Prediction of Breast Cancer Diseases by using Supervised Learning Models

Sean Chen[1] and Dong Si[2]

[1] University of Washington Bothell, USA
[1] seanyc@uw.edu [2]dongsi@uw.edu

Abstract. Breast cancer has been recognized as one of the most fatal diseases in the modern world, and it kills nearly 40, 000 people each year in the United States alone [1]. The prognosis and diagnosis play a crucial role in the treatment process of the cancer. However, it is approximately only 90% accurate in regular visual diagnosed breast fine need aspirates [2]. In this paper, the *Breast Cancer Wisconsin (Diagnostic) Dataset* that contains key cell nucleus features generated from fine needle aspirates of breast masses in text format is analyzed through Python program to make predictions for two classes: benign, or malignant breast mass through Supervised learning and Classification. The Regression model is applied to model the predictor points, and Decision trees are constructed from classification of the data cases. A variety of supervised learning techniques and algorithms are implemented, including Neural Networks, K-Nearest Neighbors, and Support Vector Machine and Random Forest to analyze the importance of each key feature of breast cell nuclei for model training.

Keywords: Supervised Learning, Classification, Regression, Random Forest, Support Vector Machine, Neural Networks, K-Nearest Neighbors

1 Introduction

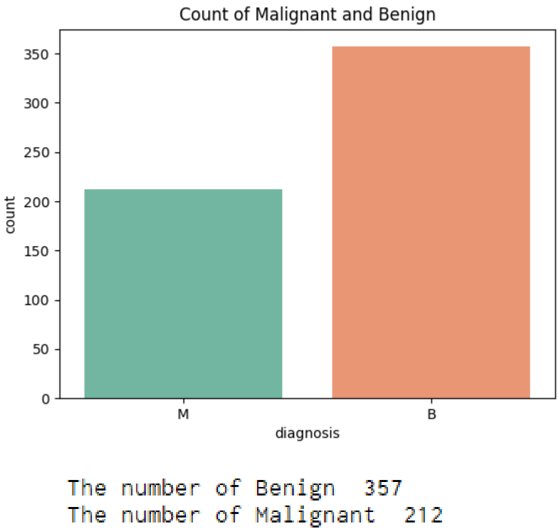
Breast cancer has become a several concern for many families and medical care practitioners. It may be found in both women and men, though mostly women are the victims of the breast cancer. According to the statistics of American Cancer Society, nearly 10% of the women population are found with breast cancer at some time of their life, and the survival rate is only 80% after 10 years from the time point of diagnosis and 88% after 5 years from the time point of diagnosis [3]. Hence, it is critical to doctors and physicians to accurately diagnose the cancer problems at the early stage of the cancer development process and follow up with timely and effective treatment plans. It requires Therefore, machine learning techniques have been developed and utilized in a wide scope of medical care practices to help doctors better understand the key characteristics of the breast cell nuclei so as to diagnose and further predict the diseases.

The purpose of this project is to predict if a breast cancer cell is malignant or benign based on featured cell nucleus properties. In the dataset, each feature's data values are computed based on a digital image of a fine needle aspirate of a breast mass. The image planes were generated through linear programming and decision trees. We will use four different models—Support Vector Machine, K-Nearest Neighbors, Neural Networks and Random Forest to conduct a comparative analysis of the data set, and find out which model performs the best.

2 Methods

2.1 Pre-Processing

I first cleaned up and pre-processed the dataset as the column labels and a few data values were missing. I also generated histogram graphs to understand the distributions of two sampling classes and a key feature epithelial cell size.



```
The number of Benign  357
The number of Malignant  212
```

Fig. 1 Count of M and B

Based on Fig. 1, we can tell the count of each of the benign and malignant diagnosis cases so that we have a general idea about the ratio of these two cases in the data set.

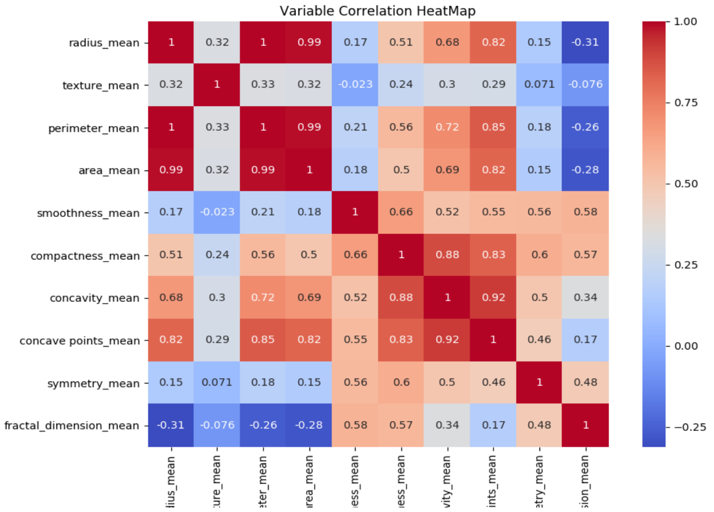


Fig. 2 Variable Correlation Heat Map

As Fig.2 and 3 illustrates, the variables are the mean value of each feature data collected. The red boxes indicate stronger variable feature correlations, and the blue boxes indicate weak variable feature correlations. Therefore, we can see that the cell radius, perimeter, area are the most correlated features, and compactness, concavity and concave points are secondary correlated features.

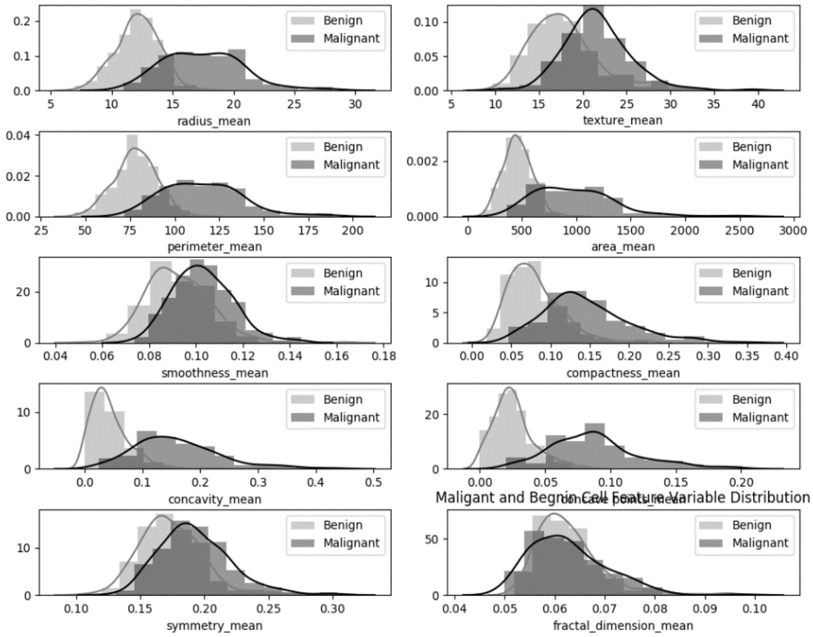


Figure 3 All Variable Distribution

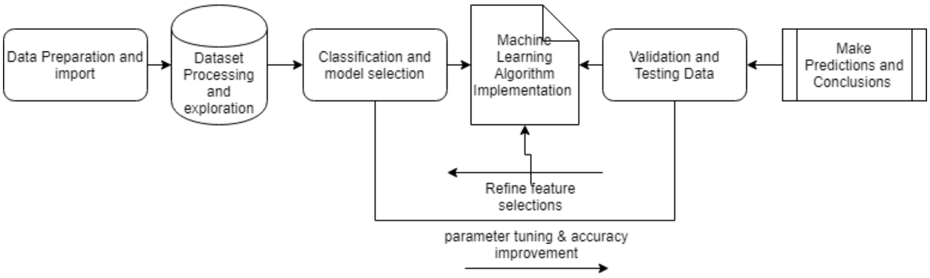

Data Preparation and import
Dataset Processing and exploration
Classification and model selection
Machine Learning Algorithm Implementation
Validation and Testing Data
Make Predictions and Conclusions
Refine feature selections
parameter tuning & accuracy improvement

Figure 4 Project Work Flowchart

The project flowchart represents the procedures of the workflow. After data exploration, four different models—Support Vector Machine, K-Nearest Neighbors, Neural Networks and Random Forest to find out which one has the best performances, then analyze the validation and accuracy to tune and improve the model, and eventually we analyze the feature selection to determine which factor plays the most crucial role in diagnosing breast cancer cells.

2.2 Decision Tree

Decision Tree is a tree-structure-based classifier in which each parent node represents the input choice and child nodes represents the outcome based on the choice. We and perform the algorithm on the dataset variables, in particular, the feature values and target values. Then the predications will be made according to the algorithm and rules we set to make a tree. It is one of the earliest and simplest machine learning classifiers to analyze categorical and numerical dataset. Despite the advantage of implementation illustration, decision tree could become very different due to a little change of rules or data values, and it could also become very complex structured tree [4].

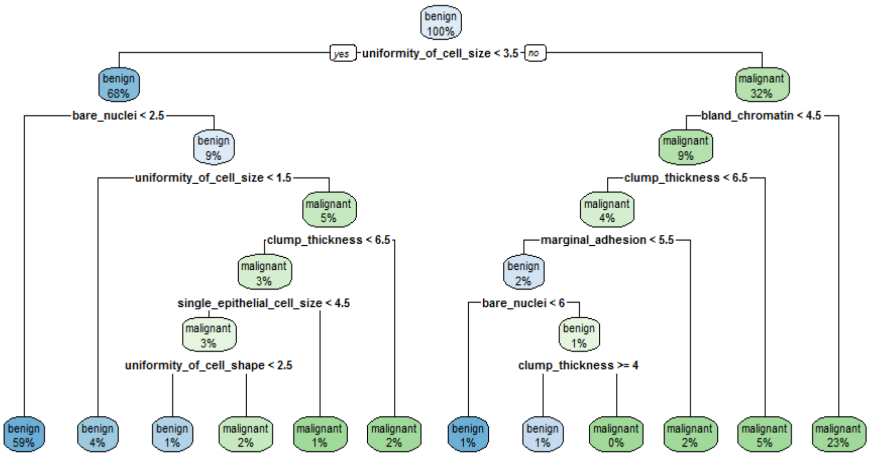


Figure 5 Decision Tree of Breast Cancer Prediction

2.3 Random Forest

Based on the decision tree constructed, random forest algorithm can be developed to find the best fitting pattern and correct the issue of overfitting to the training dataset caused by the decision tree. More importantly, it can be used to solve the variable importance problem. If possible, we may implement ExtraTrees technique to randomize the tree output and optimize the result.

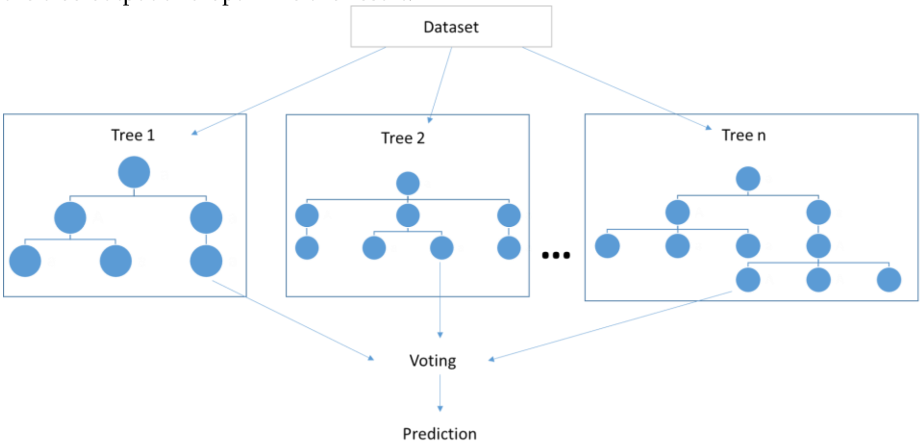


Figure 6 Random Forest Illustration

2.4 K-Nearest Neighbors

The K-Nearest Neighbors (KNN) technique is a non-parametric algorithm that tracks the k closest training data points with added weight to the contributions of the neighbors to find the feature pattern. The performance of KNN has been improved by supervised metric learning so that it can hand a large number of datasets and large margin nearest neighbor learning. It is also great for feature extraction through transferring the variable data values to certain feature sets through principal component analysis.

The error rate is also in control, and it brings accuracy and clarity of scientific computation. Based on Cover and Hart, the upper bound error rate $R_{KNN}$is

$$R^* \leq R_{KNN} \leq \left(2 - \frac{MR^*}{M-1}\right) * R^*$$

Where M is the count of class instances, and R* is the Bayes error rates.

2.5 Neural Networks

Developed from the biological neuron networks, neural networks, also often referred as Artificial Neural Networks (ANN) provides a framework for training to generate an output as a collection of the variable features. The multi-hidden layers can illustrate the connections of input and output collected variables [4,5].

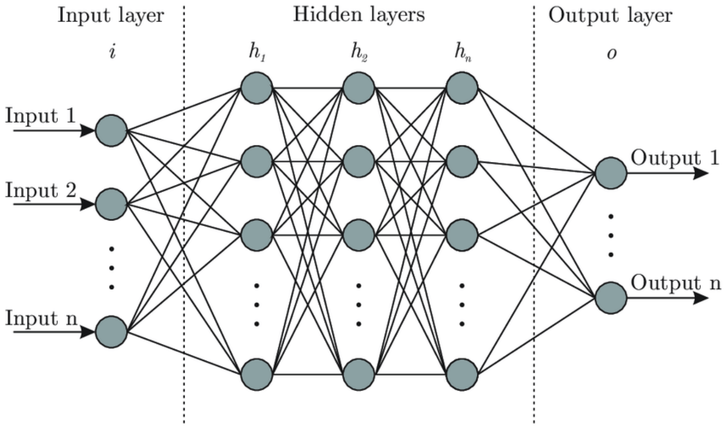


Figure 7 Artificial Neural Networks

2.6 Support Vector Machine

Support Vector Machine is a non-probabilistic linear classifier. It constructs a pattern that contains a hyperplane to divide the dataset feature space into two regions, based on the categories, to represent the data points [6]. It can be computed as below:

$$[\frac{1}{n}\sum \max(0, 1 - y_i(w * x_i - b))] + \ lamda|w|^2$$

where w is the vector.

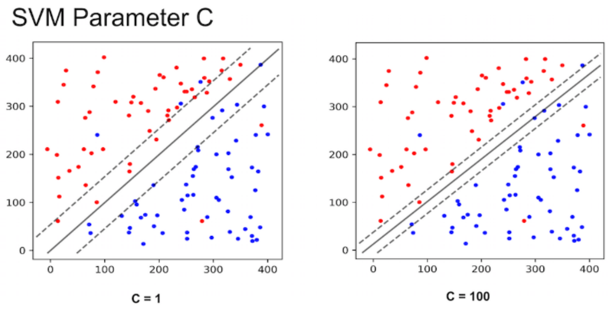
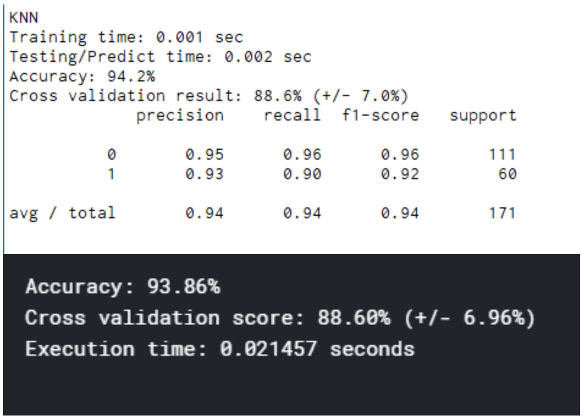


Figure 8 Support Vector Machine

3 Experiment and Result

The data set is split into two parts: training set and testing set. By implementing *train_test_split(feature_mean, diag, test_size = 0.3, random_state = 60)* in the Python program, we can randomly split the data values by setting the random state to a constant integer 60, and test size equal to 0.3. That will ensure us that two data sets can be split randomly, and 30% of the data will be used for testing and 70% will be used for training. That also help us avoid the issue of overfitting, which is to duplicate the feature of the data points with a perfect cross validation score so as to not be able to predict any unseen data points.

Then four models (SVM, Neural Networks, KNN and Random Forest) are implemented to be trained and test on the predication.

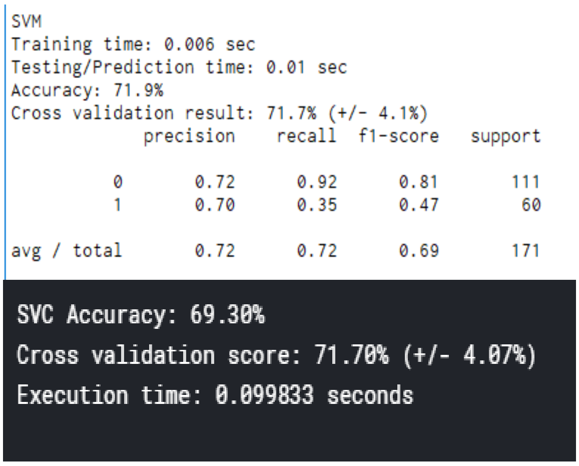


```
KNN
Training time: 0.001 sec
Testing/Predict time: 0.002 sec
Accuracy: 94.2%
Cross validation result: 88.6% (+/- 7.0%)
             precision    recall  f1-score   support

          0       0.95      0.96      0.96       111
          1       0.93      0.90      0.92        60

avg / total       0.94      0.94      0.94       171
```

```
Accuracy: 93.86%
Cross validation score: 88.60% (+/- 6.96%)
Execution time: 0.021457 seconds
```

Fig. 9 KNN Result

As Fig. 9 illustrates the result of the K-Nearest Neighbors model, the top result is generated by my program, and the bottom result is produced by Rodolfo Camrgo de Feitas who perform the similar task as mine. For the same data set, mine produces a slightly better accuracy, though we had the same cross validation score.

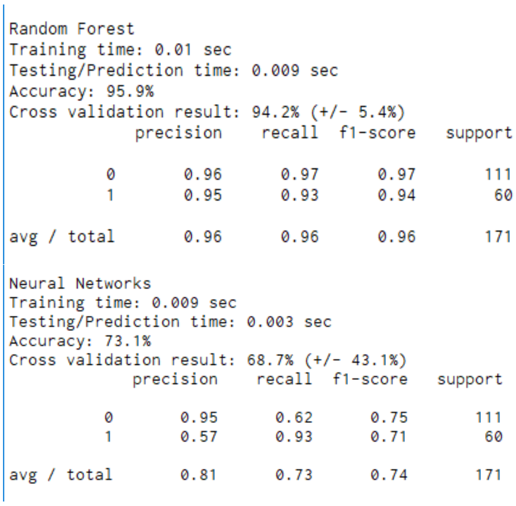


```
SVM
Training time: 0.006 sec
Testing/Prediction time: 0.01 sec
Accuracy: 71.9%
Cross validation result: 71.7% (+/- 4.1%)
             precision    recall  f1-score   support

          0       0.72      0.92      0.81       111
          1       0.70      0.35      0.47        60

avg / total       0.72      0.72      0.69       171
```

```
SVC Accuracy: 69.30%
Cross validation score: 71.70% (+/- 4.07%)
Execution time: 0.099833 seconds
```

Fig. 10 SVM Result

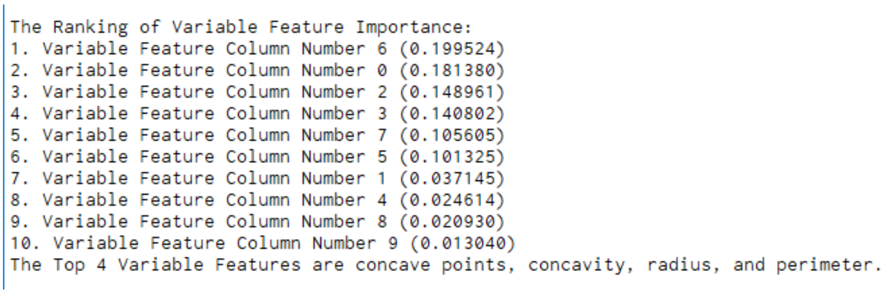
Fig. 10 represent the comparative results of support vector machine. This result ensembles the KNN model, and mine has a better accuracy and we share the same validation results. We may also tell that SVM model has a slower running time, including training and testing both, and lower accuracy rate than KNN.



```
Random Forest
Training time: 0.01 sec
Testing/Prediction time: 0.009 sec
Accuracy: 95.9%
Cross validation result: 94.2% (+/- 5.4%)
             precision    recall  f1-score   support

          0       0.96      0.97      0.97       111
          1       0.95      0.93      0.94        60

avg / total       0.96      0.96      0.96       171

Neural Networks
Training time: 0.009 sec
Testing/Prediction time: 0.003 sec
Accuracy: 73.1%
Cross validation result: 68.7% (+/- 43.1%)
             precision    recall  f1-score   support

          0       0.95      0.62      0.75       111
          1       0.57      0.93      0.71        60

avg / total       0.81      0.73      0.74       171
```

Figure 11 Random Forest and Neural Networks

Fig. 11 shows the other two models, random forest and neural networks. If we put all four models together to analyze the result, we can see that random forest has the highest accuracy and cross validation score, whereas SVM and neural networks have the worse accuracy and validation. The hidden layer of the neural networks doesn't perform well in this task.



```
The Ranking of Variable Feature Importance:
1. Variable Feature Column Number 6 (0.199524)
2. Variable Feature Column Number 0 (0.181380)
3. Variable Feature Column Number 2 (0.148961)
4. Variable Feature Column Number 3 (0.140802)
5. Variable Feature Column Number 7 (0.105605)
6. Variable Feature Column Number 5 (0.101325)
7. Variable Feature Column Number 1 (0.037145)
8. Variable Feature Column Number 4 (0.024614)
9. Variable Feature Column Number 8 (0.020930)
10. Variable Feature Column Number 9 (0.013040)
The Top 4 Variable Features are concave points, concavity, radius, and perimeter.
```

Figure 12 Feature Importance Ranking

With four model compared, we can analyze the feature importance ranking. As Fig. 12 shows, concave points and concavity, radius perimeter weigh more than the other features. The reason that we picked the top four is that they are all above 0.14, and number 5 is further down behind.

4 Conclusion

In this paper, we have investigated the advantages and disadvantages of the common supervised machine learning models—Support Vector Machine, Neural Networks, K-Nearest Neighbors and Random Forest, and implement comparative analysis to select the best-performing model to predict the breast cancer cell, based upon the given *Breast Cancer Wisconsin (Diagnostic) Dataset.* We have also conducted feature selection by implement the variable feature importance algorithm to find out which feature can be best used for the diagnosis of breast cancer cells. As a result, we see the size factor, e.g.

radius and perimeter, and concavity factor, e.g. concave points and concavity, stand out to be the more likely determining factors for the doctors and physicians to diagnose or predict breast cancer.

Certainly, there are limitations and improvement to this research project. Firstly, when I construct the classification model comparison analysis, I used all mean values features, which include all columns of variable features. I could also apply the same four model classifiers for the data set with selected features after the ranking of the variable feature importance was generated. I would be able to more deeply compare the accuracy and cross validate scores of the selected mean features with all mean features. Secondly, there could be more ways to improve the accuracy of the classifiers. I used feature selection, algorithm tuning and multiple algorithm, and bagging and boosting-like ensemble methods may also be used to improve the classifications. I will continue to investigate more machine learning algorithm and implement them on this data set to tune a better outcome of the prediction and analysis of breast cancer cell data [7,8].

5 References

1. Abraham Karplus.: Machine learning algorithms for Cancer Diagnosis. Santa Cruz County Science Fair. (2012)
2. Larissa Westerdijk.: Predicting malignant tumor cells in breasts. Faculty of Science, Business Analytics. Vrije Universiteit Amsterdam. Amsterdam, Netherland. (2018)
3. Ahmad LG., Eshlaghy AT., Poorehbrahimi A., Ebrahimi M., and Razavi AR.: Using Three Machine Learning Techniques for Predicting Breast Cancer Recurrence. Journal of Health & Medical Informatics. Islamic Azad University of Tehran-Iran, Iran (2013)
4. Konstantina Kourou., Themis P. Exarchos., Konstantinos P. Exarchos., Michalis V. Karamouzis., Dimitrios I. Fotiadis.: Machine learning applications in cancer prognosis and prediction. Computational and Structural Biotechnology Journal, Vol. 13, Pages 8-17. Greece. (2015)
5. Shikha Agrawal., Jitendra Agrawal.: Neural Network Techniques for Cancer Prediction: A Survey. Procedia Computer Science. Vol. 60, Pages 769-774. Bhopal, India. (2015)
6. Camille Biscarrat.: Using Machine learning for classification of cancer cells. www.s.u-tokyo.ac.jp. University of California, Berkeley. (2017)
7. Priyanka Gupta, Prof. Shalini L.: Analysis of Machine Learning Techniques for Breast Cancer Prediction. International Journal of Engineering and Computer Science. Vol. 7 Issue 5, Page No. 23891-23895. Tamil Nadu, India. (2015)
8. Wenbin Yue., Zidong Wang., Hongwei Chen., Annette Payne., Xiaohui Liu.: Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis. designs, MDPI. UK and China. (2018)