BREAST CANCER DETECTION

(Course Name: Introduction to Python Programming Lab)

(Course Code: 20CS3352)

A Python Project Report on Breast Cancer Detection

Submitted by

KUNCHAM MEGHANA (22501A0593)

MARIDU NAGA VEERA VENKATA

SAI BABU (22501A05A6)

JUPUDI KRISHNA PRASANTH (22501A0569)

MUTTHA VENKATESH (22501A05B8)

NALLEBOINA RAMYASRI (23505A0509)

II B. Tech I Sem

in

Computer Science and Engineering

Prasad V Potluri Siddhartha Institute of Technology

Accredited with A+ grade by NAAC, NBA Accredited,

and Autonomous ISO 9001:2015 Certified Institute

Permanently Affiliated to JNTUK-Kakinada and approved by AICTE
Kanuru, Vijayawada-520 007

Prasad V Potluri Siddhartha Institute of Technology

Accredited with A+ grade by NAAC, NBA Accredited,
and Autonomous ISO 9001:2015 Certified Institute
Permanently Affiliated to JNTUK-Kakinada and approved by AICTE
Kanuru, Vijayawada-520 007

CERTIFICATE

This is to certify that the python project report titled “Breast Cancer Detection” of Miss. Kuncham Meghana(22501A0593), Mr. Maridu Naga Veera Venkata Sai Babu(22501A05A6),Mr. Jupudi Krishna Prasanth(22501A0569),Mr. Muttha Venkatesh(22501A05B8), Miss. Nalleboina Ramyasri(23505A0509).

Signature of the Guide **Signature of the H.O.D**

TABLE OF CONTENTS

BREAST CANCER DETECTION

1.INTRODUCTION

1.1 Background

Breast cancer is a disease in which cells in the breast grow out of control. There are different kinds of breast cancer. The kind of breast cancer depends on which cells in the breast turn into cancer. Most breast cancers begin in the ducts or lobules.

Breast cancer is considered one of the most common cancers in women caused by various clinical, lifestyle, social, and economic factors. Machine learning has the potential to predict breast cancer based on features hidden in data.

A major challenge in predicting breast cancer is the creation of a model for addressing all known risk factors. Mammography-based breast cancer screening is performed at regular intervals - usually annually or every two years - for all women. Current prediction models might only focus on the analysis of mammographic images or demographic risk factors without other critical factors. In addition, these models, which are accurate enough for identifying high-risk women, could result in multiple screening and invasive sampling with magnetic resonance imaging (MRI) and ultrasound. The financial and psychological burden could be experienced by patients.

1.2 Motivation

Signs and symptoms of breast cancer may include: A breast lump or thickening that feels different from the surrounding tissue. Change in the size, shape or appearance of a breast. Changes to the skin over the breast, such as dimpling.
Early diagnosis of breast cancer increases the chance of recovery and life expectancy. Screening is the primary tool for early diagnosis and timely treatment of breast cancer in early stages.

For successful strategy to increase motivation for mass screening for detection of early breast cancer, three elements are needed: 1) women at risk must be taught the necessity of the examination; 2) they must be made aware of the existence of the facility, which must be made easily accessible.

In the realm of breast cancer detection, meticulous data preprocessing is crucial to ensure the reliability and effectiveness of predictive models. This process involves cleaning and organizing the dataset, handling missing values, and normalizing features to a consistent scale. Feature selection techniques may be employed to identify relevant attributes, reducing dimensionality and optimizing computational efficiency.

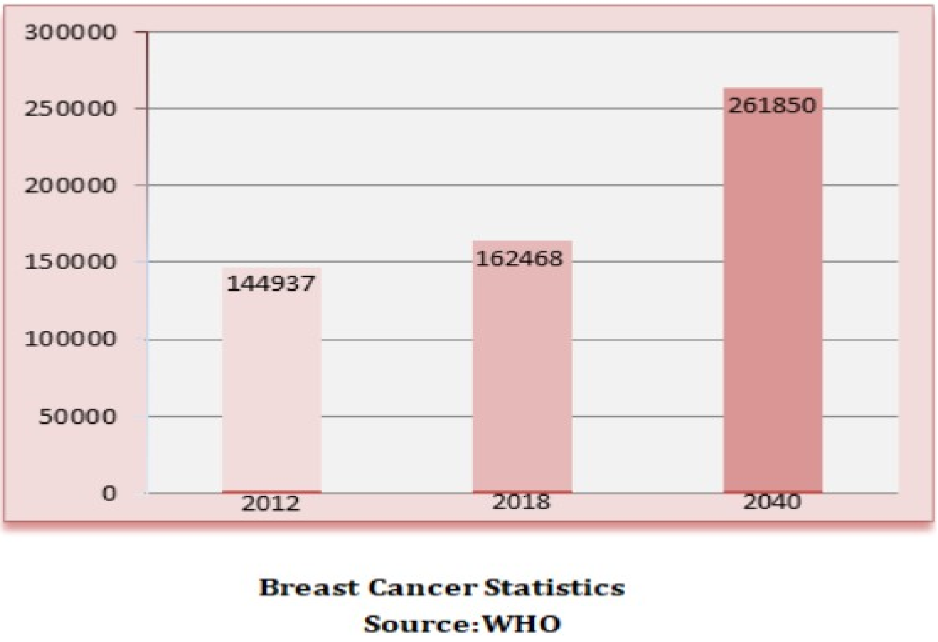


Fig 1: Death of persons due to breast cancer in India over the years

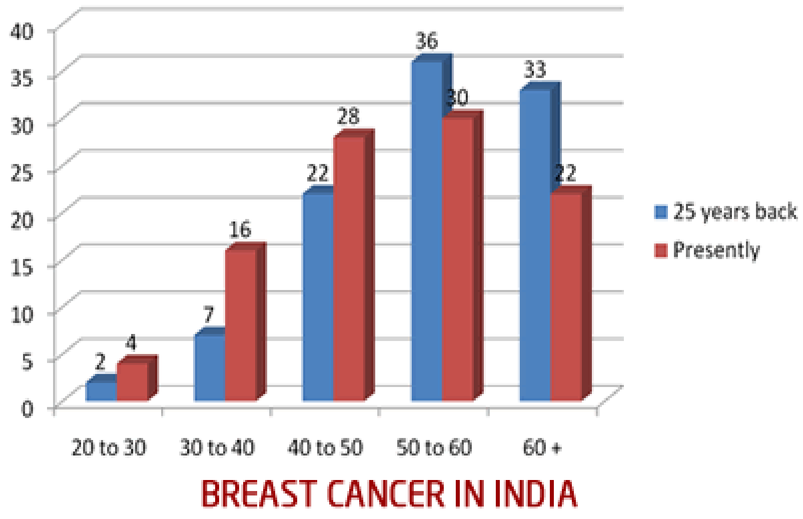


Fig 2: Changing trends in age wise distribution of Breast cancer in India

PROBLEM STATEMENT

Machine learning models offer the capability to detect the breast cancer . "The aim of this project is to develop a breast cancer detection model using logistic regression. By analyzing various features such as tumor size, shape, and texture, the model aims to accurately classify breast tumors as either malignant or benign. The objective is to be create a reliable tool that can assist healthcare professionals in the early detection and diagnosis of breast cancer, ultimately improving patient outcomes". Moreover, machine learning models can extend their analysis to include additional variables such as genetic factors and lifestyle choices.

2.LITERATURE REVIEW

Several researchers have delved into the realm of breast cancer prediction using machine learning methods, showcasing diverse approaches and datasets. Here are main key insights from notable studies:

In a study by Feld et al. to predict breast cancer, the modeling was performed on – 738 records, including demographic, genetic, and abnormal mammographic data, and was reported AUC was 0.75.

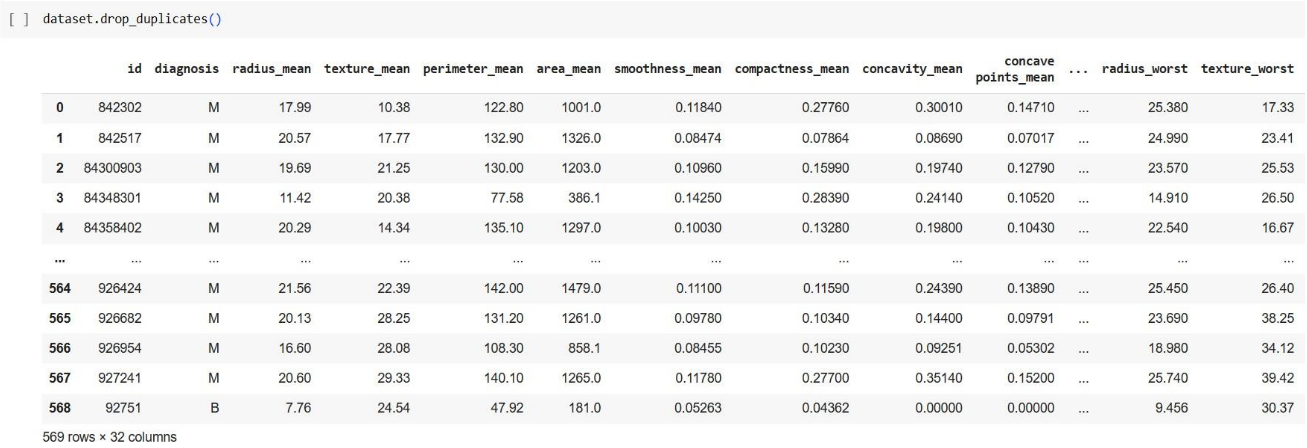
In a literature review, Leonard Fass (2008) and Safarpour Lima and colleagues (2019) found that cancer care is dependent on imaging through screening. Breast cancer can be detected early using imaging tools . The sensitivity and specificity of various techniques, however, vary .

Breast lesions are classified as either malignant or benign based on cancer stage, which is identified by breast imaging reporting and data system scores (Al-Antari et al. 2018b;Goldhirsch et al. 2006). Surgery is often the first recommendation for treating breast cancer to improve the survival rate (Fadzil et al. 2021;Nemade et al. 2022). Breast cancer screening is typically performed using X-ray mammography, MRI, and ultrasound images (Goldhirsch et al. 2006;Fadzil et al. 2021)

3. METHODOLOGY

3.1 Data Collection

The dataset for breast cancer prediction is from Kaggle [10]. This particular dataset has 569 - rows and 32columns. The columns are 'id', 'diagnosis', 'radius_mean', 'texture_mean','perimeter_ mean','area_mean','smoothness_mean','compactness_mean','concavity_mean','concavepoints_mean' 'symmetry_mean', 'fractal_dimension_mean'as the mainattributes.



```
[ ]  dataset.drop_duplicates()
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	radius_worst	texture_worst
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	...	25.380	17.33
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	...	24.990	23.41
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	...	23.570	25.53
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	...	14.910	26.50
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	...	22.540	16.67
...	...	...	...	...	...	...	...	...	...	...	...	...	...
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	25.450	26.40
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	...	23.690	38.25
566	926954	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	...	18.980	34.12
567	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	...	25.740	39.42
568	92751	B	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	...	9.456	30.37

569 rows × 32 columns

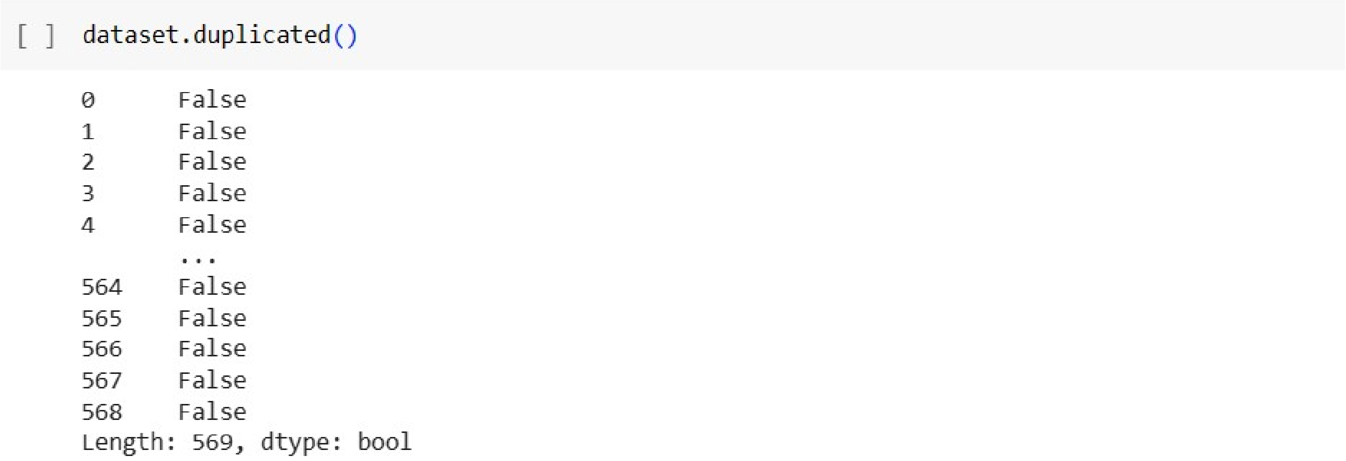
Fig.3 DROP DUPLICATES FROM DATASET

3.2 Data Pre-Processing

Data Preprocessing is required before model building to remove the unwanted noise and outliers from the dataset, resulting in a deviation from proper training. Anything that interrupts the model from performing with less efficiency is taken care of in this stage. After collecting the appropriate dataset, the next step lies in cleaning the data and making sure that it is ready for model building. The dataset taken has 12 attributes, as mentioned in Table I. Firstly, the column 'id' is dropped because its existence does not make much difference in model building. Then the dataset is checked for null values and filled if any found

The dataset chosen for the task of breast cancer prediction is highly imbalanced. The entire dataset has 569 rows, of which 32 rows are suggesting the occurrence of a breast cancer and show that there is possibility. The graphical representation of the imbalance is in Fig. 2.Training a machine-level model with such data might give accuracy, but other accuracy metrics like precision and recall are shallow. If such imbalanced data is not handled, the results are not accurate, and the prediction is inefficient.

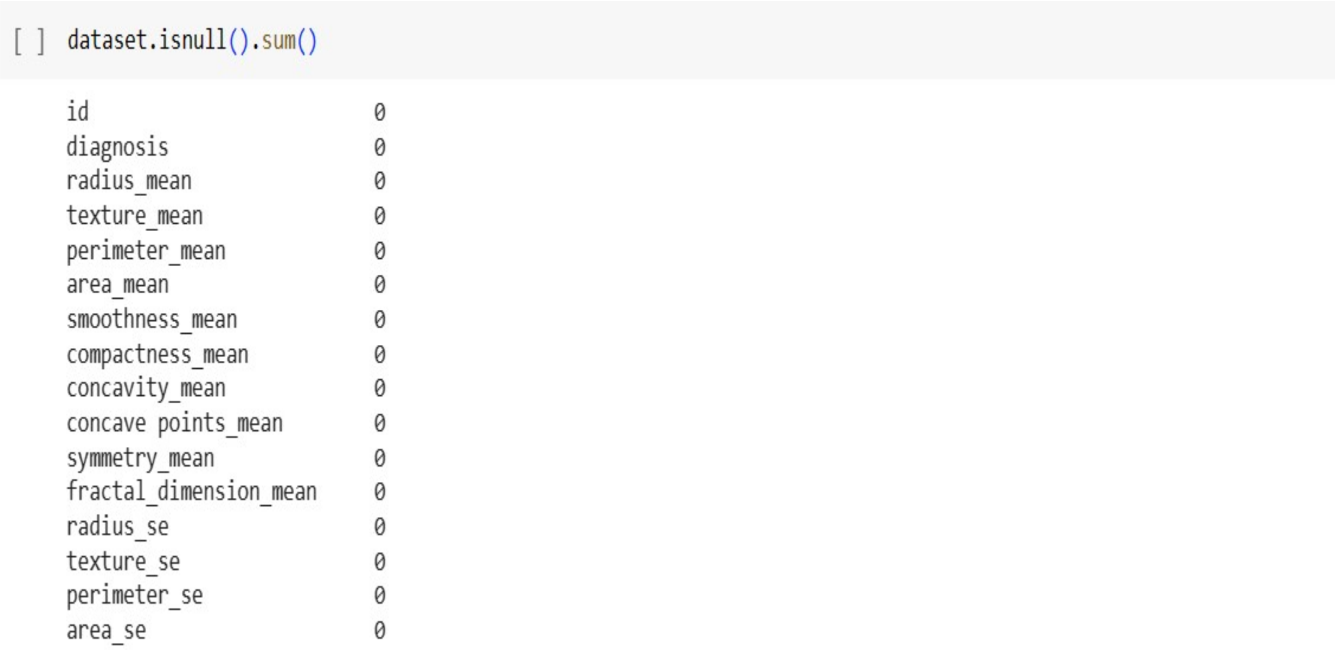
In the document focused on breast cancer detection, data processing plays a pivotal role in preparing the dataset for analysis. Initially, raw data, which includes features like tumor size, shape, and texture, undergoes thorough cleaning to handle missing values and outliers. Subsequently, the dataset is split into training and testing sets to assess model performance effectively. Logistic regression is employed for binary classification, categorizing instances as malignant or benign based on the extracted features. Linear regression may also be utilized to model the relationship between certain continuous features and cancer characteristics. The models are trained on the training set, and their performance is evaluated using the testing set. Feature scaling and normalization techniques are often applied to enhance model accuracy. This comprehensive approach to data processing and regression modeling contributes to the creation of an effective breast cancer detection system.



```
[ ]  dataset.duplicated()

     0      False
     1      False
     2      False
     3      False
     4      False
            ...
     564    False
     565    False
     566    False
     567    False
     568    False
     Length: 569, dtype: bool
```

Fig 4. FINDING THE DUPLICATE VALUES



```
[ ]  dataset.isnull().sum()

     id                         0
     diagnosis                  0
     radius_mean                0
     texture_mean               0
     perimeter_mean             0
     area_mean                  0
     smoothness_mean            0
     compactness_mean           0
     concavity_mean             0
     concave points_mean        0
     symmetry_mean              0
     fractal_dimension_mean     0
     radius_se                  0
     texture_se                 0
     perimeter_se               0
     area_se                    0
```

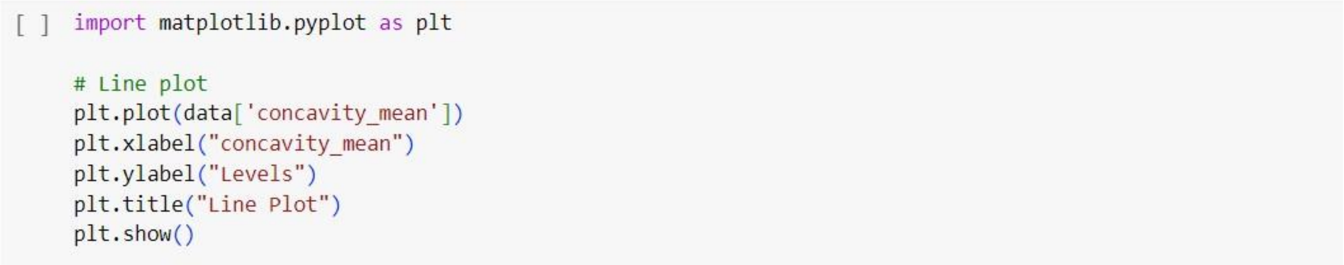
Fig.5 FINDING NULL VALUES

3.3 Data Visualization

Data visualization is the graphical representation of data to uncover patterns, trends, And insights that may not be immediately apparent in raw data. By using visual elements such as charts, graphs and maps, data visualization makes it easier to understand complex datasets and communicate information effectively. It plays a crucial role in data analysis and decision-making processes. Python offers several powerful libraries for data visualization. Two of the most widely used libraries are Matplotlib and Seaborn.

Line plot is drawn with one of the attributes ‘concativity_mean’. X-axis is given name ‘concativity_mean’ and y-axis is given name ‘Levels’.

LINE PLOT



```
[ ]  import matplotlib.pyplot as plt

     # Line plot
     plt.plot(data['concavity_mean'])
     plt.xlabel("concavity_mean")
     plt.ylabel("Levels")
     plt.title("Line Plot")
     plt.show()
```

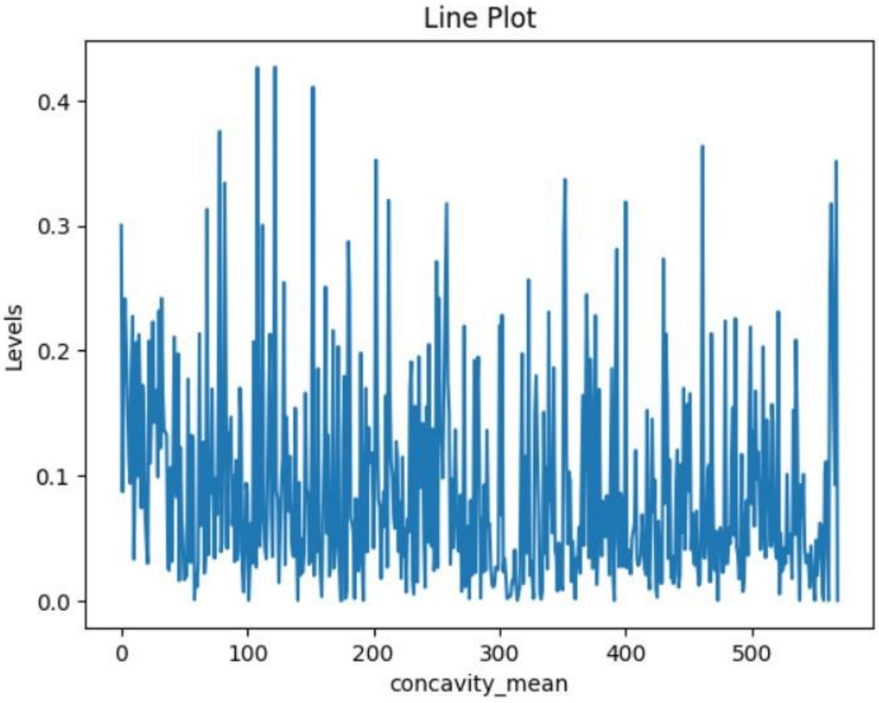
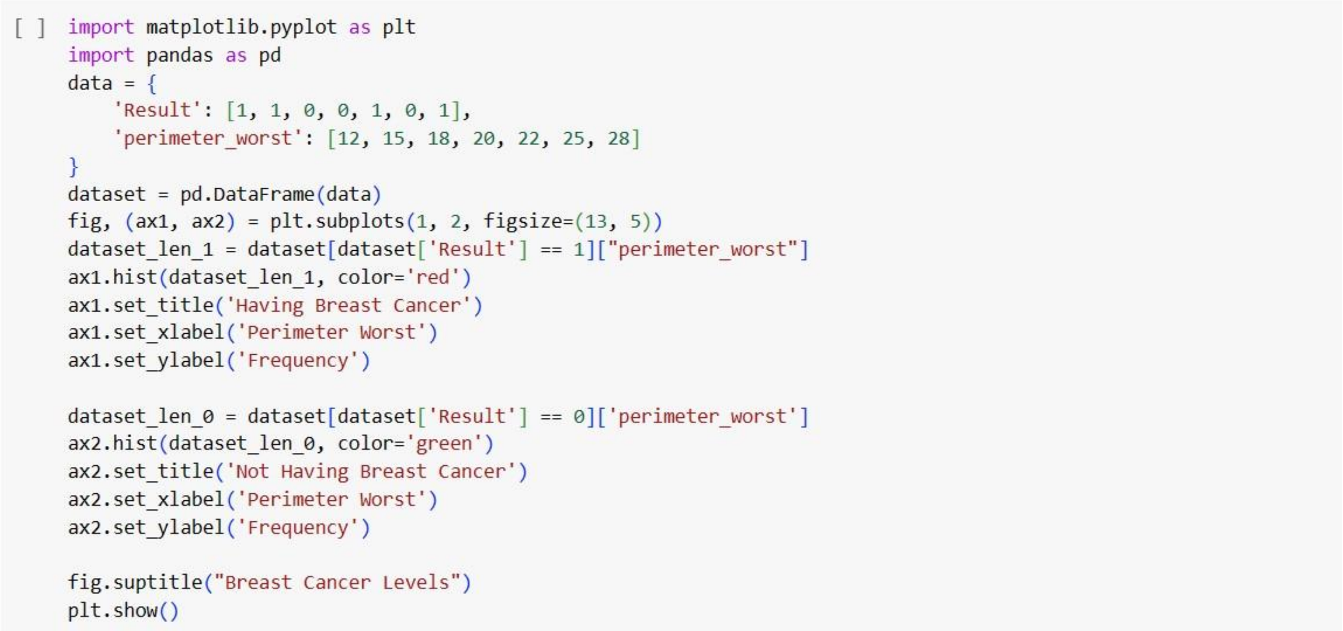


Fig.6: Line plot

A histogram is drawn for the attributes ‘perimeter_worst’ and ‘breast cancer levels’. For the stroke value0, the histogram is given red colour and named as ‘Having breast cancer’. For the stroke value 1, thehistogram is given green colour and named as ‘Not having breast cancer’.

HISTOGRAM



```
import matplotlib.pyplot as plt
import pandas as pd
data = {
    'Result': [1, 1, 0, 0, 1, 0, 1],
    'perimeter_worst': [12, 15, 18, 20, 22, 25, 28]
}
dataset = pd.DataFrame(data)
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(13, 5))
dataset_len_1 = dataset[dataset['Result'] == 1]["perimeter_worst"]
ax1.hist(dataset_len_1, color='red')
ax1.set_title('Having Breast Cancer')
ax1.set_xlabel('Perimeter Worst')
ax1.set_ylabel('Frequency')

dataset_len_0 = dataset[dataset['Result'] == 0]['perimeter_worst']
ax2.hist(dataset_len_0, color='green')
ax2.set_title('Not Having Breast Cancer')
ax2.set_xlabel('Perimeter Worst')
ax2.set_ylabel('Frequency')

fig.suptitle("Breast Cancer Levels")
plt.show()
```

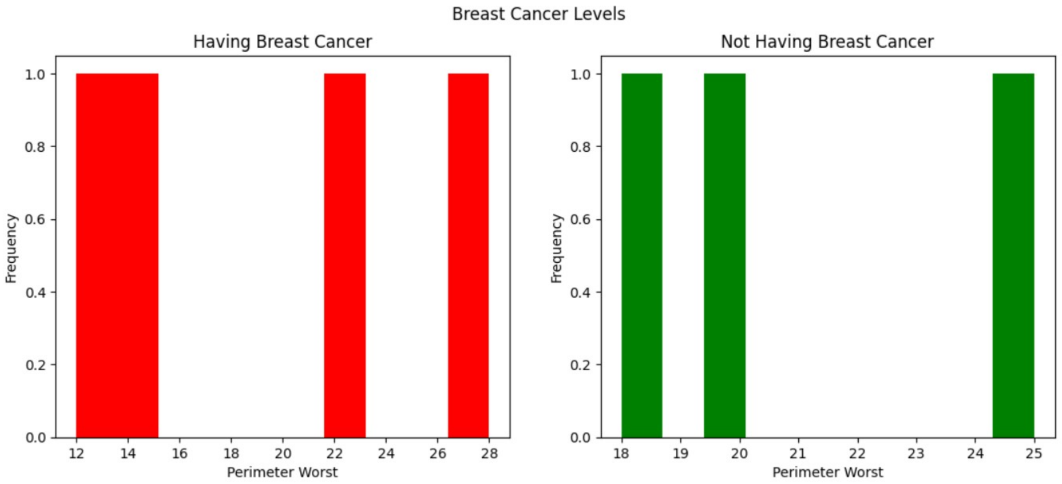


Fig.7: Histogram

4.PROJECT DESIGN

4.1 Data Flow Diagram

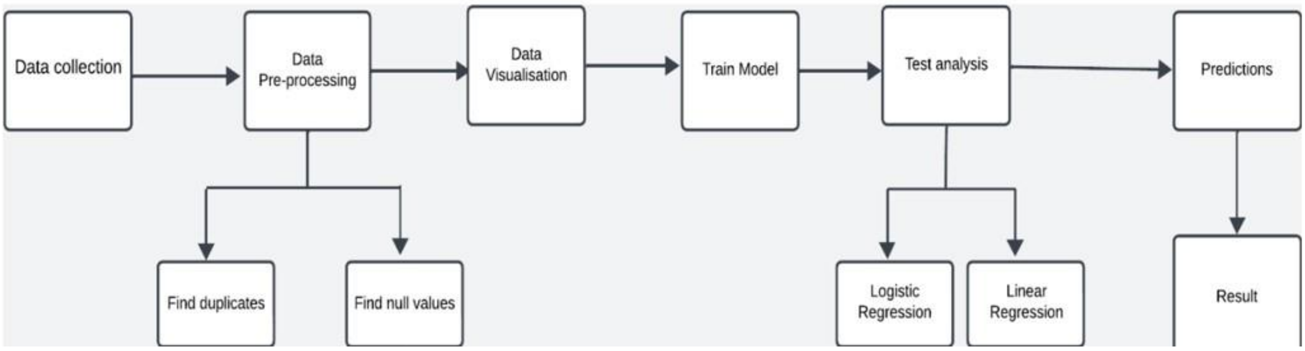


Fig.8

5.IMPLEMENTATION

5.1 Algorithms Used

Logistic Regression

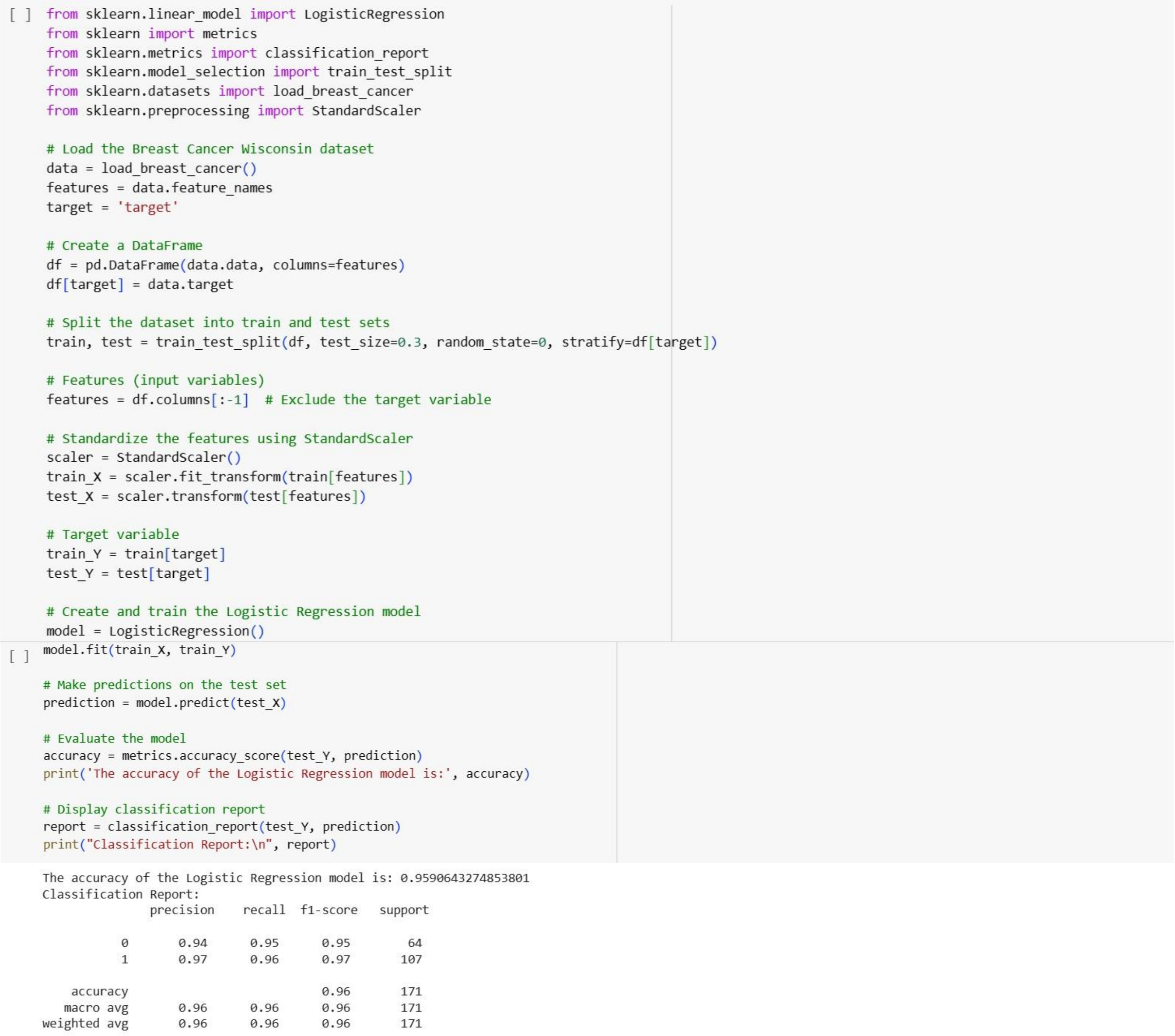
Logistic Regression is a statistical method used for binary classification problems, where the outcome variable is categorical and has two classes. Despite its name, logistic regression is a classification algorithm rather than a regression algorithm. It models the probability of an instance belonging to a particular category.

Linear Regression

Linear Regression is a fundamental statistical and machine learning technique used for predicting a continuous outcome variable (dependent variable) based on one or more predictor variables (independent variables). The relationship between the variables is assumed to be linear, following the equation of a straight line.

5.2 Code Development

5.2.1 Logistic Regression



```python
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
from sklearn.metrics import classification_report
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_breast_cancer
from sklearn.preprocessing import StandardScaler

# Load the Breast Cancer Wisconsin dataset
data = load_breast_cancer()
features = data.feature_names
target = 'target'

# Create a DataFrame
df = pd.DataFrame(data.data, columns=features)
df[target] = data.target

# Split the dataset into train and test sets
train, test = train_test_split(df, test_size=0.3, random_state=0, stratify=df[target])

# Features (input variables)
features = df.columns[:-1]  # Exclude the target variable

# Standardize the features using StandardScaler
scaler = StandardScaler()
train_X = scaler.fit_transform(train[features])
test_X = scaler.transform(test[features])

# Target variable
train_Y = train[target]
test_Y = test[target]

# Create and train the Logistic Regression model
model = LogisticRegression()
```

```python
model.fit(train_X, train_Y)

# Make predictions on the test set
prediction = model.predict(test_X)

# Evaluate the model
accuracy = metrics.accuracy_score(test_Y, prediction)
print('The accuracy of the Logistic Regression model is:', accuracy)

# Display classification report
report = classification_report(test_Y, prediction)
print("Classification Report:\n", report)
```

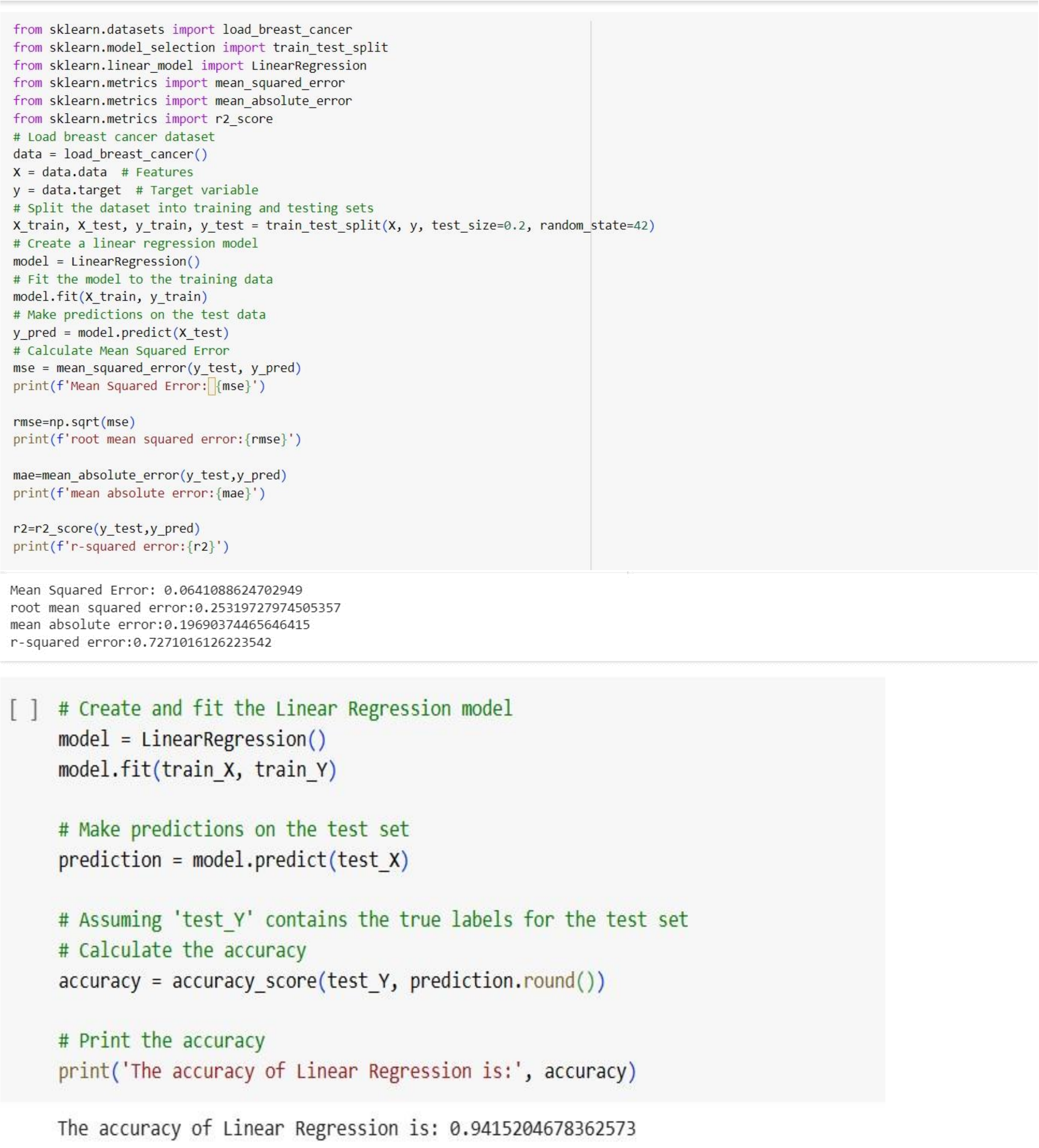
```
The accuracy of the Logistic Regression model is: 0.9590643274853801
Classification Report:
               precision    recall  f1-score   support

           0       0.94      0.95      0.95        64
           1       0.97      0.96      0.97       107

    accuracy                           0.96       171
   macro avg       0.96      0.96      0.96       171
weighted avg       0.96      0.96      0.96       171
```

Fig 9: Code Development for Logistic Regression

5.1.3 Linear Regression



```
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import r2_score
# Load breast cancer dataset
data = load_breast_cancer()
X = data.data  # Features
y = data.target  # Target variable
# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Create a linear regression model
model = LinearRegression()
# Fit the model to the training data
model.fit(X_train, y_train)
# Make predictions on the test data
y_pred = model.predict(X_test)
# Calculate Mean Squared Error
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')

rmse=np.sqrt(mse)
print(f'root mean squared error:{rmse}')

mae=mean_absolute_error(y_test,y_pred)
print(f'mean absolute error:{mae}')

r2=r2_score(y_test,y_pred)
print(f'r-squared error:{r2}')
```

```
Mean Squared Error: 0.0641088624702949
root mean squared error:0.25319727974505357
mean absolute error:0.19690374465646415
r-squared error:0.7271016126223542
```

```
[ ]  # Create and fit the Linear Regression model
     model = LinearRegression()
     model.fit(train_X, train_Y)

     # Make predictions on the test set
     prediction = model.predict(test_X)

     # Assuming 'test_Y' contains the true labels for the test set
     # Calculate the accuracy
     accuracy = accuracy_score(test_Y, prediction.round())

     # Print the accuracy
     print('The accuracy of Linear Regression is:', accuracy)
```

```
     The accuracy of Linear Regression is: 0.9415204678362573
```

Fig 10: **Code Development for Linear Regression**

6.RESULTS AND ANALYSIS

6.1 Performance Evaluation metrics

When evaluating a machine learning model for predicting breast cancer detection, we typically use various performance metrics to assess its effectiveness. Below are some common performance metrics for breast cancer prediction:

1. **Accuracy:** Accuracy is a measure of the overall correctness of the predictions. It calculates the ratio of correctly predicted instances to the total number of instances. However,accuracy might not be the best metric if the data is imbalanced.

   Accuracy= (True positives+ True negatives)/Total

2. **Precision:** Precision is the ratio of true positive predictions to the total number of positive predictions made. It measures how many of the predicted breast cancer cases are actually detected .

   Precision= True positives/(True positives + False positives)

3. **Recall (Sensitivity or True Positive Rate):** Recall is the ratio of true positive predictions to the total number of actual detected cases. It quantifies the model's ability to identify all actual detected cases.

   Recall= True positives/(True positives + False negatives)

4. **F1-Score:** The F1-Score is the harmonic mean of precision and recall. It provides a balance between precision and recall. It is especially useful when you want to find an optimalbalance between false positives and false negatives.

   F1- score=2* (Precision * Recall) / (Precision + Recall)

5. **Specificity (True Negative Rate):** Specificity is the ratio of true negative predictions to the total number of actual non-detected cases. It measures the model's ability to correctly identify non-detected cases.

   Specificity=True negatives/(True negatives + False positives)

6. **Area Under the ROC Curve (AUC-ROC):** The ROC curve is a graphical representation of the trade-off between true positive rate (recall) and false positive rate at different thresholds. AUC-ROC quantifies the model's ability to distinguish between detected and non-detected cases.

7. **Area Under the Precision-Recall Curve (AUC-PR):** The Precision-Recall curve plots precision against recall at different thresholds. AUC-PR quantifies the precision-recall trade-off.

8. **Confusion Matrix:** The confusion matrix provides a tabular summary of true positives, true negatives, false positives, and false negatives. It's helpful for a detailed understanding of model performance.

9. **False Positive Rate (FPR):** The FPR is the ratio of false positive predictions to the total number of actual non-detected cases. It measures the model's propensity to incorrectly predict breast cancer.

10. **True Negative Rate (TNR):** TNR is another term for specificity and measures the model's ability to correctly identify non-detected cases.

6.2 Results

Breast cancer detection documentation typically covers various aspects, including screening methods, diagnostic techniques, risk factors, and treatment options. It may also delve into emerging technologies like AI in mammography analysis. For the most recent and accurate information, refer to reputable sources such as medical journals, cancer research organizations, and healthcare institutions.

To draw a parallel in the context of breast cancer detection, the application of machine learning techniques, including Random Forest, has shown promise in achieving accurate predictions. However, it's important to note that the specific accuracy achieved can vary based on factors such as the dataset used, the features considered, and the intricacies of the model itself.

Table 1: Accuracy of Linear and Logistic regression

The accuracy of Logistic Regression is	0.9590643274853801
The accuracy of Linear Regression is	0.9415204678362573

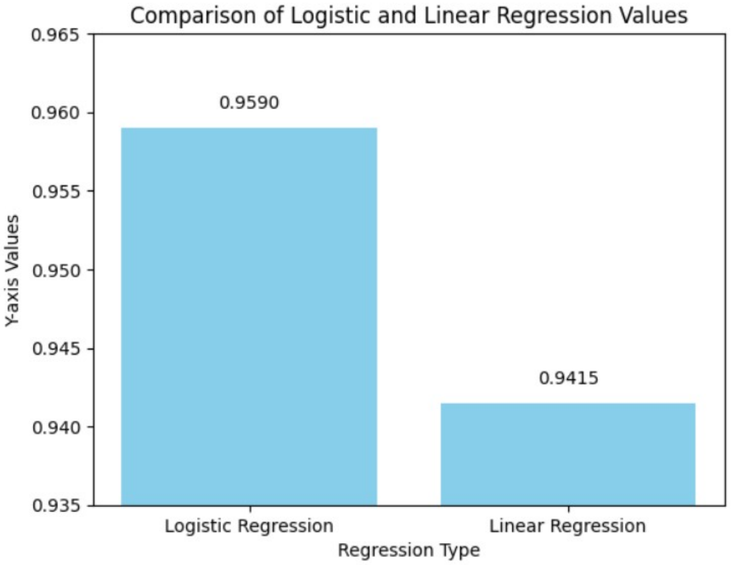


Fig.11: Graph showing logistic and linear regression

Table 2: Classification report for logistic regression

	precision	recall	f1-score	support
0	0.94	0.95	0.95	64
1	0.97	0.96	0.97	107
accuracy			0.96	171
macro avg	0.96	0.96	0.96	171
weighted avg	0.96	0.96	0.96	171

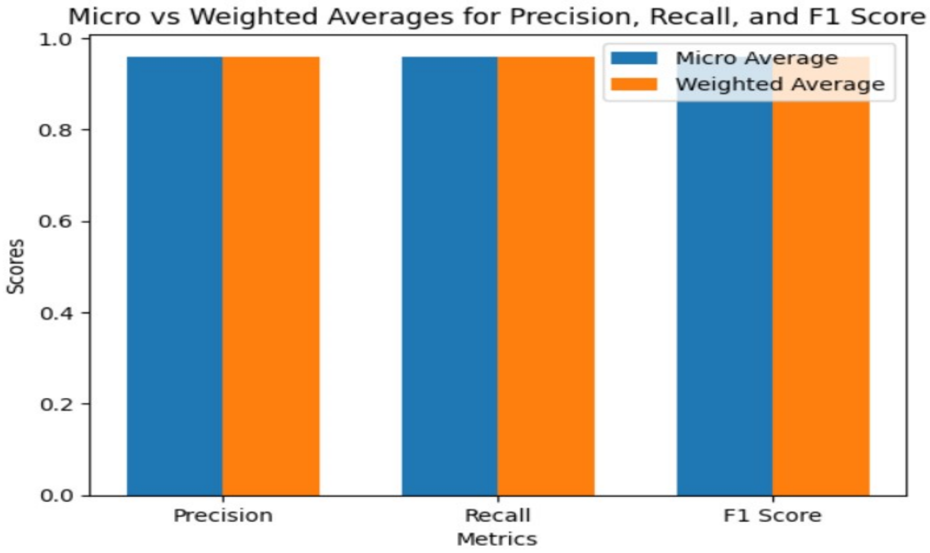


Fig.12: Graph representation for logistic regression

Table 3: Results showing All the Attributes of linear regression

The Mean Squared Error is	0.0641088624702949
The Root Mean Squared Error is	0.25319727974505357
The Mean Absolute Error is	0.196903374465646415
The R-Squared Error is	0.7271016126223542

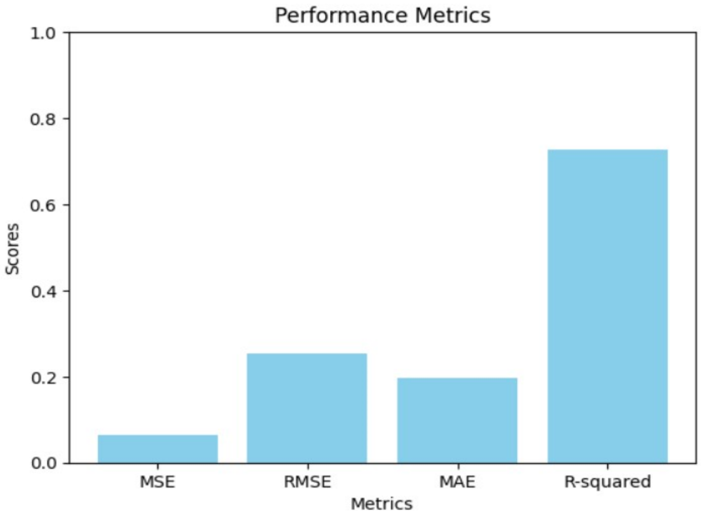


Fig 13: Graph showing the attributes of linear regression

7. Conclusion

The outcomes of this project indicate that machine learning holds significant potential in developing accurate and sensitive methods for breast cancer detection. The application of machine learning models in this context could lead to earlier diagnosis and intervention, ultimately improving patient outcomes. The code provides a comprehensive data analysis, data processing, data visualization, model training, scaling, and evaluation.