Breast Cancer Detection Using Data Mining and Machine Learning Techniques

Sayali Govilkar
Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY 14586
srg1988@cs.rit.edu

***Abstract*—Breast Cancer is one of the most frequently occurring cancers in women and it can occur to men as well. If a patient has a tumor, it could be either malignant (cancerous) or benign. If we can identify at an early stage if the tumor is benign or malignant, the chances of survival are higher. In this project, an attempt to classify the tumors accurately using different machine learning algorithms is made. Different machine learning techniques such as Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, and Naive Bayes are implemented in the project for the classification of benign and malignant tumors.**

***Index Terms*—Machine Learning; Data Mining; Classification; Hyperparameter Tuning**

I. Introduction

According to National Breast Cancer Inc., every 2 minutes, a woman is diagnosed with breast cancer in the United States[1]. It is estimated that, in 2020, in the United States, about 41,760 women would die due to this cancer [1]. Breast cancer is a disease in which cells in the breast develop uncontrollably [2]. The cells normally form a cyst that can be viewed on a radiograph or observed as a tumor on the breast [2]. Noncancerous tumors are unusual lumps and do not go outside of the breast called benign tumors [2]. Cancerous tumors are called malignant tumors [2]. A patient needs to go through a lot of expensive tests to find out whether the tumor is malignant or benign. If a machine learning model could identify that for the patient, then it will save a considerable amount of money and efforts needed to go through the medical tests and the patient can find out at an early stage whether he/she has breast cancer or not. In this project, multiple classification algorithms were used and compared at the end to find out which classification model works the best.

The paper is organized into six main sections, and each section is then divided into subsections. The first section is the introduction of the project, where the importance and necessity of the project are explained. The second section is the background, that explains the researches and experiments that have been done in the past for breast cancer detection. The third section is the most important portion of the paper, where the process followed to achieve the goals of this project is explained. The results obtained in section three are then discussed in section four, followed by the conclusion in section five. In the end, the future work of the project is described in section six.

II. Background

To date, many innovative and distinct approaches have been practiced in the past for the Detection of Breast Cancer. Some of these approaches are discussed in this section.

A. A Review of Breast Cancer Detection in Medical Images [3]

[3] focuses on using both commonly used medical imaging methods and some recently proposed methods for breast cancer detection to solve the detection problem in each of the method [3]. The paper concludes a generic approach for the breast cancer detection which includes image processing, region of interest, feature extraction, and classification. The author then applies these techniques on two different types of images, historical and mammograph (x-ray) images respectively[3]. In the end, the different methods used for the detection are compared against each other based on their performances[3].

B. Breast Cancer Detection Through Gabor Filter Based Texture Features Using Thermograms Images [4]

[4] explores thermography technique for breast cancer detection. Thermography is growing as an alternative to mammography technique [4]. Even though mammography is a most accepted method for breast cancer detection, it has some serious side effects such as exposure to harmful radiations [4]. Thermography includes, measuring the temperature patterns occurring in the breast due to increased blood flow near the affected cells [4]. Gabor filters are then used to extract the texture features of the breasts, which helps in identification of normal and abnormal cells [4]. The authors then use SVM classifier for the detection of breast cancer. The accuracy of this model is 84.5% [4].

C. Breast density classification using histogram moments of multiple resolution mammograms [5]

[5] has proposed an automatic method for the classification of breast density [5]. First, the images are pre-processed, where the background is separated from the breast tissues by

using morphology and intensity related techniques [5]. After that, breast cancer density features are extracted by collecting the statistical bits from different resolution histograms [5]. In the end, the support vector machine technique is used for the maammograms classification into different density classes [5].

D. Self-regulated multilayer perceptron neural network for breast cancer classification [6]

[6] proposed an algorithm called a self-regulated multilayer perceptron neural network for breast cancer classification (ML-NN) [6]. The ML-NN model is trained to classify benign and malignant images [6]. The model helped in reducing the efforts needed by doctors to manually analyze the suspicious breast cancer area [6]. The model used a multilayer perceptron neural network technique for the classification. The classifier achieved an accuracy of 90.59% [6].

III. Methodology

A process shown in the Figure 1 is followed for achieving the desired results in this project. The process is explained in this section.

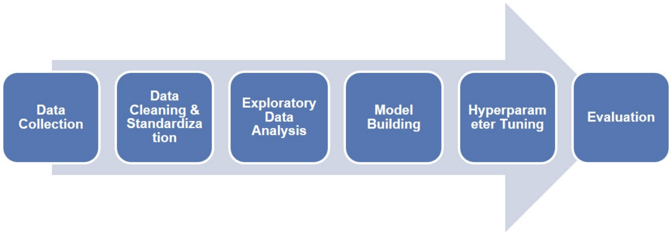
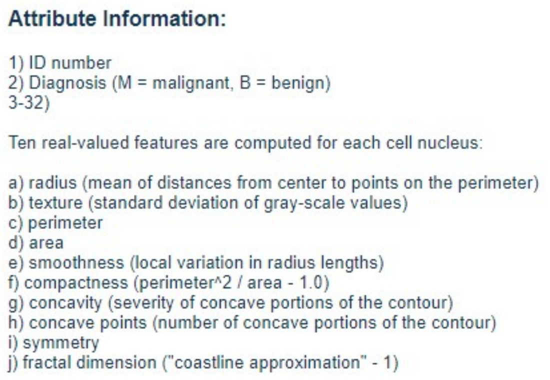


Fig. 1. Breast Cancer Classification Framework

A. Data Collection

The dataset used for the classification is taken from the Breast Cancer Wisconsin (Diagnostic) Data Set [7]. The dataset consists of 32 fields including an 'ID Number' field to identify the individual patient record and a 'Diagnosis' field that has two values, M = Malignant, B = Benign [7]. Apart from the ID number and Diagnosis fields, all the other fields describe different features of cell nuclei [7]. These traits are calculated by taking a digitised picture of a thin needle aspirate of the breast tumor [7]. The attribute information of the first ten attributes is shown in Figure 2. There is a total of 569 records in the dataset, out of which 57 records belong to the benign class, and 212 records belong to the malignant class [7].



Attribute Information:

1) ID number
2) Diagnosis (M = malignant, B = benign)
3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

Fig. 2. Attribute Information [7]

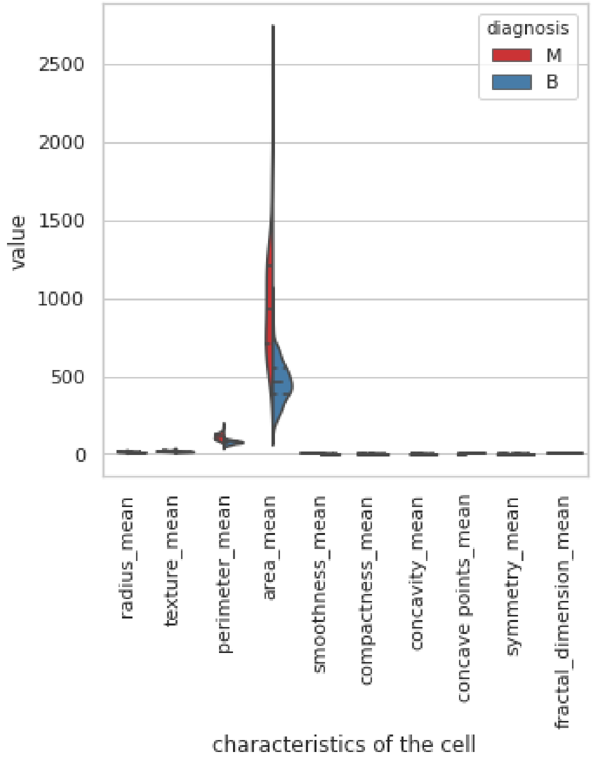


Fig. 3. Violin Plot Before Standardization [8]

B. Data Cleaning and Standardization

After looking at the dataset, it was realized that the last attribute of the dataset had a lot of null values, so that attribute was excluded from the further process. Again, during exploratory data analysis, it was realized that the dataset had huge differences in their ranges. We can see these differences by looking at the violin plot shown below in the in fig 3.

So, standardization of the data was necessary before proceeding with the exploratory data analysis. Standardization helped in bringing the data into a common format, and it was easy to interpret the plots after that. The 'sklearn' library in Python is used to perform the standardization[8]. The library uses the below formula to get the standardized values for each attribute[8].

$$z = (x - u)/s \tag{1}$$

where,
X → sample
U → mean of the training samples

S → Std. deviation of the training samples [8]

C. Exploratory Data Analysis

The main purpose of Exploratory Data Analysis is to identify patterns, spot irregularities, test hypothesis and to examine assumptions in the data using various visual methods [9]. Different visualizations like Violin Plot, Swarm Plot, and Heatmap are considered in this project for investigating the data. All the visualizations are done using the 'Seaborn' library in Python[10].

1) Violin Plots: Violin plots are essentially a blend of histograms and box plots[11]. Violin plots show the distribution of the data like histograms as and median, upper, and lower quartiles in the attributes like box plots[11]. A violin with a large shape at a value shows that there are more data points at that value and a small violin shows fewer data points at that value[11].

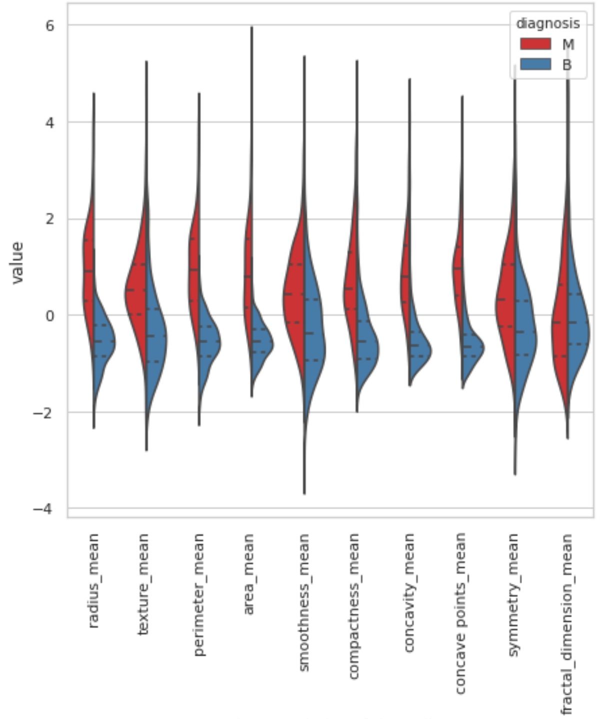


Fig. 4. Violin Plot of the First Ten Attributes of the Dataset[10]

Figure 4 shows the violin plot of the first 10 attributes of the dataset[10]. The red color in the figure shows the malignant records, and blue color shows the benign records for each feature. We can observe in figure 4 that, for some features such as fractle_dimention_mean, the median of the benign and malignant records is not distinguishable, so it might not be a helpful feature for the classification. Whereas, features like compactnes_mean would be beneficial for the classification[10].

2) Swarm Plots: Swarm plots are also known as beeswarm plots, where we can see all the data points[11]. Swarm plots are similar to strip plots, except they do not add jitter to the data points[11].

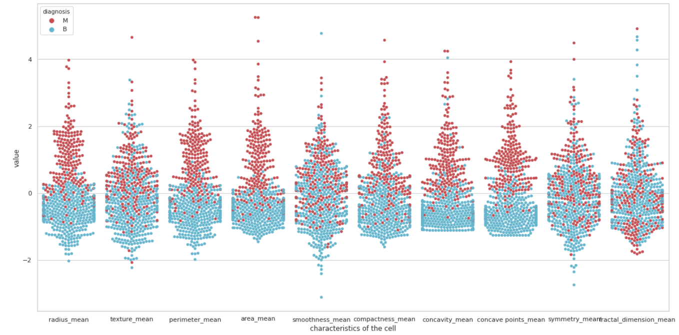


Fig. 5. Swarm Plot of the First Ten Attributes of the Dataset [10]

In Figure 5, the swarm plot for the first ten attributes is shown. Here, the red color represents the malignant tumor records, and blue color represents benign records. From the figure, we can observe that the malignant and benign data points in some of the fields such as smoothness_mean and summary_mean are not well separated. Whereas, data points in radius_mean are well separated, hence, it will be useful for the classification.

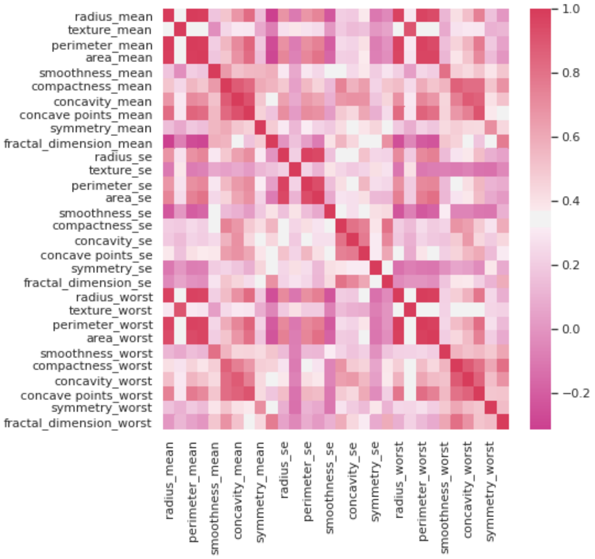


Fig. 6. Heatmap [10]

3) Heatmap: Heatmap is shown in Figure 6. Here the correlation between the two attributes is directly proportional to the darkness of the color. So, from the Figure 6, we can observe that attributes like radius mean, parameter_mean, and area_mean are highly correlated, and attributes like radius_se and concavity_worst are not correlated to each other.

D. Model Building

After exploratory data analysis, various machine learning models such as Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Naïve Bayes, Decision Tree, and Random Forest were explored for the classification of benign and malignant tumors. The Sklearn module in Python is used to perform the classification[8]. For the evaluation of the above-mentioned models, Recall is used instead of Precision. The Recall and Precision formulas are shown below Figures 7 and 8:

$$\textbf{\textit{Recall}} = \frac{TP}{TP + FN}$$

Fig. 7. Recall Formula[12]

$$\textbf{\textit{Precision}} = \frac{TP}{TP + FP}$$

Fig. 8. Precision Formula[12]

Where,
TP (True Positive) → We predicted cancer as positive and it is true.[12]
TN (True Negative) → We predicted cancer as negative and it is true.[12]
FP (False Positive) → We predicted cancer as positive and it is false.[12]
FN (False Negative) → We predicted cancer as negative and it is false.[12]

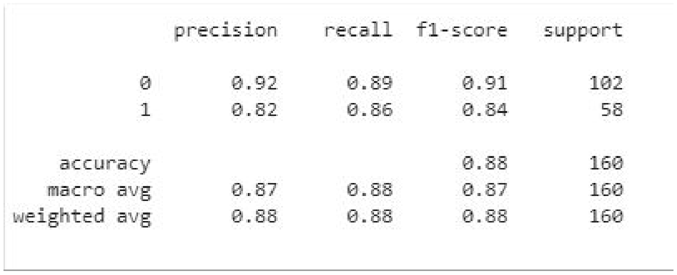
Since we are dealing with breast cancer here, we can afford a few false positives, i.e., we can have patients that do not have breast cancer, but they are identified with breast cancer. But, we do not want to miss out on any patient that has breast cancer. Hence, we must have a high recall to reduce false negatives.

1) Logistic Regression: Logistic Regression is comparable to Linear Regression. The difference is that, in Logistic Regression, the dependent variable is always categorical[13]. Logistic Regression can be used for both binary and multiclass classification[13]. In our case, the classification is binary, as we have only two classes benign and malignant. The conditional probability in the case of binary regression is shown as[13]:

$$P(Y = 1|X) or P(Y = 0|X)[13] \tag{2}$$

Where, y is the dependent variable and x is the independent variable.
The classification report of the model is shown in Figure 9.

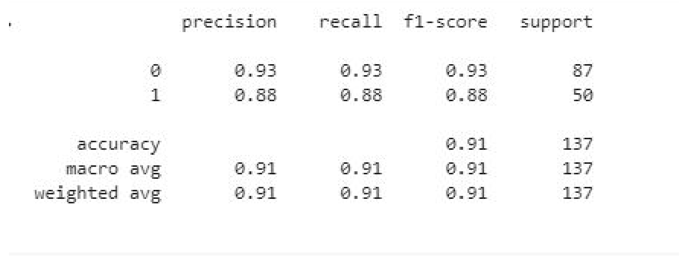
The recall score of the Logistic Regression model is 86.20%



```
              precision    recall  f1-score   support

           0       0.92      0.89      0.91       102
           1       0.82      0.86      0.84        58

    accuracy                           0.88       160
   macro avg       0.87      0.88      0.87       160
weighted avg       0.88      0.88      0.88       160
```

Fig. 9. Classification Report of Logistic Regression [8]

2) K-Nearest Neighbors (KNN): K-Nearest Neighbors is one of the simplest and effective algorithms in machine learning that is used for classification [14]. As the name suggests, in KNN, the data point is classified into a group where its K nearest neighbors belong [14]. Most of the time, Euclidean Distance is utilized to measure the space between the data point and its neighbors [14]. The classification report of the KNN model is shown in Figure. Figure 10.

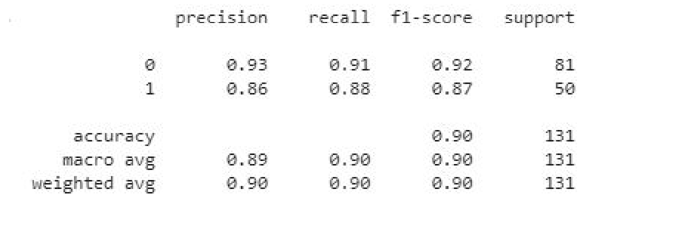


```
              precision    recall  f1-score   support

           0       0.93      0.93      0.93        87
           1       0.88      0.88      0.88        50

    accuracy                           0.91       137
   macro avg       0.91      0.91      0.91       137
weighted avg       0.91      0.91      0.91       137
```

Fig. 10. Classification Report of KNN [8]

The recall score of the KNN model is 88%

3) Support Vector Machine: The Support Vector Machine produces a hyperplane to separate the data points [15]. The hyperplane is also called a decision boundary that has one class at one side of the boundary and another class on the other side of the boundary [15]. The classification report of the SVM model is shown in Figure 11.



```
              precision    recall  f1-score   support

           0       0.93      0.91      0.92        81
           1       0.86      0.88      0.87        50

    accuracy                           0.90       131
   macro avg       0.89      0.90      0.90       131
weighted avg       0.90      0.90      0.90       131
```

Fig. 11. Classification Report of SVM [8]

The recall score of the SVM model is 88%

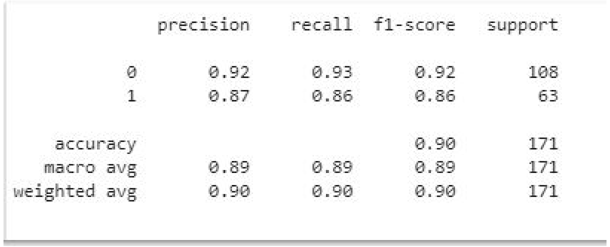
4) Naive Bayes: Naïve Bayes classification model is based on Bayes theorem [16]. Bayes theorem states that, given a probability of an event that has occurred, we can find out the probability of another event [16]. Bayes theorem is shown in Figure 12.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Fig. 12. Bayes Theorem [16]

Where,
P(A|B) → given that B has occurred, identify probability of A happening.
A → hypothesis
P(A) → prior probability
B → evidence [16]

The classification report of the Naïve Bayes model is shown in Figure 13.

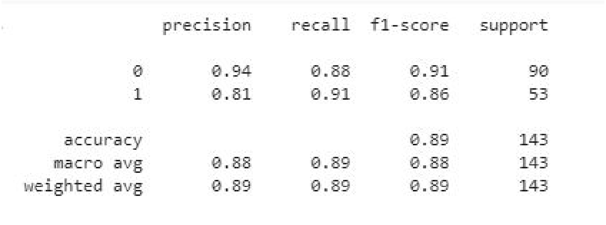


```
              precision    recall  f1-score   support

           0       0.92      0.93      0.92       108
           1       0.87      0.86      0.86        63

    accuracy                           0.90       171
   macro avg       0.89      0.89      0.89       171
weighted avg       0.90      0.90      0.90       171
```

Fig. 13. Classification Report of Naïve Bayes [8]

The recall score of the Naïve Bayes model is 85.71%.

5) Decision Trees: In Decision Trees, a tree-like structure is formed to perform the classification. At each point, the tree divides itself into subtrees based on the given if-then-else decision rules. The leaf nodes of the tree represent the classification.If the tree is deeper, then the results are more accurate. The classification matrix of the Decision Tree model is shown in Figure 15.



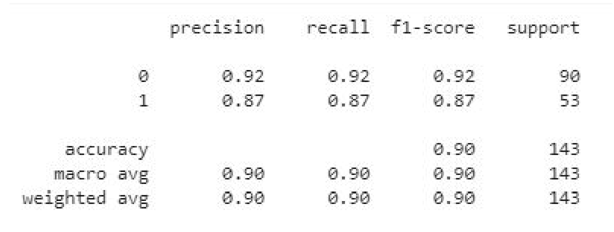
```
              precision    recall  f1-score   support

           0       0.94      0.88      0.91        90
           1       0.81      0.91      0.86        53

    accuracy                           0.89       143
   macro avg       0.88      0.89      0.88       143
weighted avg       0.89      0.89      0.89       143
```

Fig. 14. Classification Report of Decision Tree Model [8]

The recall score of the Decision Tree model is 90.5%.

6) Random Forest: The Random Forest classifier is built using multiple Decision Trees except for a few key points that are different from Decision Tree concepts[17]. In Random Forest, while building the trees, the data points are sampled randomly, and while dividing the nodes, random subsets of the characteristics are considered[17]. The classification matrix of the Random Forest model is shown in Figure 15.

The recall score of the Random Forest model is 86.7%.



```
              precision    recall  f1-score   support

           0       0.92      0.92      0.92        90
           1       0.87      0.87      0.87        53

    accuracy                           0.90       143
   macro avg       0.90      0.90      0.90       143
weighted avg       0.90      0.90      0.90       143
```

Fig. 15. Classification Report of Random Forest Model [8]

E. Hyperparameter Tuning

After building the classification models, it was necessary to improve the performances of the models. The Hyperparameter optimization method was explored to achieve that. Hyperparameter tuning is the process of finding the optimal parameters for the classification models [18]. While building the above-mentioned models, the parameters used were Default, and they might not give us the optimal results. Trying all the available parameters for every model manually would have been a tiresome process, and that is when Hyperparameter Tuning came into the picture. Hyperparatmer Tuning has two major types mentioned below:

1) Grid Search: In the Grid Search approach, all the combinations of the parameters set given are tried, and the best combination is returned [18]. The performance is measured by the cross-validation technique in the Grid Search technique [18].

2) Random Search: In the Random search technique, the search space is randomly sampled instead of an exhaustive search, and a time budget has to be specified to end the algorithm [18].

In this project, the Grid Search technique is used to find the optimal parameters. The classification models were run again with the best parameters to see the difference. The optimal parameters obtained for each of the models are shown below along with their improved performances.

Logistic Regression:
Optimal parameters → {'C': 1, 'multi_class': 'auto', 'penalty': 'l1', 'random_state': 0, 'solver': 'liblinear'}
Recall Score → 87.9% [8]

Support Vector Machine:
Optimal parameters → {'C': 1, 'degree': 1, 'kernel': 'linear'}
Recall Score → 88% [8]

Naïve Bayes:
Optimal parameters → {'var_smoothing': 1e-09}
Recall Score → 85.7% [8]

K-Nearest Neighbors:

Optimal parameters → {'algorithm': 'auto', 'leaf_size': 10, 'n_neighbors':9, 'p': 1, 'weights': 'uniform' }
Recall Score → 90% [8]

Decision Tree:
Optimal parameters → {'criterion': 'gini', 'max_depth': 5, 'random_state': 0, 'splitter': 'random' }
Recall Score → 98.11% [8]

Random Forrest:
Optimal parameters → {'criterion': 'gini', 'max_depth': 4, 'n_estimators': 100, 'random_state': 0 }
Recall Score → 86.79% [8]

IV. Results and Discussion

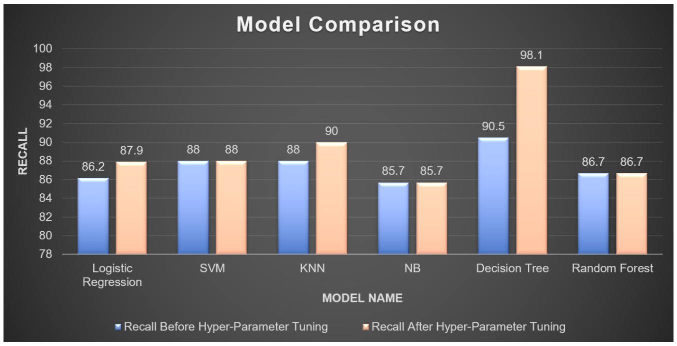


Fig. 16. Model Comparison Before and After Hyperparameter Tuning

In Figure 16, we can see the difference between the performances of the models before and after hyperparameter tuning and it helps us to compare the individual models as well. We can observe in Figure 16, that the Decision Tree model performed better than all the other models. The hyperparameter tuning helped in improving the performance of some of the models by 2 to 8%. For some of the models such as Random Forest or SVM, the hyperparameter tuning did not make any difference because the optimal parameters turned out to be the default parameters that were used before. Medical data is always expensive due to the human power involved in the process of collection of the data [19]. As we are also dealing with the medical data in this project, the records available from the Breast Cancer Wisconsin (Diagnostic) Data Set for the classification are limited. So it is necessary to address the issues related to the data set size in this case because a small dataset tends to increase the accuracy of the model by over-fitting to the data [19]. According to the results obtained by some of the experiments done with the dataset sizes for different classification models, even though the complexity of the learned theories is low for the small datasets, the variance and the error rates are high [20]. The over-fitting problem is resolved in this project by using the cross-validation technique, but some preprocessing techniques could also be utilized in the future to avoid these issues.

V. Conclusion

A patient needs to go through a lot of expensive tests to find out whether the tumor is malignant or benign. The goal of building classification models using machine learning to classify the malignant and benign tumor has been completed. The dataset is taken from the Breast Cancer Wisconsin (Diagnostic) Data Set. The data collection process is followed by data cleaning and standardization to bring the data into a common format. Exploratory Data Analysis helped in identifying the important features in the data that would be useful in the classification between benign and malignant tumors. Various classification models like Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, and Naive Bayes are built in this project followed by hyperparameter tuning to improve their performances and to find out the best performing model. Hyperparameter tuning improved the performance of the classification models by 2 to 8%. Decision Tree classification model performed better than other models, even before and after hyperparameter tuning (98.1%).

VI. Future Work

As mentioned in the discussion section, to address the issues related to the small dataset, some precautionary measures can be taken in future to improve the performance of the models such as removing the features that do not contribute to the prediction, removing the outliers, using a combination of models, and using SMOTE method to adjust the data [21].

In the future, a pipeline could be built in the Google Cloud Platform. The best performing model, which is Decision Trees in our case, can be deployed into GCP to streamline the process. The pipeline would be able to handle input with different formats. Also, it would help in redacting the patient's information, as we would want to maintain the privacy of the patients if the data is coming from a hospital. Once the data is ready, it will go through the classification model and the results could be stored in the BigQuery database, so that the results could be sent back to the hospital or the patients. The database would also help us in handling the results of large datasets.

Acknowledgment

I would like to express my sincere gratitude to my advisor Dr. Carol Romanowski, for continuous guidance for this project. Her support, patience, and encouragement kept me motivated throughout the project. I would also like to acknowledge the Computer Science department, which helped me to learn the essential skills needed for the execution of this project. Lastly, I would like to thank my class instructor Prof. Joe Geigel, for his continuous direction and valuable feedback during the presentations.

References

[1] I. National Breast Cancer Foundation. (2020) Facts about breast cancer in the united states. [Online]. Available: https://www.nationalbreastcancer.org/breast-cancer-facts

[2] A. C. Society. What is breast cancer? [Online]. Available: https://www.cancer.org/cancer/breast-cancer/about/what-is-breast-cancer.html

[3] Y. Lu, J. Li, Y. Su, and A. Liu, "A review of breast cancer detection in medical images," in *2018 IEEE Visual Communications and Image Processing (VCIP)*, 2018, pp. 1–4.

[4] A. A. Khan and A. S. Arora, "Breast cancer detection through gabor filter based texture features using thermograms images," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, 2018, pp. 412–417.

[5] L. Liu, J. Wang, and K. He, "Breast density classification using histogram moments of multiple resolution mammograms," in *2010 3rd International Conference on Biomedical Engineering and Informatics*, vol. 1, 2010, pp. 146–149.

[6] F. F. Ting and K. S. Sim, "Self-regulated multilayer perceptron neural network for breast cancer classification," in *2017 International Conference on Robotics, Automation and Sciences (ICORAS)*, 2017, pp. 1–5.

[7] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[9] P. Patil. (2018) What is exploratory data analysis? [Online]. Available: https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15

[10] M. Waskom. seaborn.violinplot. [Online]. Available: https://seaborn.pydata.org/index.html

[11] A. Gude. Visualizing multiple data distributions. [Online]. Available: https://alexgude.com/blog/distribution-plots/

[12] S. Narkhede. (2018) Understanding confusion matrix. [Online]. Available: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[13] R. Khandelwal. (2020) Quick and easy explanation of logistic regression. [Online]. Available: https://towardsdatascience.com/quick-and-easy-explanation-of-logistics-regression-709df5cc3f1e

[14] M. Schott. (2019) K-nearest neighbors (knn) algorithm for machine learning. [Online]. Available: https://medium.com/capital-one-tech/k-nearest-neighbors-knn-algorithm-for-machine-learning-e883219c8f26

[15] MonkeyLearn. An introduction to support vector machines(svm). [Online]. Available: https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/

[16] geeksforgeeks. Naive bayes classifiers. [Online]. Available: https://www.geeksforgeeks.org/naive-bayes-classifiers/

[17] W. Koehrsen. (2018) An implementation and explanation of the random forest in python. [Online]. Available: https://towardsdatascience.com/an-implementation-and-explanation-of-the-random-forest-in-python-77bf308a9b76

[18] Prabhu. (2018) Understanding hyperparameters and its optimisation techniques. [Online]. Available: https://towardsdatascience.com/understanding-hyperparameters-and-its-optimisation-techniques-f0debba07568

[19] Daniyal, W. Wang, M. Su, S. Lee, C. Hung, and C. Chen, "A guideline to determine the training sample size when applying big data mining methods in clinical decision making," in *2018 IEEE International Conference on Applied System Invention (ICASI)*, 2018, pp. 678–681.

[20] D. Brain and G. Webb, "On the effect of data set size on bias and variance in classification learning," in *Proceedings of the Fourth Australian Knowledge Acquisition Workshop, University of New South Wales*, 1999, pp. 117–128.

[21] R. Alencar. (2019) Dealing with very small datasets. [Online]. Available: https://www.kaggle.com/rafjaa/dealing-with-very-small-datasets