A PROJECT REPORT

on

"BREAST-CANCER DETECTION USING MACHINE LEARNING CLASSIFIER"

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of BACHELOR'S DEGREE IN INFORMATION TECHNOLOGY

BY

PIYUSH MOHANTY	2005949
SANJANA SUBUDHI	2005961
ANKITA SAMAL	2005922

UNDER THE GUIDANCE OF
Prof. (Dr.) Alok Kumar Jagadev

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

May 2023

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

“BREAST-CANCER DETECTION USING MACHINE LEARNING CLASSIFIER“

submitted by

PIYUSH MOHANTY	2005949
SANJANA SUBUDHI	2005961
ANKITA SAMAL	2005922

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Sci- ence & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2022-2023, under our guidance.

Date: / /

(Prof. (Dr.) Alok Kumar Jagadev)
Project Guide

Acknowledgements

We are profoundly grateful to **Dr.Alok Kumar Jagadev** of **Affiliation** for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. **....................**

PIYUSH MOHANTY
SANJANA SUBUDHI
ANKITA SAMAL

ABSTRACT

In today's world we know how big of a disease cancer is which causes highest number of deaths. Cancer is a collection of more than 100 different and individual diseases. It can evolve in any part of the body and have many different forms in each body part. Breast Cancer is one among them and it is the most widespread type of cancer among women all over the world. The diagnosis of breast cancer manually takes long time and there is low availability of systems, we require to develop the automatic diagnosis system for early detection of breast cancer. So we are developing a diagnosis model in our project for which we studied the different supervised machine learning classifiers and compared them to get to know which classifier is giving the best accuracy. For that we have taken the dataset from the UCI ML Wisconsin breast cancer database which is the benchmark database for comparing the results through different algorithms. We will use following classification techniques of machine learning for our project implementation like Logistic Regression, Support Vector Classifier, KNN Classifier, Naive Bayes, Decision trees, Extract Tree Classifier, Random Forest, Gradient Boost, Ada boost Classifier, XG Boost, Stacking Classifier and Voting Classifier for the classification of benign and malignant tumor in which the machine is learned from the past data and can predict the category of new input.

Keywords- WBCD, Logistic Regression, Support Vector Classifier, KNN Classifier, Naive Bayes, Decision trees, Extract Tree Classifier, Random Forest, Gradient Boost, Ada boost Classifier, XG Boost, Stacking Classifier and Voting Classifier.

Contents

List of Figures

Chapter 1

Introduction

Breast cancer is a serious public health concern and one of the main causes of cancer-related deaths in women across the world.The importance of early detection and precise diagnosis in enhancing patient outcomes and survival rates cannot be overstated. Breast cancer diagnosis, on the other hand, can be difficult, and it frequently necessitates the skill of radiologists tointerpret mammography pictures. This procedure can be time- consuming and error-prone, putting a substantial strain on
healthcare staff and negatively harming patient care.

Machine learning has showed considerable potential in improving the identification and diagnosis of breast cancer. Thegoal of this study is to investigate the potential of various machine learning algorithms in detecting breast cancer from mammogram images. The project is divided into stages, which includes data pretreatment, feature extraction, model selection, and assessment. To train and assess the performance of several machine learning algorithms, we will use a publicly available dataset comprising mammography pictures, demographic, and clinical data.

Our ultimate objective is to create an automated breast cancer detection tool that can properly anticipate the existence of breastcancer and help radiologists with the diagnostic process. The suggested technology can possibly improve the accuracy and efficiency of breast cancer detection by greatly reducing the time and effort necessary for manual diagnosis. Furthermore, theproject can provide insights into the most effective machine learning algorithms for breast cancer detection and diagnosis, which will help to develop better models in the future.

This project has the potential to significantly improve patient outcomes, lower healthcare costs, and advance cancer research.An automated breast cancer screening tool can help healthcare providers make prompt and precise choices, thereby improving patient outcomes and survival rates. Furthermore, the findings ofthe project can assist researchers and healthcare professionals in identifying the most effective machine learning algorithms for breast cancer detection and diagnosis, resulting in further advances in cancer research and treatment.

Literature Review

Breast cancer has become one of women's most prevalent causes of death. It can be diagnosed by categorizing tumours. There are two sorts of tumors: malignant and benign tumors. To distinguish between these tumors, doctors require an accurate diagnostic process. However, even experts have difficulty distinguishing tumours. As a result, diagnosis requires the automation of diagnostic systems. Breast cancer, the most prevalent disease in women, has historically had a high incidence and death rate. Breast cancer is expected to account for 25% of all new cancer diagnoses and 15% of all cancer deaths among women worldwide, according to the most recent cancer data. Patients usually go to the doctor right soon if they notice any signs or symptoms, and the doctor may refer them to an oncologist if required. The oncologist can diagnose breast cancer by evaluating the patient's medical history, checking both breasts, and inspecting any lymph nodes in the armpit for swelling or hardness. We employed the Wisconsin Breast Cancer Dataset (WBCD) and the fine needle aspiration biopsy method in this effort and with that dataset, we used machine learning algorithms to predict whether or not the patient has breast cancer.

Matplotlib is a Python data visualisation toolkit that provides tools for constructing many sorts of charts and plots, such as line plots, scatter plots, bar charts, and histograms.

2.1 Python Libraries used:

Pandas is a Python library used for data manipulation and analysis providing tools for data cleaning, filtering and merging data frames, among other functionalities.

NumPy is a Python library that supports multi-dimensional arrays and matrices, as well as mathematical operations like linear algebra, Fourier transformations, and random number

generation.
Seaborn is a Python library used for data visualization and statistical analysis, providing tools for creating more complex and visually appealing charts and plots, such as heatmaps, box plots, and violin plots.

Scikit-learn (sklearn) is a Python library used for machine learning, providing tools for data preprocessing, feature selection, model selection, and evaluation, as well as a wide range of machine learning algorithms for classification, regression, clustering, and dimensionality reduction.

2.2 ML algorithms used:

Scikit-learn (sklearn) is a Python machine learning toolkit that includes data preparation, feature selection, model selection, and assessment tools, as well as a variety of machine learning algorithms for classification, regression, clustering, and dimensionality reduction.

2.2 ML methods employed:

Logistic Regression is a sort of linear model used in binary classification issues. It predicts the likelihood of a binary answer variable using one or more predictor variables. The model is trained using the maximum likelihood method to estimate the parameters that best suit the data.

SVC is a type of algorithm that is widely used for classification issues. It operates by locating the hyperplane that best divides the data into distinct classes. SVC tries to maximize the margin between the hyperplane and the closest data points, which can improve the model's generalization performance.

The K-Nearest Neighbour Classifier (KNN) is a non-parametric classifier technique. It operates by locating a data point's k nearest neighbours and assigning it to the class with the greatest number of neighbours. KNN requires no training and may be used to solve both binary and multi-class classification problems.

The Naive Bayes method is a probabilistic technique that is used to solve classification issues. It works by assuming that given the class, the characteristics are independent of each other, which simplifies the computation of the posterior probability. Naive Bayes is a computationally efficient method for dealing with high-dimensional data.

Decision Trees are algorithms that are used to solve classification and regression issues. It operates by splitting the data recursively into subgroups based on the values of the predictor variable.The decision tree is constructed by choosing the predictor variable that provides the most information gain at each step. These can be prone to overfitting and can be improved by using ensemble methods such as Random Forest and Gradient Boosting.

ExtraTree is a variant of the decision tree algorithm that selects random split points and thresholds for the decision nodes. This introduces additional randomness into the tree construction process and can reduce overfitting.

Random Forest is an ensemble learning approach that mixes numerous decision trees to increase model accuracy while minimising overfitting. It operates by building numerous decision trees using bootstrapped data samples and random selections of predictor variables. The ultimate forecast is determined by averaging all of the trees' projections.

Gradient Boosting is yet another ensemble learning approach that combines numerous weak learners to increase the accuracy of the model. It works by incrementally adding decision trees to the model and modifying the weights of the data points based on the faults of the preceding trees. Gradient Boosting is computationally costly and susceptible to overfitting.

AdaBoost is an ensemble learning technique that combines the accuracy of the model by combining many weak learners. AdaBoost can be used with any base learner and is computationally efficient.

XGBoost is an optimized version of Gradient Boosting that uses a more efficient algorithm for constructing decision trees and includes regularization techniques to prevent overfitting. It also supports parallel processing and can handle large datasets with high-dimensional features.

2.3 Feature scaling

Feature scaling is the process of standardizing or normalizing the values of input features in a dataset. It is used to ensure that the features are on the same scale and have a similar magnitude, which can improve the performance and convergence of some machine learning algorithms.

2.4 Parameter Tuning

Parameter tuning is the process of selecting the best values for the hyperparameters of a machine learning algorithm. It is used to optimize the performance of the algorithm on the specific task or dataset at hand, by finding the hyperparameters that result in the best validation or test scores. Parameter tuning is essential for improving the accuracy and generalization performance of machine learning models

Chapter 3

Requirement Specifications

Problem Statement :
To identify and classify a breast cancer tumor as malignant / benign and also to select the machine learning classifier which gives the maximum accuracy to determine the same.

2.1 Project Planning:

The model we have used for the implementation of our project is iterative waterfall model (fig.1) by including the required changes to the classical waterfall model to make it usable in practical software development projects. The iterative waterfall model is a software development life cycle method in which initial task of development is executed based on initial requirements which are clearly defined, and then additional features are added to the basic software product through iterations until the final product is completed and desired requirements are met. Feedback feature is also an addition here to the classical waterfall model.

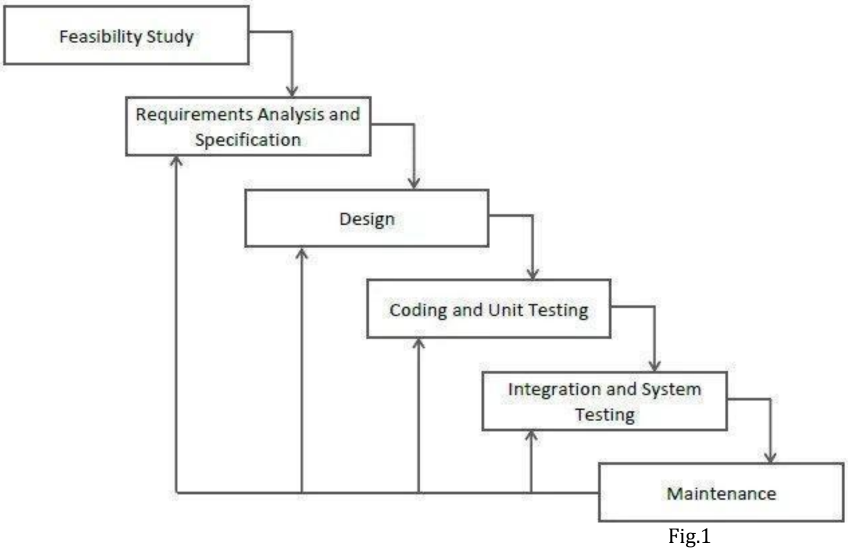

Feasibility Study
Requirements Analysis and Specification
Design
Coding and Unit Testing
Integration and System Testing
Maintenance

Fig.1

2.2 Project Analysis :

Functional Requirements:

The functional requirements of the Breast cancer cell classification using Supervised learning project are as follows:

- Data Loading:

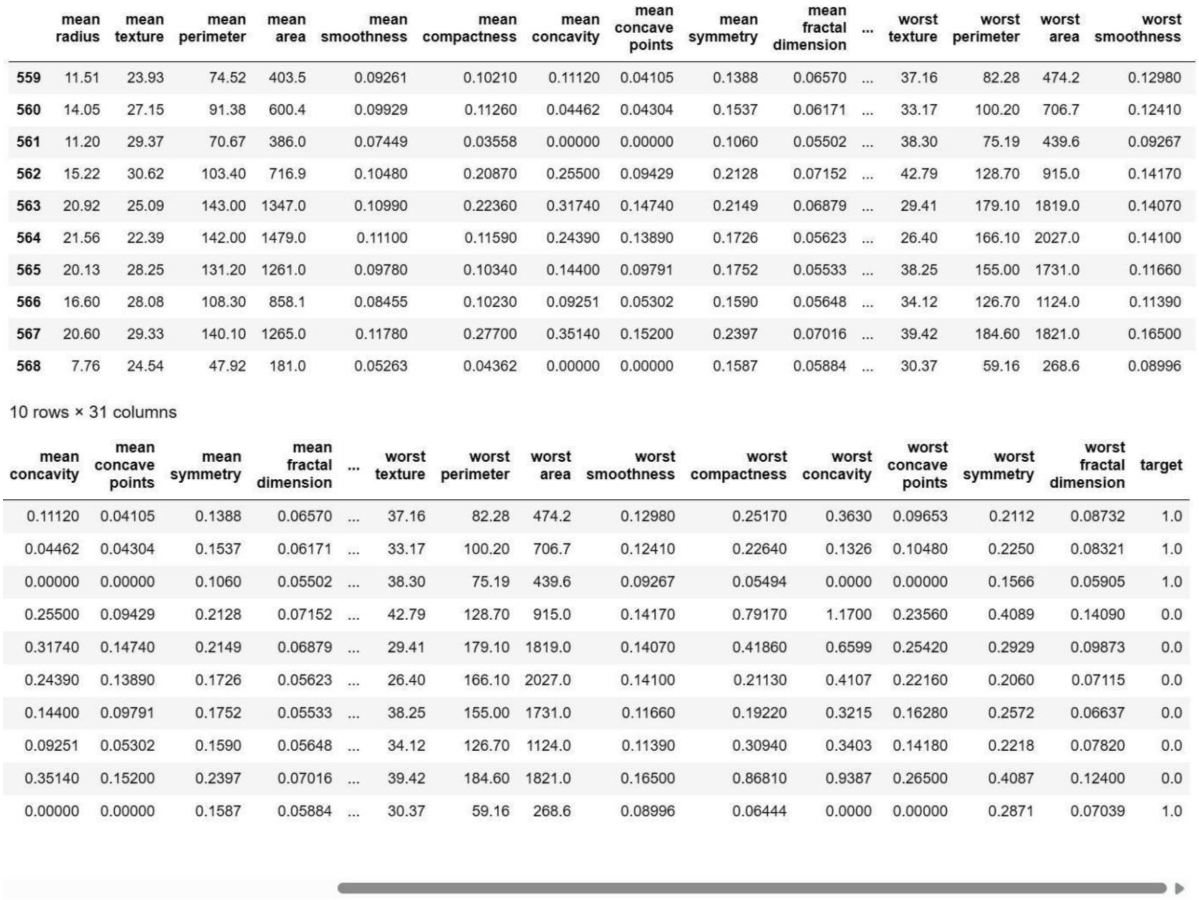
System must be able to fetch the dataset from Wincinsin Breast Cancer database and load the dataset using sklearn, which is an open source data analysis library.

- Data Manipulation:

We have the data in the form of dictionary. The system must be able to display the keys of te dataset which are 'data', 'target', 'frame', 'target_names', 'DESCR', 'feature_names', 'filename' and 'data_module' and values which are in 2D Array format. ‘Target’ is the tumor which can either be malignant or benign (0:malignant, 1:benign). If the patient has cancer it means tumor is malignant else it is benign. We have 569 instances with 30 features.

- Dataframe:

System must be able to create the dataframe by combining the key-value pairs (data and target). It should give the column name by using the ‘feature_name’ and ‘target’ and then we store that into the file for future use. It must be able to check for null values in the dataset. Finally it should describe the numeric distribution of our dataset. (fig.2)



	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness
559	11.51	23.93	74.52	403.5	0.09261	0.10210	0.11120	0.04105	0.1388	0.06570	...	37.16	82.28	474.2	0.12980
560	14.05	27.15	91.38	600.4	0.09929	0.11260	0.04462	0.04304	0.1537	0.06171	...	33.17	100.20	706.7	0.12410
561	11.20	29.37	70.67	386.0	0.07449	0.03558	0.00000	0.00000	0.1060	0.05502	...	38.30	75.19	439.6	0.09267
562	15.22	30.62	103.40	716.9	0.10480	0.20870	0.25500	0.09429	0.2128	0.07152	...	42.79	128.70	915.0	0.14170
563	20.92	25.09	143.00	1347.0	0.10990	0.22360	0.31740	0.14740	0.2149	0.06879	...	29.41	179.10	1819.0	0.14070
564	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	0.1726	0.05623	...	26.40	166.10	2027.0	0.14100
565	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	0.1752	0.05533	...	38.25	155.00	1731.0	0.11660
566	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	0.1590	0.05648	...	34.12	126.70	1124.0	0.11390
567	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	0.2397	0.07016	...	39.42	184.60	1821.0	0.16500
568	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	0.1587	0.05884	...	30.37	59.16	268.6	0.08996

10 rows × 31 columns

mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness	worst compactness	worst concavity	worst concave points	worst symmetry	worst fractal dimension	target
0.11120	0.04105	0.1388	0.06570	...	37.16	82.28	474.2	0.12980	0.25170	0.3630	0.09653	0.2112	0.08732	1.0
0.04462	0.04304	0.1537	0.06171	...	33.17	100.20	706.7	0.12410	0.22640	0.1326	0.10480	0.2250	0.08321	1.0
0.00000	0.00000	0.1060	0.05502	...	38.30	75.19	439.6	0.09267	0.05494	0.0000	0.00000	0.1566	0.05905	1.0
0.25500	0.09429	0.2128	0.07152	...	42.79	128.70	915.0	0.14170	0.79170	1.1700	0.23560	0.4089	0.14090	0.0
0.31740	0.14740	0.2149	0.06879	...	29.41	179.10	1819.0	0.14070	0.41860	0.6599	0.25420	0.2929	0.09873	0.0
0.24390	0.13890	0.1726	0.05623	...	26.40	166.10	2027.0	0.14100	0.21130	0.4107	0.22160	0.2060	0.07115	0.0
0.14400	0.09791	0.1752	0.05533	...	38.25	155.00	1731.0	0.11660	0.19220	0.3215	0.16280	0.2572	0.06637	0.0
0.09251	0.05302	0.1590	0.05648	...	34.12	126.70	1124.0	0.11390	0.30940	0.3403	0.14180	0.2218	0.07820	0.0
0.35140	0.15200	0.2397	0.07016	...	39.42	184.60	1821.0	0.16500	0.86810	0.9387	0.26500	0.4087	0.12400	0.0
0.00000	0.00000	0.1587	0.05884	...	30.37	59.16	268.6	0.08996	0.06444	0.0000	0.00000	0.2871	0.07039	1.0

Fig.2

- Data Visualization:

The system must be able to visualize our data which is in numerical format into several visualization formats for better visual analysis such as pairplot (fig.3), counterplot, barplot, heatmap, etc.

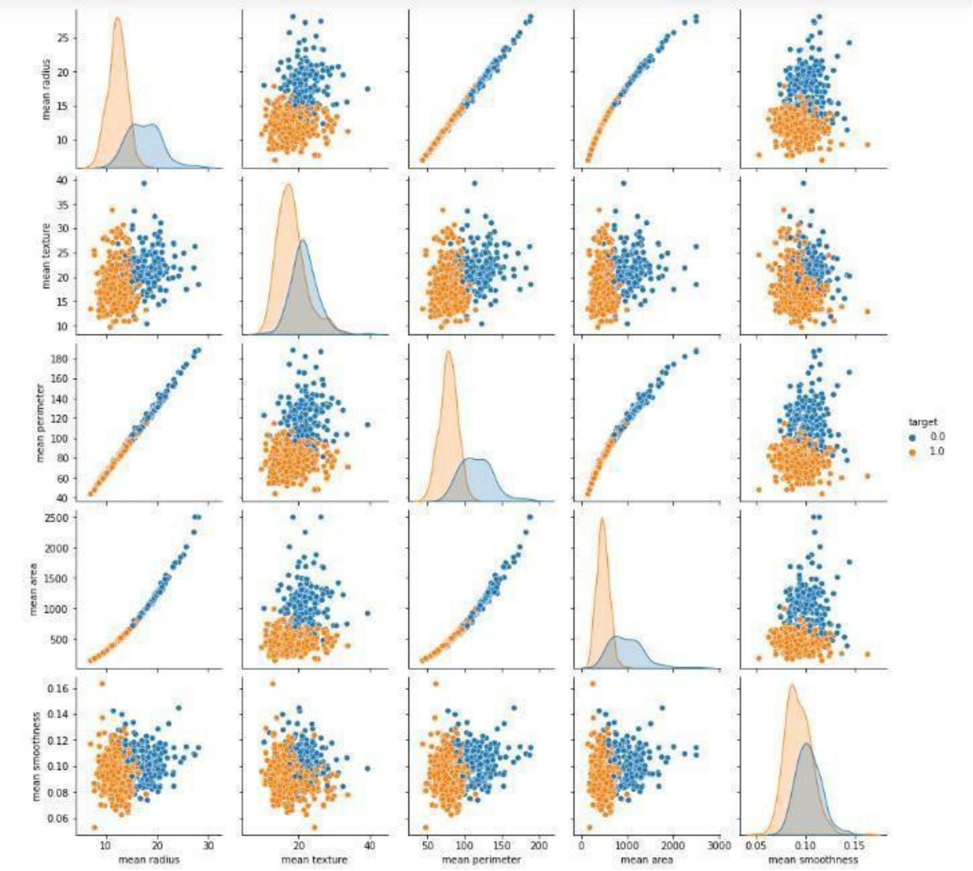


Fig.3

- Data Splitting:

The system must be able to split data appropriately, in our case it should be able to split data into 75% data as trained data and 25% data as test data.

- Data Scaling:

The system must be able to scale the data using Standard scaler technique to ensure that all attributes have a similar scale or range.

- Machine learning model building:

The system must be able to train a classification model using supervised machine learning techniques like Logistic Regression, Support Vector Classifier, KNN Classifier, Naive Bayes, Decision trees, Extract Tree Classifier, Random Forest, Gradient Boost, Ada boost Classifier, XG Boost, Stacking Classifier and Voting Classifier to predict if the tumor is benign or malignant

- Parameter tuning:

The system must be able to tune the hyperparameters of the trained model using GridSearchCV technique to improve the performance of the model. (fig.4)

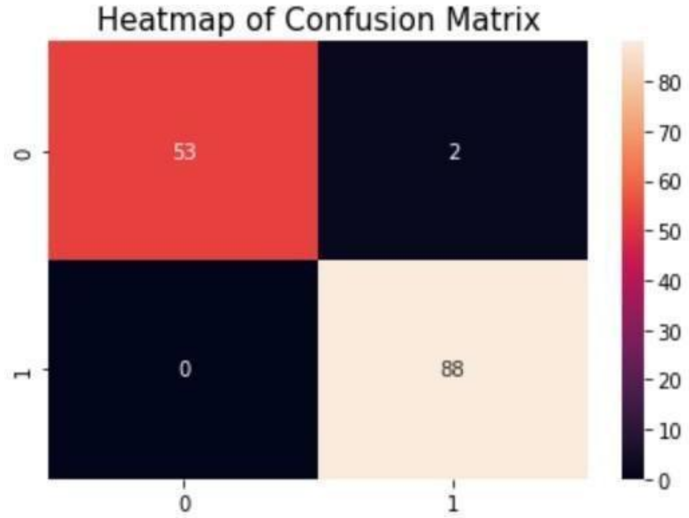


Fig.4

- Metric Analysis:

The system must be able to evaluate the performance of the trained model using metrics such as confusion matrix, classification report, precision score, recall score, f1 score, accuracy score, and ROC curve (fig.5).

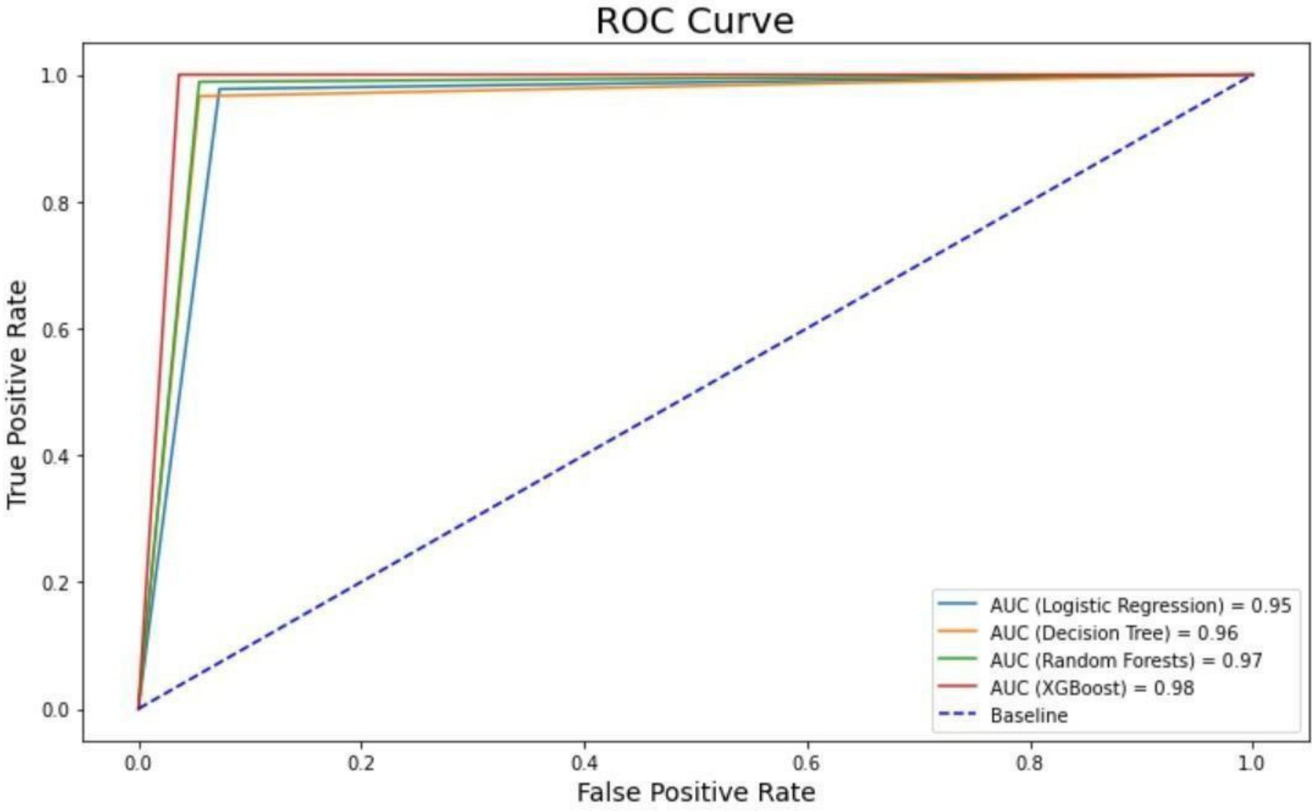


Fig.5

Non-Functional Requirements:

The non-functional requirements of the Breast cancer cell classification using Supervised Learning project are as follows:

3.1 Performance:

The system must be able to process large amounts of data efficiently and provide predictions in a timely manner.

3.2 Usability:

The system must be user-friendly and easy to use, even for users who do not have a background in machine learning.

3.3 Reliability:

The system must be reliable and provide accurate predictions consistently.

3.4 Security:

The system must be secure and protect the data from unauthorized access or disclosure.

3.5 Scalability:

The system must be scalable and able to handle an increasing amount of data and users.

Constraints:

The constraints of the Breast cancer cell classification using Supervised Learning project areas follows:

3.1. Hardware Requirements:

The system requires a computer with a minimum of 4GB RAM and a processor with a speed of at least 2.5GHz.

3.2. Software Requirements:

The system requires Python 3.7 or above, along with the following libraries:

4 Pandas
5 Matplotlib
6 Numpy
7 Seaborn
8 Scikit-learn
9 Shap-hypetune

Chapter 4

Implementation

We have implemented multiple machine learning algorithms through scikit-learn library to find the highest accuracy and build the most efficient model . After finding the highest efficiency using XGboost algorithm we further performed hyper-paramter tuning and feature selection to find the best parameters , for this we first used scikit-learn's Random search and Grid Search. We also tried to combine both hyperparameter tuning and feature selection into one using shap-hypetune which couldn't give satisfactory results. We then performed ensembling of classification models using stacking and voting . Using an ensemble of logistic regression , gradient boost and random forest classifier in soft voting resulted in obtaining similar overall results to XGboost model both showing highest accuracy among all models.

4.1 Methodology

The initial phase of the project involves importing the essential libraries for analyzing and visualizing data. The libraries that are commonly utilized include pandas, numpy, matplotlib, seaborn, and scikit-learn, which are imported to execute different tasks.

Once all the required libraries have been imported, the data is loaded into memory using pandas and explored using various functions such as head(), tail(), info(), describe(), and value_counts(). This exploration step is crucial in comprehending the data's size, variable types, value ranges, and the presence of any missing or inconsistent values.To visualize the data, we utilized seaborn's pairplot, countplot, heatmap, and barplot to uncover details about the data and their interrelationships.

The data was split into a 25/75 ratio for training and testing purposes. Feature scaling was implemented using StandardScaler to normalize the feature range, particularly mean_area and worst_area, which had the highest range among all features and could have had an adverse impact on the overall outcome if left unaddressed.

Scikit-learn was used to implement the following algorithms :
Logistic Regression
Support Vector Classifier
KNearest Neighbour
Naive Bayes
Random forest
Extratree
Decision tree
Gradient boost
Adaboost
Xgboost

The performance of the models are assessed throughout the using a variety of techniques, including as accuracy, recall, precision from classification report and cross-validation.The amount of false positives, false negatives, true positives, and true negatives are represented in the confusion matrix, which is used to show how well the model is doing.After all assessments Xgboost algorithm gave the highest efficiency of 98.6 percentage.

For improving its efficiency we performed hyperparamter tuning using scikit-learn's Grid Search and Randomized search . After using both searching method we used best estimator function to find the best parameters and refit the model to test with new hyperparameters . Both showed similar results to xgboost's initial efficiency.Then boost search was implemented using Shap-hypetune's library which couldn’t show good results.

Then to find more models for classification we used ensembling through different combinations of stacking and voting classifiers. Stacking makes it possible to train different models to address similar issues and then create a new model with superior performance based on their combined output. The voting classifier basically combines the results from every classifier input into it and determines the output class according to the highest majority of votes cast. We obtained similar findings to the xgboost approach during soft voting of an ensemble of logistic regression, random forest and gradient boost classifiers. Finally, the most efficient model was saved using pickle.

4.2 Testing

The project's testing and validation strategy aims to assess how well the machine learning model performs when applied to fresh, unexplored data. This evaluation's objective is to assess the model's precision, accuracy, and other performance indicators that may be used to gauge how well the model predicts the outcome of tumour classification, whether benign or malignant.

Validation and testing are the two main phases of the testing method. A fraction of the data that wasn't used during training is used to evaluate the model's performance during the validation phase. This is done to ensure that the model can generalise effectively to new data and does not overfit the training set of data.

Using a function from the scikit-learn library, the data was divided into training and testing portions at a ratio of 75:25 for the validation phase. The validation set is used to evaluate the model's performance after it has been trained using the training set. The model is then enhanced for accuracy utilizing randomised search and grid search and boost seach methods for hyperparameter tuning

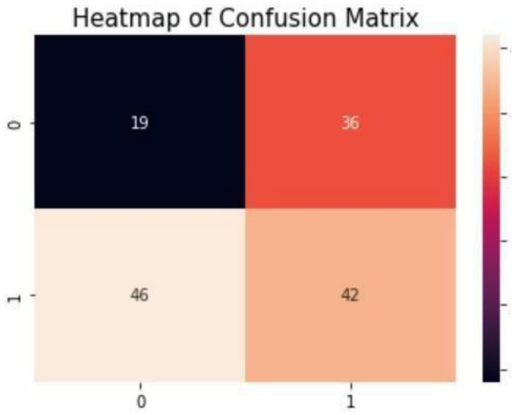
The model is assessed on the testing set once it has been optimised. The testing set is a whole separate collection of information that the model has not seen during training or validation. This ensures that the model may be used to real-world circumstances and can generalise successfully to new, fresh data.
The performance of the model is assessed throughout the testing phase using a variety of techniques, including as accuracy, recall, and precision.The amount of false positives, false negatives, true positives, and true negatives are represented in the confusion matrix, which is used to show how well the model is doing.

The trained model is saved to a file for later use as the last stage in the testing and verification strategy. This was accomplished by utilizing the Python pickle package, which enables serialization and saving of the model on disk. The stored model may then be imported and used in new data to produce predictions.

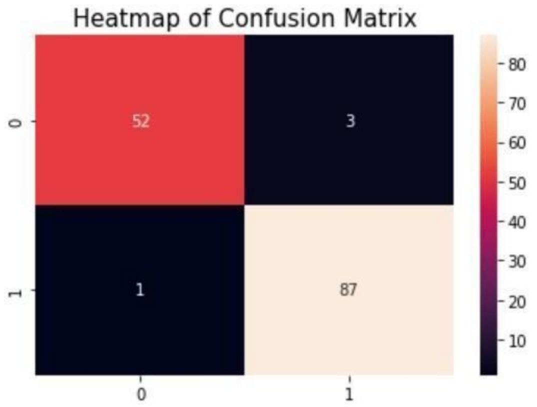
4.3 Result Analysis

The performance of all the used algorithma are evaluated using various metrics including confusion matrix, classification report, and accuracy_score.
It was observed that only a few algorithm i.e SVC , Naive Bayes ,Adaboost and Xgboost could perform well with scaled data as compared to non-scaled data .Confusion matrix report of each algorithm tested with scaled data in[Table1]
Fig.Table 1

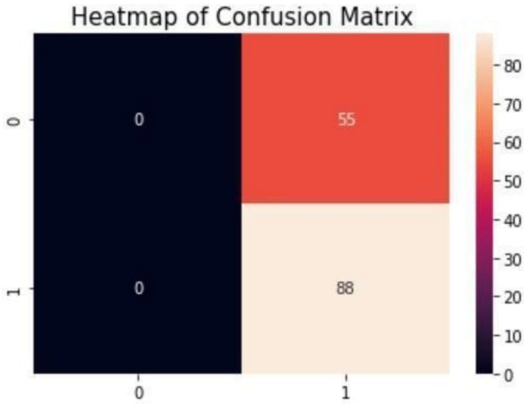
Logistic Regression



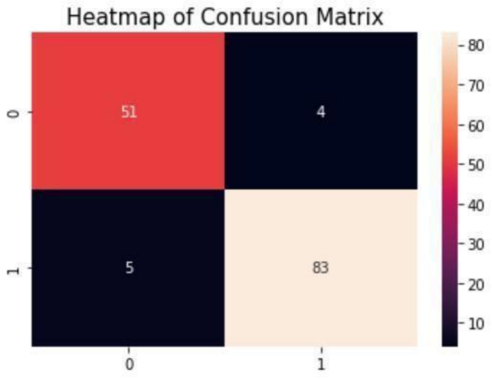
SVC



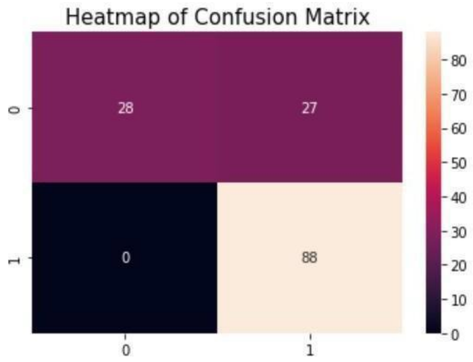
KNN



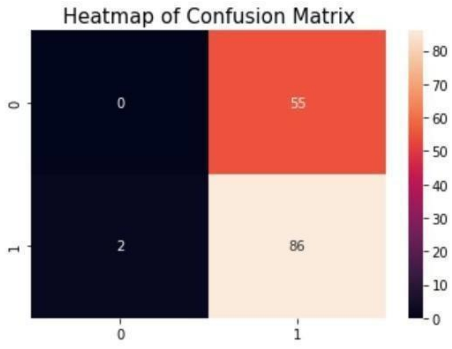
Naive Bayes



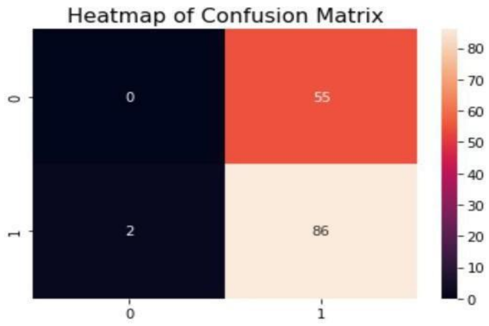
Random forest



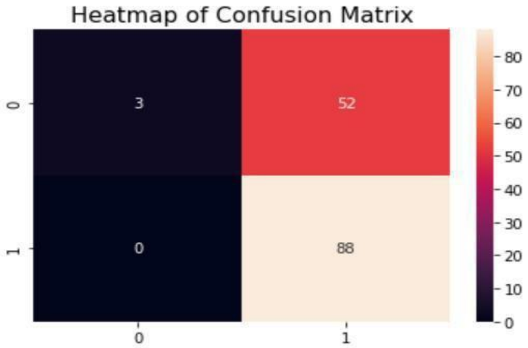
Extra tree



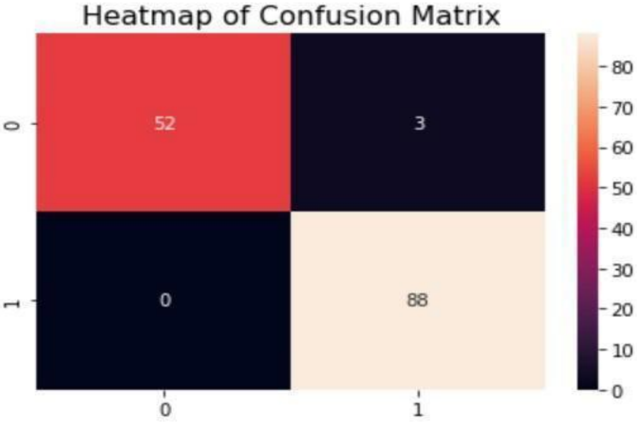
Decision tree



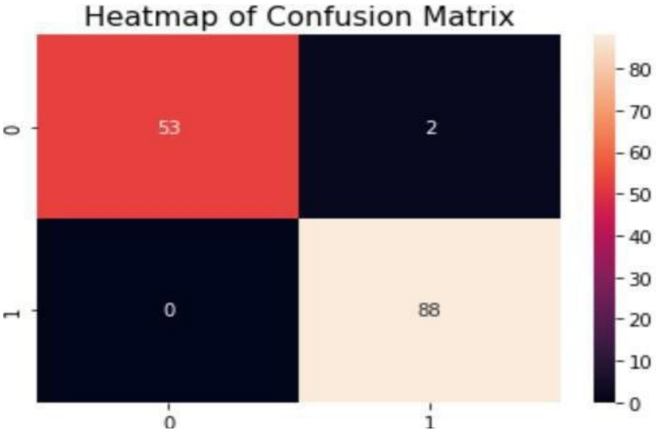
Gradient boost

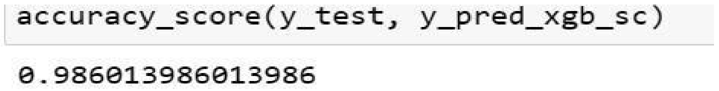


Adaboost



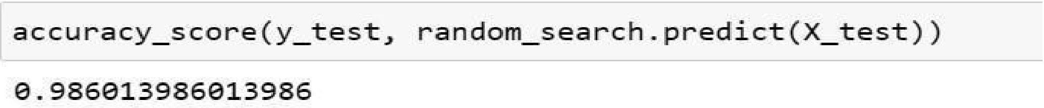
Xgboost



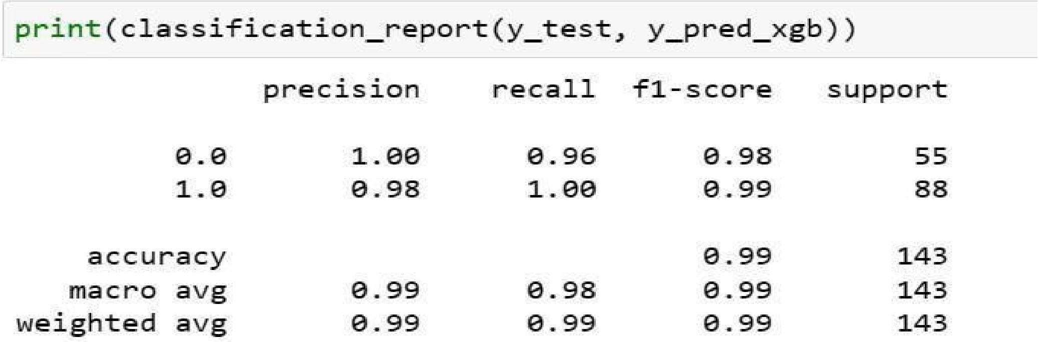


```
accuracy_score(y_test, y_pred_xgb_sc)

0.986013986013986
```

Figure. Screenshots



```
accuracy_score(y_test, random_search.predict(X_test))

0.986013986013986
```

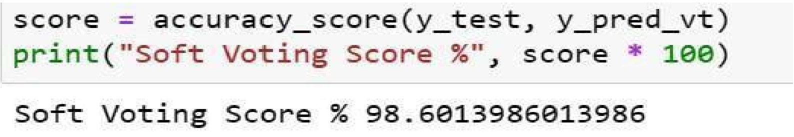


```
print(classification_report(y_test, y_pred_xgb))

              precision    recall  f1-score   support

         0.0       1.00      0.96      0.98        55
         1.0       0.98      1.00      0.99        88

    accuracy                           0.99       143
   macro avg       0.99      0.98      0.99       143
weighted avg       0.99      0.99      0.99       143
```

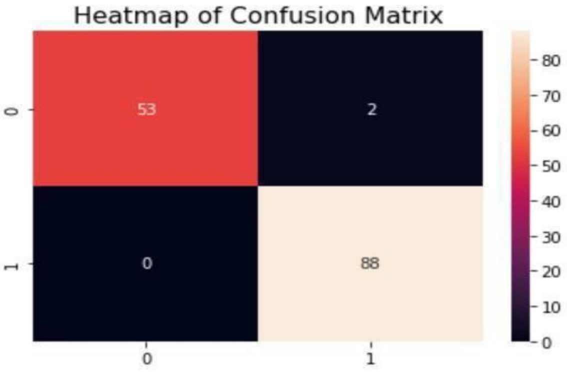
Hence , it was clear that xgboost performed the best among all algorithms in both standard and scaled data as shown in [Screenshots]

After performing stacking and voting we obtained a similar result to xgboost during soft voting of an ensemble of logistic regressor, random forest and gradient boost models [Screenshot2] .

Figure.
Screenshot2



```
score = accuracy_score(y_test, y_pred_vt)
print("Soft Voting Score %", score * 100)

Soft Voting Score % 98.6013986013986
```



Comparison of Model accuracy with non scaled data shown in [Table 2]:

Figure . Table 2

Logistic Regression	0.958
SVC	0.951
KNN	0.951
Naive Bayes	0.951
Decision tree	0.916
Random forest	0.972
extratree	0.958
gradientboost	0.979
adaboost	0.923
Xgboost	0.986

As mentioned earlier , there is huge difference in results with scaled data . Only SVC , Naive Bayes, Adaboost and Xgboost were able to perform well with scaled data.

Comparison of Model accuracy results with scaled data shown in [Table3]
Figure. Table 3

Logistic Regression	0.426
SVC	0.972
KNN	0.615
Naive Bayes	0.937
Decision tree	0.769
Random forest	0.81
extratree	0.601
Gradient boost	0.636
Adaboost	0.923
Xgboost	0.986

4.4 Quality Assurance

Quality assurance is an important aspect of any project as it ensures that the final product meets the required standards and expectations. In the context of the diabetes detection project using supervised learning, it refers to the measures taken to ensure that the machine learning model isaccurate, reliable, and consistent in its predictions.

One of the key steps in quality assurance is data exploration and engineering. This includes understanding the characteristics of the data, identifying any missing or inconsistent values, and transforming the features of the dataset to improve the performance of the machine learning algorithms. In this project, this was done by exploring the datasetusing functions such as head(), tail(), info(), describe(), isnull().sum(),andvalue_counts(). Visualizations such as pairplot, countplot,correlation matrix and heat maps were also used to identify patterns and outliers in the data. Scaling of the data was done using the StandardScaler function to ensure that the features of the dataset have a similar scale or range.

Most important part is selection of machine learning algorithms and hyperparameters. This involves testing and evaluating different algorithms and their corresponding hyperparameters to determine the bestcombination that yields the most accurate predictions. In the project, several algorithms were tested, including Logistic Regression, Decision Tree, Random Forest , extratree, SVC, Gradient boost, Adabosst and XGboost using both randomized and grid search cross- validation techniques from scikit-learn and lastly boostsearch
from shap-hypetune.After stacking and voting an ensemble of logisticregressor,random forest and gradient boost models was also obtained which showed similar results to xgboost.

Quality assurance also includes saving the model and using it for future predictions. In the breast cancer classification project, the best model wassaved using the pickle library, which allows for easy loading and re-use of the model for future predictions.

Chapter 5

Standards Adopted

The project mentioned above involves the use of supervised learning techniques to classify the tumor. In this project, various design standards have been adopted to ensure the quality of the project.

5.1. Design Standards

The usage of a well-defined data exploration approach is one of the design criteria employed in this project. Before creating a model, it is necessary to examine the data in order to comprehend its characteristics, such as its size, variable types, keys and values, target and range of values. This is accomplished through the use of procedures such as head(), tail(), info(), describe(), and value_counts(). Data visualization techniques such as pair plots, bar plots, counterplots, scatter plots, heatmaps, and box plots are also used to find trends and outliers in data.

Another design standard adopted in this project is the use of data scaling techniques. Scaling is a data preprocessing technique that involves transforming the features of a dataset so that they have a similar scale or range. The goal of scaling is to improve the performance of the machine learning algorithms thatare sensitive to the scale of the features, such as distance-based algorithms like K- Nearest Neighbors (KNN). In this project, the standard scaling technique is used to transform the features of the dataset.

Feature selection is another important design standard adopted in this project. Feature selection is a machine learning technique that involves selecting the most relevant features from a dataset to improve the accuracy of a model. The objective of feature selection is to decrease the dimensionality of the data by removing features that are irrelevant or redundant, while retaining those that are most important for the prediction task. In this project, an inbuilt-feature-selection technique is used to automatically select the most relevant features from the dataset

In addition to these design standards, the project also adopts the use of various machine learning algorithms for modelling, such as Logistic Regression, Support Vector Classifier, KNN Classifier, Naive Bayes, Decision trees, Extract Tree Classifier, Random Forest, Gradient Boost, Ada boost Classifier, XG Boost, Stacking Classifier and Voting Classifier to predict if the tumor is benign or malignant. The performance of these models is evaluated using various metrics such as precision, recall, F1-score, and accuracy.

Finally, the project adopts the use of hyper tuning to fine-tune the parameters of the machine learning algorithms. Hyperparameter tuning is the process of searching for the best combination of hyperparameters that maximize the performance of the model. In this project, GridSearchCV is used to perform hyperparameter tuning.

Overall, the project adopts various design standards to ensure the quality of the project. These design standards include well-defined data exploration processes, the use of data scaling techniques, the adoption of feature selection techniques, the use of various machine learning algorithms, and the use of hyperparameter tuning techniques.

5.2. Coding Standards

Indentation and spacing: The code has been properly indented and spaced for better readability. Use of a consistent number of spaces or tabs to indent each block of code has been done. This helps in visualizing the hierarchy of the code and make it more readable.

Naming conventions: The variables, functions, and classes have been named in a consistent and meaningful manner. Use of descriptive names has been done for variables and functions that accurately describe their purpose. Using of abbreviations and single-letter names has been avoided.

Comments and documentation: Comments have been added whenever required to explain the logic behind the code. These are not a repetition of the code but rather provide additional information that is not obvious from the code itself. Additionally, documentation of the functions is done using docstrings to provide information about the parameters, return type, and purpose of the function.

Consistency in code style: The coding style is consistent throughout the project. For example, if one function is using single quotes for strings, then all the other functions also use the same style.

Use of libraries: Reinventing the wheel by using standard libraries instead of writing custom code for common functionalities has been avoided. This not only saves time but also improves the readability of the code.

Error handling: The code handles the errors gracefully, i.e. the code catches exceptions and provides a meaningful error message to the user. Additionally, assertions are used to check the validity of the inputs and outputs.

Function length: The length of the functions is kept to a reasonable limit. Large functions can be difficult to read and understand. It is recommended to keep the length of a function to a maximum of 30 lines of code.

Code readability: The code has been written such that it is easily readable and understandable by other developers. Meaningful variable and function names have been used, sufficient comments have been provided and the code has been broken into logical blocks for better readability.

By following these coding standards, the code in the Breast cancer cell classification using Supervised Learning project has become more consistent, maintainable, and scalable. Additionally, it improves quality of the code and makes it easier to understand for other developers. In conclusion, coding standards play a crucial role in software engineering, and following them can help to create better code.

5.3. Testing Standards

The testing standards used in this project include data splitting, cross-validation, and hyperparameter tuning. These techniques are commonly used in supervised learning to evaluate the performance of a model and optimize its parameters.

Data splitting- It is the process of splitting the dataset into two subsets: the training dataset and the testing dataset. The training dataset is used to train the machine learning model, while the testing dataset is used to check its performance on new, unseen data. In this project, the train-test split function from scikit-learn has been used to split the dataset randomly into a training set and a testing set in the ratio of 75:25.

Cross-validation-It is a method used to evaluate the performance of a model by dividing the dataset into k-folds and using each fold as both the training dataset and the testing dataset. In this project, the k-fold cross- validation technique has been used to evaluate the performance of various machine learning models. The best performing model is then selected for further analysis.

Hyperparameter tuning- It is the process of optimizing the parameters of a machine learning model to improve its performance on new data. In this project, hyperparameter tuning has been performed using the Randomized Search CV and GridSearchCV functions from scikit-learn. These functions allow for the systematic exploration of the hyperparameter space of a model to find the combination of hyperparameters that gives the best accuracy on the validation set.

The trained model has been them evaluated by using various parameters like accuracy, precision, recall, F1-score, and the ROC curve. These parameters provide a quantitative measure of the performance of the model and help in identifying areas where the model can be improved. For instance, accuracy is the proportion of correctly classified instances, while precision is the proportion of true positives among all predicted positives, recall is the proportion of true positives among all actual positives, and F1-score is the harmonic mean of precision and recall.

In addition to these testing standards, it has been ensured that the code is well-documented, maintainable, and follows best coding practices. This includes writing clear and concise comments, using meaningful variable names, adhering to PEP-8 coding standards, and conducting code reviews to identify and fix potential issues.

In conclusion, testing standards are essential for ensuring that the trained machine learning model performs well on new, unseen data. The use of data splitting, cross-validation, and hyperparameter tuning, as well as the evaluation of performance metrics, ensures that the model is robust and reliable. Furthermore, adhering to best coding practices and conducting code reviews ensures that the code is maintainable and extensible.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Building more precise and computationally efficient classifiers for medical applications is an essential requirement in machine learning. Therefore, in this project to classify breast cancer, we used scikit-learn's machine learning classifier algorithms on the Wisconsin Breast Cancer datasets in an effort to asess their effectiveness. To find the most efficient classification method, we compared algorithms based on a variety of metrics, including accuracy score, recall, f1-score, and confusion matrix. The results showed that the Xgboost model and a Voting classifier model (ensemble of logistic regressor, random forest, and gradient boost) performed the best, with an accuracy rate of 98.6 percent in identifying the type of breast cancer and whether it was malignant or not.

6.2 Future Scope

The project can be improved and extended in various ways such as by using more advanced techniques to create new features. For example, interaction features can be created by combining two or more features to capture the relationship between them and by using deep learning models to improve the performance of the model.The model can be deployed in a web application for predicting the type of tumor quickly and then refer to further medical testing.

References

[1]Nayan Kumar Sinha, Menuka Khulal, Manzil Gurung, Arvind Lal, Developing A Web based System for Breast Cancer Prediction using XGboost Classifier,International Research Journal of Engineering and Technology (IRJET) Volume 09, Issue 06 ,June 2020.

[2]R. Chtihrakkannan, P. Kavitha, T. Mangayarkarasi, R. Karthikeyan, “Breast Cancer Detection using Machine Learning”, International Journal of Innovative Technology and Exploring Engineering (IJITEE) Volume-8 Issue-11, September 2019.

[3]Priyanka Gupta, Prof. Shalini L, “Analysis of Machine Learning Techniques for Breast Cancer Prediction”, International Journal Of Engineering And Computer Science (IJECS) Volume 7 Issue 5 ,May 2018

[4]Madhuri Gupta, Bharat Gupta, “A Comparative Study of Breast Cancer Diagnosis Using Supervised Machine Learning Techniques”, 978-1-5386-3452-3/18/31.00 ,2018 IEEE

[5]Varsha J. Gaikwad, “Detection of Breast Cancer in Mammogram using Support Vector Machine”, International Journal of Scientific Engineering and Research (IJSER) Volume 3 Issue 2, February 2015.

[6](2023) The IEEE website. [Online]. Available: http://www.ieee.org/

INDIVIDUAL CONTRIBUTION REPORT:

BREAST-CANCER DETECTION USING MACHINE LEARNING CLASSIFIER

PIYUSH MOHANTY 2005949
SANJANA SUBUDHI 2005961
ANKITA SAMAL 2005922

Abstract: In this project we developed a diagnosis model for which we have done the comparative study of the supervised machine learning classifiers to get to know which classifier is giving the best accuracy using scikit-learn's library with stacking and voting classifiers on UCI ML Wisconsin breast cancer database .

Individual contribution and findings:
Piyush Mohanty: contributed to the data modeling phase of the project by selecting appropriate machine learning algorithms, such as random forest , support vector classifier,Xgboost and training the model on the breast cancer dataset . Also performed hyperparameter tuning.
Sanjana Subudhi:contributed to finding the dataset and required libraries, data visualization phase of the project by visualizing the breast cancer data using techniques such as pair plots,scatter plots,barplots, and implemented logistic regression model in modeling phase.
Ankita Samal: contributed to the data exploration phase of the project by exploring the phishing dataset to identify any potential data quality issues, such as missing data or outliers. Implemented feature scaling and used heatmap for data visualization.

Individual contribution to project report preparation:
Ankita Samal: Prepared chapter 1 and chapter 2
Sanjana Subudhi: Prepared chapter 3 and chapter 5
Piyush Mohanty: Prepared chapter 4 and chapter 6

Full Signature of Supervisor: Full signature of the student:
……………………………. ……………………………..

BREAST-CANCER DETECTION USING MACHINE LEARNING CLASSIFIER

ORIGINALITY REPORT

22%	13%	8%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijert.org Internet Source	5%
2	www.researchgate.net Internet Source	1%
3	Submitted to South Bank University Student Paper	1%
4	Submitted to American University in the Emirates Student Paper	1%
5	Submitted to Letterkenny Institute of Technology Student Paper	1%
6	Submitted to Midlands State University Student Paper	1%
7	Avnish Goel, Apoorv Kashyap, B. Devesha Reddy, Rochak Kaushik, S Nagasundari, Prasad B Honnavali. "Detection of VPN Network Traffic", 2022 IEEE Delhi Section Conference (DELCON), 2022 Publication	1%

8	Submitted to Broward Community College Student Paper	1%
9	scholar.dsu.edu Internet Source	1%
10	Submitted to De Montfort University Student Paper	1%
11	Submitted to University of Birmingham Student Paper	1%
12	Submitted to University of Wales Institute, Cardiff Student Paper	1%
13	Submitted to SVKM International School Student Paper	<1%
14	Submitted to Roehampton University Student Paper	<1%
15	Submitted to University of Hertfordshire Student Paper	<1%
16	Submitted to BPP College of Professional Studies Limited Student Paper	<1%
17	Submitted to Queen Mary and Westfield College Student Paper	<1%
18	Submitted to International Islamic University Malaysia	<1%

	Student Paper	
19	Submitted to UCSI University Student Paper	<1%
20	Submitted to Kennesaw State University Student Paper	<1%
21	Mohammed Mijanur Rahman, Asikur Rahman, Swarnali Akter, Sumiea Akter Pinky. "Hyperparameter Tuning Based Machine Learning Classifier for Breast Cancer Prediction", Journal of Computer and Communications, 2023 Publication	<1%
22	Submitted to Westcliff University Student Paper	<1%
23	Submitted to Saint John's Preparatory School Student Paper	<1%
24	www.ncbi.nlm.nih.gov Internet Source	<1%
25	Submitted to Liverpool John Moores University Student Paper	<1%
26	Submitted to University of Greenwich Student Paper	<1%
27	www.slideshare.net Internet Source	<1%

28	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1%
29	Chengang Lyu, Yuxin Chen, Zhijuan Chen, Yuheng Liu, Zengguang Wang. "Automatic Epilepsy Detection Based on Generalized Convolutional Prototype Learning", Measurement, 2021 Publication	<1%
30	Afshin Jamshidi, Jean-Pierre Pelletier, Johanne Martel-Pelletier. "Machine-learning-based patient-specific prediction models for knee osteoarthritis", Nature Reviews Rheumatology, 2018 Publication	<1%
31	Submitted to Erasmus University of Rotterdam Student Paper	<1%
32	Submitted to Middlesex University Student Paper	<1%
33	Submitted to Southampton Solent University Student Paper	<1%
34	Zhaosong Fang, Huiyu He, Yudong Mao, Xiwen Feng, Zhimin Zheng, Zhisheng Guo. "Investigating an accurate method for measuring the outdoor mean radiation	<1%

	temperature", International Journalof Thermal Sciences, 2023 Publication	
35	Submitted to Coventry University Student Paper	<1%
36	Ting Sun. "Chapter 43 Diagnosis of Hepatitis C Patients via Machine Learning Approach: XGBoost and Isolation Forest", Springer Science and Business Media LLC, 2023 Publication	<1%
37	Submitted to University of Stirling Student Paper	<1%
38	blog.codinginvaders.com Internet Source	<1%
39	tudr.thapar.edu:8080 Internet Source	<1%
40	Submitted to SAMRAT ASHOK TECHNOLOGICAL INSTITUTE VIDISHA M.P Student Paper	<1%
41	ethesis.nitrkl.ac.in Internet Source	<1%
42	Anuj Kinge, Yash Oswal, Tejas Khangal, Nilima Kulkarni, Priyanka Jha. "Chapter 12 Comparative Study on Different Classification Models for Customer Churn Problem",	<1%

Springer Science and Business Media LLC, 2022
Publication

43 Huiming Lu, Jiazheng Wu, Yingjun Ruan, Fanyue Qian, Hua Meng, Yuan Gao, Tingting Xu. "A multi-source transfer learning model based on LSTM and domain adaptation for building energy prediction", International Journal of Electrical Power & Energy Systems, 2023
Publication
<1%

44 dokumen.pub
Internet Source
<1%

45 www.mdpi.com
Internet Source
<1%

46 Chahid Ahabchane, Martin Trépanier, André Langevin. "Street-segment-based salt and abrasive prediction for winter maintenance using machine learning and GIS", Transactions in GIS, 2018
Publication
<1%

47 Tazar Hussain, Alfie Beard, Liming Chen, Chris Nugent, Jun Liu, Adrian Moore. "From Machine Learning Based Intrusion Detection to Cost Sensitive Intrusion Response", 2022 6th International Conference onCryptography, Security and Privacy (CSP), 2022
Publication
<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		