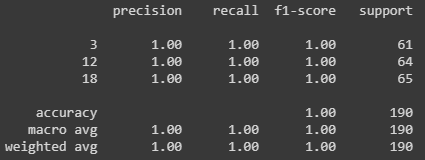
Figure (13): Adaptive Boosting (AB)

Figure (14): Decision Tree Classifier (DTC)

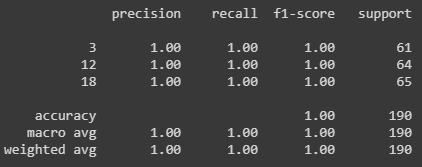


Figure (15): Decision Tree 2

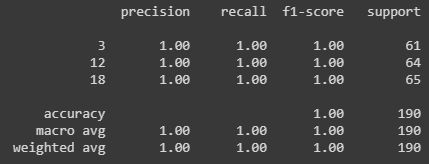


Figure (16): Decision Tree 3

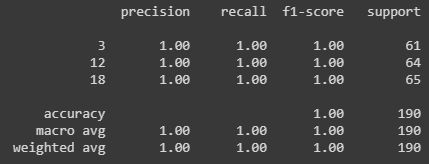


Figure (17): Bagging classifier Decision tree

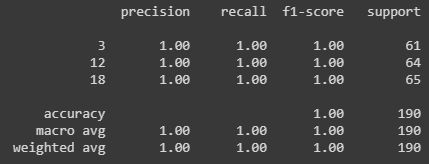


Figure (18): Gradient boosting with grid search cv.

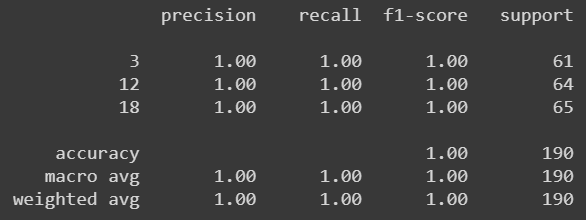


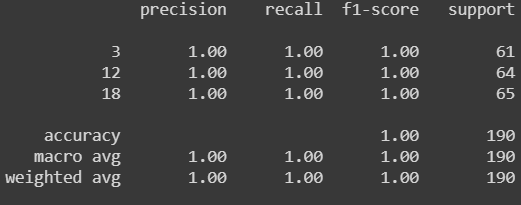
Figure (19): Voting-Ensembling (VE)

Figure (20): Ordinal Logistic Regression (OLR)

##### **FIFTH OUTPUT ANALYSIS**

As can be recognized from the above implementation, most of the classifiers got 100% true prediction of the mouse age according to its body measurement, and they are 14 classifiers:

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| Gradient Boosting with DTC | 100% |
| Gradient boosting with grid search and Cross-Validation | 100% |
| Support Vector Machines (SVM)-with RBF Function | 100% |
| Random forest (RF) | 100% |
| Naive Bayes (NB) | 100% |
| Bagging with Naive Bayes (BNB) | 100% |
| Adaptive Boosting (AB). | 100% |
| Decision Tree (default parameters) | 100% |
| Decision Tree 2 (default parameters) | 100% |
| Bagging Classifier Decision tree | 100% |
| Gradient boosting with grid search cv | 100% |
| Voting-Ensembling (VE) | 100% |
| Ordinal Logistic Regression (OLR) | 100% |
| Support Vector Machines (SVM) | 49% |
| SVM with Linear kernel | 99% |
| SVM with RBF kernel | 48% |
| Gaussian Naive Bayes (GBN) | 96% |
| Multilayer Perceptron (MLP) | 75% |
| Logistic Regression (LR) | 96% |
| Ordinal Logistic Regression (OLR) |  |

#### Sixth Deep Learning

Implementing the Neural Network, to predict the age for DORIAN mice, would be an essential phase, since deep learning algorithms deals better with the features and can handle a huge amount of dataset, in the meanwhile, the .csv file gold few number of redundant mice rows

It becomes more efficient to depend on the DL to gain benefits, especially for a large trend bank like the cirulli lab tests.