OCTOBER 22, 2021

BREAST CANCER PREDICTION

CAPSTONE PROJECT 1

YOHEITA YOSHIMURA

Table of Contents

Introduction

If breast cancer can be diagnosed at an early stage, the treatment can be highly effective. Breast Cancer is the second cause of death for women in United States. Among many diagnostic methods, a fine needle aspirate (FNA) is one of the quick procedures for diagnosis. A thin needle is inserted into a suspicious area to collect a sample of cells for examination to determine if the cell is malignant. However, the result is not 100% accurate. The study shows the false positive of FNA is 0-2.5% and false negative is 5-10% or even higher than 15% also has been documented [1]. The positive prediction is greater than 90%, but the negative prediction is variable and can sometimes go as low as 67% [1].

We will explore machine learning algorithms through this report to find the model with higher accuracy than the current study of FNA statistics. Thus, this study may aid doctors to give final diagnosis decisions decreasing the number of incorrect decisions of “benign” to increase the number of early breast cancer diagnoses for effective treatment.

We will be using the dataset provided by The University of Wisconsin to explore machine learning classification models. The dataset was produced from digitalized images of fine needle aspirate (FNA) with the characteristic and diagnosis results of the cells of the specimen.

It consists of the outcome (a.k.a. target) and predictors (a.k.a. features), and the type of machine learning algorithms that will be used belongs to the class of supervised algorithms. This is because our desired prediction outcome is the diagnosis result (Malignant or Benign) which is the form of a binary classification problem. The models will be trained using the given dataset.

We will analyze the dataset for deeper understanding to apply multiple algorithms to compare accuracy to conclude the most accurate model found with this report and compare the result with the study. The whole process was conducted using Python. We used the following Python libraries: panda, matplotlib, numpy, seaborn, scikit-learn, and shap.

The data of fine-needle aspiration can be found from:

LINK: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

Python code can be found at the following link:

LINK:

https://github.com/yoheita/Springboard/blob/6a8bb6f20dcba1cc7997139c6de38023cd4c6a5c/Capstone%20Project%201/Actual_Capstone_Project_1_final.ipynb

Approach

Data Acquisition and Wrangling

Understanding the dataset is a fundamental initial process before using the dataset. We will first understand the architecture of the data. After understanding, we clean data, data type conversion, or replace null data. This process will make it more convenient for data extraction and more organized for an easier understanding of the data.

We studied the dataset by first converting it into a Pandas data frame. Dataset with Pandas data frame enabling to visually understand the general data information. We discovered the dataset contains 569 patient data and 13 features (containing 11 float data, 1 char, and 1 null data). The features of the dataset contained the following:

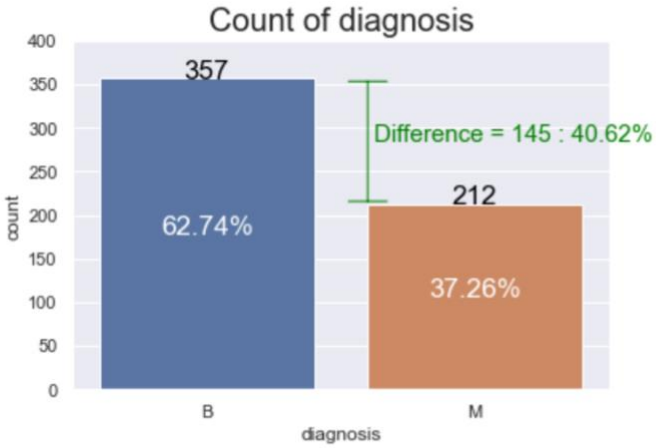
- 1 char type data: diagnosis result
- 11 float type data: contain patient's ID and computed specimen sample features:
- 1 null data all null data. The feature was called "Unnamed: 32"

The prediction is based on the result and specimen sample features. Therefore, we removed unnecessary features (ID and Unnamed: 32) to clean the dataset. Other than the feature which only contained null values (Unnamed: 32), there was no null data within other features. Therefore, none of the data had to be replaced or removed.

Storytelling and Inferential Statistics1

Overview of our target feature

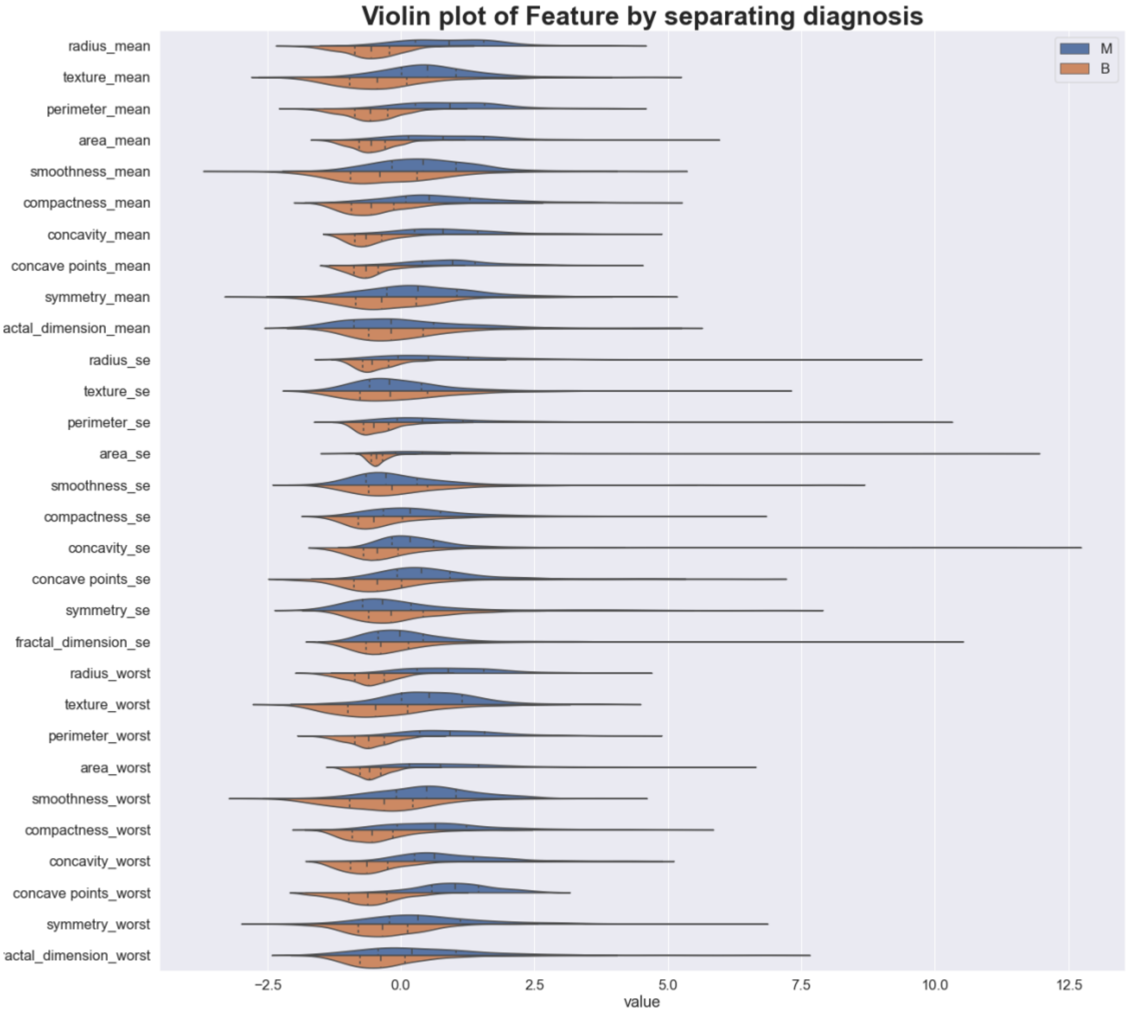
Our Machine Learning algorithm model's target data is a feature was called 'diagnosis'. The data was displayed either 'M' as malignant or 'B' as benign. The below graph shows the total count of the 'diagnosis' feature.

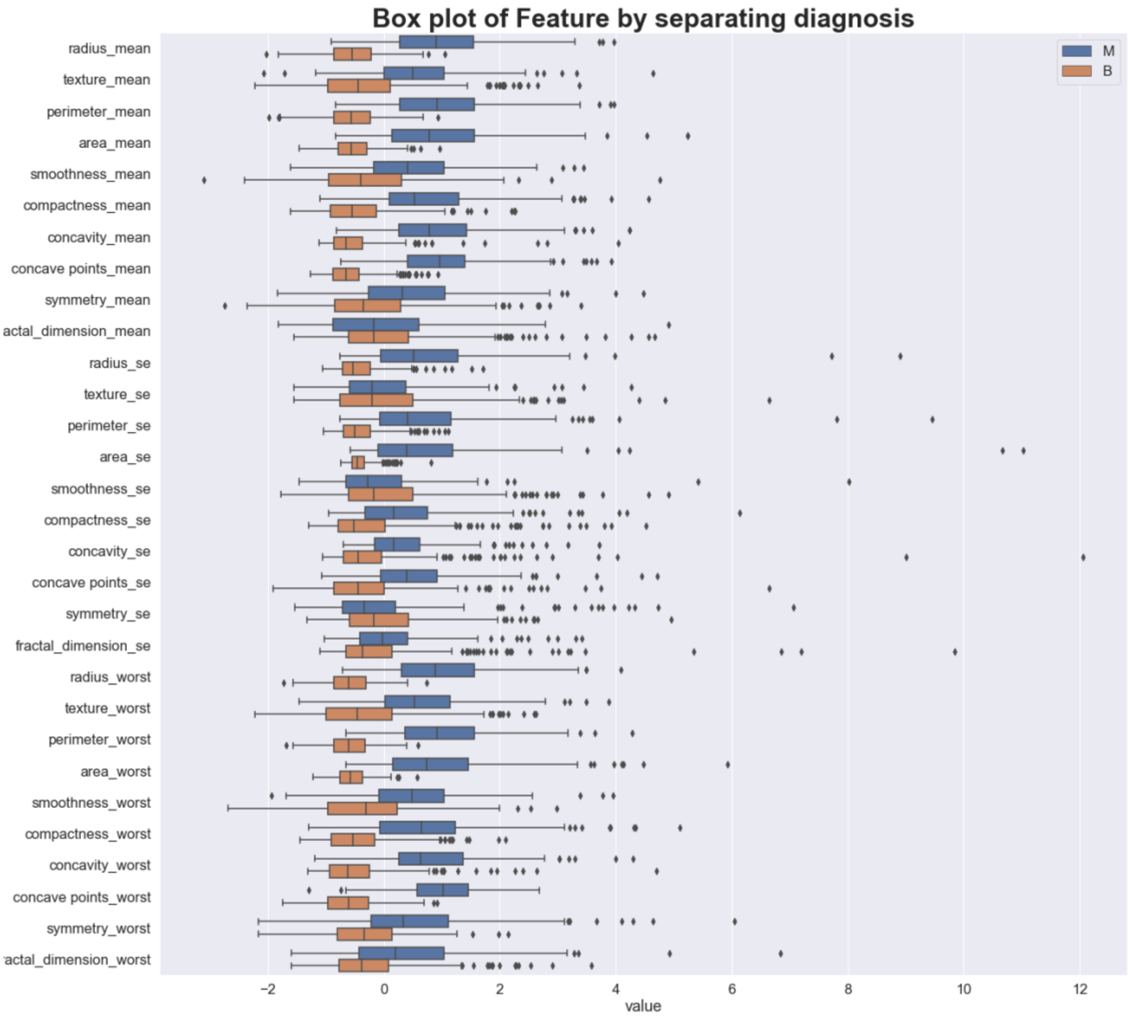


The bar graph represents the target variable has an imbalanced number of B and M. There is a 145 difference between B and M with 40.62% compared to B. There are 212 patients (37.2%) diagnosed as malignant and 357 patients (62.74%) were diagnosed as benign.

Characteristic of Features

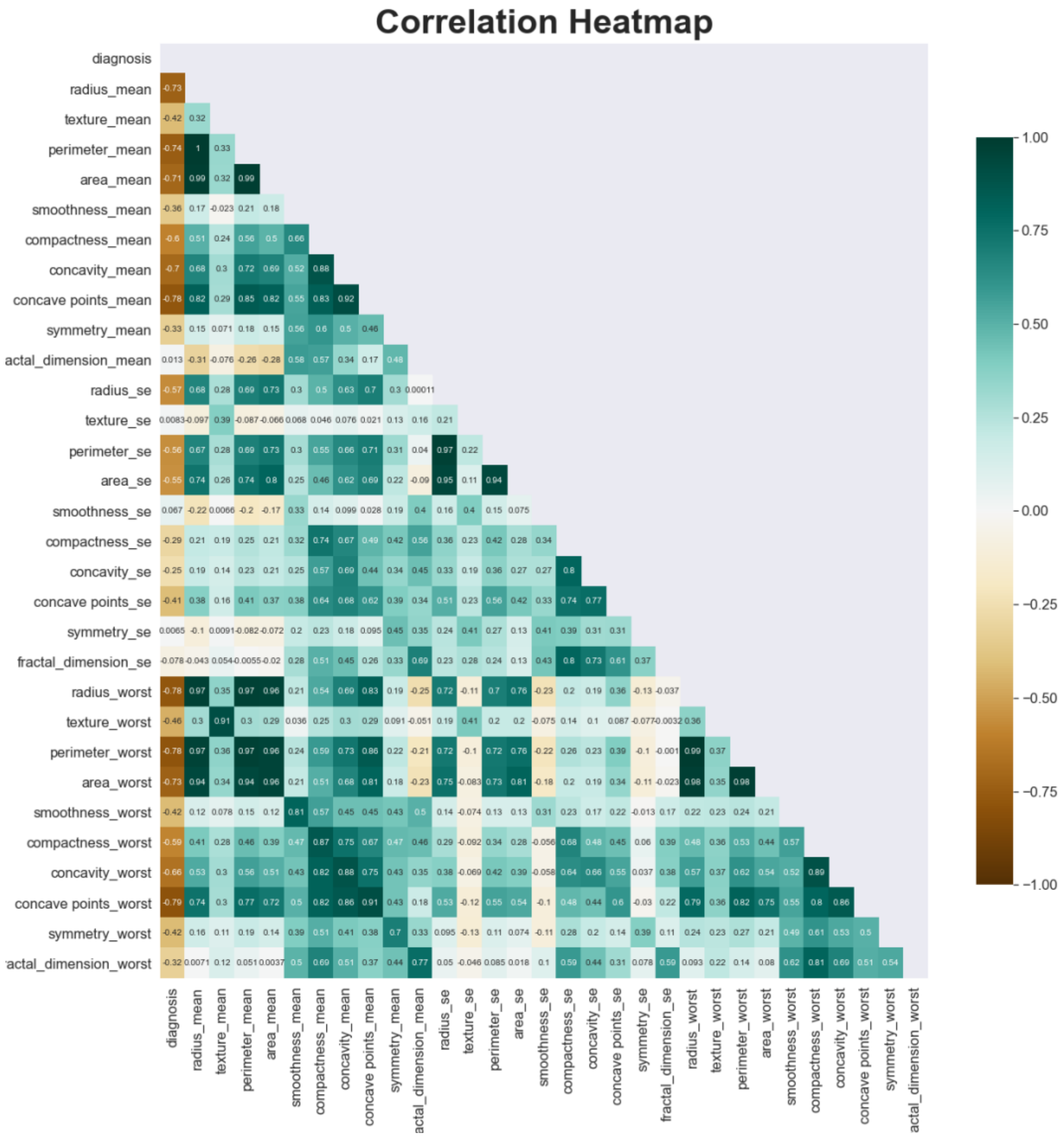
For checking the distribution of each feature with separating diagnosis results, we used standardization to compare each feature. Standardization allows us to plot all the features to the same scale. This will visually help us to compare the distribution and characteristics of all the features. We plotted violin-plot for distribution comparison and box-plot for visualizing the distribution of quantitative values by displaying the median, Q1, Q3, maximum, and minimum.

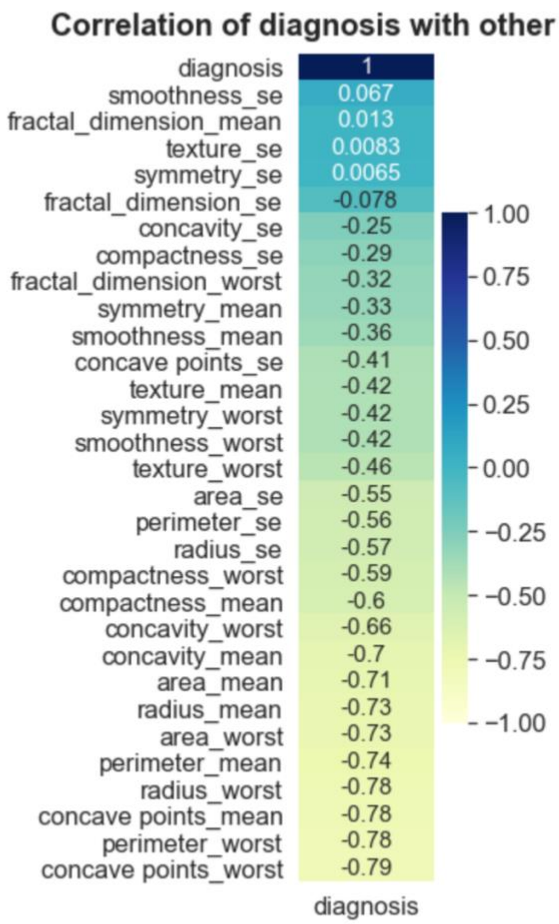

Violin plot of Feature by separating diagnosis
M
B
radius_mean
texture_mean
perimeter_mean
area_mean
smoothness_mean
compactness_mean
concavity_mean
concave points_mean
symmetry_mean
actal_dimension_mean
radius_se
texture_se
perimeter_se
area_se
smoothness_se
compactness_se
concavity_se
concave points_se
symmetry_se
fractal_dimension_se
radius_worst
texture_worst
perimeter_worst
area_worst
smoothness_worst
compactness_worst
concavity_worst
concave points_worst
symmetry_worst
:actal_dimension_worst
−2.5
0.0
2.5
5.0
7.5
10.0
12.5
value



The graph presents some of the features that have similar distribution such as "smoothness_se" and "symmetry_se". Both features have a similar distribution pattern of ‘M’ and ‘B’ which could mean a low correlation to the target variable. Whereas, 'M’ and ‘B’ of “concave points_worst” have a similar pattern of distribution with a different mean. Since ‘M’ and ‘B’ means are separated, the correlation might be high.

A different analysis was conducted to find the correlation by comparing the target features with the other features. We first plot a heatmap of correlations by comparing each then extracted correlation of target feature as shown below. We have used a heatmap to plot correlation. Heat map aids with a color scale which visually helps the correlation relationship of all other features.

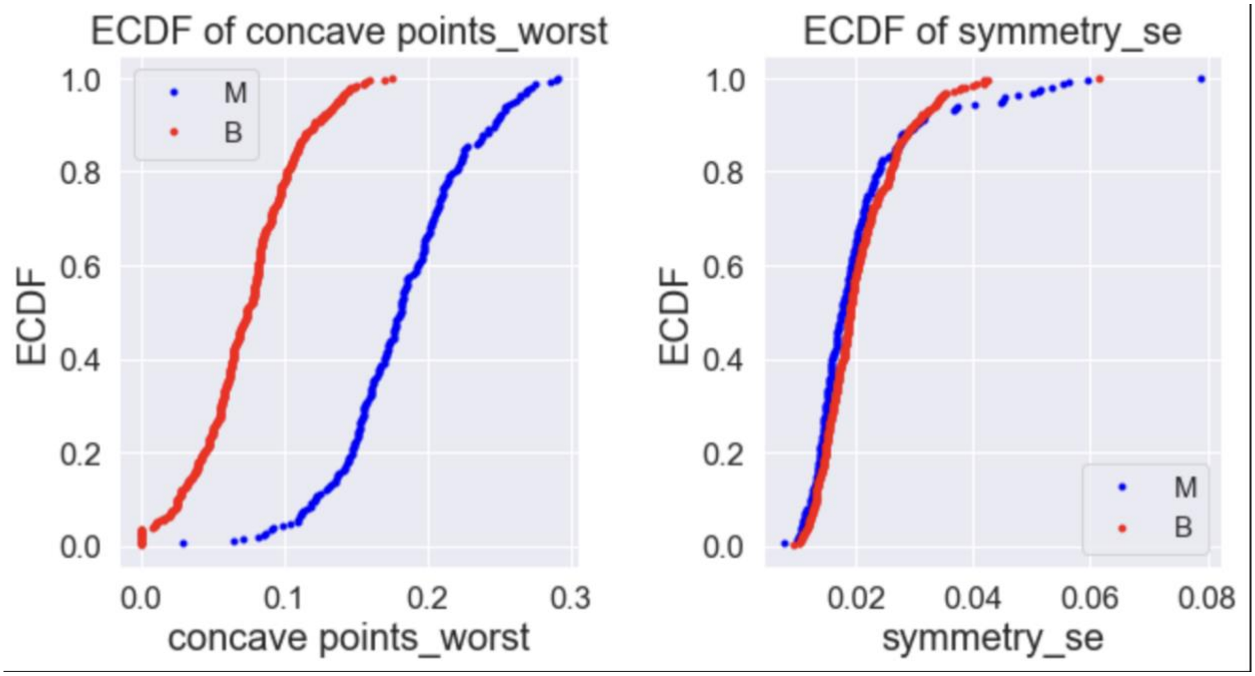




The highest negative correlation was “concave points_worst" with a negative correlation of -0.79, and the lowest correlation of feature is "symmetry_se" with a positive correlation of 0.0065. This describes the relationship between the distribution of the diagnosis and correlation.

Inferential Statistics

Based on the correlation result from heat-map, we have compared the highest and lowest correlation with the empirical cumulative distribution of features (ECDF). ECDF will allow us to see the distribution pattern across the dataset.



We can observe that “symmetry_se” Malignant and Benign have a very similar distribution unable to distinguish diagnosis from this data. Whereas, “concave points_worst” has separate distribution between Malignant and Benign.

Baseline Modeling

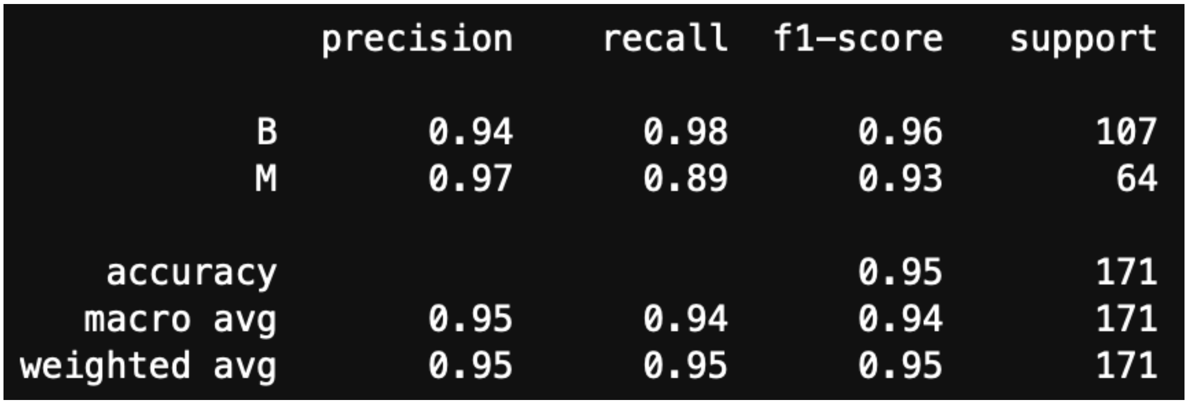
Before our evaluation with different Machine Learning Algorithms to the dataset, we first must separate the data into test data and train data. Train data will be used to train an algorithm then applied to test data to create predicted target data. Predicted data will be compared with target data in test data to compute the accuracy.

To separate the data, we applied 70% for training data and the rest of 30% for testing data. As we recall from section “Overview of our target feature” the ratio of M and B count is imbalanced. Therefore, I made sure to separate train and test data with the same ratio to produce the closest environment to overall data.

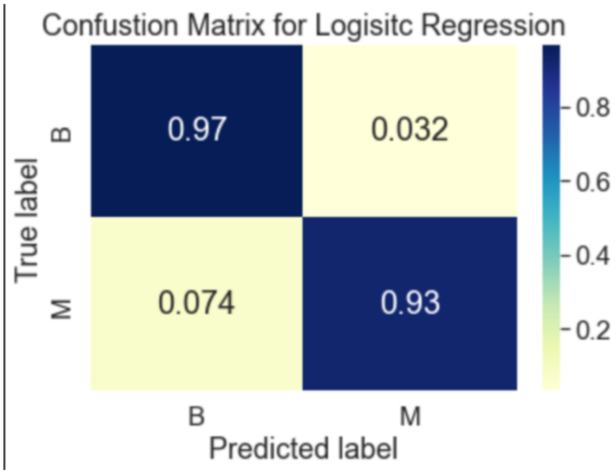
Logistic Regression

Since the target data is categorical with two possible values, we will be using binary classification algorithms. Our first approach was using Logistic Regression Algorithm. Logistic Regression Algorithm is one of the algorithms that is implemented in scikit-learn library that can be used when the target feature is categorical. From many types of Logistic Regression, we will use Binary Logistic Regression because of the binary variable in the target feature “diagnosis”.

After applying Logistic Regression, the result was the following (using the test set):



```
              precision    recall  f1-score   support

           B       0.94      0.98      0.96       107
           M       0.97      0.89      0.93        64

    accuracy                           0.95       171
   macro avg       0.95      0.94      0.94       171
weighted avg       0.95      0.95      0.95       171
```



I have outputted the classification report which outputs the basic statistic result of the model that I have used.

The table’s terms are follow:

precision: It is the ratio of true values and a total number of predicted true values. True value is the outcome of predicted data is equal to the target data (diagnosis in this case).

Recall: it is the ratio of true values and an actual number of true in the data.

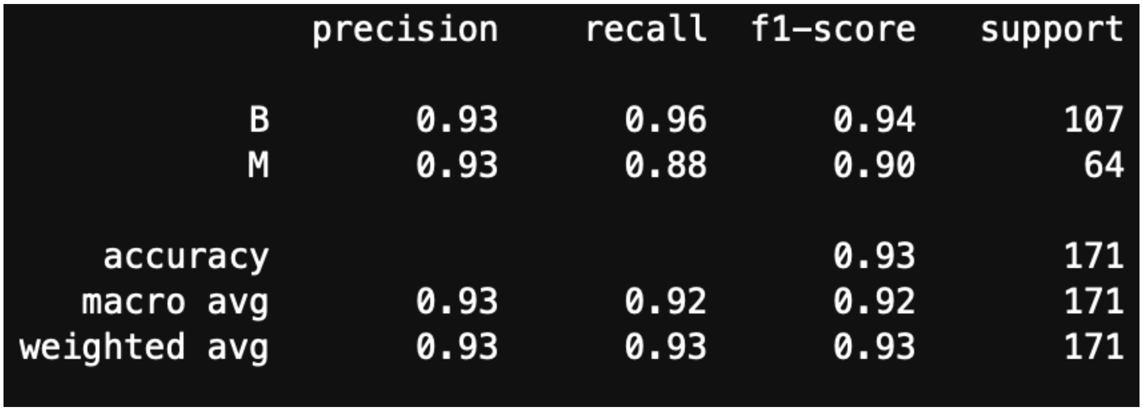
F1 score: It is a combination of precision and recall, it presents the overall score of the model.

Support: it is the number of data points that belong to classes M and B data from the test set

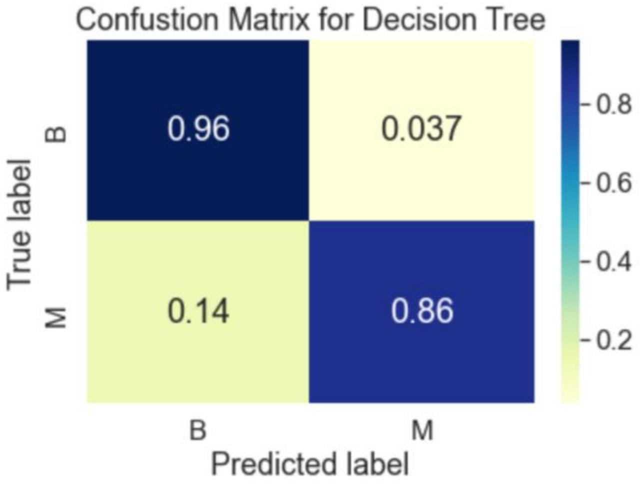
Extended Modeling

Since the target data is categorical, we used the following classification algorithms: Decision Trees, SVMs, and Random Forest Classifiers implemented in scikit-learn library. We will be using the same testing and training data from Logistic Regression.

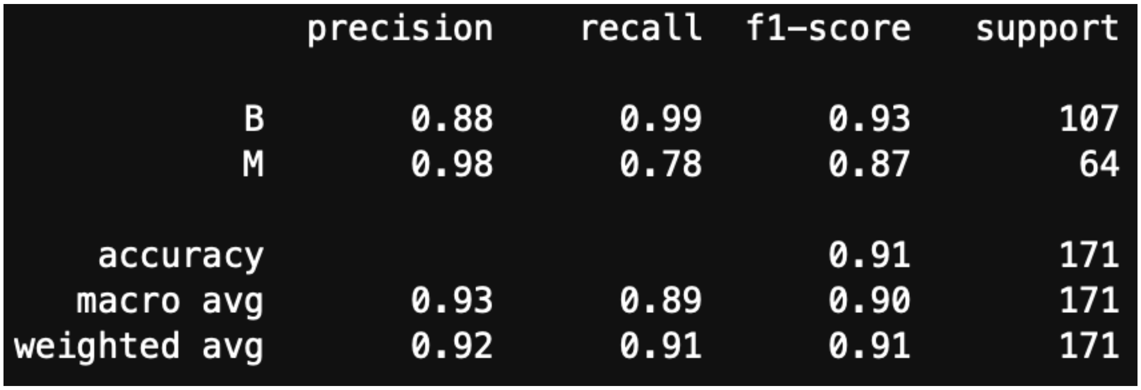
Decision Tree



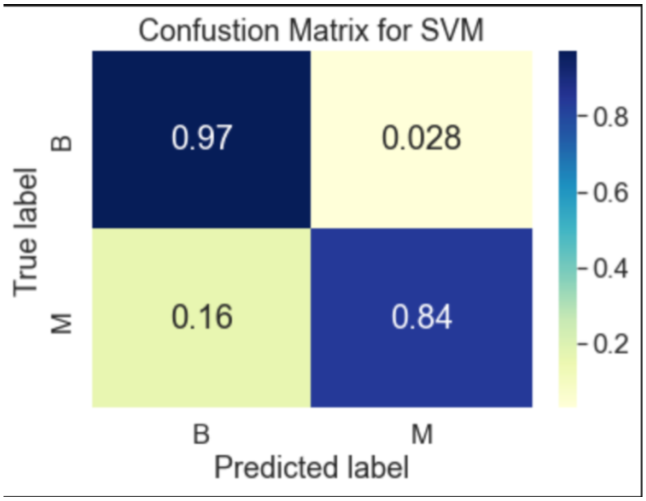
```
              precision    recall  f1-score   support

           B       0.93      0.96      0.94       107
           M       0.93      0.88      0.90        64

    accuracy                           0.93       171
   macro avg       0.93      0.92      0.92       171
weighted avg       0.93      0.93      0.93       171
```



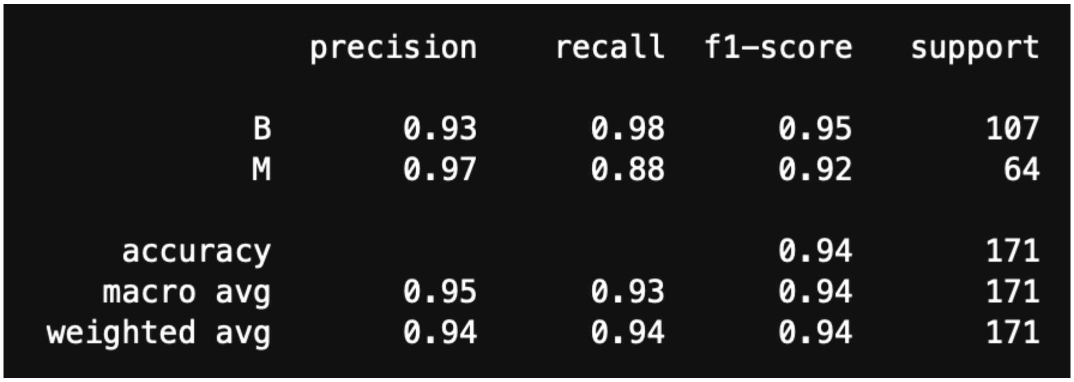
SVMs (Support Vector Machine)



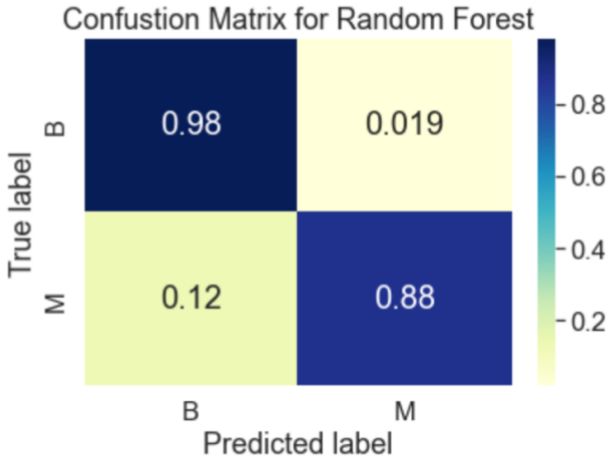
```
              precision    recall  f1-score   support

           B       0.88      0.99      0.93       107
           M       0.98      0.78      0.87        64

    accuracy                           0.91       171
   macro avg       0.93      0.89      0.90       171
weighted avg       0.92      0.91      0.91       171
```



Random Forests



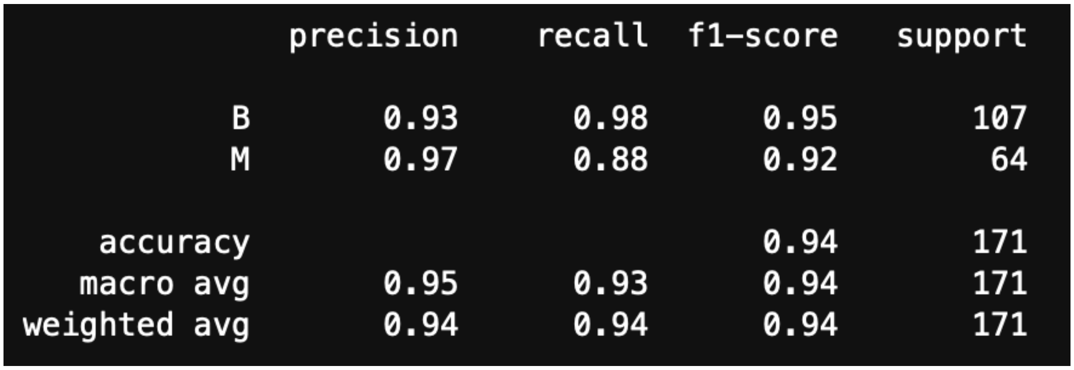
```
              precision    recall  f1-score   support

           B       0.93      0.98      0.95       107
           M       0.97      0.88      0.92        64

    accuracy                           0.94       171
   macro avg       0.95      0.93      0.94       171
weighted avg       0.94      0.94      0.94       171
```



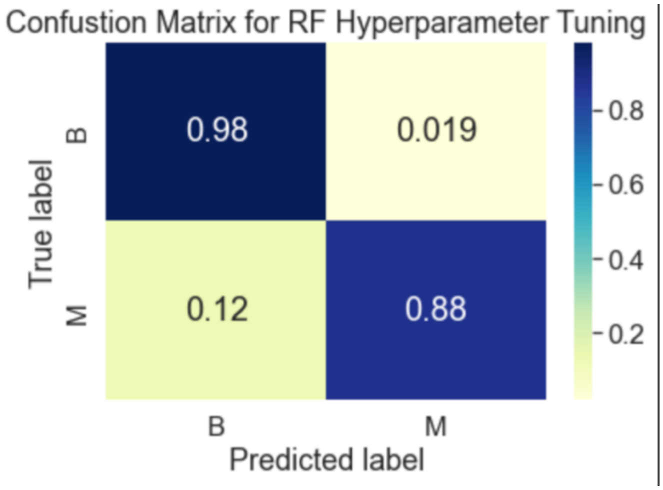
Hyperparameter Tuning for Random Forests

Random Forests have hyperparameters that can be optimized to increase the accuracy. We have conducted Hyperparameter Tunning to Random Forest to search for better parameters. However, the result was not improved.

The result became the following:

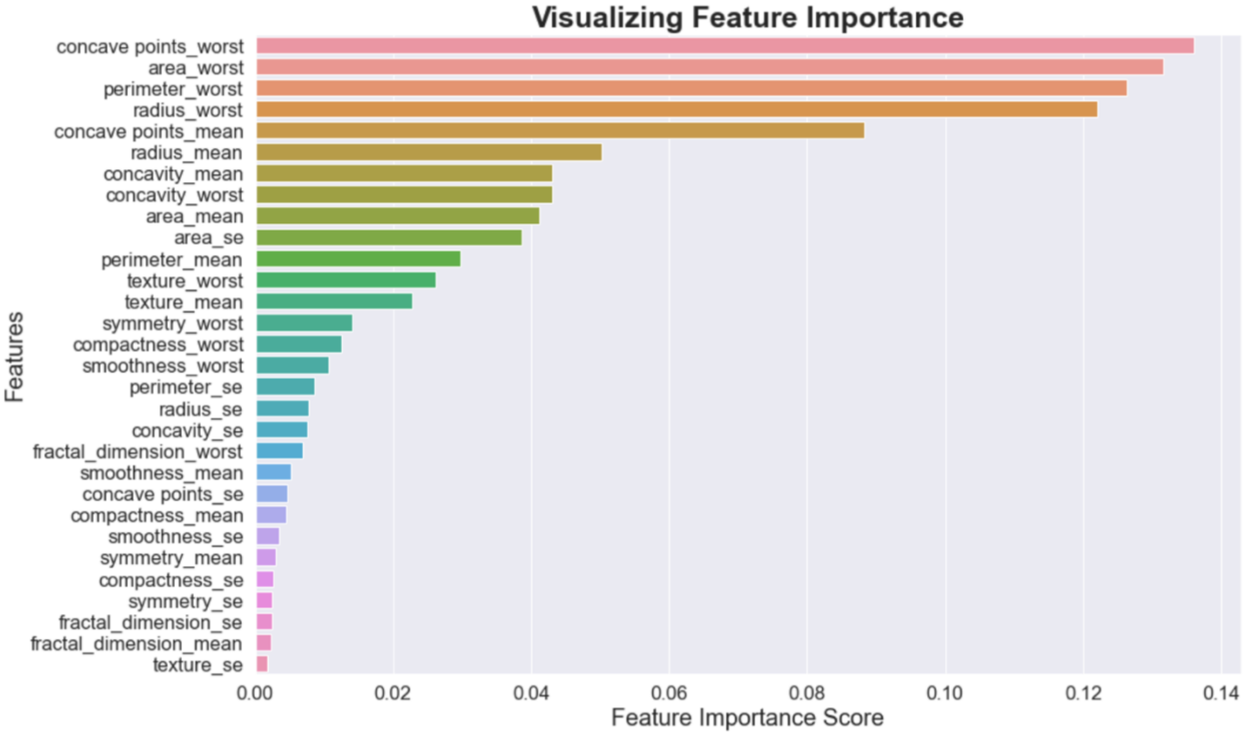


```
              precision    recall  f1-score   support

           B       0.93      0.98      0.95       107
           M       0.97      0.88      0.92        64

    accuracy                           0.94       171
   macro avg       0.95      0.93      0.94       171
weighted avg       0.94      0.94      0.94       171
```



Findings

By using Random Forests from scikit-learn library, the model enabled to present a list of important features which impacted the result of diagnosis from the model. This enabled us to understand "concave points_worst" has the highest feature importance overall.

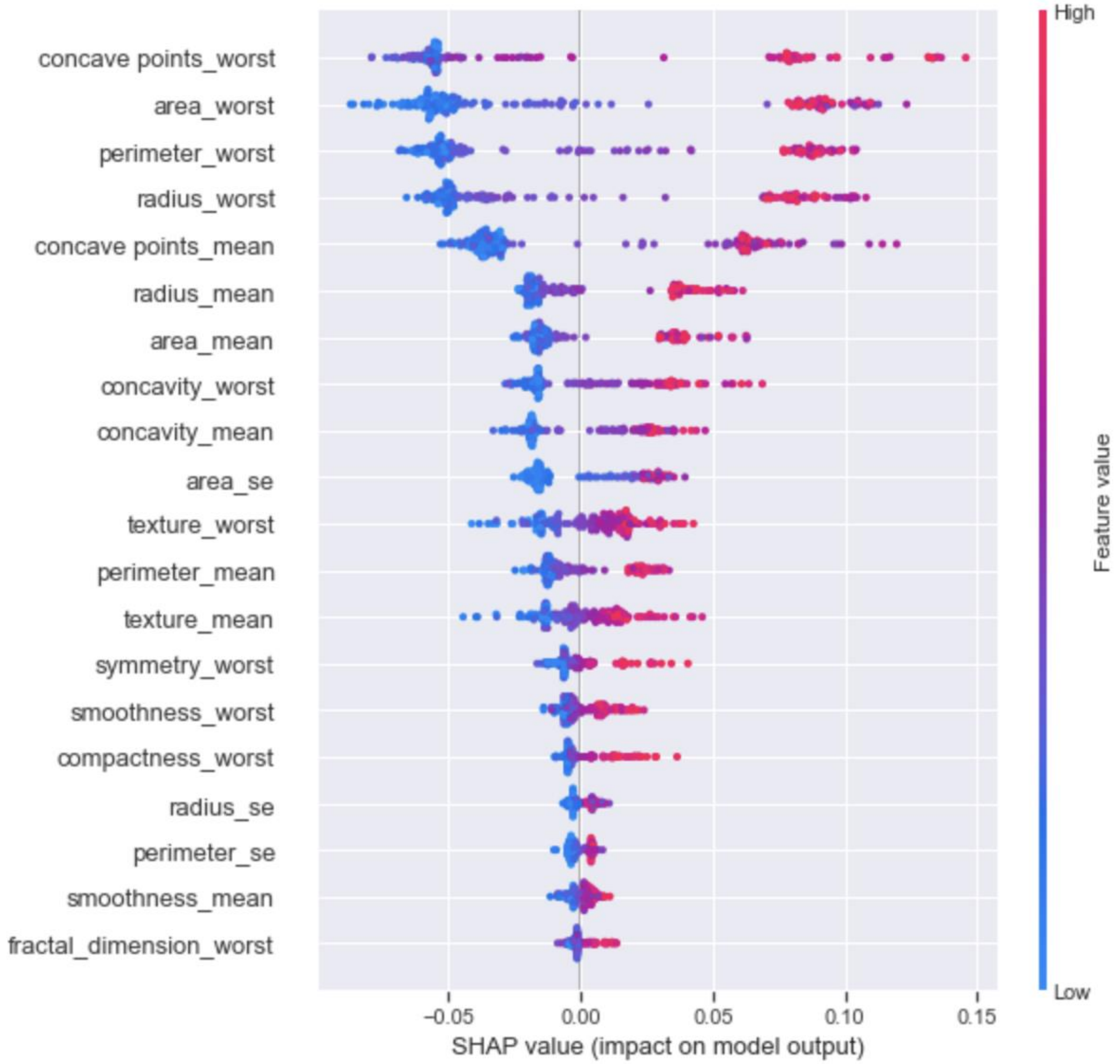


For further investigation, we imported the SHAP library[1] to evaluate the impact of features predicting diagnosis results. We evaluated different approaches to scikit-learn library for deeper understanding. SHAP outputs for any feature measures how much it has contributed to the outcome.

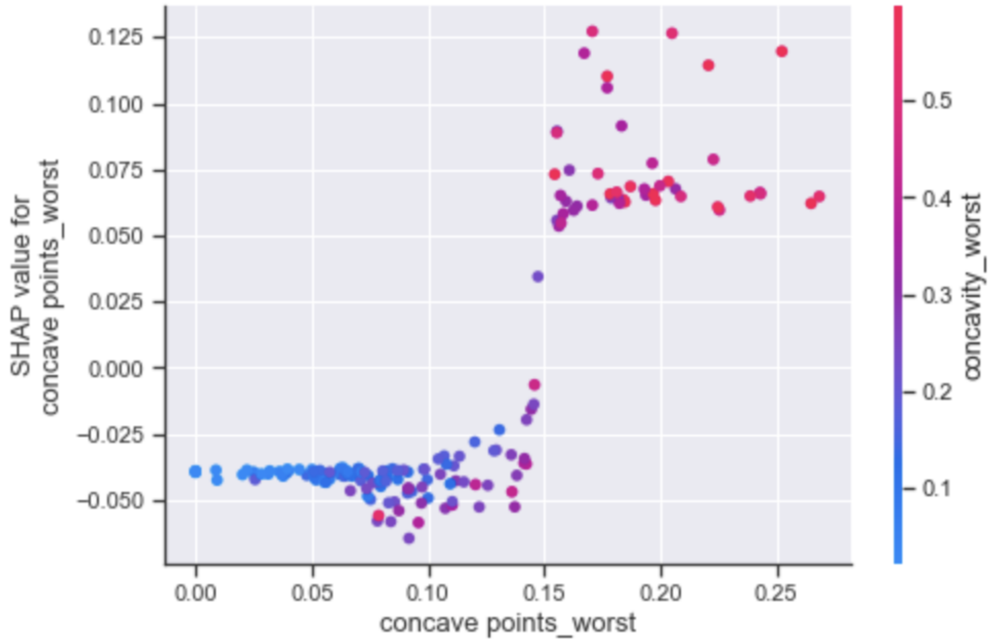
Summary_plot in SHAP library generated graph shown below. It represents how each feature data has impacted the predicted result. The color bar indicates the feature importance level. Each

[1] https://github.com/slundberg/shap

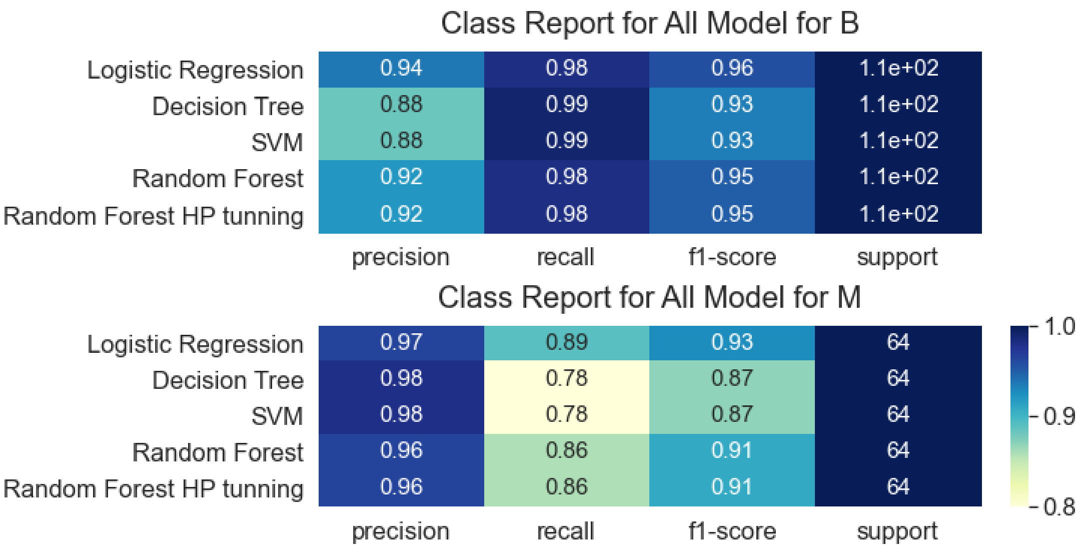
dot represents each data point in the dataset, and the color of the dot towards red means a higher SHAP value (higher odds) and blue a lower SHAP value (lower odds). The X-axis represents the level of impact on the model using the SHAP value.



Since “concave points_worst” has the highest feature importance, the graph below shows that (a) as “concave-worst” grows, the odds of the tumor being malignant also grows (from about 0.15); and (b) in that region, the values of “concavity_worst” also grow, making the odds of the tumor being malignant even higher.



We will now compare each classification report to the table to understand which algorithm had the highest performance score.



Class Report for All Model for B

	precision	recall	f1-score	support
Logistic Regression	0.94	0.98	0.96	1.1e+02
Decision Tree	0.88	0.99	0.93	1.1e+02
SVM	0.88	0.99	0.93	1.1e+02
Random Forest	0.92	0.98	0.95	1.1e+02
Random Forest HP tunning	0.92	0.98	0.95	1.1e+02

Class Report for All Model for M

	precision	recall	f1-score	support
Logistic Regression	0.97	0.89	0.93	64
Decision Tree	0.98	0.78	0.87	64
SVM	0.98	0.78	0.87	64
Random Forest	0.96	0.86	0.91	64
Random Forest HP tunning	0.96	0.86	0.91	64

We can see that Logistic Regression Algorithm has the highest scores (precision, recall, and f1 score) for Benign and Malignant with an f1 score of 0.96 with 0.93.

Conclusions and Future Work

In this project, we were able to explore the dataset by cleaning and wrangling and using multiple graphs to analyze the characteristics of each feature. We were able to experience multiple classical algorithms to find the accuracy using precision, recall, and f1 score and compare the accuracy of each model.

Upon investigation, we managed to find Logistic Regression has the highest accuracy score with an f1-score of 0.96. F1-score over 0.96 defines 96% of the prediction is the same result as the current dataset, and the values of the false positive and false negative of prediction were less than 1%. Producing higher accuracy than the statistic found from FNA where false positive was 0 – 2.5% and false negative was 5-10% [1].

Furthermore, I managed to discover the most impacted feature “concave points_worst” for prediction using SHAP and Random Forest. This project can conclude that the highest algorithm model is Logistic Regression and the most important feature for prediction is “concave points_worst”.

If more time was allowed, I would have investigated deeper into this project. Firstly, we would have used more algorithms and understood the concept deeper. There could have been an algorithm with higher accuracy scores and understanding the algorithm deeper will allow us to use hyperparameters of each algorithm to explore higher accuracy combinations. Second, we would have used more the SHAP library to understand prediction importance of each feature. Many other graphs are useful to investigate the data which had low SHAP values but high feature values. Therefore, with more time allowed, I would have tested more algorithms and SHAP for a deeper understanding of the data.

Consulted Resources

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5019018/