



Breast Cancer Prediction Using Machine Learning

Tanmay Nagori : 2021UCD2141

Avi Vaswani : 2021UCD2160

Deepanshu : 2021UCD2138

Pravesh Gupta : 2021UCD2119

ABSTRACT

Breast cancer stands as a significant global health concern, responsible for a substantial number of fatalities. It is the most prevalent form of cancer affecting women worldwide. This research report aims to establish a predictive model for breast cancer utilizing a range of machine learning classification algorithms, including k Nearest Neighbor (kNN), Support Vector Machine (SVM), Logistic Regression (LR), and Random Forest Classifier (RFC). The study endeavors to rigorously evaluate and compare the performance of these diverse classifiers, focusing on metrics such as accuracy, precision, recall, and the F1-Score.

The breast cancer dataset will be partitioned into an 80% portion allocated for the training phase and a 20% segment for the testing phase, in order to ensure robust model evaluation. Notably, the Random Forest Classifier algorithm demonstrates exceptional performance across all assessed parameters.

Keywords: Breast Cancer, Machine Learning, Classification, Accuracy, Precision, Random Forest Classifier, Predictive Model, Model Evaluation, Health Research.

I. INTRODUCTION

Around the world, Breast cancer is the most widely recognized type of cancer alongside lung and bronchus cancer, prostate cancer, colon cancer, and pancreatic cancer among others. Breast cancer might be a prevalent reason for death, and it's the main kind of malignant growth that is boundless among ladies in the around the world. Breast Cancer causes are multifactorial and include family ancestry, weight hormones, radiation treatment, and even reproductive factors. As indicated by the report of the world health organization every year, 2.1 million ladies are recently affected by breast cancer, and furthermore cause the highest number of cancer-

related deaths among ladies. In 2018, it is assessed that 627,000 ladies died from breast cancer - that is roughly 15% of all cancer deaths among ladies. While breast cancer growth rates are higher among ladies in extra developed areas, rates are expanding in about each locale internationally.

Many imaging techniques are developed for early identification and treatment of breast cancer and to scale back the amount of death and lots of aided breast cancer diagnosis methods are wont to increase the symptomatic precision.

Machine Learning algorithms are widely utilized in intelligent human services frameworks,

particularly for breast cancer diagnosis and guess. There are many many machine learning classification and algorithms for prediction of breast cancer outcomes but during this report, we are comparing various sorts of classification algorithms like k Nearest Neighbors, Support Vector Machine, Logistic Regression, and Random forest. And furthermore, assess and compare the performance of the varied classifiers as far as accuracy, precision, recall, f1-Score. The outcomes obtained during this report provide a summary of the condition of modern Machine Learning strategies for breast cancer detection.

II. MACHINE LEARNING ALGORITHMS

Figure 1 shows the bosom breast cancer classification model with machine learning calculations, where the breast cancer dataset is loaded, features need to be extracted and therefore the classification model is often trained and used for prediction of benign and malignant. Benign cases are considered noncancerous, which is non-perilous. Harmful cancer begins with irregular cell development and may quickly spread or attack close-by tissue all together that it is regularly hazardous.

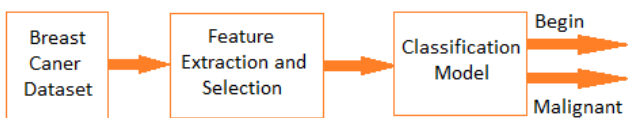


Figure 1. Breast Cancer Classification Model

A. k Nearest Neighbor (kNN)

k Nearest Neighbors algorithm utilizes 'feature similarity' to foresee the estimations of the most recent snippets of data which further methods the new information point will be assigned a value upheld how closely it matches the points inside the training set.

B. Support Vector Machine (SVM)

Support Vector Machine is of the Supervised Machine Learning characterization strategies that are broadly applied inside the field of cancer malignant growth determination and guess. Support Vector Machine works by choosing basic examples from all classes referred to as help vectors and isolating the classes by creating a linear function that partitions them as comprehensively as conceivable utilizing these help vectors. In this way, it is regularly said that planning between an input vector to a high dimensionality space is framed utilizing Support Vector Machine that intends to search out the preeminent reasonable hyperplane that separates the data set into classes. This linear classifier intends to expand the space between the decision hyperplane and along these lines the closest data, which is named the minimal distance, by finding the most appropriate hyperplane.

C. Logistic Regression (LR)

Logistic Regression is a key machine learning classification procedure. It has a place with the gathering of linear classifiers and is fairly practically like polynomial and statistical regression. Logistic regression is quick and similarly simple, and it's helpful for you to decipher the outcomes. In spite of the fact that it's basically a path for binary classification, it additionally can be applied to multi-class issues. This is frequently not the same as statistical regression, as statistical regression contemplates with the forecast of consistent qualities. Logistic regression models the likelihood that reaction falls into a specific classification. A logistic regression model helps us solve, via the Sigmoid function, for situations where the output can take but only two values, 0 or 1.

D. Random Forest Classifier (RFC)

The Random Forest algorithm is a versatile machine learning method that leverages ensemble learning. It combines multiple decision trees, each constructed with randomized subsets of data and features, to enhance predictive accuracy and reduce overfitting. This approach, along with majority voting for classification and averaging for regression, results in robust and accurate predictions. Random Forest also provides insights into feature importance, making it a valuable tool for understanding the factors driving its decisions. It's particularly well-suited for complex, high-dimensional datasets and can be parallelized for efficiency. In summary, Random Forest is a powerful and flexible algorithm that excels in various machine learning tasks. Its ensemble approach, feature randomness, and feature importance analysis contribute to its popularity and effectiveness in predictive modeling.

III. ABOUT DATASET

This Report is based on a dataset that is openly accessible from Sklearn Library [2]. The dataset comprises of a few hundred human cell test records, every one of which contains the estimations of a gathering of cell qualities. The dataset having over thirty attributes. Some of them are:

- i. Radius
- ii. Texture
- iii. Concave points
- iv. Fractioal dimension
- v. Compactness
- vi. Symmetry
- vii. Smoothness
- viii. Perimeter
- ix. Area
- x. Circumference

The ID Number attribute contains the patient identifiers. The qualities of the cell tests from every patient are contained in attribute Clump Thickness to Mitoses. The values are evaluated from 1 to 10, with 1 being the nearest to benign. the class field contains the conclusion, as affirmed by isolated clinical procedures, on whether the tests are benign (value = 2) or malignant (value = 4).

Table . shows the statistics of classes in the dataset.

Class	Instances	% Distribution
Benign	259	45.52
Malignant	310	54.48
Total	569	100

IV. LITERATURE REVIEW

We have use classification experimentation to call attention to that the most straightforward accuracy inside the report was accomplished by the Neural Network calculation, which had, in its best configuration, 97.51% of exactness.

It uses two classification algorithms Random forest and Multi-Layer Perceptron and after analyzing the performance of both algorithm found that Random forest gives the more accurate results.

We also used two different classifiers namely Random forest and K Nearest Neighbors for breast cancer classification on comparing accuracy using cross-validation and RFC achieved that 97.51% accuracy with lowest error rate then KNN 96.19% accuracy.

Also we used three different classifiers namely Random forest, Support Vector Machine, and Decision Tree to classify a Wisconsin breast cancer dataset and

got the best outcome by utilizing a support vector machine with an accuracy score of 96.99%.

e have looked at the performance of supervised learning classifiers by utilizing a Wisconsin breast cancer growth dataset and Random forest, Support Vector Machine, Neural Networks, Decision Tree techniques applied. reliable with the investigation results, the Support Vector Machine gave the chief the exact outcome with a score of 96.84%.

V. OBSERVATION

Confusion matrix is a table that's frequently wont to depict the performance of a classification model on a gathering of test information that truth values are known.

TABLE II
CONFUSION MATRIX

Actual Class	Predicted Class	
	Class=Yes	Class=No
	True Positive (TP)	False Negative (FN)
Class=Yes	82	8
Class=No	3	47

In Table II TP and FP are the observations that are accurately predicted and hence shown in blue shading. we might want to decrease false positives and false negatives all together that they have appeared in red shading.

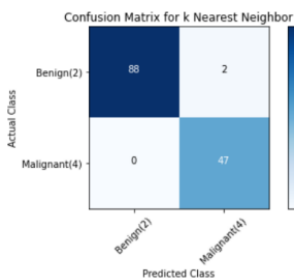


Figure 2(a)

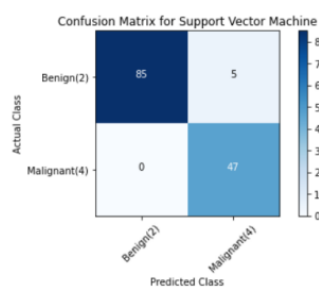


Figure 2(b)

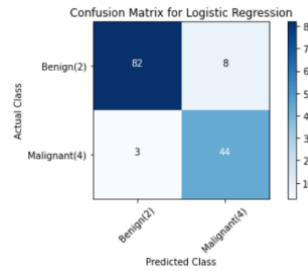


Figure 2(c)

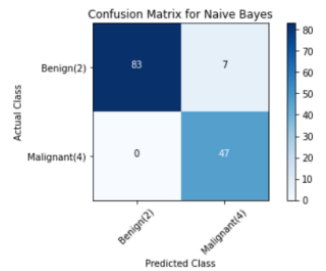


Figure 2(d)

Figure 2. Graphical Representation of Confusion Matrix

A. Accuracy

The classifier exactness is a proportion of how well the classifier can accurately predict cases into their right classification. it's the number of right forecasts separated by the whole number of instances within the data set. it's significant that the accuracy is extremely reliant on the edge picked by the classifier and may, hence, change for different testing sets. Along these lines, it's not the ideal technique to check various classifiers but rather may give a rundown of the classification. Hence, accuracy are often calculated using the following equation:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

Table III shows the accuracy values for all four machine learning algorithms.

TABLE III. ACCURACY VALUES

Algorithms	Accuracy
RFC	0.99
SVM	0.96
LR	0.97
KNN	0.95

B. Recall

Recall, likewise generally referred to as sensitivity, is that the pace of the positive predictions that are effectively predicted as positive. This measure is attractive, particularly within the clinical field because of what level of the observations are accurately analyzed. during this examination, it's progressively imperative to appropriately recognize a threatening neoplasm than it's to inaccurately distinguish a considerate one.

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

Table IV shows the recall values for all four machine learning algorithms.

TABLE IV. RECALL VALUES

Algorithms	Benign	Malignant	Average
RFC	0.98	1.00	0.99
SVM	0.94	1.00	0.97
LR	0.96	1.00	0.98
KNN	0.92	1.00	0.96

C. Precision

Precision, additionally generally referred to as confidence, is that the pace of both true positive and true negative that are distinguished as obvious positive. This shows how well the classifier handles the positive observations however doesn't say a lot of regarding the negative ones.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

Table V shows the precision values for all four machine learning algorithms.

TABLE V. PRECISION VALUES

Algorithms	Benign	Malignant	Average
RFC	1.00	0.96	0.98
SVM	1.00	0.90	0.95
LR	1.00	0.92	0.96
KNN	1.00	0.87	0.94

D. F1-Score

F1-Score is the weighted harmonic mean of Precision and Recall. Subsequently, this score takes both false positive and false negative into thought.

$$\text{F1-Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

Table VI shows the f1-score values for all four machine learning algorithms.

TABLE VI. F1-SCORE VALUES

Algorithms	Benign	Malignant	Average
RFC	0.99	0.98	0.98
SVM	0.97	0.95	0.96
LR	0.98	0.96	0.97
KNN	0.96	0.93	0.95

E. Jaccard Index

Jaccard Index likewise referred to as the Jaccard similarity score is a measurement used in understanding the similarities between test sets. The estimation underscores the similarity between limited test sets and is officially defined because of the fact that the size of the crossing point partitioned by the size of the union of the test sets.

Table VI shows the Jaccard index for all four machine learning algorithms.

TABLE III. JACCARD INDEX VALUES

Algorithms	Jaccard
RFC	0.98
SVM	0.96
LR	0.97
KNN	0.94

VI. RESULTS AND DISCUSSION

Table VII shows all five parameter values for all four machine learning algorithms.

TABLE IVI. PARAMETER VALUES

Algorithms	Accuracy	Precision	Recall	F1-Score	Jaccard
RFC	0.99	0.98	0.99	0.98	0.98
SVM	0.96	0.95	0.97	0.96	0.96
LR	0.97	0.96	0.98	0.97	0.97
KNN	0.95	0.94	0.96	0.95	0.94

In Table-VII k nearest neighbor accomplishes the critical performance as far as accuracy, precision, recall, f1-score and Jaccard index are 0.99, 0.98, 0.99, 0.98, and 0.98 respectively. Logistic Regression accomplishes the second performance as far as accuracy, precision, recall, f1 score and Jaccard index are 0.97, 0.96, 0.98, 0.97 and 0.97 respectively. Support Vector Machine accomplishes the third performance as far as accuracy, precision, recall, f1 score and Jaccard index are 0.96, 0.95, 0.97, 0.96 and 0.96 respectively. Random forest accomplishes the fourth performance as far as accuracy, precision,

recall, f1 score and Jaccard index are 0.97, 0.96, 0.98, 0.97 and 0.97 respectively.

VII. CONCLUSION

In this report, we have compared the classification parameters as far as four Machine Learning algorithms, in particular, k Nearest Neighbor, Support Vector Machine, Logistic Regression, and Random forest available on UCI Machine Learning Repository Wisconsin breast cancer dataset. The target of this comparative analysis was to search out the foremost accurate machine learning algorithm which will act as a tool for the diagnosis of breast cancer which is consistent with the prediction results, Random Forest Classification has the very best accuracy for the given dataset. This shows this machine learning algorithm is regularly better for the prediction of breast cancer as compared with others.

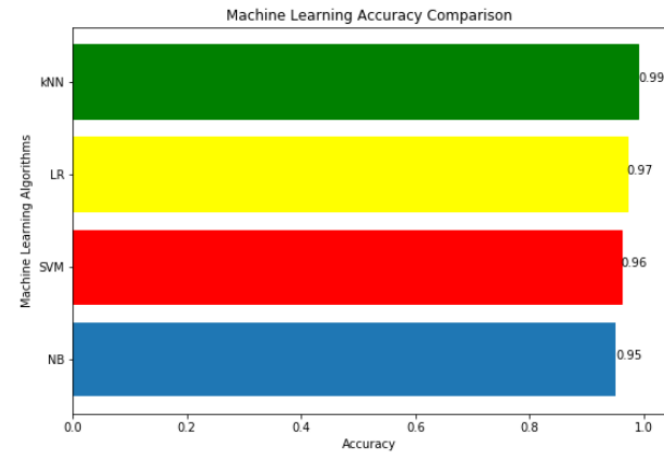


Figure 2. Graphical Representation of Accuracy Comparison

Figure 2 shows that Random Forest Classifier gives the more accurate algorithm having classified the samples with 98.51% accuracy in conventional validation. Logistic Regression, Support Vector Machine and KNN comes second, third, and fourth respectively in classification accuracy.

This comparative investigation shows that the classification accuracy, precision, recall, f1-score and Jaccard index of k Nearest Neighbor is above Support Vector Machine, Logistic Regression and Random forest classification algorithm within the predictive breast cancer data from the Machine Learning Repository Wisconsin breast cancer dataset. we have seen that RFC gives critical performance classification algorithm as far as accuracy, precision and recall.

The limitation of this analysis is that the size of the information used. the amount of samples used for training and testing is low. The analysis of information with respect to the clinical settings should be administered with a bigger dataset.

VIII. ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Dr. MPS Bhatia, Professor and Head of the Department of Computer Science and Engineering at NSUT Engineering College, Dwarka, Delhi, for his exceptional guidance during the completion of this project assignment. The support and mentorship provided by Dr. Bhatia have been invaluable, and this project has significantly enriched our academic journey. We would also like to acknowledge the contributions of our fellow students and the department for creating a conducive learning environment. This experience has been pivotal in strengthening our knowledge and skills in the field of computer science and engineering.

CODE :

App.py

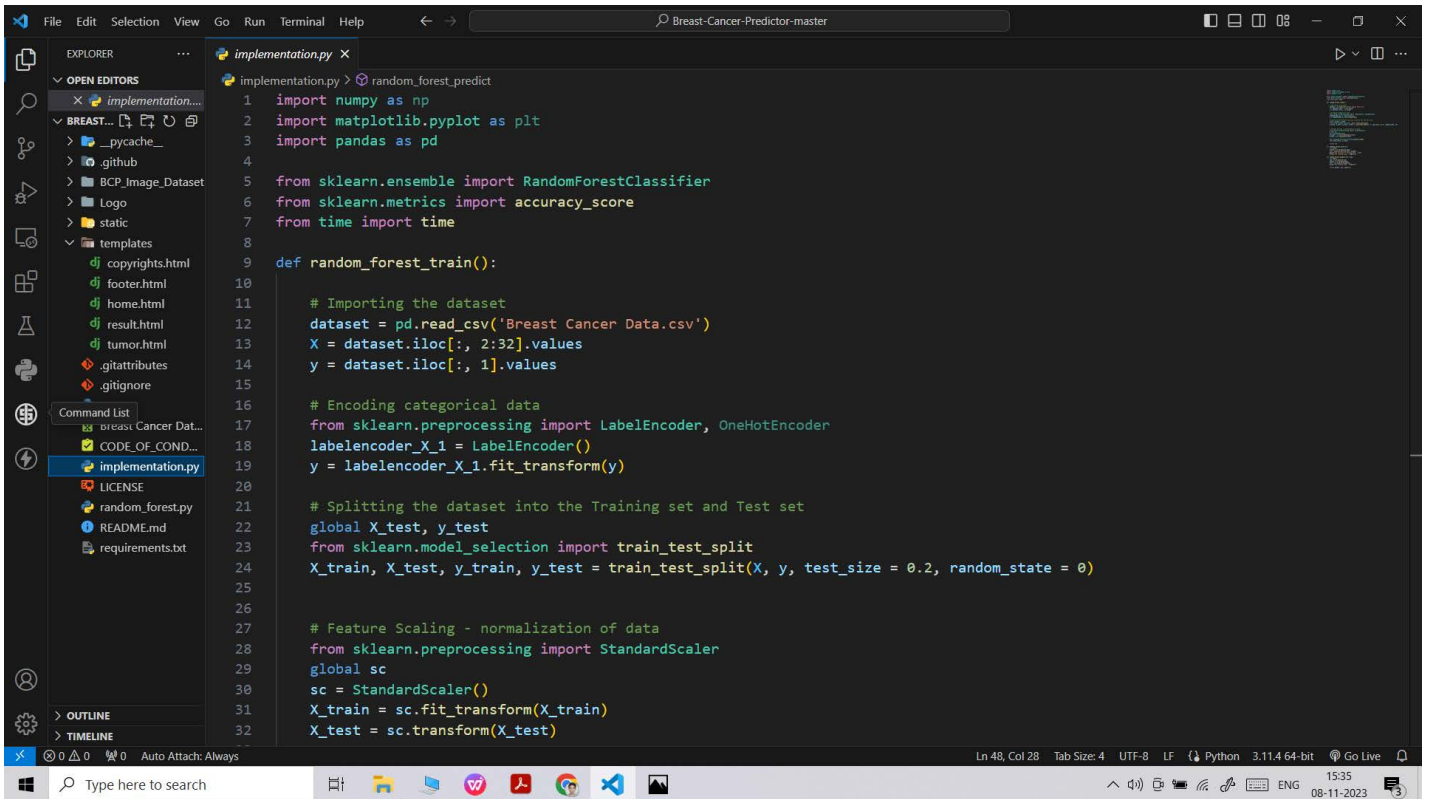
```
File Edit Selection View Go Run Terminal Help
Breast-Cancer-Predictor-master

app.py 1 x
app.py > ...
1 from flask import Flask, render_template, request
2 from implementation import random_forest_test, random_forest_train, random_forest_predict
3 from sklearn.preprocessing import StandardScaler
4 import numpy as np
5 import matplotlib.pyplot as plt
6 import pandas as pd
7 from random_forest import accuracy
8 from sklearn.metrics import accuracy_score
9 from time import time
10
11
12 app = Flask(__name__)
13 app.url_map.strict_slashes = False
14
15 @app.route('/')
16 def index():
17     return render_template('home.html')
18
19 @app.route('/predict', methods=['POST'])
20 def login_user():
21
22     data_points = list()
23     data = []
24     string = 'value'
25     for i in range(1,31):
26         data.append(float(request.form['value'+str(i)]))
27
28     for i in range(30):
29         data_points.append(data[i])
30
31     print(data_points)
32
```

```
File Edit Selection View Go Run Terminal Help
Breast-Cancer-Predictor-master

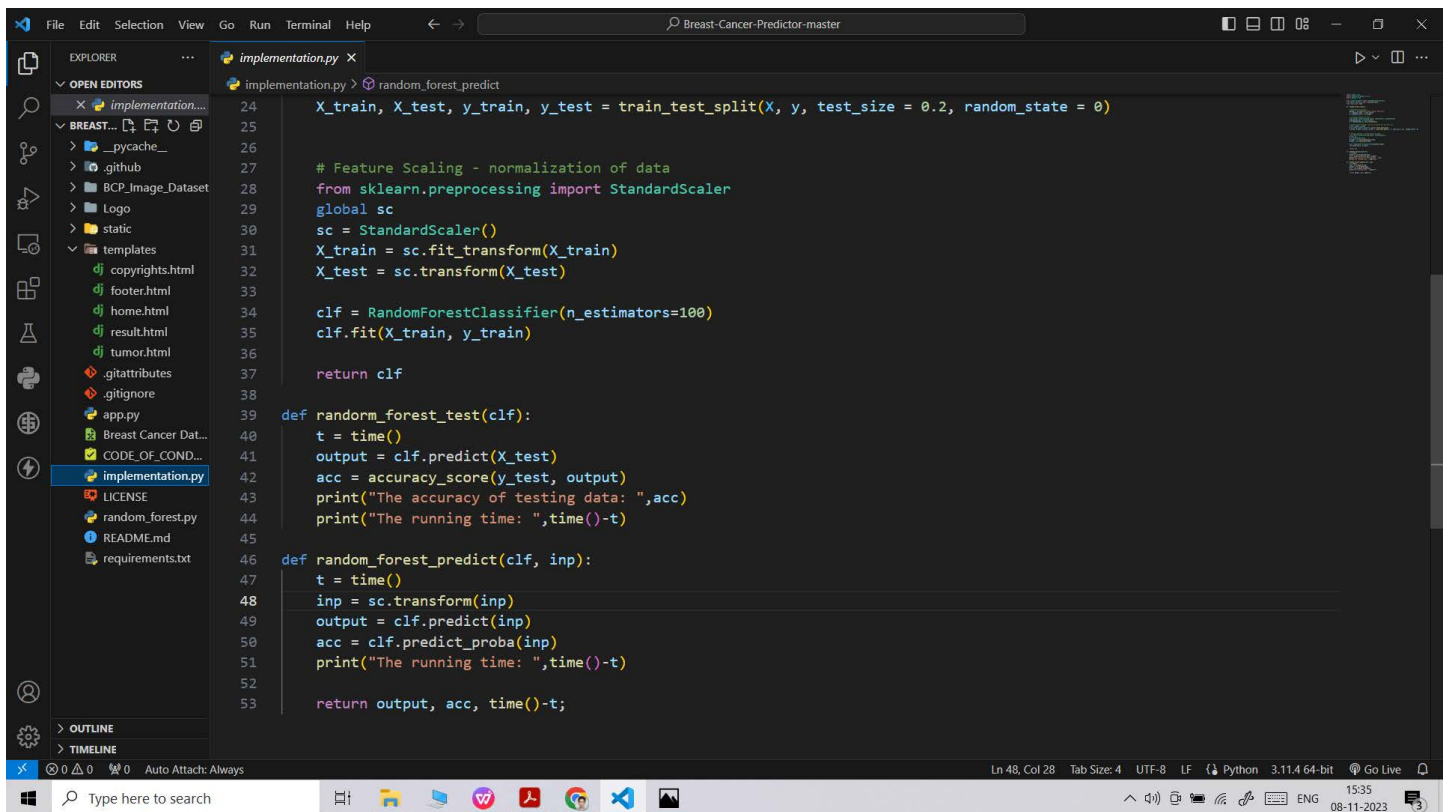
app.py 1 x
app.py > login_user
27
28     for i in range(30):
29         data_points.append(data[i])
30
31     print(data_points)
32
33     data_np = np.asarray(data, dtype = float)
34     data_np = data_np.reshape(1,-1)
35     out, acc, t = random_forest_predict(clf, data_np)
36
37     if(out==1):
38         output = 'Malignant'
39     else:
40         output = 'Benign'
41
42     acc_x = acc[0][0]
43     acc_y = acc[0][1]
44     if(acc_x>acc_y):
45         acc1 = acc_x
46     else:
47         acc1=acc_y
48     return render_template('result.html', output=output, accuracy=accuracy, time=t)
49
50
51
52 if __name__=='__main__':
53     global clf
54     clf = random_forest_train()
55     randomm_forest_test(clf)
56     #print("Done")
57     app.run(debug=True)
58
```


Implementation.py:



This screenshot shows the first 32 lines of the `implementation.py` file in a VS Code editor. The Explorer sidebar on the left shows the project structure, including files like `breast_Cancer_Data.csv`, `CODE_OF_CONDUCT.md`, `implementation.py`, `LICENSE`, `random_forest.py`, `README.md`, and `requirements.txt`. The main editor window displays the following code:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 from sklearn.ensemble import RandomForestClassifier
6 from sklearn.metrics import accuracy_score
7 from time import time
8
9 def random_forest_train():
10
11     # Importing the dataset
12     dataset = pd.read_csv('Breast Cancer Data.csv')
13     X = dataset.iloc[:, 2:32].values
14     y = dataset.iloc[:, 1].values
15
16     # Encoding categorical data
17     from sklearn.preprocessing import LabelEncoder, OneHotEncoder
18     labelencoder_X_1 = LabelEncoder()
19     y = labelencoder_X_1.fit_transform(y)
20
21     # Splitting the dataset into the Training set and Test set
22     global X_test, y_test
23     from sklearn.model_selection import train_test_split
24     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)
25
26
27     # Feature Scaling - normalization of data
28     from sklearn.preprocessing import StandardScaler
29     global sc
30     sc = StandardScaler()
31     X_train = sc.fit_transform(X_train)
32     X_test = sc.transform(X_test)
```



This screenshot shows the continuation of the `implementation.py` file, starting from line 24. The code defines functions for testing and predicting using the trained model.

```
24 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)
25
26
27 # Feature Scaling - normalization of data
28 from sklearn.preprocessing import StandardScaler
29 global sc
30 sc = StandardScaler()
31 X_train = sc.fit_transform(X_train)
32 X_test = sc.transform(X_test)
33
34 clf = RandomForestClassifier(n_estimators=100)
35 clf.fit(X_train, y_train)
36
37 return clf
38
39 def random_forest_test(clf):
40     t = time()
41     output = clf.predict(X_test)
42     acc = accuracy_score(y_test, output)
43     print("The accuracy of testing data: ",acc)
44     print("The running time: ",time()-t)
45
46 def random_forest_predict(clf, inp):
47     t = time()
48     inp = sc.transform(inp)
49     output = clf.predict(inp)
50     acc = clf.predict_proba(inp)
51     print("The running time: ",time()-t)
52
53     return output, acc, time()-t;
```