BREAST CANCER CLASSIFICATION

PROJECT CODE: OP03

TEAM NUMBER: 5

Students Details:

1.Student Names:

Sneha Das
Debolina Chakraborty
Anandi Roy Chowdhury
Devanshi Bhattacharjee

2.Course and Batch:

B.Sc. (Hons) Computer Science
Batch:2023-27

3.Institute Name:

St.Xavier's College (Autonomous), Kolkata

Project Guide/Mentor:

Snehalika Lall

Report Submitted To:

IDEAS - Institute of Data Engineering, Analytics and Science Foundation, ISI Kolkata.

1.Abstract:

Breast cancer is a common cause of female mortality in developing countries. Thus, the detection of breast cancer is a very crucial task even the most seasoned doctors can't perform with a hundred percent accuracy. Late detection of the tumor has caused a significant number of deaths each year. Fortunately, the introduction of artificial intelligence has helped to solve this worry in the field.

Thereby, with the help of **machine learning** and **algorithm** which is realized in **Python environment**, our study is mainly aiming to develop a more **accurate way of diagnosing breast cancer using machine learning**.

2.Introduction:

Breast cancer is a major health concern, making early and accurate diagnosis essential. This project uses machine learning to classify breast tumors as **benign** or **malignant**, and predicts whether the disease will **recur** or not helping to support timely clinical decisions. Using the appropriate dataset, we apply supervised learning algorithms like **logistic regression** and **support vector machines**.

The process includes **data preprocessing**, **model training**, and **evaluation using accuracy metrics**. A background review of similar studies guides model selection and methodology. The project aims to demonstrate that **ML can achieve over 95% accuracy**, highlighting its relevance and effectiveness in enhancing medical diagnostics through technology using **logistic regression**, **random forest**, **XGBoost** and **SVM**.

3.Project Objective:

This project aims to classify breast cancer tumors and demonstrate machine learning's role in accurate, early diagnosis. The objective of our project mainly focuses on:

- Building a model to classify breast tumors as **benign** or **malignant**.
- Build a model to determine whether the disease will **recur** or **not recur.**
- Showing how machine learning aids **early cancer detection**.
- Comparing classification models using **accuracy, precision**, and **recall**.
- Testing the hypothesis: **ML models can achieve ≥95% accuracy**.

4.Methodology:

This project employs qualitative and quantitative methods to gather, **analyze**, and **interpret data**, ensuring a comprehensive approach to address the research objectives. Our methodology mainly includes:

- **Importing** the appropriate breast cancer classification datasets from **UCI machine learning repository**.
- **Preprocessing the data** by cleaning out noise, normalizing values, removing outliers, etc.
- **Dividing the dataset** into train set, validation set and test set.
- **Training the binary classification model** by selecting the appropriate classification algorithm, parameters and hyperparameters.
- **Testing the model** with the test data and observing the evaluation metrics like confusion matrix, f-score, etc.
- **Improve accuracy** as much as possible.

Training and Validation:

We train the classification models namely - **Logistic Regression**, **Random Forest**, **XGBoost** and **Support Vector Machine** on the three subsets of data after cleaning and preprocessing. After training and validation we shall ultimately choose the proper subset of the data and the model that gives the best performance.

Diagnostic Model:

Steps Followed:

- Normalize the data using **Standard Scaler** of **scikit-learn** library.
- Divide the dataset into train set, validation set and test set using **train_test_split** of scikit-learn. (**60% - train, 20% validation, 20% test**)
- Train the model on the **train set**.
- Fine tune the model using **validation set**.
- Test the model on the **test set**.

Prognostic Model:

Steps followed:

- Normalize the data using **Standard Scaler** of **scikit-learn** library.
- Divide the dataset into train set and test set using **train_test_split** of scikit-learn. (**80% - train, 20% test**)
- Train and set the hyper-parameters of the model on the **train set**.
- Test the model on the **test set**.

Observations:

Performance of the prognostic model is weaker than the diagnostic model due to lack of sufficient data.

5.Data Analysis and Results:

This project delves into data analysis to uncover meaningful patterns in breast cancer diagnostic data. By exploring relationships among **clinical features**, we aim to enhance understanding of **tumor behavior**. The insights gained serve as a foundation for building predictive models, contributing to more accurate and timely breast cancer detection. Ultimately, the project aspires to support early detection, enhance diagnostic precision and contribute to more informed clinical decision-making in the fight against breast cancer.

This project includes certain work plans assisted and conducted by team members:

Processing and cleaning the datasets executed by Sneha Das:

Cleaning and pre-processing of the dataset is an important step before training the model for breast cancer classification. This process begins by handling missing or inconsistent data, either by removing the records or by inputting them with appropriate methods.

After checking whether there are any inconsistent data in the dataset, the **Heatmapping** is done to visualize the **correlation matrix**, which shows how strongly the different features(columns) are related to each other. If the two features are highly related to each other, then they are very similar and keeping both of them is unnecessary making the model more complex. Based on this, we identify and drop the repetitive features. This makes the model faster as it focuses on the most important features.

Diagnostic: In this analysis, we examined feature **correlations** within a diagnostic dataset consisting of **10** features. After confirming the absence of missing values, we computed correlation matrices using

the **.corr()** method with **numeric_only=True**. Heatmaps were generated via **Seaborn** to visualize feature relationships within and across the subsets.

Features showing high correlation **(threshold: 0.90–1.0)** were identified and removed to reduce any possible **redundancy** and improve **model efficiency**.

- **<u>The following features were dropped</u>:**

  1. **Perimeter** and **area** from all subsets.

  2. **Texture** and **concave_points** from subset 3.

- **<u>Finally we trained the datasets with three sets of data to determine the most relevant one</u>:**

  1. The **original** data

  2. Data without the attributes – **perimeter1, area1, perimeter2, area2, perimeter3, area3**

  3. Data without the features in (2.) **plus texture and concave_points.**

Prognostic: In this model, a small proportion **(2.02%)** of missing values was observed in the **lymph_node_status** feature. Given, the low percentage, the feature was retained, and missing values were imputed using **forward-fill(ffill)** method. A comparison of distribution plots before and after imputation showed minimal difference, confirming the validity of this approach. Since, **Time** doesn't play a role in prognosis of the disease, so we eventually drop it.

Following this, correlation heatmaps were generated for **three** subsets of the dataset using correlation matrices, similar to the diagnostic

model. Based on high correlation **(threshold: 0.90–1.0)**, the following features were removed to reduce **redundancy**:

- **perimeter1, area1, perimeter3, area3 and radius2**
- **area2 and concavity1**

Finally, we trained the datasets with three sets of data to determine the most relevant one:

- The **original** data.
- Data without the attributes - **perimeter1**, **area1**, **perimeter3**, **area3** and **radius2**.
- Data without the features in (2.) plus **area2** and **concavity1.**

Training the Diagnostic Model executed by Debolina Chakraborty:

A diagnostic model uses **data-driven algorithms** to identify or predict diseases, assisting in early detection, accurate classification and improved medical decision-making.

Steps Followed:

- Normalize the data using **Standard Scaler** of **scikit-learn** library.
- Divide the dataset into train set, validation set and test set using **train_test_split** of scikit-learn. (**60% - train, 20% validation, 20% test**)
- Train the model on the **train set**.
- Fine tune the model using **validation set**.
- Test the model on the **test set**.

Observations:

Performance of the diagnostic model is more **efficient** as compared to the prognostic model due to the lack of **sufficient data** in the prognostic model.

Therefore, diagnostic model accurately identifies disease presence, supporting **timely intervention** and enhancing **clinical decision-making**, ultimately contributing to improved patient care and early detection. Its intelligent analysis empowers healthcare professionals, enhances confidence in diagnosis, and paves the way for more responsive, personalized treatment strategies—bridging the gap between data and compassionate, life-saving care.

Training the Prognostic Model executed by Anandi Roy Chowdhury:

A prognostic model predicts **future clinical** outcomes by analysing patient data, aiding in personalized treatment decisions and improving healthcare through early risk assessment and informed medical planning.

Steps followed:

- Normalize the data using **Standard Scaler** of **scikit-learn** library.
- Divide the dataset into train set and test set using **train_test_split** of scikit-learn. (**80% - train, 20% test**)
- Train and set the hyper-parameters of the model on the **train set**.
- Test the model on the **test set**.

Observations:

Performance of the prognostic model is **weaker** than the diagnostic model due to lack of **sufficient data**.

Though the prognostic model is less **robust** than its diagnostic counterpart due to **limited** data, it still offers valuable foresight, highlighting the promise of predictive analytics in guiding future care and personalized treatment pathways. As data collection improves and more comprehensive clinical features are integrated, the model's predictive power will strengthen, offering deeper insights into disease progression.

Validating and testing the models executed by Devanshi Bhattacharjee:

Validating and testing the models for this project ensures their reliability and accuracy. This crucial step evaluates performance on unseen data, confirming the model's effectiveness in making precise, real-world predictions for early and accurate diagnosis.

Testing:

- Diagnostic Model:
  LOGISTIC REGRESSION

    1)X-Data:

      Train set:

        Accuracy: 99.12023460410558

Confusion Matrix: [[216 0]
[3 122]]

Validation set:

Accuracy: 98.35164835164835

Confusion matrix: [[109 1]
[2 70]]

Test set:

Accuracy: 100.0

Confusion Matrix: [[31 0]
[0 15]]

2) X1 Data:

Train Set:

Accuracy: 98.53372434017595

Confusion Matrix: [[215 1]
[4 121]]

Validation Set:

Accuracy: 98.35164835164835

Confusion Matrix: [[109 1]
[2 70]]

Test Set:

Accuracy: 97.82608695652173

Confusion Matrix: [[31 0]
[1 14]]

3)X2 Data:

Train Set:

Accuracy: 98.82697947214076

Confusion Matrix: [[215 1]

[3 122]]

Validation Set:

Accuracy: 98.35164835164835

Confusion Matrix: : [[109 1]

[2 70]]

Test Set:

Accuracy: 9 100.0

Confusion Matrix: [[31 0]

[0 15]]

CONCLUSION: We select the X data.

RANDOM FOREST

1)X Data

Train Set:

Accuracy: 99.41348973607037

Confusion Matrix: [[216 0]

[2 123]]

Validation Set:

Accuracy: 99.45054945054946

Confusion Matrix: [[110 0]

[1 71]]

Test Set:

Accuracy:100.0

Confusion Matrix: [[31 0]

[0 15]]

2)X1 Data

Train Set:

Accuracy: 98.53372434017595

Confusion Matrix: [[215 1]

[4 121]]

Validation Set:

Accuracy: 98.9010989010989

Confusion Matrix: [[109 1]

[1 71]]

Test Set:

Accuracy: 93.47826086956522

Confusion Matrix: [[28 3]

[0 15]]

3)X2 Data

Train Set:

Accuracy: 98.82697947214076

Confusion Matrix: [[215 1]

[3 122]]

Validation Set:

Accuracy: 98.9010989010989

Confusion Matrix: [[109 1]

[1 71]]

Test Set:

Accuracy: 93.47826086956522

Confusion Matrix: [[29 2]

[1 14]]

CONCLUSION: We select the X data.

XGBOOST

1)X Data

Train Set:

Accuracy: 99.70674486803519

Confusion Matrix: [[216 0]

[1 124]]

Validation Set:

Accuracy: 96.15384615384616

Confusion Matrix: [[108 2]

[5 67]]

Test Set:

Accuracy: 100.0

Confusion Matrix: [[31 0]

[0 15]]

2)X1 Data

Train Set:

Accuracy: 100.0

Confusion Matrix: [[216 0]

[0 125]]

Validation Set:

Accuracy: 94.5054945054945

Confusion Matrix: [[107 3]

[7 65]]

Test Set:

Accuracy: 97.82608695652173

Confusion Matrix: [[31 0]

[1 14]]

3)X2 Data

Train Set:

Accuracy: 99.70674486803519

Confusion Matrix: [[216 0]

[1 124]]

Validation Set:

Accuracy: 95.6043956043956

Confusion Matrix: [[108 2]

[6 66]]

Test Set:

Accuracy: 97.82608695652173

Confusion Matrix: [[31 0]

[1 14]]

CONCLUSION: We select the X data.

SVM

1)X Data

Train Set:

Accuracy: 99.12023460410558

Validation Set:

Accuracy: 100.0

Test Set:

Accuracy: 100.0

2)X1 Data

Train Set:

Accuracy: 97.94721407624634

Validation Set:

Accuracy: 100.0

Test Set:

Accuracy: 100.0

3)X2 Data

Train Set:

Accuracy: 98.24046920821115

Validation Set:

Accuracy: 100.0

Test Set:

Accuracy: 97.82608695652173

CONCLUSION: We select the X data.

Comparing the accuracy of 'X Data' of the four models, namely Logistic Regression, Random Forest, XGBoost and SVM we select the SVM model as the Diagnostic Model.

- Prognostic Model:
  LOGISTIC REGRESSION

  1)X1 Data

  Train Set:

  Accuracy: 86.70886075949366

  Confusion Matrix: [[116 3]
  [18 21]]

  Test Set:

  Accuracy: 85.0

  Confusion Matrix: [[29 3]
  [3 5]]

  2)X1_newData

  Train Set:

  Accuracy: 78.48101265822784

  Confusion Matrix: [[111 8]

[26 13]]

Test Set:

Accuracy: 80.0

Confusion Matrix: [[29 3]

[5 3]]

3)X2_newData

Train Set:

Accuracy: 79.11392405063292

Confusion Matrix: [[112 7]

[26 13]]

Test Set:

Accuracy: 75.0

Confusion Matrix: [[29 3]

[7 1]]

CONCLUSION: We select the X1 data.

RANDOM FOREST

1)X1 Data

Train Set:

Accuracy: 100.0

Confusion Matrix: [[136 0]

[0 42]]

Test Set:

Accuracy: 90.0

Confusion Matrix: [[15 0]

[2 3]]

2)X1_new Data

Train Set:

Accuracy: 87.5

Confusion Matrix: [[126 0]

[21 21]]

Test Set:

Accuracy: 86.66666666666667

Confusion Matrix: [[25 0]

[4 1]]

3)X2_new Data

Train Set:

Accuracy: 88.09523809523809

Confusion Matrix: [[126 0]

[20 22]]

Test Set:

Accuracy: 86.66666666666667

Confusion Matrix: [[25 0]

[4 1]]

CONCLUSION: We select the X1 data.

XGBOOST

1)X1 Data

Train Set:

Accuracy: 100.0

Confusion Matrix: [[126 0]

[0 42]]

Test Set:

Accuracy: 93.33333333333333

Confusion Matrix: [[24 1]

[1 4]]

2)X1_new Data

Train Set:

Accuracy: 83.33333333333334

Confusion Matrix: [[126 0]

[28 14]]

Test Set:

Accuracy: 80.0

Confusion Matrix: [[24 1]

[5 0]]

3)X2_new Data

Train Set:

Accuracy: 83.33333333333334

Confusion Matrix: [[126 0]

[28 14]]

Test Set:

Accuracy: 80.0

Confusion Matrix: [[24 1]

[5 0]]

CONCLUSION: We select the X1 data.

SVM

1)X1s Data

Train Set:

Accuracy: 99.40476190476191

Confusion Matrix: [[126 0]

[1 41]]

Test Set:

Accuracy: 96.66666666666667

Confusion Matrix: [[25 0]

[1 4]]

2)X1_new Data

Train Set:

Accuracy: 89.28571428571429

Confusion Matrix: [[126 0]

[18 24]]

Test Set:

Accuracy: 86.66666666666667

Confusion Matrix: [[25 0]

[4 1]]

3)X2_new Data

Train Set:

Accuracy: 89.28571428571429

Confusion Matrix: [[126 0]

[18 24]]

Test Set:

Accuracy: 9 86.66666666666667

Confusion Matrix: [[25 0]

[4 1]]

CONCLUSION: We select the X1 data.

Comparing the accuracy of 'X1 Data' of the four models, namely Logistic Regression, Random Forest, XGBoost and SVM we select the SVM model as the Prognostic Model.

6.Conclusion:

"Prevention is better than the cure"-indeed it's true. Thus, propr detection if done beforehand, can be handled and the respective person can see through their remission.

Breast cancer classification using AI offers an accurate and efficient method for early diagnosis and outcome prediction. By preprocessing the data and training both diagnostic and prognostic models using the UCI Wisconsin dataset, the system helps in identifying cancer types and predicting disease progression. This approach supports faster, more reliable, and personalized treatment decisions in healthcare.

Thereby by testing and training the models, we can come to the conclusion that the performance of the diagnostic model is more efficient as compared to the prognostic model due to the lack of sufficient data in the prognostic model. Though the prognostic model is less robust yet it can be used to get a predictive analysis in near future with required datasets being provided.

The project underscores the significance of data-driven approaches in modern medicine, paving the way for more efficient, and accessible diagnostic solutions. Ultimately, it reflects how technology, when thoughtfully applied, can empower life-saving advancements and foster hope in the fight against cancer.

7.Appendices:

For the completion of this project on **breast cancer prediction**, extensive data analysis, model building, and validation were undertaken to develop a reliable tool aimed at enhancing early detection and supporting informed clinical decisions. Some notables which are used:

- Datasets obtained from: **UCI Machine Learning Repository**.
- **Diagnostic Model**: https://colab.research.google.com/drive/1hZOPx7fIlaAVud6HbaMcww_InlSCgt-K?usp=sharing
- **Prognostic Model**: https://colab.research.google.com/drive/1Ux6I_xnNGjcZw-bkZ-PDzt1Fjv_UfNiC?usp=sharing
  Libraries Used: **Numpy**, **Pandas**, **Scikit-Learn**, **Matplotlib** and **Seaborn**.
- Cloud Platform Used: **Google Colab Notebook**.
- Code Editor: **Visual Studio Code**.
- Programming Language: **Python**.