Breast Cancer Tumor Classification Using Machine Learning Models

CMPE 257 Machine Learning, Fall, 2020
Instructor: Dr Mahima Agumbe Suresh,
Computer Engineering Department
San Jose State University, San Jose, CA

CODE LINK: https://colab.research.google.com/drive/1LZiFg-xswo2VYVsSwtwTAUrqxES-DBaJ?usp=sharing

Akash Aggarwal
*a*kash.aggarwal@sjsu.edu

Amitha Shreshta Papetla
*a*mithashreshta.papetla@sjsu.edu

Sowmya Mruthyunjaya
*s*owmya.mruthyunjaya@sjsu.edu

Srijoni Chakraborty
*s*rijoni.chakraborty@sjsu.edu

***Abstract*—Breast cancer is one of the leading types of cancer in women in terms of morbidity and mortality rates as of 2017 worldwide [12]. Encouragingly, it is also among the most curable cancer types if detection and diagnosis happens early in the disease trajectory [5]. This project is aimed at training various machine learning models for classifying whether a tumor is malignant or benign. We have utilised the Wisconsin Diagnostic Breast Cancer data-set from UCI Machine Learning Repository that consists of images of breast mass from various patients [1].**

I. Introduction

With an estimated 276,480 new cases in 2020, that constitutes 15.3% of all cancer cases [7], breast cancer is one of the most prevalent diseases in women [9] globally, with high mortality in women under 60 years of age in high-income countries. Breast cancer is typically characterized by abnormal growth of cells in the breast tissue, and can further be classified based on the type of cells involved and location of growth. Uncontrolled growth of cells leads to the formation of a mass of tissues called tumors [13]. Breast tumors can be of two types- those that are non-cancerous are called benign while the cancerous tumors are termed malignant that can invade other cells. A malignant tumor, when left untreated can metastasize throughout the body and is often fatal. There are different tools like mammography, ultrasound, and physical examination by qualified medical professionals to diagnose breast cancer. Early diagnosis will help reduce the mortality rate. The goal of this project is to leverage various machine learning algorithms to classify the tumor as benign or malignant. After comprehending the nature of the attributes of breast cancer the healthcare community can conduct research and analysis corresponding to these features to reduce the pervasion of breast cancer. In this project, we will be following the steps mentioned below to train various machine learning models to classify a given tumor as benign or malignant.

II. Environment setup

The project was implemented using Python in Google Colaboratory. The different Python libraries used are as follows:

- Numpy for mathematical analysis
- Pandas for the various data structures used
- Scikitlearn for various machine learning algorithms and pre-processing
- Matplotlib and Seaborn for data visualization

III. Methodology

- Data visualization and preprocessing
- Implementing the various machine learning algorithms
- Observing the accuracy and results

IV. Data visualization and Preprocessing

A. Data visualization

A well-annotated data set is an important requirement to leverage machine learning models for the detection of breast cancer. The breast cancer data set used in this project is available at the UCI machine learning repository. It was created by Dr. William H. Wolberg, an Oncologist at the University Of Wisconsin Hospital at Madison, Wisconsin, USA. The features are computed from a digitized image of a fine needle aspirate (FNA) which explains the features of the cell nuclei from a given breast mass image.

There are 10 different real-valued features that are computed from the nuclei, which are given as follows:

- radius (mean of distances from the center to points on the perimeter)
- texture (standard deviation of gray-scale values)
- perimeter
- area
- smoothness (local variation in radius lengths)
- compactness (perimeter**2 / area - 1.0)
- concavity (severity of concave portions of the contour)
- concave points (number of concave portions of the contour)
- symmetry
- fractal dimension (”coastline approximation” - 1)

For the above 10 features, their mean, standard error, and ”worst” or largest (mean of the three largest values) were computed for each image, which resulted in a total of 30 features per row. The data set was found to have no missing attribute values and was with a class distribution of 357 benign, 212 malignant tumors.

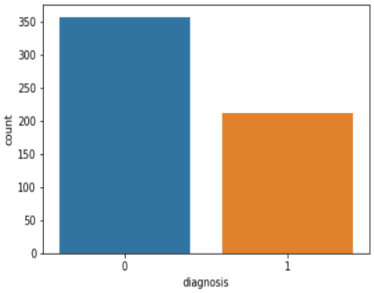


Fig. 1: Bar graph of features vs Count

Visualization of data is an important aspect of data science as it helps to understand data better, identify patterns, detect errors and outliers, help find a correlation between features. In the exploratory data analysis phase, we visualized the tumor characteristics for malignant and benign tumors.

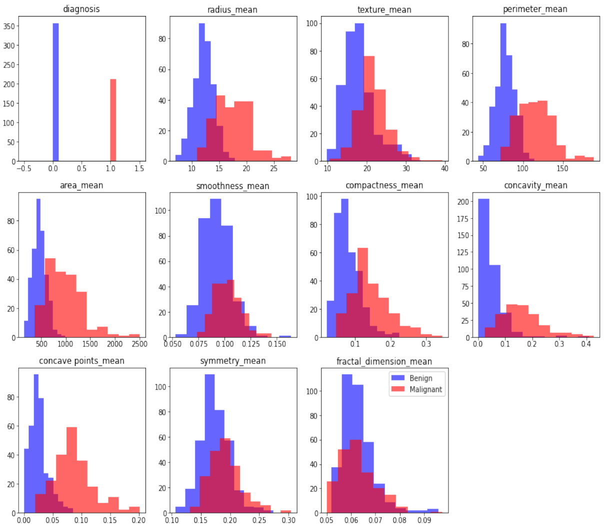


Fig. 2: Diagnosis Count

Mean radius, perimeter, compactness concave points, and concavity most clearly segregate between diagnoses. We also generated a correlation matrix heat map to visualize the correlation between various features. Radius, perimeter, and area are highly correlated. compactness_mean, concavity_mean and concave point_mean are highly correlated. Features like fractal_dimension_mean, fractal_dimension_se, texture_se, smoothness_se, symmetry_se are not correlated with respect to diagnosis. Based on our observation of the correlation matrix, we comprehended the various features that are not significant with respect to the diagnosis, and also performed dimensionality reduction based on that. We dropped the features that had the least correlation.

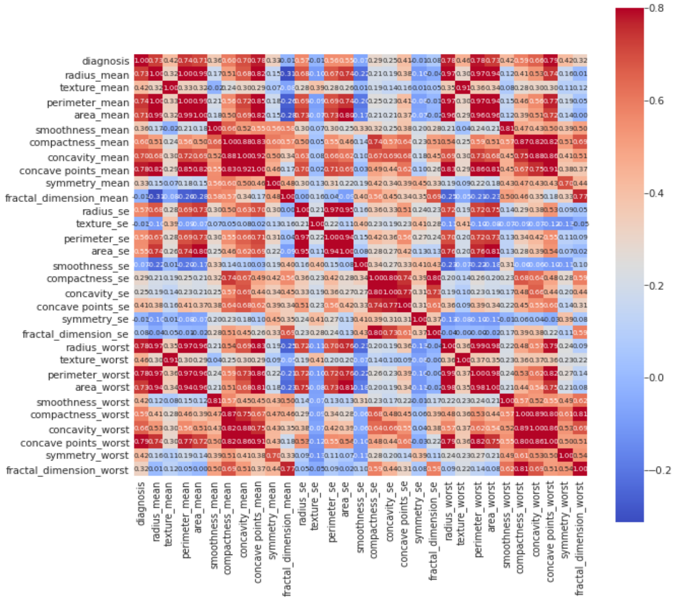


Fig. 3: Heat Map to depict correlation between features

B. Data Preprocessing

After visualizing the tumor characteristics and understanding the various features of the data set and their correlation, we preprocessed the data. A clean data set is very important to help a model perform best. The ratio between benign and malignant samples was 60-40 which was decently balanced. The data preprocessing we performed are as follows - transformed categorical values to 1 and 0 for Malignant and Benign diagnosis, dropped ID and Unnamed Nan columns which did not add any value to the prediction task. There were no missing values eliminating the need for imputation using mean. Dimensionality reduction of the data set was performed through feature selection which brought down the original feature set from 31 to 26 by comprehending the various correlated and non-correlated features. The data set was split into 80% training data and 20% test data sets. This was done by importing the train_test_split method from the model_selection from sklearn libraries. The data set was split into X_test, X_train, Y_test, and Y_train respectively. Grid search cross-validation using K fold was performed on the training set to get the best hyperparameters for the parametric models used in the project. By tuning these parameters, the accuracy of the model was improved and predicted on the test set. The data set features were scaled in order to account for the feature scaling. Feature scaling brings all the values of the feature set under the same scale thus increasing the computation speed of some of the algorithms. This was done by importing the standardscaler method from the preprocessing from the sklearn [11]library. Thus the data set was prepared for machine learning modeling.

V. Machine learning Algorithms

We have implemented the following machine learning algorithms to train the models.

- Logistic regression

- Nearest neighbor
- Random forest classification
- Support vector machine
- Decision tree

A. *Logistic Regression*

Under Supervised Learning, Logistic Regression is widely used algorithm for classification that predicts the probability of a target variable and provides only two outcomes that is either 0 or 1. It is a parametric learning model that is independent of number of training samples and takes set of parameters of fixed size. Logistic Regression is fast in terms of learning from training data, simple to understand and interpret and requires less data to train. For our data set on Breast Cancer Analysis, Logistic Regression is useful because we are training our data set to predict categorical output that is whether the tumor is Malignant or Benign.

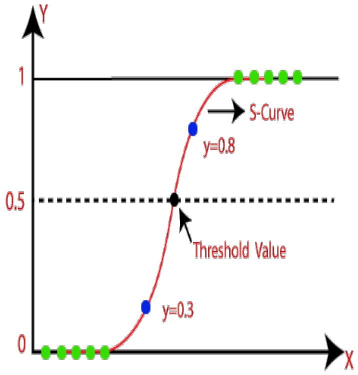


Fig. 4: Logistic Regression. Adapted from [8]

1) Implementation: On our Training set, we did Logistic Regression using grid search from scikit-learn library to find out the best parameters. The hyper parameter of Logistic Regression consists of 'C' that is the penalty for miss classification. The following list of C=[1,10,100] was used. We also performed 5 Fold Cross Validation using 'cross_val_score' from scikit-learn to determine the estimators. The cross validation accuracy obtained for train set was 98.24 and for test set, it was recorded as 97.36.

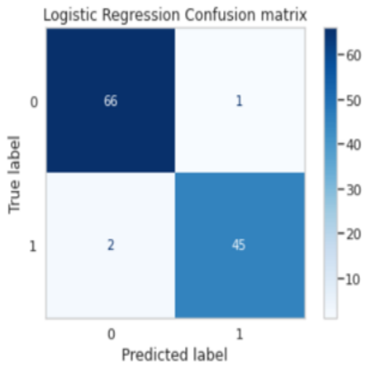
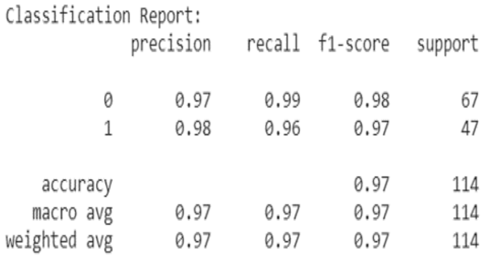


Fig. 5: Confusion Matrix of Logistic Regression.



```
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.99      0.98        67
           1       0.98      0.96      0.97        47

    accuracy                           0.97       114
   macro avg       0.97      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```

Fig. 6: Classification Report of Logistic Regression.

B. *K-Nearest neighbor*

Under Supervised Learning, K-Nearest Neighbor or KNN is a distance based recognition algorithm. This learning model uses 'feature similarity' and predict the values of new data points based on its nearest neighbors. It is a non-parametric model that is very useful for huge data and no prior knowledge of features. As the name suggests, K stands for number of neighbors and it can be any positive integer . The value of K is very important as higher value of K can make boundaries less distinct causing worse classification and smaller value of K can lead to incompetence to classification. The algorithm is as follows:

- Find a value of K. If unknown, use 1,2,4 etc.
- After choosing K, for each point of data, calculate the distance between points using Euclidean, Hamming or Manhattan distance measures.
- Once distance is computed, sort all the values from lowest to highest.
- Finally, test point is assigned a class based on the frequency.

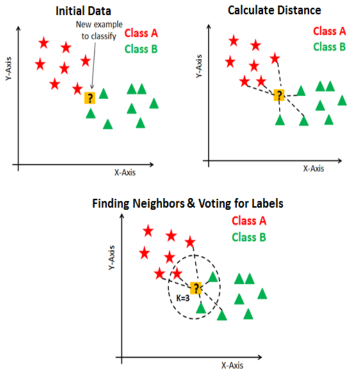


Fig. 7: K-Nearest Neighbors Depiction. Adapted from [2]

1) Implementation: For our Project, we again used 5 fold cross validation using 'cross_val_score' from sci-kit learn to find the best estimators. Initially we ran grid search from neighbors from 1 to 30 to find the best value of K. The other hyper parameter used was 'metric' that defines the type of distance measure. For KNN implementation, we used metric= Euclidean. After running the grid search, we found K=4 that provided the best accuracy. We used K=4 to train our model and obtained an accuracy of 96.02. For the test set using same parameters, we recorded an accuracy 96.49. The closeness in test and train accuracy depicts the effectiveness of this model.

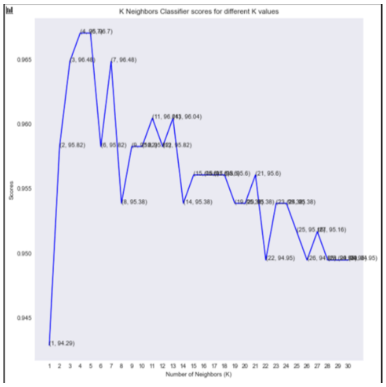


Fig. 8: Number of Features vs Accuracy

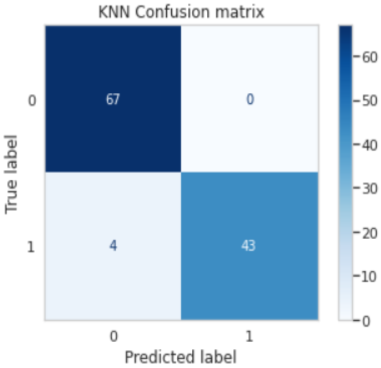
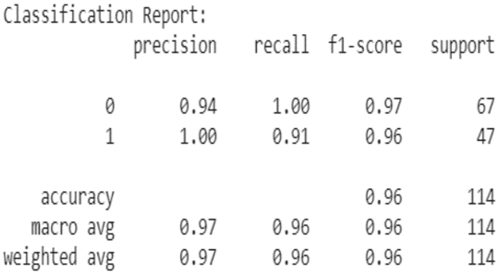


Fig. 9: Confusion Matrix of K-Nearest Neighbor.



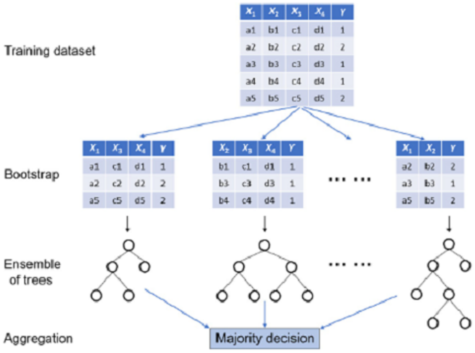
```
Classification Report:
              precision    recall  f1-score   support

           0       0.94      1.00      0.97        67
           1       1.00      0.91      0.96        47

    accuracy                           0.96       114
   macro avg       0.97      0.96      0.96       114
weighted avg       0.97      0.96      0.96       114
```

Fig. 10: Classification Report of K-Nearest Neighbor.

C. Random Forest classification

Random forest is an ensemble machine learning model which is used for both classification and regression. The model is a collection of several decision trees. The algorithm trains decision trees in parallel and aggregates the decisions of individual trees. The final result is chosen by selecting the most voted prediction. Each tree in the random forest is grown as follows:

- If the number of cases in the training set is N, sample N cases is drawn from the original data at random with replacement (bootstrapping). This sample is considered as the training set for growing the tree.
- At each node, m variables are selected at random out of M input variables. The best split on these m variables is used for splitting each node. This value of m is held constant during forest growth.



X_1	X_2	X_3	X_4	Y
a1	b1	c1	d1	1
a2	b2	c2	d2	2
a3	b3	c3	d3	1
a4	b4	c4	d4	1
a5	b5	c5	d5	2

X_1	X_3	X_4	Y
a1	c1	d1	1
a2	c2	d2	2
a5	c5	d5	2

X_2	X_3	X_4	Y
b1	c1	d1	1
b3	c3	d3	1
b4	c4	d4	1

X_1	X_2	Y
a2	b2	2
a3	b3	1
a5	b5	2

Fig. 11: The RF model trains several decision trees in parallel with bootstrapping and aggregation. Adapted from [10]

1) Implementation: The random forest classifier was trained and validated using 5-fold cross validation technique for selecting the best parameters. The helper function 'cross_val_score' from scikit-learn was used for determining the estimators. The main parameters to adjust when using RF classifier from scikit-learn are n_estimators and max_features. n_estimator refers to the number of trees in the forest. We passed 6 values to this parameter (estimators = [10, 50, 100, 200, 500, 1000]). The cross-validation score for n_estimator: 50 was the highest. We obtained an accuracy of 96.5% on the training data and 94.7% on the test data using this estimator for our RF classifier. From our test dataset, 64 out of 67 benign samples were predicted correctly and 44 out of 47 malignant samples were predicted correctly using Random forest model.

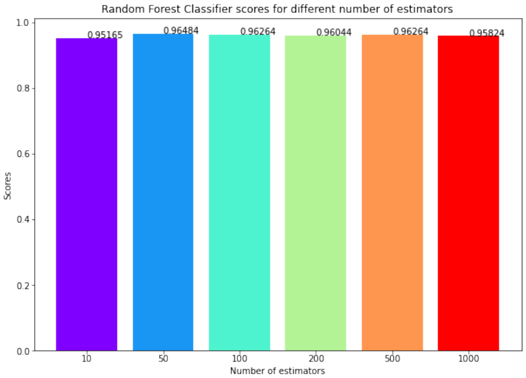


Fig. 12: Cross-validation score vs n_estimators

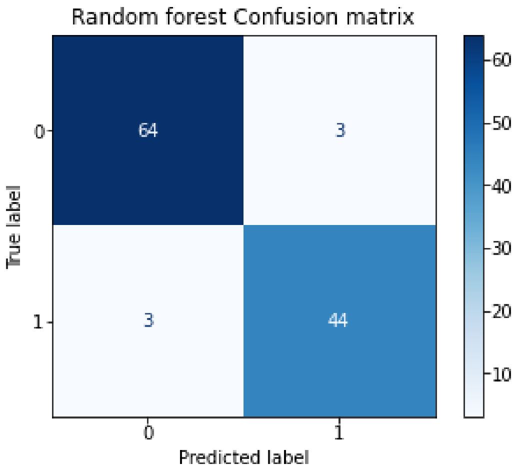
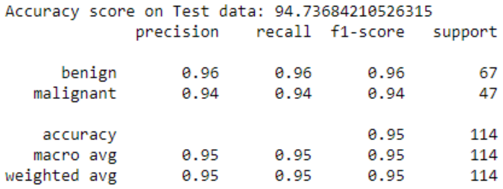


Fig. 13: Confusion matrix of Random Forest Classifier



```
Accuracy score on Test data: 94.73684210526315
              precision    recall  f1-score   support

      benign       0.96      0.96      0.96        67
   malignant       0.94      0.94      0.94        47

    accuracy                           0.95       114
   macro avg       0.95      0.95      0.95       114
weighted avg       0.95      0.95      0.95       114
```

Fig. 14: Classification report of Random Forest Classifier

D. Support vector machine

Support vector machine [3] is widely used in classification problems as it gives a very high accuracy with less computational power. The objective is to find a hyper-plane which is a decision boundary to help classify data points in an N-dimensional space. Data points can be attributed to different classes based on which side of the hyper plane they fall. Data points closer to the hyper plane influence the position and orientation of the hyper plane by maximizing the margin of the classifier. They are known as support vectors. Maximum margin is defined as the the maximum distance between data points of both classes. Higher the the margin, more confidently can future data points can be classified.

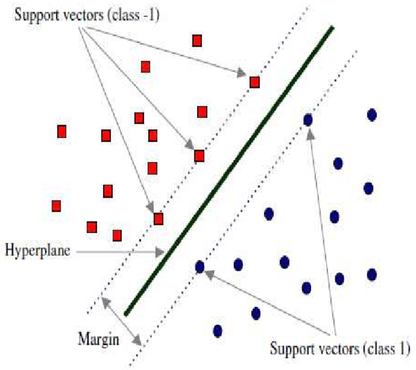


Fig. 15: Support Vector Machine Depiction,Adapted from [6]

1) Implementation: 3 most important hyper parameters of SVM are C which is the penalty for miss classification and gamma , which is the amount of curvature allowed in the classification boundary of kernel and the type of Kernel used. The following list of values for C = [0.001, 0.1, 10, 100], kernel = [’rbf’ ,’linear’, ’poly’], gamma = [1, 0.1, 0.01, 0.001, 0.0001] have been used in the gridsearch to get the best parameters. The best parameters obtained were C’: 10, ’gamma’: 0.01, ’kernel’: ’rbf’. Using this the training set accuracy of 97.58 and and test accuracy of 96.49 was obtained.

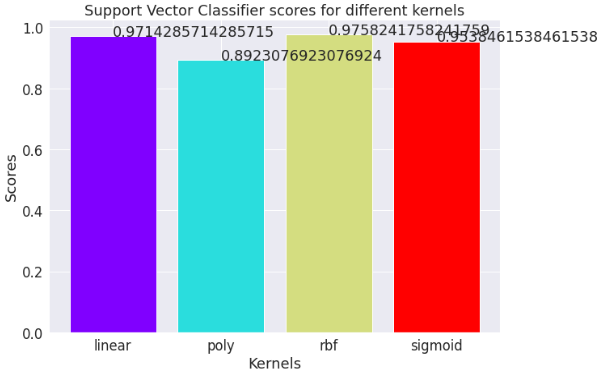
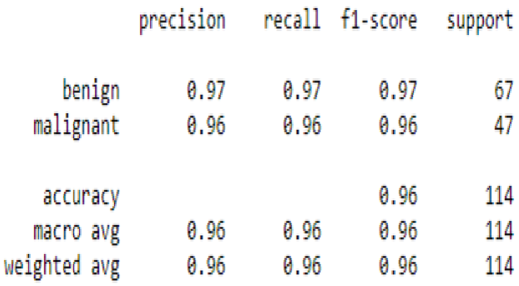


Fig. 16: SVM Classifier score for different kernels



```
              precision    recall  f1-score   support

      benign       0.97      0.97      0.97        67
   malignant       0.96      0.96      0.96        47

    accuracy                           0.96       114
   macro avg       0.96      0.96      0.96       114
weighted avg       0.96      0.96      0.96       114
```

Fig. 17: Classification Report of Support Vector Machine

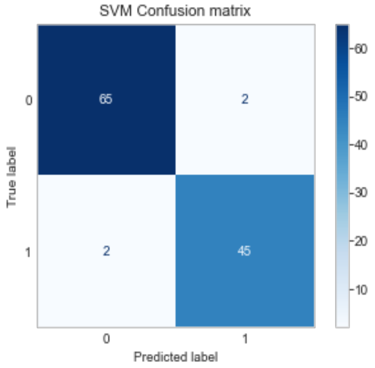


Fig. 18: Confusion Matrix of Support Vector Machine

E. Decision Tree Classifier

Decision Tree classifier [4] is one of the classification algorithms which is a top-down based method. As the name suggests it is used for making decisions from the given data features and create a model to predict the target. A decision tree may be analogous to a flow chart where each internal node is a test on the attribute, a branch represents the result of the test, and the leaf node is the one that holds the class label. The decision tree follows a recursive partitioning.

- It takes attribute selection measures and splits the records.
- Makes that attribute a decision node and breaks the data set into smaller subsets.
- It builds the tree using recursive partitioning for each child until one of the following conditions will match: All the tuples belong to an equivalent attribute value There are no more remaining attributes. There are no more instances.

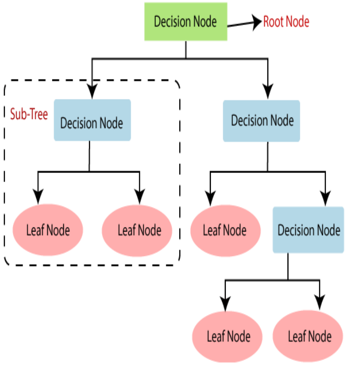


Fig. 19: Decision Tree Representation; Adapted from [2]

1) Implementation: The implementation of the algorithm included a few techniques; such as K-fold cross-validation where we observed the maximum features for which the highest accuracy was obtained.For the maximum feature of 3 we obtained the highest cross-validation training accuracy of 93.84.

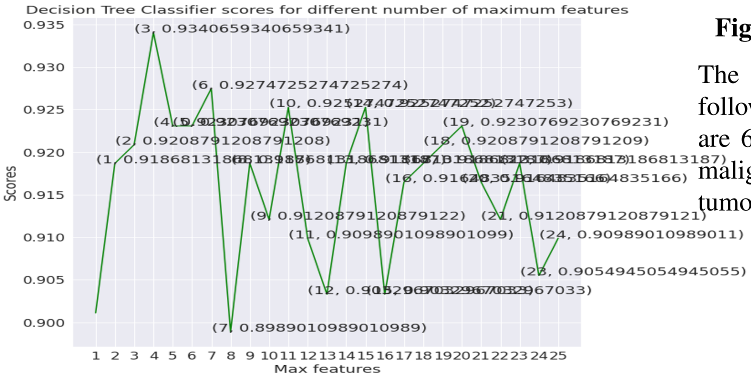


Fig. 20: Max_features vs accuracies

We also performed grid search cross validation in order to perform hyper-parameter optimization.The hyper parameter list given are for criterion: ['Entropy','Gini']. The max features given were all the 25 features. The max_depth:[3, 5, 10, 20, 50, 100]. From the grid search cross-validation we obtained the best parameters and predicted the test data set using those parameters ad observed the training and test accuracies respectively. We also plotted the decision tree utilising the Gini loss function and entropy loss function for our given data set. The best parameters obtained are entropy as the loss criterion and maximum features of 5. Decision Tree using the entropy as loss function:

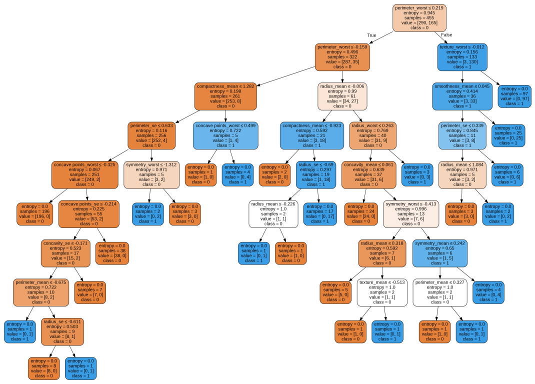
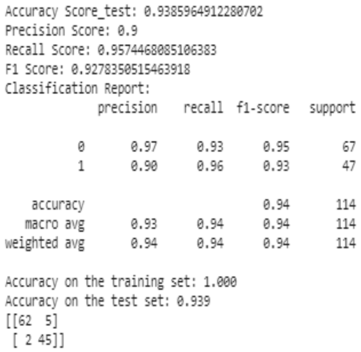


Fig. 21: Decision Tree for our data set

The classification report of the model is given below which gives us the accuracy of the complete model along with the F1 score, precision and recall of the both benign and malignant classifications.



```
Accuracy Score_test: 0.9385964912280702
Precision Score: 0.9
Recall Score: 0.9574468085106383
F1 Score: 0.9278350515463918
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.93      0.95        67
           1       0.90      0.96      0.93        47

    accuracy                           0.94       114
   macro avg       0.93      0.94      0.94       114
weighted avg       0.94      0.94      0.94       114

Accuracy on the training set: 1.000
Accuracy on the test set: 0.939
[[62  5]
 [ 2 45]]
```

Fig. 22: Classification Report for Decision Tree Classifier

The confusion matrix of the decision tree classifier is as follows and from the image below we can conclude that there are 61 benign tumours that were classified correctly; and 45 malignant tumours that were classified correctly of the 114 tumours in the test data set.

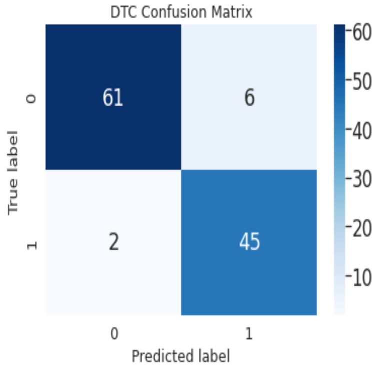


Fig. 23: Confusion Matrix for Decision Tree

VI. Results and Observations

Evaluation metrics like confusion matrix, Accuracy, F1 score, precision, and recall were adopted to gauge the performance of the various machine learning algorithms that were implemented on the data set. It was observed that logistic regression performed exceptionally well with an accuracy of 98.24 on the training set and an accuracy of 97.36 on the test set, as it a pure classification model. This was closely followed by KNN, random forest, support vector machines, and decision tree classifier. The results are tabulated below. We also observed the confusion matrix for each model that gave us a clear idea about the true positives, true negatives, false positives, and false negatives. This helped us conclude the efficiency of the model. The training and test accuracies found for the various algorithms are found and described below in the table. The below table is created from the classification report of each algorithm.

Model	Train Accuracy	Test Accuracy
Logistic Regression	98.24	97.36
KNN	96.02	96.49
RFC	96.48	94.74
SVM	97.58	96.49
DTC	94.06	92.98

TABLE I: Train and Test Accuracy of Machine Learning Models

The Precision, F1 score and recall have the same values for both weighted and macro averages.

Model	F1	Prec	Recall
Logistic Regression	0.97	0.97	0.97
KNN	0.96	0.97	0.96
RFC	0.95	0.95	0.95
SVM	0.96	0.96	0.96
DTC	0.93	0.93	0.93

TABLE II: F1 score, Recall and Precision of Machine Learning Models

Thus, these were the various evaluation metrics that were observed in the project for the breast cancer classification.

VII. Conclusion

A breast cancer diagnosis can be predicted with roughly 95% accuracy.Machine learning algorithms that performed the best in classifying the tumor as benign or malignant based on the tumor features include linear models like logistic regression followed by SVM, and Random forest classifier. A decision support system for predicting breast cancer helps and assists physicians in making optimum, accurate, and timely decisions to improve the quality of life of patients. By predicting breast cancer at an early stage of development, it is possible to reduce the overall cost of treatment and most importantly help reduce the fatality of this disease. This will also aid in the future research analysis that can be conducted in this field.

VIII. Future work

This is our first work on breast cancer prediction using the FNA image samples. This data set consists of image samples collected from only Wisconsin. It would be interesting to find a larger data set to assess how these machine learning models perform in predicting the tumor class. It would also be interesting to find the human accuracy in this prediction task. Finding a cheaper and more efficient set of breast cancer tumor data predictors such as results from patients' routine blood analysis tests would be a better approach. If a model consistently performs well using this blood data, the model can be used to build an Artificial Intelligence tool that may be used by clinicians in identifying breast cancer in patients. This would be a more efficient process and a potential first-step for screening and detecting breast cancer in patients compared to the traditional methods that are expensive and invasive.

IX. Roles and Responsibilities

Task Distribution:

- Akash: LR, KNN, Presentation and Report
- Amitha: EDA, Decision Tree Classifier, Presentation and Report
- Sowmya: EDA, Data preprocessing, Support Vector Machine, Presentation and Report
- Srijoni: Random Forest Classifier, Presentation and Report

References

[1] https://www.kaggle.com/uciml/breast-cancer-wisconsin-data?select= data.csv.

[2] https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm.

[3] https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learni

[4] https://towardsdatascience.com/decision-tree-fundamentals-388f57a60d2a.

[5] Mehmet Fatih Akay. Support vector machines combined with feature selection for breast cancer diagnosis. *Expert systems with applications*, 36(2):3240–3247, 2009.

[6] Hamid Reza Baghaee, Dragan Mlakić, Srete Nikolovski, and Tomislav Dragičević. Support vector machine-based islanding and grid fault detection in active distribution networks. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2019.

[7] National Cancer Institute. Cancer stat facts: Female breast cancer. https://seer.cancer.gov/statfacts/html/breast.html.

[8] Javatpoint. Linear vs logistic regression. https://www.javatpoint.com/linear-regression-vs-logistic-regression-in-machine-learning.

[9] Johns Hopkins Medicine. Diseases more prevalent in women. https://www.hopkinsmedicine.org/womans_path_wellness/for_women/disease_more_prevalent.html.

[10] Siddharth Misra and Hao Li. Noninvasive fracture characterization based on the classification of sonic wave travel times. *Machine Learning for Subsurface Characterization*, page 243, 2019.

[11] 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*, 12:2825–2830, 2011.

[12] Max Roser and Hannah Ritchie. Cancer. *Our World in Data*, 2015. https://ourworldindata.org/cancer.

[13] American Cancer Society. About breast cancer. https://www.cancer.org/cancer/breast-cancer/about/what-is-breast-cancer.html.