A Project Report

On

Breast Cancer is Benign or Malignant

BY

Padala Vijaya Reddy -- 17331A05B6

Madabathula Venkata Sai Pavan – 17331A0590

D Karunakar Rao

Under the supervision of

Dr. Aruna

(June 2020)

ACKNOWLEDGMENTS

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organisations. We would like to extend my sincere thanks to all of them. We are highly indebted to Goal Street for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project. We would like to express my gratitude towards my parents & member of Goal Street for their kind co-operation and encouragement which help me in completion of this project. We would like to express my special gratitude and thanks to industry persons for giving me such attention and time. My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

ABSTRACT

Breast cancer is associated with the highest morbidity rates for cancer diagnoses in the world and has become a major public health issue. Breast cancer is the second most leading cancer occurring in women compared to all other cancers. Early diagnosis can increase the chance of successful treatment and survival. However, it is a very challenging and time-consuming task that relies on the experience of pathologists. The automatic diagnosis of breast cancer by analysing histo-pathological images plays a significant role for patients and their prognosis. Around 1.1 million cases were recorded in 2004. Observed rates of this cancer increase with industrialization and urbanization and also with facilities for early detection. It remains much more common in high-income countries but is now increasing rapidly in middle- and low-income countries including within Africa, much of Asia, and Latin America. Breast cancer is fatal in under half of all cases and is the leading cause of death from cancer in women, accounting for 16% of all cancer deaths worldwide. The objective of this project is to apply multiple classification algorithms and be able to classify the test data. For the implementation of the ML algorithms, the dataset was partitioned in the following fashion: 70% for training phase, and 30% for the testing phase. The hyper-parameters used for all the classifiers were manually assigned. Results show that all the presented ML algorithms performed well (all exceeded 90% test accuracy) on the classification task. The SVM algorithm stands out among the implemented algorithms with a test accuracy of ~95.04%.

CONTENTS

Title page……………………………………………………..……….….….……..1

Acknowledgements…………………………………………………….….….……...2

Abstract……………………….……………………………………………..…….3

1. Introduction.....………………………….……………………...…….….……......5
2. Data Collection...………………………….….……..…………….….…..….6
3. Data Visualization…………………………..…………………….……….....8
4. Feature Selection………………………………………………………………12
5. Fitting ML Models.…………………………………………………….…….....
   5.1. Logistic Regression………………………………………………….......15
   5.2. Support Vector Machines……………………………………….………....17
   5.3. Random Forests………………………………………………….………...19
   5.4. Decision Tree...…………………………….………………………………22
   5.5. Naïve Bayes….……………………………..…………………………….24
   5.6. K-Nearest Neighbors.……………………….……………………………..26
6. Conclusion……………………………………...…………….…………..30

References………………………………………….……………………..………

1. INTRODUCTION

Cancer is the principle wellspring of death around the globe with 2.09 million cases so far in 2018 [1]. Around 627000 deaths accounting to 6.6% are caused because of female breast cancer and it ranks five amongst the list of top causes for deaths, the prime reason being prognosis being favourable in developed countries. Breast cancer is the most common malignancy among women, accounting for nearly 1 in 3 cancers diagnosed among women in the United States, and it is the second leading cause of cancer death among women. Breast Cancer occurs as a results of abnormal growth of cells in the breast tissue, commonly referred to as a Tumour. A tumour does not mean cancer - tumours can be benign (not cancerous), pre-malignant (pre-cancerous), or malignant (cancerous). Tests such as MRI, mammogram, ultrasound and biopsy are commonly used to diagnose breast cancer performed.

The best possible recognizable proof of breast cancer disease and the process of characterizing into benign and malignant groups is that the main concern of a ton of investigation and research. When thrown light on its particular advantages in significant alternatives recognition from the datasets of entangled breast cancer, the generally perceived option is Machine Learning, because of the philosophy of determination in breast cancer to arrange pattern and forecast modelling.

2. Data Collection

Breast Cancer Diagnostic Dataset was acquired from an open Kaggle database, a collection of 569 records of breast mass computed from digitalized images.

This data set was created by Dr.William H. Wolberg, physician at the University of Wisconsin Hospital at Madison, Wisconsin, USA. To create the dataset Dr.Wolberg used fluid samples, taken from patients with solid breast masses and an easy-to-use graphical computer program called Xcyt, which is capable of perform the analysis of cytological features based on a digital scan. The program uses a curve-fitting algorithm, to compute ten features from each one of the cells in the sample, than it calculates the mean value, extreme value and standard error of each feature for the image, returning a 30 real-valuated vector.

From the dataset we can see these things clearly:

1) There is an id that cannot be used for classification

2) Diagnosis is our class label

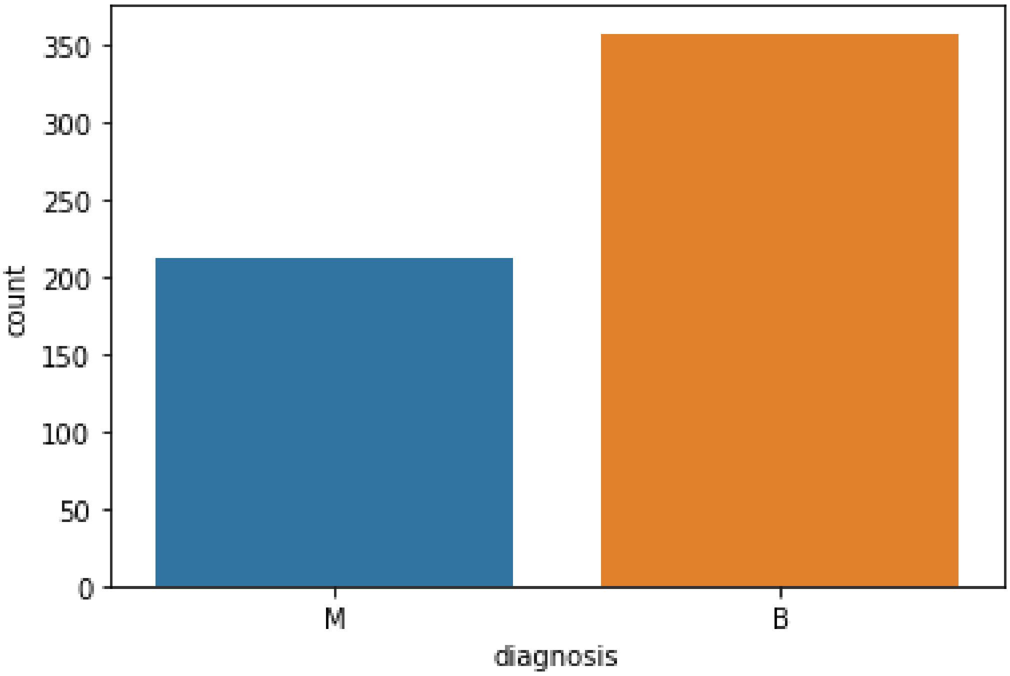
3) Unnamed: 32 feature includes NaN so we do not need it.

We need to apply Feature Selection methods to reduce the number of features to be used when training a machine learning model. More number of irrelevant features may cause our model to not perform well.

Some of the problems we find with many features are:

- Increased computer throughput.
- Too complex visualization problems.
- Decrease efficiency by including variables that have no effect on the analysis.
- Make data interpretation difficult.

The dataset Collected have only two classes: Benign and Malignant. Out of those two classes Benign class have 357 records out of 569 whereas Malignant class have 212 records out of 569.



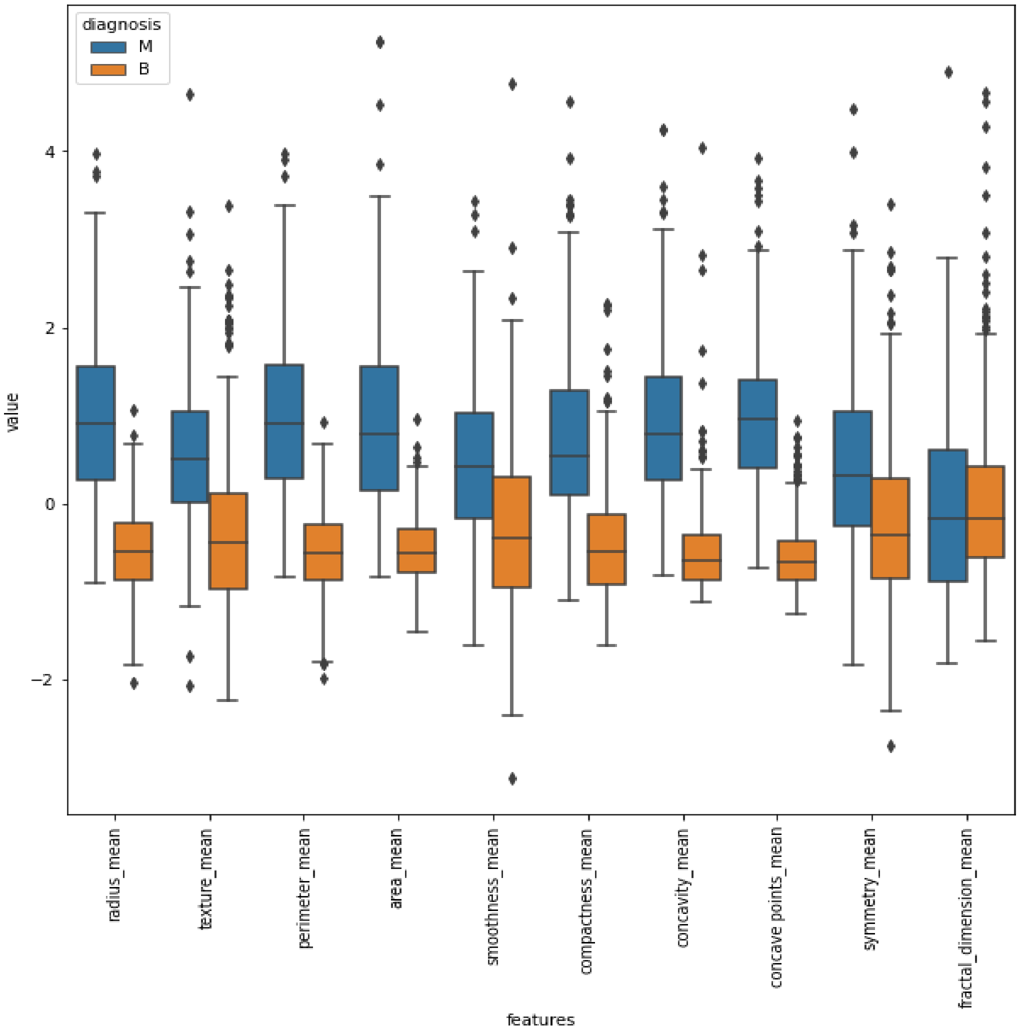
Now the dataset will be processed to remove the NaN values and the features that are irrelevant for the classification. The NaN valued Column is “Unnamed: 32” which contains every row as NULL so we remove the entire column. After that we have to apply feature selection methods on the dataset to obtain the features necessary for our machine learning model.

3. Data Visualization

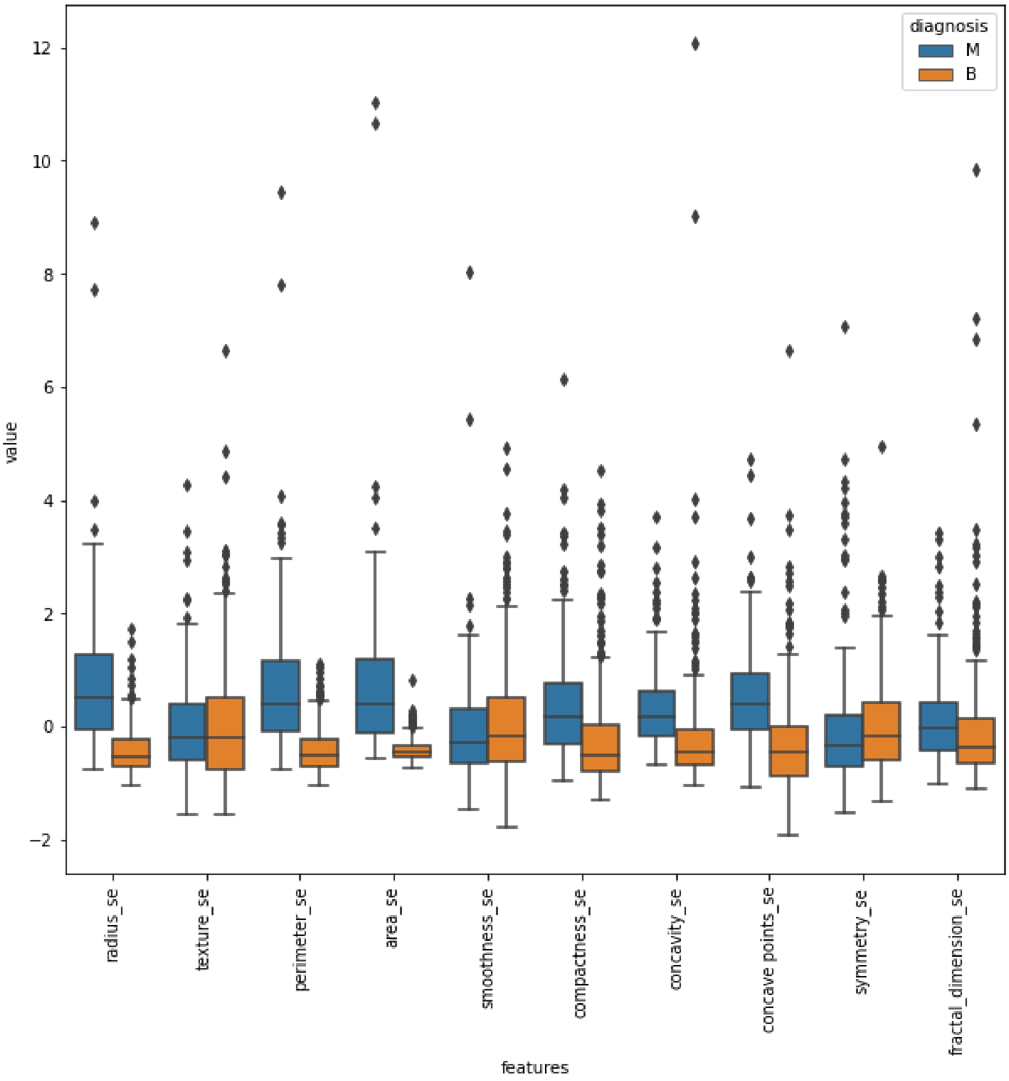
Boxplots:

In order to visualize data we are going to use seaborn plots. We need to normalization or standardization. Because differences between values of features are very high to observe on plot. W plot features in 3 group and each group includes 10 features to observe better.

A bar plot represents an estimate of central tendency for a numeric variable with the height of each rectangle and provides some indication of the uncertainty around that estimate using error bars. This bar plot shows only the mean (or other estimator) value, but in many cases it may be more informative to show the distribution of values at each level of the categorical variables.

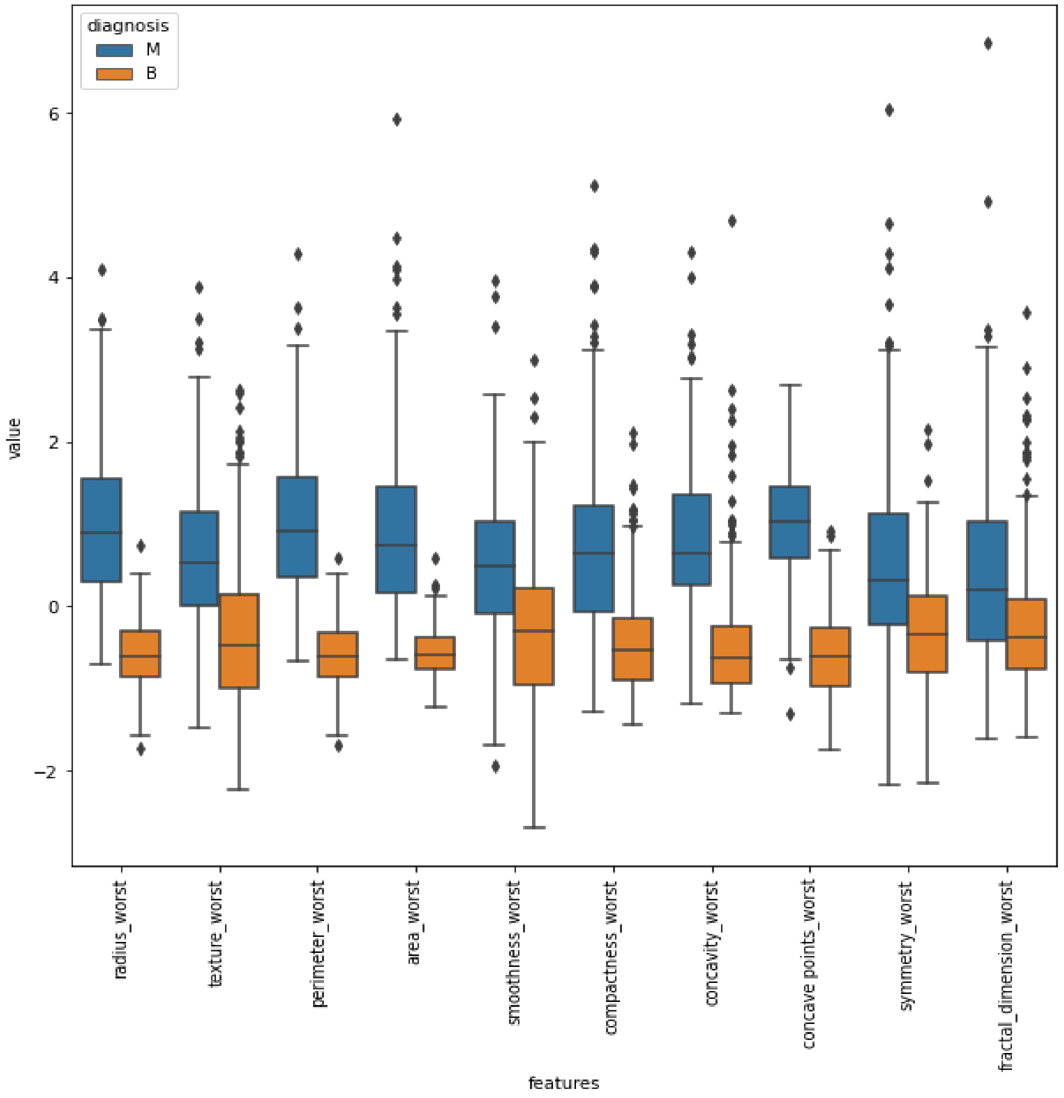


Let’s interpret the plot above. For example, in texture_mean feature, median of the *Malignant* and *Benign* looks like separated so it can be good for classification. However, in fractal_dimension_mean feature, median of the Malignant and Benign does not looks like separated so it does not gives good information for classification.



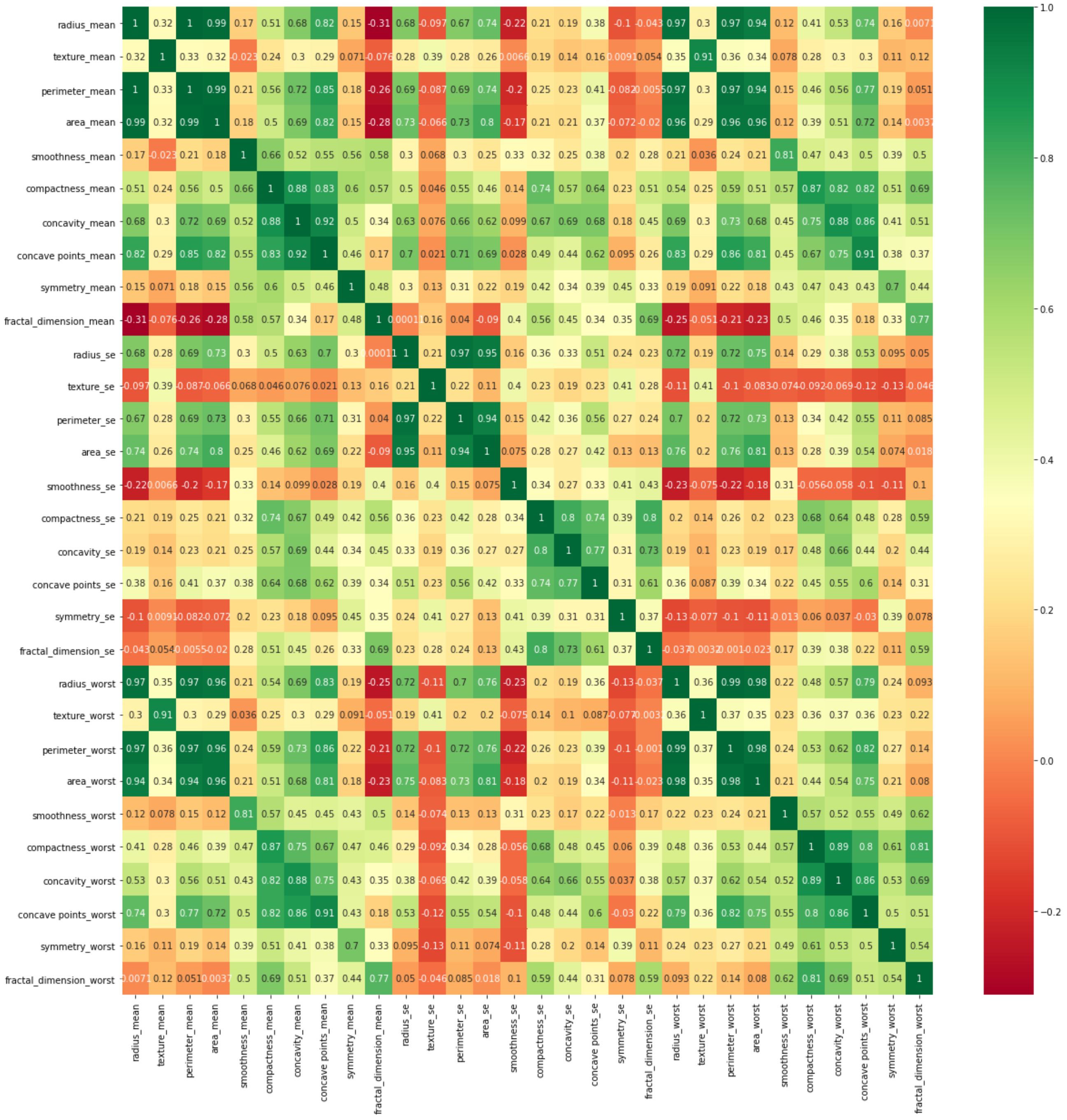
Let’s interpret one more thing about plot above:

Variable of concavity_worst and concave_point_worst looks like similar but how can we decide whether they are correlated with each other or not. (Not always true but, basically if the features are correlated with each other we can drop one of them)



HeatMap:

A heat map is a data visualization technique that shows magnitude of a phenomenon as colour in two dimensions. The variation in colour may be by hue or intensity, giving obvious visual cues to the reader about how the phenomenon is clustered or varies over space. There are two fundamentally different categories of heat maps: the cluster heat map and the spatial heat map. In a cluster heat map, magnitudes are laid out into a matrix of fixed cell size whose rows and columns are discrete phenomena and categories, and the sorting of rows and columns is intentional and somewhat arbitrary, with the goal of suggesting clusters or portraying them as discovered via statistical analysis. The size of the cell is arbitrary but large enough to be clearly visible. By contrast, the position of a magnitude in a spatial heat map is forced by the location of the magnitude in that space, and there is no notion of cells; the phenomenon is considered to vary continuously.



From the above Heatmap we can find some correlations among the features of the dataset that we are using. These correlations help us to find which feature is required for the classification along with which other features.

4. Feature Selection

In this part we will select feature with different methods that are feature selection with correlation, univariate feature selection, and tree based feature selection.

4.1 Feature Selection using Random Forest Classifier:

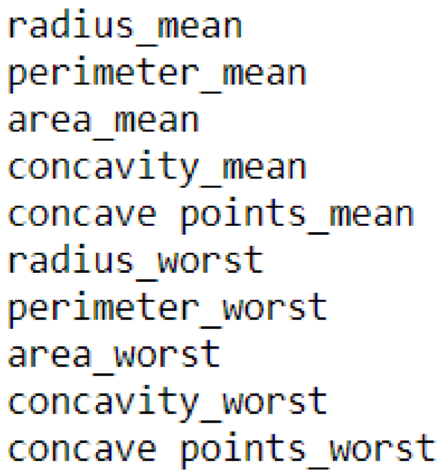
Random Forests are often used for feature selection in a data science workflow. The reason is because the tree-based strategies used by random forests naturally ranks by how well they improve the purity of the node. This mean decrease in impurity over all trees. Nodes with the greatest decrease in impurity happen at the start of the trees, while notes with the least decrease in impurity occur at the end of trees. Thus, by pruning trees below a particular node, we can create a subset of the most important features.

Feature selection using Random forest comes under the category of embedded methods. Embedded methods combine the qualities of filter and wrapper methods. They are implemented by algorithms that have their own built-in feature selection methods. Some of the benefits of embedded methods are:

- They are highly accurate.
- They generalize better.
- They are interpretable

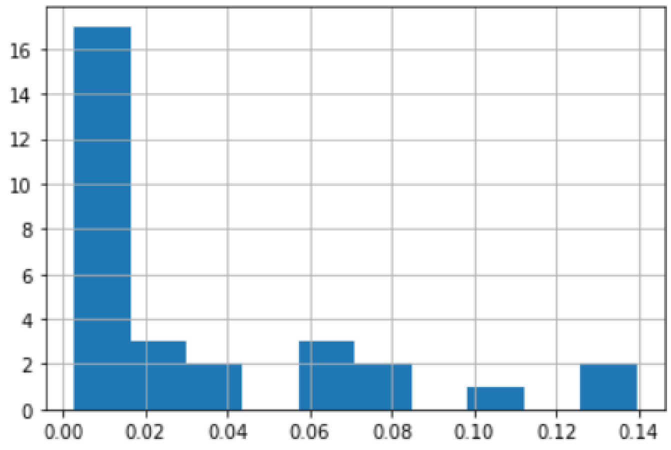
Random forests consist of 4 –12 hundred decision trees, each of them built over a random extraction of the observations from the dataset and a random extraction of the features. Not every tree sees all the features or all the observations, and this guarantees that the trees are de-correlated and therefore less prone to over-fitting. Each tree is also a sequence of yes-no questions based on a single or combination of features. At each node (this is at each question), the three divides the dataset into 2 buckets, each of them hosting observations that are more similar among themselves and different from the ones in the other bucket. Therefore, the importance of each feature is derived from how “pure” each of the buckets is.

Of all the 32 features the Random Forest Classifier obtains the following features as the most important features that play a major role in classification.



```
radius_mean
perimeter_mean
area_mean
concavity_mean
concave points_mean
radius_worst
perimeter_worst
area_worst
concavity_worst
concave points_worst
```

The Histogram for the above selected features from the above dataset is as follows:

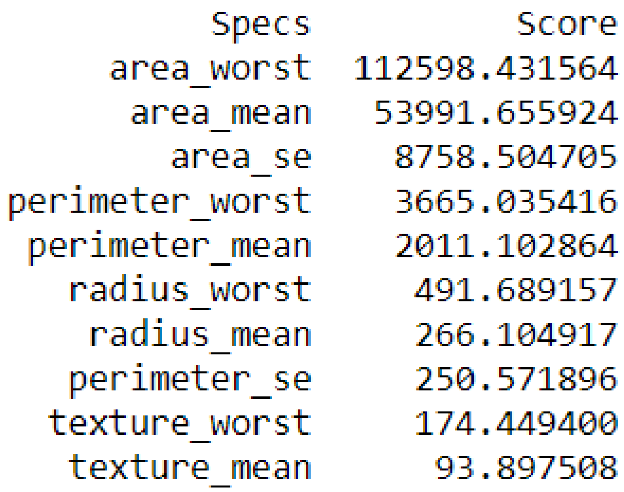


4.2 Feature Selection using Univariate Method:

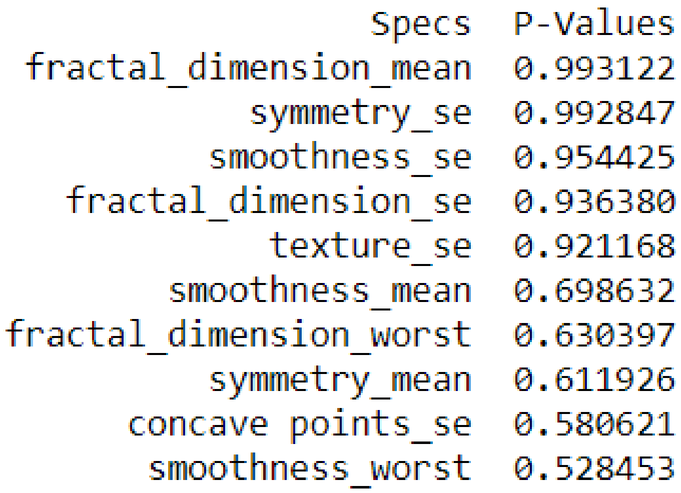
In univariate feature selection, we will use SelectKBest that removes all but the k highest scoring features.

Univariate feature selection examines each feature individually to determine the strength of the relationship of the feature with the response variable. These methods are simple to run and understand and are in general particularly good for gaining a better understanding of data (but not necessarily for optimizing the feature set for better generalization). There are lot of different options for univariate selection.

In this method we need to choose how many features we will use. For example, will k (number of features) be 5 or 10 or 15? WE have selected the best 10 features for the dataset to apply our ML model.

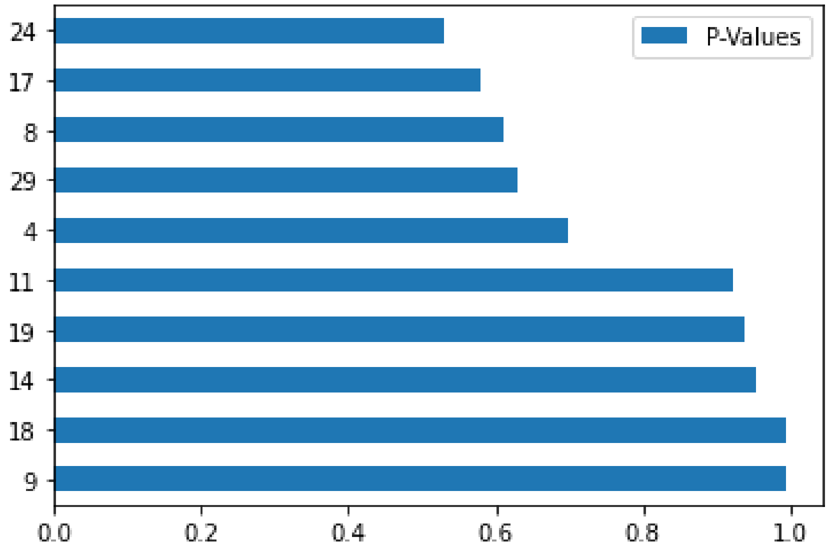


```
             Specs          Score
        area_worst  112598.431564
         area_mean   53991.655924
           area_se    8758.504705
   perimeter_worst    3665.035416
    perimeter_mean    2011.102864
      radius_worst     491.689157
       radius_mean     266.104917
      perimeter_se     250.571896
     texture_worst     174.449400
      texture_mean      93.897508
```



```
                  Specs  P-Values
 fractal_dimension_mean  0.993122
            symmetry_se  0.992847
          smoothness_se  0.954425
   fractal_dimension_se  0.936380
             texture_se  0.921168
        smoothness_mean  0.698632
fractal_dimension_worst  0.630397
          symmetry_mean  0.611926
      concave points_se  0.580621
       smoothness_worst  0.528453
```

Using the SelectKBest in finding the features that are important we found the above features based on their respective scores and their p-values. The Barplot for p-values of the respective features selected is as follows:



5. Fitting Machine Learning Models to Data

Classification is technique to categorize our data into a desired and distinct number of classes where we can assign label to each class. The present Dataset is of 2 distinct classes and hence it is a binary classification. The training dataset and test dataset from above data pre-processing is used to train and check the reliability of our model. The Accuracy on the test set are reported for each model.

1. Logistic Regression:

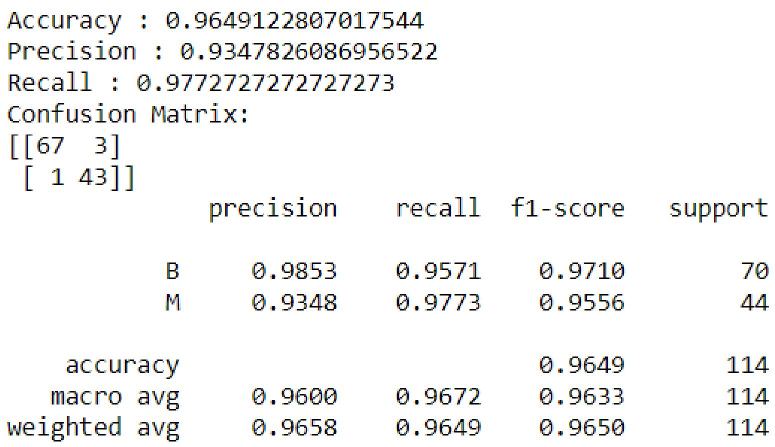
A Logistic Regression algorithm is used to create a binary classifier that is optimized on our training dataset. The Logistic Regression function from sklearn library is used to create and train our classifier.

1.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply "Label Encoding" on the output labels.

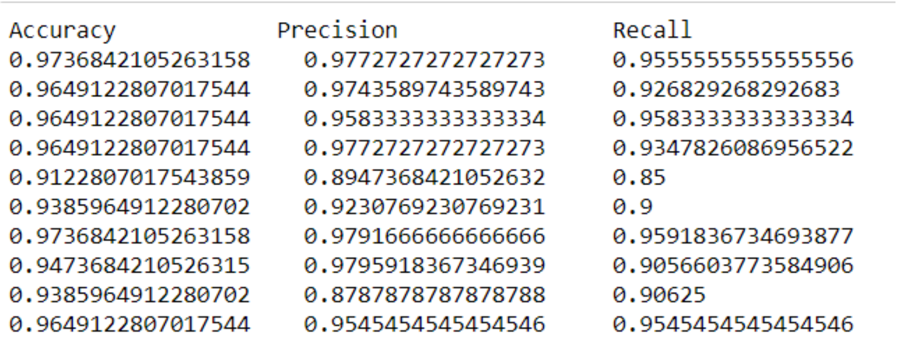


```
Accuracy : 0.9649122807017544
Precision : 0.9347826086956522
Recall : 0.9772727272727273
Confusion Matrix:
[[67  3]
 [ 1 43]]
              precision    recall  f1-score   support

           B     0.9853    0.9571    0.9710        70
           M     0.9348    0.9773    0.9556        44

    accuracy                         0.9649       114
   macro avg     0.9600    0.9672    0.9633       114
weighted avg     0.9658    0.9649    0.9650       114
```

-The max accuracy was about 96% when we use the features selected using random forest classifier method.



```
Accuracy              Precision              Recall
0.9736842105263158    0.9772727272727273     0.9555555555555556
0.9649122807017544    0.9743589743589743     0.926829268292683
0.9649122807017544    0.9583333333333334     0.9583333333333334
0.9649122807017544    0.9772727272727273     0.9347826086956522
0.9122807017543859    0.8947368421052632     0.85
0.9385964912280702    0.9230769230769231     0.9
0.9736842105263158    0.9791666666666666     0.9591836734693877
0.9473684210526315    0.9795918367346939     0.9056603773584906
0.9385964912280702    0.8787878787878788     0.90625
0.9649122807017544    0.9545454545454546     0.9545454545454546
```

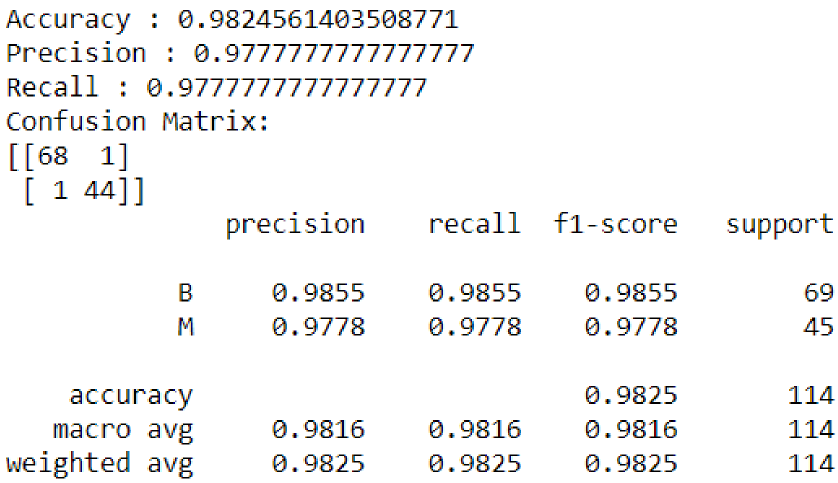
- On an average the accuracy is about 96% when the model is run for about 10 times.

1.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

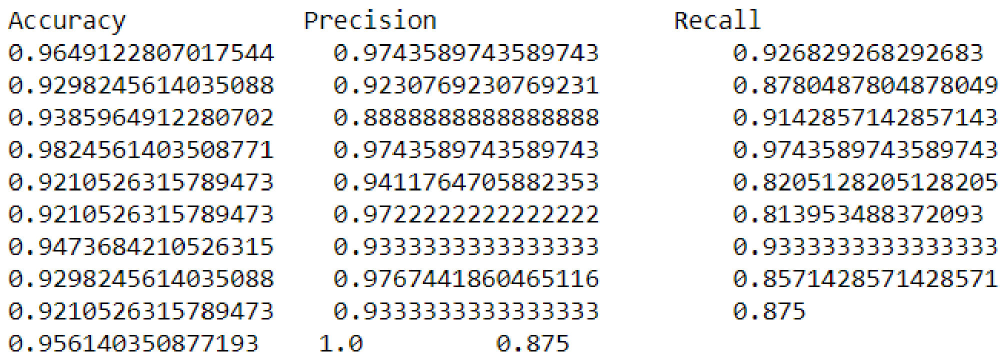


```
Accuracy : 0.9824561403508771
Precision : 0.9777777777777777
Recall : 0.9777777777777777
Confusion Matrix:
[[68  1]
 [ 1 44]]
              precision    recall  f1-score   support

           B     0.9855    0.9855    0.9855        69
           M     0.9778    0.9778    0.9778        45

    accuracy                         0.9825       114
   macro avg     0.9816    0.9816    0.9816       114
weighted avg     0.9825    0.9825    0.9825       114
```

- The max accuracy was about 98% when we use the features selected using random forest classifier method.



```
Accuracy             Precision              Recall
0.9649122807017544    0.9743589743589743        0.926829268292683
0.9298245614035088    0.9230769230769231        0.8780487804878049
0.9385964912280702    0.8888888888888888        0.9142857142857143
0.9824561403508771    0.9743589743589743        0.9743589743589743
0.9210526315789473    0.9411764705882353        0.8205128205128205
0.9210526315789473    0.9722222222222222        0.813953488372093
0.9473684210526315    0.9333333333333333        0.9333333333333333
0.9298245614035088    0.9767441860465116        0.8571428571428571
0.9210526315789473    0.9333333333333333        0.875
0.956140350877193    1.0          0.875
```

-On an average the accuracy is about 93% when the model is run for about 10 times.

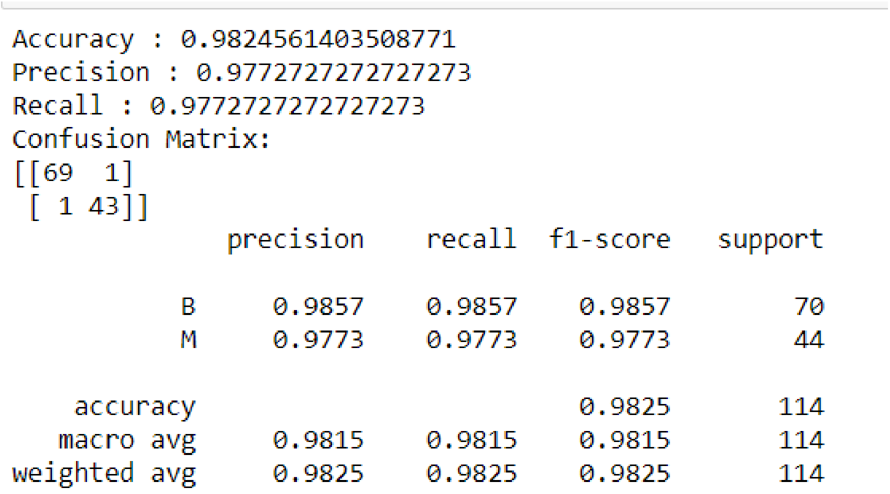
2. **Support Vector Machine (SVM) :**

The SVM training algorithm builds a model that assigns an datapoint to one class or the other. The SVM model is the representation of sample points in n dimensional space, mapped so that examples of different class are divided by a clear boundary or gap that is as wide as possible. The test or unseen examples are then mapped to the same space belonging to one category or the other based on which side of the boundary they fall. This boundary solve both linear or non linear classification problems based on the kernel methods used for training. A Support Vector Classifier is created using the SVC function of sklearn library. The parameters of the function used :

- **kernel** : Specifies the kernel type to be used in the algorithm. It must be one of ‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’, ‘precomputed’ or a callable. WE used the ‘rbf’ (radial basis function) kernel.

2.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.
-The dataset is feature scaled and is then trained with logistic regression model.
-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

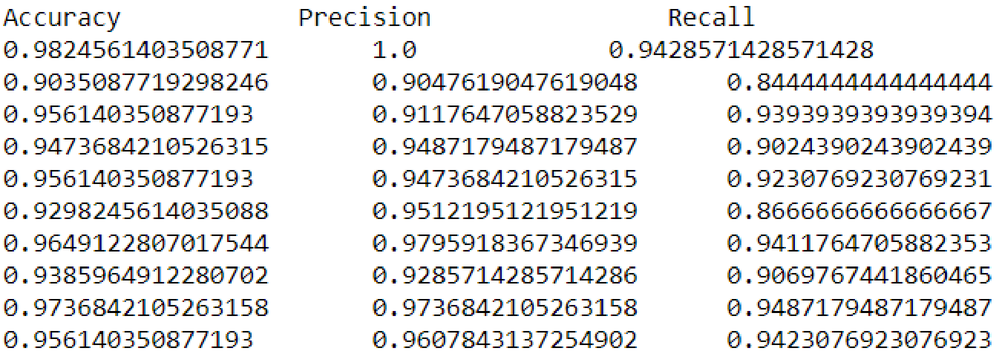


```
Accuracy : 0.9824561403508771
Precision : 0.9772727272727273
Recall : 0.9772727272727273
Confusion Matrix:
[[69  1]
 [ 1 43]]
              precision    recall  f1-score   support

           B     0.9857    0.9857    0.9857        70
           M     0.9773    0.9773    0.9773        44

    accuracy                         0.9825       114
   macro avg     0.9815    0.9815    0.9815       114
weighted avg     0.9825    0.9825    0.9825       114
```

-The max accuracy that has obtained using this model is about 98%.



```
Accuracy                  Precision                 Recall
0.9824561403508771        1.0             0.9428571428571428
0.9035087719298246        0.9047619047619048        0.8444444444444444
0.956140350877193         0.9117647058823529        0.9393939393939394
0.9473684210526315        0.9487179487179487        0.9024390243902439
0.956140350877193         0.9473684210526315        0.9230769230769231
0.9298245614035088        0.9512195121951219        0.8666666666666667
0.9649122807017544        0.9795918367346939        0.9411764705882353
0.9385964912280702        0.9285714285714286        0.9069767441860465
0.9736842105263158        0.9736842105263158        0.9487179487179487
0.956140350877193         0.9607843137254902        0.9423076923076923
```

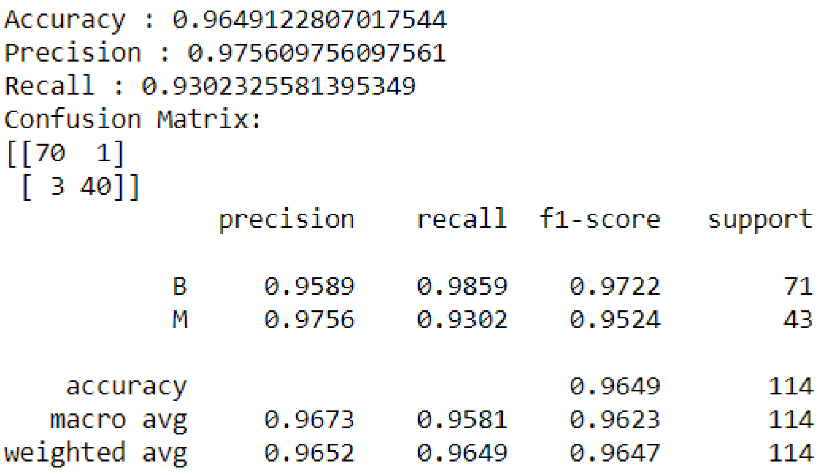
- On an average the accuracy is about 96% when the model is run for about 10 times.

2.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

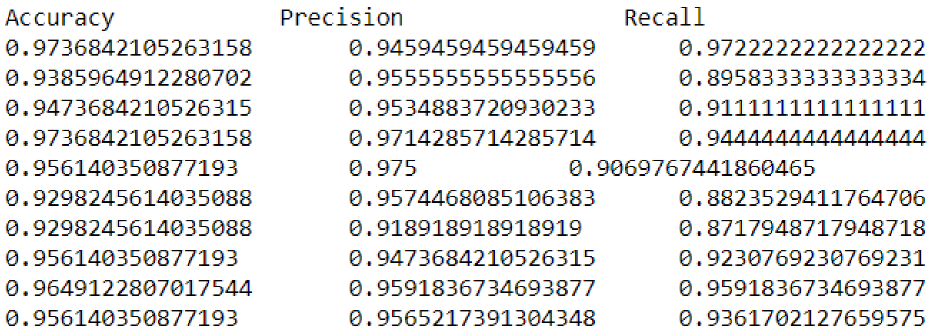


```
Accuracy : 0.9649122807017544
Precision : 0.975609756097561
Recall : 0.9302325581395349
Confusion Matrix:
[[70  1]
 [ 3 40]]
              precision    recall  f1-score   support

           B     0.9589    0.9859    0.9722        71
           M     0.9756    0.9302    0.9524        43

    accuracy                         0.9649       114
   macro avg     0.9673    0.9581    0.9623       114
weighted avg     0.9652    0.9649    0.9647       114
```

-The max accuracy obtained when using SVM is about 96%



```
Accuracy              Precision              Recall
0.9736842105263158       0.9459459459459459       0.9722222222222222
0.9385964912280702       0.9555555555555556       0.8958333333333334
0.9473684210526315       0.9534883720930233       0.9111111111111111
0.9736842105263158       0.9714285714285714       0.9444444444444444
0.956140350877193        0.975            0.9069767441860465
0.9298245614035088       0.9574468085106383       0.8823529411764706
0.9298245614035088       0.918918918918919        0.8717948717948718
0.956140350877193        0.9473684210526315       0.9230769230769231
0.9649122807017544       0.9591836734693877       0.9591836734693877
0.956140350877193        0.9565217391304348       0.9361702127659575
```

-On average the accuracy of the model is about 96% when it run for about 10 times.

3. Random Forest:

The random forest algorithm creats a forest with a number of Decision Trees. It is a type of Ensemble machine learning algorithm, which use a divide-and-conquer approach. The main principle behind ensemble algorithms is **boosting**, that is a group of weak learners (single estimator or a decision tree) can work together to form a strong learner (group of estimators or a forest) to classify the data. The random decision forests can correct for the decision trees' habit of overfitting to the training dataset. Hence, random forest algorithm comprises of **bagging** (Bootstrap aggregating), which is the approach to reduce overfitting by combining the classifications of randomly generated training sets, together with the random selection of features to construct a collection of decision forests. The Random Forest Classifier is created using the RandomForestClassifier function of sklearn library. The parameters of the function used :

- **n_estimators** : The number of decision trees in the forest, WE selected 100.

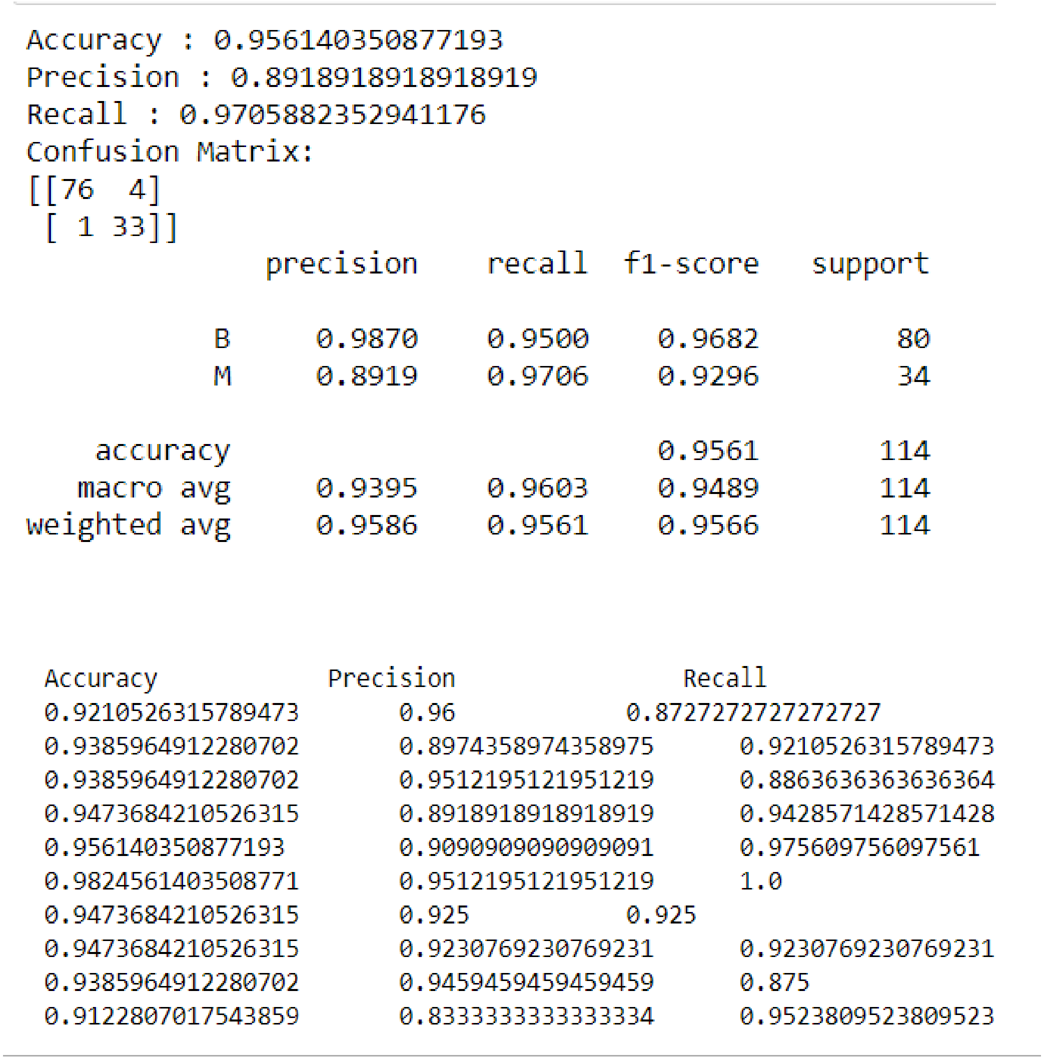
- **Criterion** : WE have used 'gini' importance criterion or the Mean Decrease in Impurity (MDI), which calculates each feature importance as the sum over the number of splits (across all tress) that include the feature, proportionally to the number of samples it splits.

3.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply "Label Encoding" on the output labels.



```
Accuracy : 0.956140350877193
Precision : 0.8918918918918919
Recall : 0.9705882352941176
Confusion Matrix:
[[76  4]
 [ 1 33]]
              precision    recall  f1-score   support

           B     0.9870    0.9500    0.9682        80
           M     0.8919    0.9706    0.9296        34

    accuracy                         0.9561       114
   macro avg     0.9395    0.9603    0.9489       114
weighted avg     0.9586    0.9561    0.9566       114
```

```
Accuracy                Precision                   Recall
0.9210526315789473          0.96            0.8727272727272727
0.9385964912280702          0.8974358974358975        0.9210526315789473
0.9385964912280702          0.9512195121951219        0.8863636363636364
0.9473684210526315          0.8918918918918919        0.9428571428571428
0.956140350877193           0.9090909090909091        0.975609756097561
0.9824561403508771          0.9512195121951219        1.0
0.9473684210526315          0.925           0.925
0.9473684210526315          0.9230769230769231        0.9230769230769231
0.9385964912280702          0.9459459459459459        0.875
0.9122807017543859          0.8333333333333334        0.9523809523809523
```

-The max accuracy achieved is 98% when using this model.

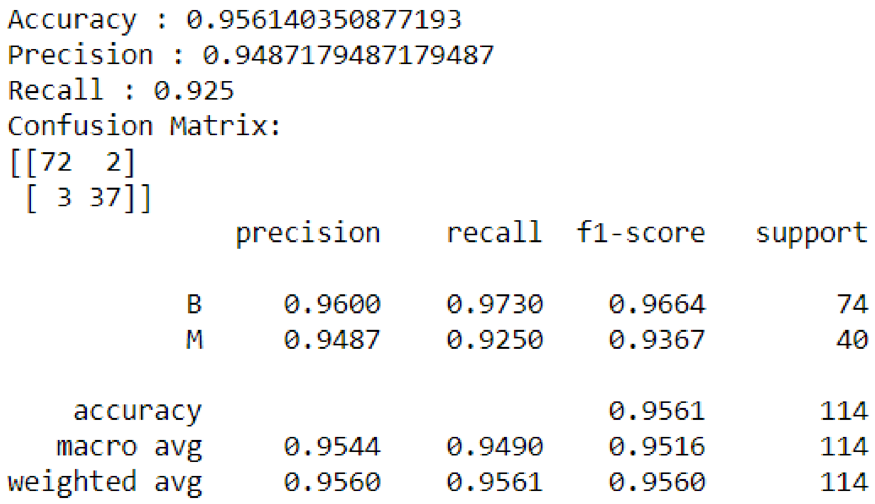
-On consecutive runs the accuracy average is about 96% on running it for about 10 times.

3.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

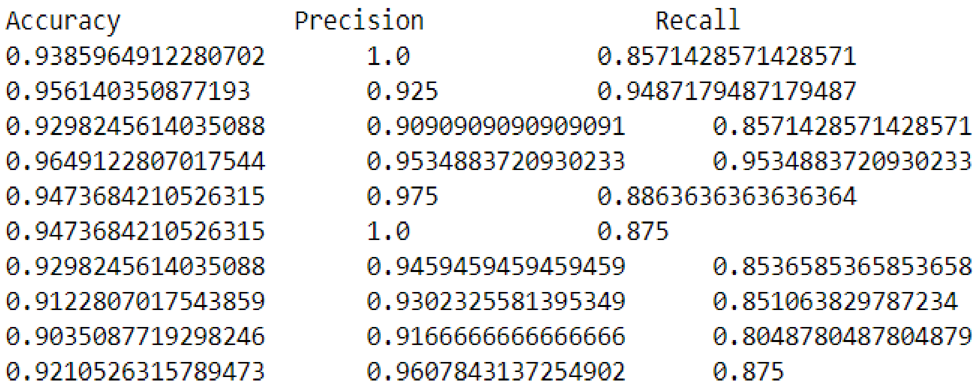
-For applying the logistic regression we have to apply “Label Encoding” on the output labels.



```
Accuracy : 0.956140350877193
Precision : 0.9487179487179487
Recall : 0.925
Confusion Matrix:
[[72  2]
 [ 3 37]]
              precision    recall  f1-score   support

           B     0.9600    0.9730    0.9664        74
           M     0.9487    0.9250    0.9367        40

    accuracy                         0.9561       114
   macro avg     0.9544    0.9490    0.9516       114
weighted avg     0.9560    0.9561    0.9560       114
```



```
Accuracy                  Precision                Recall
0.9385964912280702        1.0               0.8571428571428571
0.956140350877193         0.925             0.9487179487179487
0.9298245614035088        0.9090909090909091        0.8571428571428571
0.9649122807017544        0.9534883720930233        0.9534883720930233
0.9473684210526315        0.975             0.8863636363636364
0.9473684210526315        1.0               0.875
0.9298245614035088        0.9459459459459459        0.8536585365853658
0.9122807017543859        0.9302325581395349        0.851063829787234
0.9035087719298246        0.9166666666666666        0.8048780487804879
0.9210526315789473        0.9607843137254902        0.875
```

-The accuracy when using this model is about 95%.

-On consecutive runs of the machine learning model the accuracy varied among 93% - 94%.

4. Decision Tree:

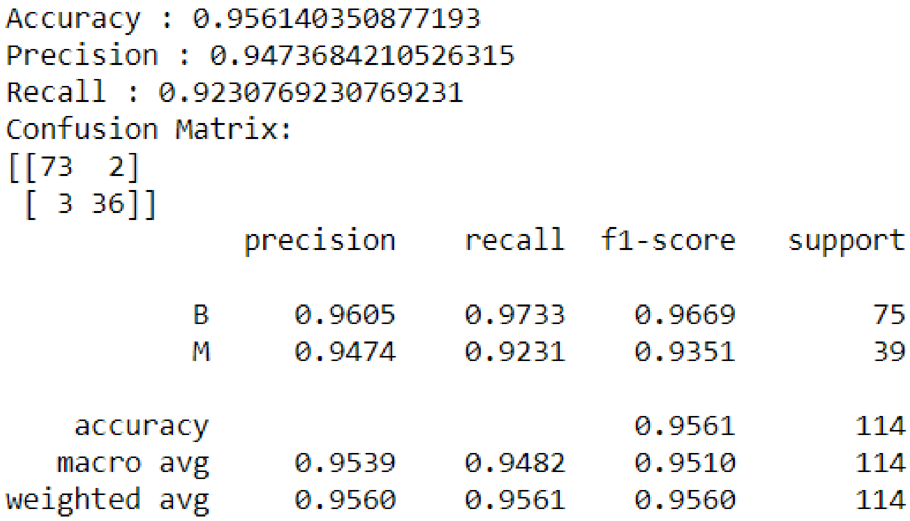
A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements. A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions.

4.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply "Label Encoding" on the output labels.

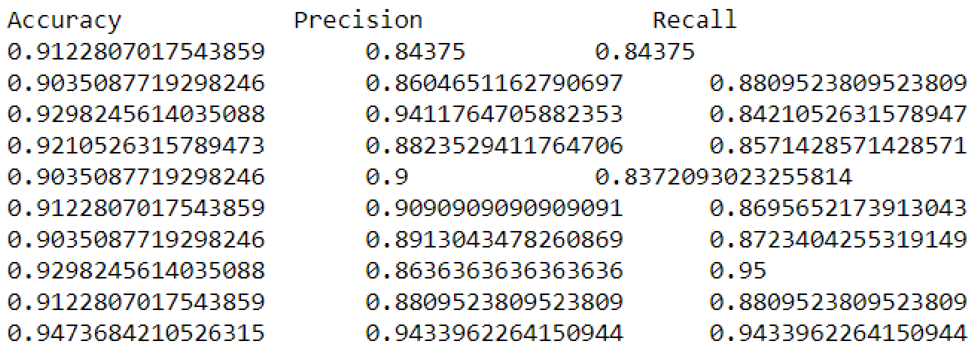


```
Accuracy : 0.956140350877193
Precision : 0.9473684210526315
Recall : 0.9230769230769231
Confusion Matrix:
[[73  2]
 [ 3 36]]
              precision    recall  f1-score   support

           B     0.9605    0.9733    0.9669        75
           M     0.9474    0.9231    0.9351        39

    accuracy                         0.9561       114
   macro avg     0.9539    0.9482    0.9510       114
weighted avg     0.9560    0.9561    0.9560       114
```

-The max accuracy achieved from the model is about 95% when using these features.



```
Accuracy            Precision               Recall
0.9122807017543859       0.84375         0.84375
0.9035087719298246       0.8604651162790697       0.8809523809523809
0.9298245614035088       0.9411764705882353       0.8421052631578947
0.9210526315789473       0.8823529411764706       0.8571428571428571
0.9035087719298246       0.9             0.8372093023255814
0.9122807017543859       0.9090909090909091       0.8695652173913043
0.9035087719298246       0.8913043478260869       0.8723404255319149
0.9298245614035088       0.8636363636363636       0.95
0.9122807017543859       0.8809523809523809       0.8809523809523809
0.9473684210526315       0.9433962264150944       0.9433962264150944
```

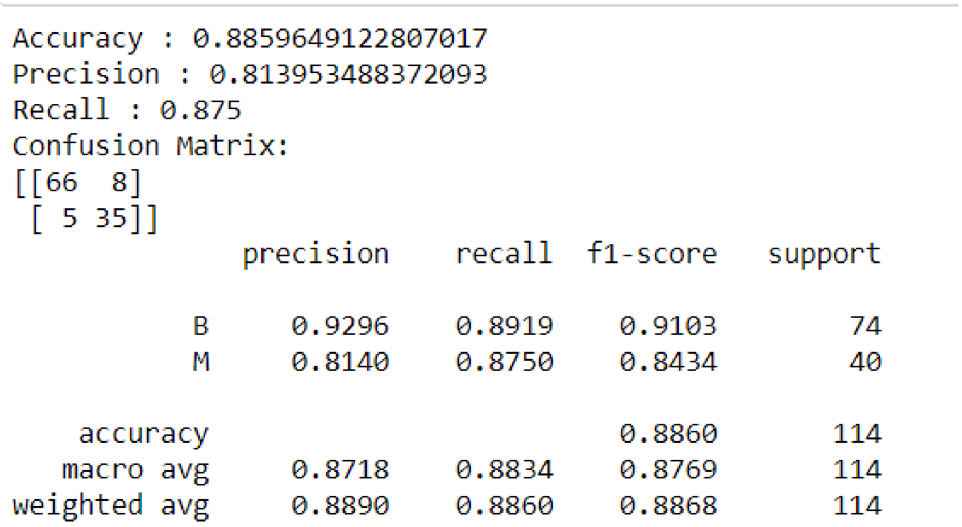
-On consecutive runs the model accuracy mean is about 91% when it is run 10 times.

4.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

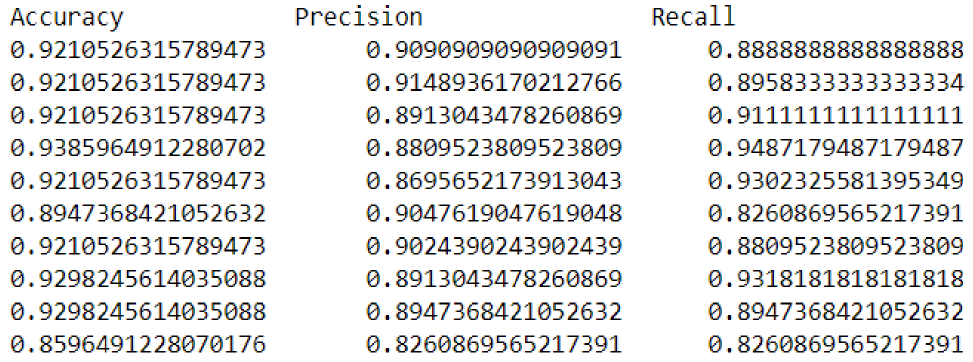


```
Accuracy : 0.8859649122807017
Precision : 0.813953488372093
Recall : 0.875
Confusion Matrix:
[[66  8]
 [ 5 35]]
              precision    recall  f1-score   support

           B     0.9296    0.8919    0.9103        74
           M     0.8140    0.8750    0.8434        40

    accuracy                         0.8860       114
   macro avg     0.8718    0.8834    0.8769       114
weighted avg     0.8890    0.8860    0.8868       114
```

-The accuracy is about 88% first time when we train the model.

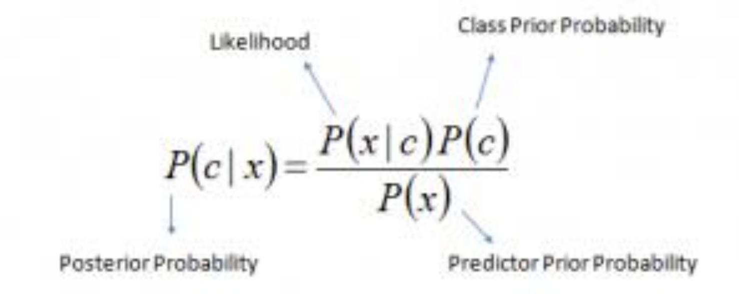


```
Accuracy            Precision               Recall
0.9210526315789473       0.9090909090909091      0.8888888888888888
0.9210526315789473       0.9148936170212766      0.8958333333333334
0.9210526315789473       0.8913043478260869      0.9111111111111111
0.9385964912280702       0.8809523809523809      0.9487179487179487
0.9210526315789473       0.8695652173913043      0.9302325581395349
0.8947368421052632       0.9047619047619048      0.8260869565217391
0.9210526315789473       0.9024390243902439      0.8809523809523809
0.9298245614035088       0.8913043478260869      0.9318181818181818
0.9298245614035088       0.8947368421052632      0.8947368421052632
0.8596491228070176       0.8260869565217391      0.8260869565217391
```

-On consecutive runs, the accuracy is about 91% when we run about 10 times.

5. Naïve Bayes :

It is a classification algorithm based on Bayes' Theorem with an assumption of independence among predictors. A Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods. If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as "Zero Frequency".



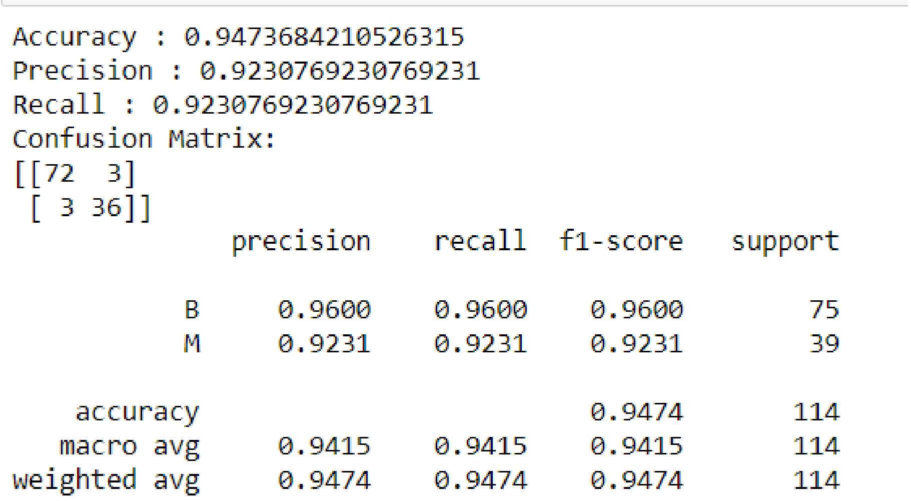
$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

5.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply "Label Encoding" on the output labels.

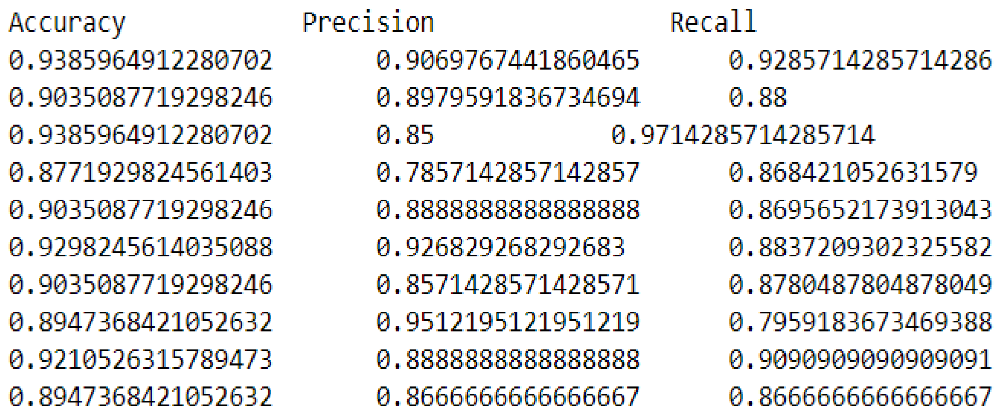


```
Accuracy : 0.9473684210526315
Precision : 0.9230769230769231
Recall : 0.9230769230769231
Confusion Matrix:
[[72  3]
 [ 3 36]]
              precision    recall  f1-score   support

           B     0.9600    0.9600    0.9600        75
           M     0.9231    0.9231    0.9231        39

    accuracy                         0.9474       114
   macro avg     0.9415    0.9415    0.9415       114
weighted avg     0.9474    0.9474    0.9474       114
```

-The max accuracy achieved when running the model is about 94%.



```
Accuracy              Precision                Recall
0.9385964912280702         0.9069767441860465        0.9285714285714286
0.9035087719298246         0.8979591836734694        0.88
0.9385964912280702         0.85              0.9714285714285714
0.8771929824561403         0.7857142857142857        0.868421052631579
0.9035087719298246         0.8888888888888888        0.8695652173913043
0.9298245614035088         0.926829268292683         0.8837209302325582
0.9035087719298246         0.8571428571428571        0.8780487804878049
0.8947368421052632         0.9512195121951219        0.7959183673469388
0.9210526315789473         0.8888888888888888        0.9090909090909091
0.8947368421052632         0.8666666666666667        0.8666666666666667
```

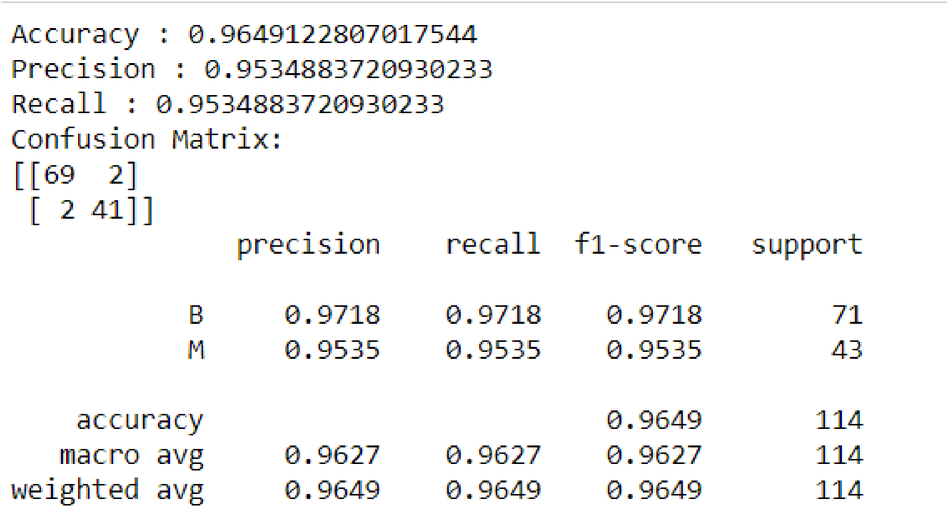
-On an average the accuracy is about 91% when using the features selected by this method.

5.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

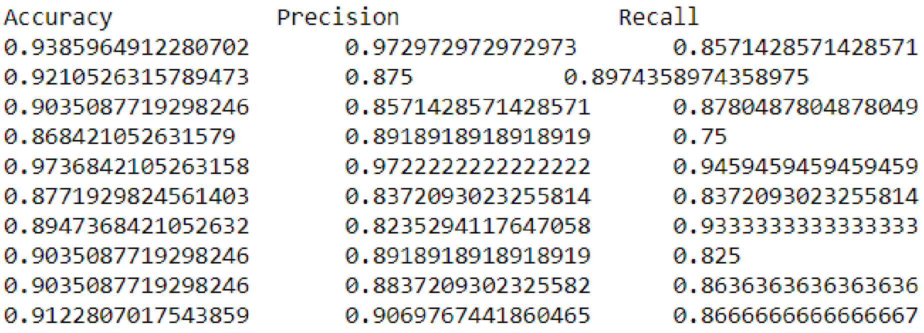


```
Accuracy : 0.9649122807017544
Precision : 0.9534883720930233
Recall : 0.9534883720930233
Confusion Matrix:
[[69  2]
 [ 2 41]]
              precision    recall  f1-score   support

           B     0.9718    0.9718    0.9718        71
           M     0.9535    0.9535    0.9535        43

    accuracy                         0.9649       114
   macro avg     0.9627    0.9627    0.9627       114
weighted avg     0.9649    0.9649    0.9649       114
```

-The max accuracy achieved is about 96% when using these features.



```
Accuracy           Precision            Recall
0.9385964912280702     0.972972972972973      0.8571428571428571
0.9210526315789473     0.875          0.8974358974358975
0.9035087719298246     0.8571428571428571     0.8780487804878049
0.868421052631579      0.8918918918918919     0.75
0.9736842105263158     0.9722222222222222     0.9459459459459459
0.8771929824561403     0.8372093023255814     0.8372093023255814
0.8947368421052632     0.8235294117647058     0.9333333333333333
0.9035087719298246     0.8918918918918919     0.825
0.9035087719298246     0.8837209302325582     0.8636363636363636
0.9122807017543859     0.9069767441860465     0.8666666666666667
```

-On an average the accuracy is about 91% when using the features selected by this method.

6. **K-Nearest Neighbors:**

K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems.

Lazy learning algorithm − KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.

Non-parametric learning algorithm − KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.

Process:-

Step 1 − For implementing any algorithm, we need dataset. So during the first step of KNN, we must load the training as well as test data.

Step 2 − Next, we need to choose the value of K We.e. the nearest data points. K can be any integer.

Step 3 − For each point in the test data do the following –

3.1− Calculate the distance between test data and each row of training data with the help of any of the method namely: Euclidean, Manhattan or Hamming distance. The most commonly used method to calculate distance is Euclidean.

3.2 − Now, based on the distance value, sort them in ascending order.

3.3 − Next, it will choose the top K rows from the sorted array.

3.4 − Now, it will assign a class to the test point based on most frequent class of these rows.

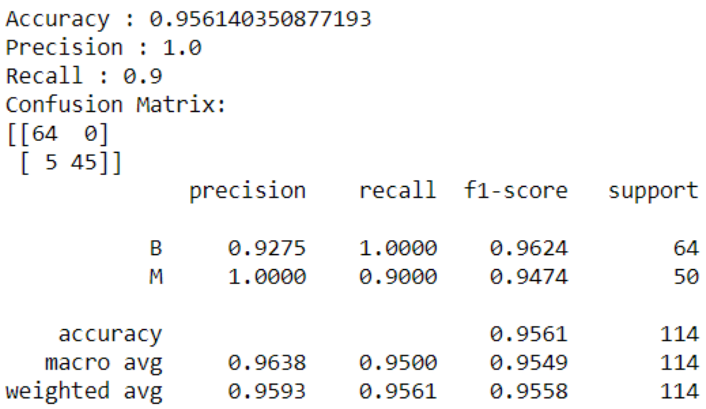
Step 4 – End

6.1 Applying Model on Random Forest Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply “Label Encoding” on the output labels.

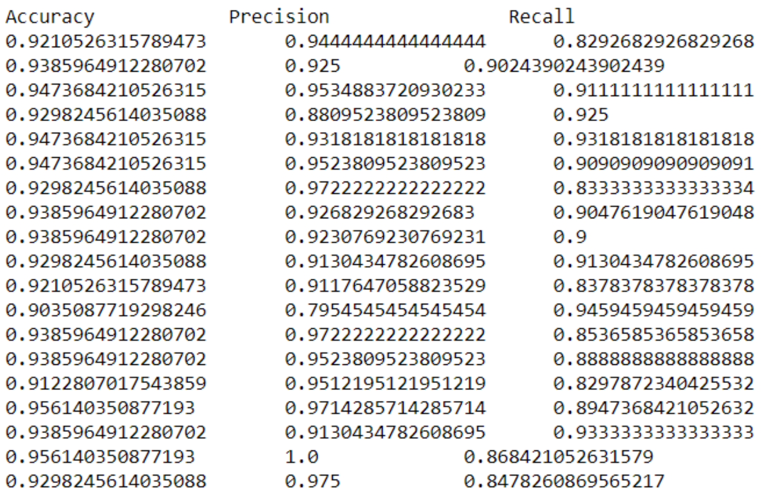


```
Accuracy : 0.956140350877193
Precision : 1.0
Recall : 0.9
Confusion Matrix:
[[64  0]
 [ 5 45]]
              precision    recall  f1-score   support

           B     0.9275    1.0000    0.9624        64
           M     1.0000    0.9000    0.9474        50

    accuracy                         0.9561       114
   macro avg     0.9638    0.9500    0.9549       114
weighted avg     0.9593    0.9561    0.9558       114
```

-The accuracy obtained on K = 2 is about 95%.



```
Accuracy             Precision                Recall
0.9210526315789473       0.9444444444444444       0.8292682926829268
0.9385964912280702       0.925            0.9024390243902439
0.9473684210526315       0.9534883720930233       0.9111111111111111
0.9298245614035088       0.8809523809523809       0.925
0.9473684210526315       0.9318181818181818       0.9318181818181818
0.9473684210526315       0.9523809523809523       0.9090909090909091
0.9298245614035088       0.9722222222222222       0.8333333333333334
0.9385964912280702       0.926829268292683        0.9047619047619048
0.9385964912280702       0.9230769230769231       0.9
0.9298245614035088       0.9130434782608695       0.9130434782608695
0.9210526315789473       0.9117647058823529       0.8378378378378378
0.9035087719298246       0.7954545454545454       0.9459459459459459
0.9385964912280702       0.9722222222222222       0.8536585365853658
0.9385964912280702       0.9523809523809523       0.8888888888888888
0.9122807017543859       0.9512195121951219       0.8297872340425532
0.956140350877193        0.9714285714285714       0.8947368421052632
0.9385964912280702       0.9130434782608695       0.9333333333333333
0.956140350877193        1.0              0.868421052631579
0.9298245614035088       0.975            0.8478260869565217
```

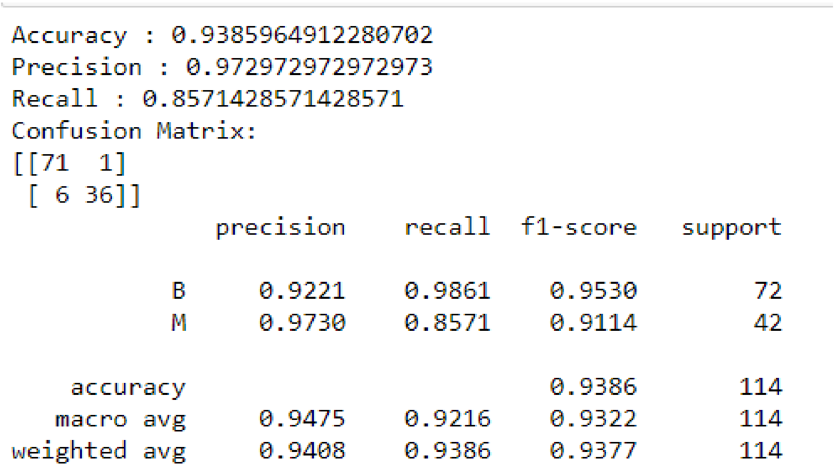
-The accuracies are as above when we use different neighbors between 2 to 20.

6.2 Applying Model on Univariate Method Feature selection:

-The Features Selected from the dataset are: radius_mean, perimeter_mean, area_mean, concavity_mean, concave points_mean, radius_worst, perimeter_worst, area_worst, concavity_worst, concave points_worst.

-The dataset is feature scaled and is then trained with logistic regression model.

-For applying the logistic regression we have to apply "Label Encoding" on the output labels.

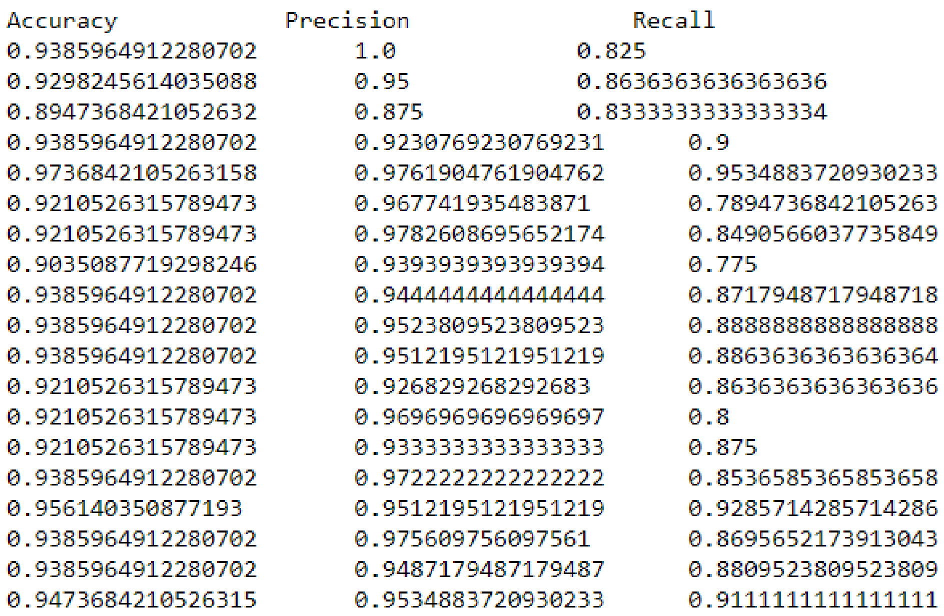


```
Accuracy : 0.9385964912280702
Precision : 0.972972972972973
Recall : 0.8571428571428571
Confusion Matrix:
[[71  1]
 [ 6 36]]
              precision    recall  f1-score   support

           B     0.9221    0.9861    0.9530        72
           M     0.9730    0.8571    0.9114        42

    accuracy                         0.9386       114
   macro avg     0.9475    0.9216    0.9322       114
weighted avg     0.9408    0.9386    0.9377       114
```

-The accuracy obtained when the neighbours are 2 is 93% using the features seleted by univariate feature selection method.



```
Accuracy            Precision                Recall
0.9385964912280702       1.0             0.825
0.9298245614035088       0.95            0.8636363636363636
0.8947368421052632       0.875           0.8333333333333334
0.9385964912280702       0.9230769230769231      0.9
0.9736842105263158       0.9761904761904762      0.9534883720930233
0.9210526315789473       0.967741935483871       0.7894736842105263
0.9210526315789473       0.9782608695652174      0.8490566037735849
0.9035087719298246       0.9393939393939394      0.775
0.9385964912280702       0.9444444444444444      0.8717948717948718
0.9385964912280702       0.9523809523809523      0.8888888888888888
0.9385964912280702       0.9512195121951219      0.8863636363636364
0.9210526315789473       0.926829268292683       0.8636363636363636
0.9210526315789473       0.9696969696969697      0.8
0.9210526315789473       0.9333333333333333      0.875
0.9385964912280702       0.9722222222222222      0.8536585365853658
0.956140350877193        0.9512195121951219      0.9285714285714286
0.9385964912280702       0.975609756097561       0.8695652173913043
0.9385964912280702       0.9487179487179487      0.8809523809523809
0.9473684210526315       0.9534883720930233      0.9111111111111111
```

-The accuracies are as above when we use different neighbours between 2 to 20.

6. Conclusion

The aim of this project is to apply multiple classification algorithms and be able to classify the test data. We collected our breast cancer dataset from Kaggle website which is actually a work done by some researchers in Wisconsin Town. The data contains 32 Features of which many may not be helpful in classifying the cancer. So we have applied Feature Selection Methods to extract the features that are actually useful in classification process. The data then have fewer number of features and we have applied Feature Scaling to the data to make the values normalized and we have even applied label encoder to convert the categorical string data to integral type of data. The data is then fed into the Machine Learning Models and results are having higher accuracy with Random Forest Feature Selection Method, so we have considered those features. The result accuracy for all models are nearly the same but SVM, Random Forest, and KNN are having more accuracy. The results extrapolated from the experiment are as follows:

Classifier Name	Precision	Recall	F1-Score	Accuracy
Logistic Regression	94.5%	92%	97%	94.3%
Gaussian Naïve Bayes	88%	87%	96%	94%
SVC Linear Kernel	94%	94%	95%	96.7%
Decision Tree	90%	89.4%	96%	91%
Random Forest	95%	96%	95%	95.34 %
KNN	92%	91%	96.2%	95%

The accuracy of SVM increased with a different set of Kernel, an Hyper Parameter available in sklearn’s SVC library. This gives the utmost accuracy as 98.24%, while on training with the same data again from the scratch gives approximately the previous accuracy. On repeating the process again and again we have an accuracy range of about 96% - 97.3%. Thus the conclusion SVM Classifier with some change in parameters performs better than the traditional SVM Classifier and predicts whether Breast Cancer is Benign or Malignant with higher accuracy and confidence Levels.

References

1. Kaggle, Breast Cancer Diagnostic Dataset**,** ***"https://www.kaggle.com/uciml/breast-cancer-wisconsin-data*** *", URL obtained on June 3, 2020.*
2. Feature Selection Algorithms, ***"https://towardsdatascience.com/the-5-feature-selection-algorithms-every-data-scientist-need-to-know-3a6b566efd2"***, URL obtained on June 15, 2020.
3. Algorithm References, ***"https://scikit-learn.org/stable/supervised_learning.html#supervised-learning"***, URL obtained on June 17, 2020.