BREAST CANCER PREDICTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

ANITHA R	**[513220104302]**
YAMUNA S	**[513220104010]**
KIRUTHIKA S	**[513220104003]**

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

THIRUMALAI ENGINEERING COLLEGE, KANCHPURAM

ANNA UNIVERSITY: CHENNAI – 600 025

MAY 2024

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certificate that this project report titled "**BREAST CANCER PREDICTION USING MACHINE LEARNING**" is the bonafide work of "**ANITHA. R [513220104302], YAMUNA. S [513220104010], KIRUTHIKA. S [513220104003]**" who Carried out the project work under my supervision.

SIGNATURE OF HOD

V. VIJAYABHASKAR M.C.A., M.Tech.,

HEAD OF THE DEPARTMENT,

Associate Professor,

Department of CSE,

Thirumalai Engineering College,

Kanchipuram – 631 551.

SIGNATURE OF SUPERVISOR

S. VENMAL M.Tech.,

SUPERVISOR,

Assistant Professor,

Department of CSE,

Thirumalai Engineering College,

Kanchipuram – 631 551.

Submitted for the Project Viva Voce held on ________________

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

I profoundly thank our **Chairman and trust members of Kanchipuram Educational Trust** for providing adequate facilities.

I would like to express my hearty thanks to our respectable Principal. Incharge **Mr.T.MohanRaj M.Tech.,** for allowing us to have the extensive use of our colleges facilities to our colleges facilities to have precious advice regarding the project.

I extend our thanks to Associate Professor **Mr.V.VIJAYABHASKAR M.C.A., M.Phil., M.Tech., Head of the Department, Information Technology** for this precious advice regarding the project.

I would like to express my deep and unbounded gratefulness to my project Guide **Mrs.S.VENMAL M.Tech.,** Department of Information Technology, for his valuable guidance and encouragement throughout the project. He has been a constant source of inspiration and has provided the precious suggestion throughout this project.

I thank all facilities and supporting staff for the help they extended in completing this project. I also express my sincere thanks to our parents, and all my friends for their continuous support.

TABLE OF CONTENTS

ABSTRACT

Breast cancer affects the majority of women worldwide, and it is the second most common cause of death among women. However, if cancer is detected early and treated properly, it is possible to be cured of the condition. Early detection of breast cancer can dramatically improve the prognosis and chances of survival by allowing patients to receive timely clinical therapy. Furthermore, precise benign tumour classification can help patients avoid unneeded treatment. This paper study uses Convolution Neural Networks for Image dataset and K-Nearest Neighbour (KNN), Decision Tree (CART), Support Vector Machine (SVM), and Naïve Bayes for numerical dataset, whose features are obtained from digitised image of breast mass, as to forecast and analyse cancer databases in order to improve accuracy. The dataset will be analysed, evaluated, and model is trained as part of the process. Finally, both image and numerical test data will be used for prediction.

Keywords: IDC (Invasive Ductal Carcinoma), FNA (Fine Needle Aspirate), Breast cancer prediction, Classifier algorithms, CNN (Convolutional neural network).

LIST OF ABBREVATIONS

ACRONYM	ABBREVATIONS
WHO	WORLD HEALTH ORGANIZATION
NLP	NATURAL LANGUAGE PROCESS
DS	DATA SCIENCE
EDA	EXPLORATORY DATA ANALYSIS
CSV	COMMA SEPERATE VALUE
KNN	K-NEAREST NEIGHBOR
ROC	RECEIVER OPERATING CHARACTER
API	APPLICATION PROGRAMMABLE INTERFACE
NOSQL	NOT ONLY SQL
VOC	VARIENCES OF CONCERN
SVM	SUPPORT VECTOR MECHINE
BDV	BIG DATA VISUALIZATION

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Breast cancer has become the most recurrent type of health issue among women especially for women in middle age. Early detection of breast cancer can help women cure this disease and death rate can be reduced. In the present-day scenario, to observe breast cancer mammograms are used and they are known be the most effective scanning technique.

In this paper the detection of cancer cells is done by machine learning technique. Image processing is a method to convert an image into digital form and perform some operations on it, to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image.

Usually, Image Processing system includes treating images as two-dimensional signals while applying already set signal processing methods to them.

Importing the image with optical scanner or by digital photography. Analyzing and manipulating the image which includes data compression and image enhancement and spotting patterns that are not to human eyes like satellite photographs.

Output is the last stage in which result can be altered image or report that is based on image analysis. Digital Processing techniques help in manipulation of the digital images by using computers.

Digital Image is composed of a finite number of elements, each of which elements have a particular value at a particular location. These elements are referred to as picture elements, image elements and pixels.

A Pixel is most widely used to denote the elements of a Digital Image. As raw data from imaging sensors from satellite platform contains deficiencies. To get over such flaws and to get originality of information, it has to undergo various phases of processing.

The 7 three general phases that all types of data have to undergo while using digital technique are Preprocessing, enhancement and display, information extraction. The first step is to analyses the images and represent them. The method of image representation will fix fundamental problems that are the ineffectiveness of capturing textural information and the weak capacity for classification of features that result in low performance of retrieval.

In Content-based Image Retrieval, similarity estimation is a part of the primary task and has a greater effect on the accuracy of retrieval and time of retrieval. The project aims to solve problems such as "Which similarity measure is appropriate for particular feature type and how to reduce the similarity calculation computation? "And" The texture function is more representative and discriminatory to describe the mammogram of the given query?

In this project we use classification techniques such as Fuzzy C-means and KNN to employ feature extraction. The FCM clustering is used for image segmentation once the mammographic image was collected. Data point in this system be held by to various clusters with changing degrees of membership and is based on objective criteria.

The segmented area is rigorously examined using Multi-level Discrete Wavelet Transform to get edge details to which is then used as a feature. PCA is then used for this data to analyze along with GLCM. After performing the analysis, 13 features extracted in the proposed framework and their pixel values in matrix form are stored in database. Machine learning is an application of artificial intelligence that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

The basic premise of machine learning is to build algorithms that can receive input data and use statistical analysis to predict an output while updating outputs as new data becomes available. The process of learning begins with observations or data, such as examples, direct experience, or instruction, to look for patterns in data and make better decisions in the future based on the 8 examples that we provide.

The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly. Supervised algorithms with machine learning skills to provide both input and desired output, in addition to furnishing feedback about the accuracy of predictions during algorithm training.

A classification problem is when the output variable is a category or a group. Using the KNN classifier method, which primarily relies on the shape of the cancer cells in the image, the algorithm classifies the image into Benign, Malignant and Normal once the features are extracted & fully educated. ‘

By performing appropriate morphological operations, the device calculates the appropriate region properties, such as size, Euler number, etc., and displays the detected boundary image along with the tumor area.

1.2 PROBLEM STATEMENT

Breast cancer is one of the most common types of cancer among women worldwide. Early detection and accurate diagnosis are crucial for effective treatment and improved patient outcomes.

Machine learning techniques offer a promising approach for predicting breast cancer based on various clinical and demographic features.

The objective of this project is to develop a machine learning model that can accurately predict the likelihood of breast cancer in patients based on relevant factors such as age, family history, mammography results, and other clinical indicators. The model should be able to distinguish between benign and malignant tumors with high sensitivity and specificity.

1.3 OBJECTIVE OF THE PROJECT

The objective of breast cancer prediction using machine learning is to develop models that can accurately classify whether a given patient has breast cancer or not based on various features such as medical history, genetic information, and imaging data. The ultimate goal is early detection and diagnosis of breast cancer, which can significantly improve treatment outcomes and save lives.

Early Detection: Machine learning models aim to detect breast cancer at an early stage when it's more treatable and chances of survival are higher.

Accuracy: Develop models that can accurately predict breast cancer, minimizing false positives (misclassifying a healthy person as having cancer) and false negatives (misclassifying a person with cancer as healthy).

Personalized Medicine: Tailoring treatment plans based on individual patient characteristics and risk factors identified by machine learning models.

Risk Assessment: Assessing the risk of developing breast cancer for individuals based on various factors such as age, family history, lifestyle, and genetic predisposition.

Feature Selection: Identifying the most relevant features or biomarkers that contribute to breast cancer prediction, which can aid in understanding the underlying mechanisms of the disease.

Improving Healthcare Access: Providing cost-effective and scalable solutions for breast cancer screening, particularly in regions with limited access to healthcare resources.

Decision Support: Assisting healthcare professionals in making informed decisions regarding diagnosis, treatment planning, and patient management.

Overall, the primary aim is to leverage machine learning techniques to enhance the accuracy, efficiency, and accessibility of breast cancer diagnosis and management, ultimately improving patient outcomes and reducing mortality rates.

1.4 SCOPE OF THE PROJECT

The scope of breast cancer prediction using machine learning is broad and encompasses various aspects of research, development, and application. Here are some key areas within the scope

Data Collection and Preprocessing: Gathering diverse datasets containing patient demographics, medical history, genetic information, mammography images, and histopathological reports. Preprocessing involves cleaning, normalizing, and transforming data to make it suitable for machine learning algorithms.

Feature Selection and Engineering: Identifying relevant features or biomarkers that contribute to breast cancer prediction. This may involve extracting features from medical images, genetic sequences, or clinical variables, as well as engineering new features to enhance model performance.

Model Development: Designing, training, and evaluating machine learning models for breast cancer prediction. This includes selecting appropriate algorithms such as logistic regression, support vector machines, random forests, or deep learning architectures like convolutional neural networks (CNNs).

Performance Evaluation: Assessing the performance of machine learning models using metrics such as accuracy, sensitivity, specificity, area under the receiver operating characteristic curve (AUC-ROC), and precision-recall curve. Cross-validation and independent testing on unseen datasets are commonly used to validate model generalization.

Integration with Clinical Workflow: Integrating predictive models into clinical workflows to assist healthcare professionals in decision-making processes. This may involve developing user-friendly interfaces or integrating models into existing electronic health record (EHR) systems.

CHAPTER 2

LITERATURE SURVEY

2.1 PAPER 1

Siyabend Turgut et al., "Microarray Breast Cancer Data Classification Using Machine Learning Methods" [IEEE 2018]

The paper uses microarray breast cancer data for classification of the patients using machine learning methods. In the first case, eight different machine learning algorithms are applied to the dataset and the results of classification were noted. Then in the second case, two different feature selection methods such as Recursive Feature Elimination (RFE) and Randomized Logistic Regression (RLR) were applied on the microarray breast cancer dataset and 50 features were chosen as stop criterion. Again, the same eight machine learning algorithms were applied on the modified dataset. The results of the classifications are compared with each other and with the results of the first case. The methods applied are SVM, KNN, MLP, Decision Trees, Random Forest, Logistic Regression, Ad boost and Gradient Boosting Machines. After applying the two different feature selection methods, SVM gave the best results. MLP is applied using different number of layers and neurons to examine the effect of the number of layers and neurons on the classification accuracy.

2.2 PAPER 2

Varalatchoumy M et al., "Four Novel Approaches for Detection of Region of Interest in Mammograms - A Comparative Study" [ICISS 2017].

The paper compares Four Novel approaches used for detection of Region of Interest in Mammographic images based on database and Real time images. In Approach I histogram equalization and dynamic thresholding techniques were used for preprocessing. Region of Interest (ROI) was partitioned from the preprocessed image by using particle swarm optimization and kmeans clustering methods. In Approach II preprocessing was done using various morphological operations like erosion followed by dilation. For the identification of ROI, a modified approach of 10 watershed segmentation was used. Approach III uses histogram equalization for preprocessing and an advanced level set approach for performing segmentation. Approach IV, which is considered to be the most efficient approach that uses different morphological operations and contrast limited adaptive histogram equalization for image preprocessing. A very novel algorithm was developed for detection of Region of Interest. Approaches I and II were applicable for Mammographic Image Analysis Society (MIAS) database images alone. Approaches III and IV were applicable for MIAS and Real time hospital images. The various graphs presented in the comparative study, clearly depicts that the novel approach that used a novel algorithm for detection of ROI is proved to be the most efficient, accurate and highly reliable approach that can be used by radiologists to detect tumors in MRM images.

2.3 PAPER 3

Ammu P K et al., "Review on Feature Selection Techniques of DNA Microarray Data" [IJCA 2013]

This paper reviews few major feature selection techniques employed in microarray data and points out the merits and demerits of various approaches. Feature selection from DNA microarray data is one of the most important procedures in bioinformatics. Biogeography Based Optimization (BBO) is an optimization algorithm which works on the basis of migration of species between different habitats and the process of mutation. Particle Swarm Optimization (PSO) is an algorithm which works on the basis of movement of particles in a search space. Redundancy based feature selection approaches can be used to remove redundant genes from the selected genes as the resultant gene set can achieve a better representation of the target class. A two-stage hybrid filter wrapper method where, in the first stage a subset of the original feature set is obtained by applying information gain as the filtering criteria. In the second stage the genetic algorithm is applied to the set of filtered genes. Gene selection based on dependency of features where the features are classified as 11 independent, half dependent and dependent features. Independent features are those features that doesn’t depend on any other features. Half dependent features are more relevant in correlation with other features and dependent features are fully dependent on other features.

2.4 PAPER 4

Bing Lan Li et al., “Integrating spatial fuzzy clustering with level set methods for automated medical image segmentation” [ELSEVIER 2010].

A new Fuzzy Level Set algorithm is proposed in this paper to facilitate automated medical image segmentation. It can directly evolve from the initial segmentation by spatial fuzzy clustering where centroid and the scope of each subclass are estimated adaptively in order to minimize a pre-defined cost function. The controlling parameters of Level Set evolution are also estimated from the results of fuzzy clustering. The level set methods utilize dynamic variational boundaries for image segmentation. The new Fuzzy Level Set algorithm automates the initialization and parameter configuration of the level set segmentation, using spatial fuzzy clustering. It employs a Fuzzy-C means (FCM) with spatial restrictions to determine the approximate contours of interest in a medical image. Moreover, the Fuzzy Level Set algorithm is enhanced with locally regularized evolution. Such improvements facilitate level set manipulation and lead to more robust segmentation. Performance evaluation of the proposed algorithm was carried on medical images from different modalities. The results confirm its effectiveness for image segmentation [6].

2.5 PAPER 5

Bayrak, E.A.; Kırcı, P.; Ensari, T. Comparison of machine learning methods for breast cancer diagnosis. In Proceedings of the 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), Istanbul, Turkey, 24–26 April 2019.

Cancer is the common problem for all people in the world with all types. Particularly, Breast Cancer is the most frequent disease as a cancer type for women. Therefore, any development for diagnosis and prediction of cancer disease is capital important for a healthy life. Machine learning techniques can make a huge contribute on the process of early diagnosis and prediction of cancer. In this paper, two of the most popular machine learning techniques have been used for classification of Wisconsin Breast Cancer (Original) dataset and the classification performance of these techniques have been compared with each other using the values of accuracy, precision, recall and ROC Area. The best performance has been obtained by Support Vector Machine technique with the highest accuracy.

2.6 PAPER 6

Loomans-Kropp, H.A.; Umar, A. Increasing Incidence of Colorectal Cancer in Young Adults. *J. Cancer* Epidemiol. 2019, *2019*, 9841295.

Colorectal cancer (CRC) incidence and mortality has been declining in the United States for over a decade. Despite the decreasing trend, CRC remains the third most incident and fatal cancer. In 2019, CRC will be responsible for an estimated 78,500 new cases in men and 67,100 new cases in women, an increase of approximately 3.8% from 2018 for men and women. Fortunately, there are well-established screening guidelines that allow for the prevention and early detection of CRC. Despite the recommendations issued by the United States Preventive

Services Task Force (USPSTF) and the American Cancer Society, adherence to these guidelines remains low despite national efforts to improve screening rates. Existing CRC screening guidelines do not sufficiently address a newly emerging high-risk group: early-onset colorectal cancers (EOCRCs). EOCRC is defined as a cancer diagnosis occurring in individuals under the current recommended screening age of 50.

2.7 PAPER 7

Fatima, N.; Liu, L.; Hong, S.; Ahmed, H. Prediction of Breast Cancer, Comparative Review of Machine Learning Techniques, and Their Analysis. IEEE Access 2020, 8, 150360–150376. [Google Scholar] [CrossRef].

Breast cancer is type of tumor that occurs in the tissues of the breast. It is most common type of cancer found in women around the world and it is among the leading causes of deaths in women. This paper presents the comparative analysis of machine learning, deep learning and data mining techniques being used for the prediction of breast cancer. Many researchers have put their efforts on breast cancer diagnoses and prognoses, every technique has different accuracy rate and it varies for different situations, tools and datasets being used. Our main focus is to comparatively analyze different existing Machine Learning and Data Mining techniques in order to find out the most appropriate method that will support the large dataset with good accuracy of prediction. The main purpose of this review is to highlight all the previous studies of machine learning algorithms that are being used for breast cancer prediction and this paper provides the all necessary information to the beginners who want to analyze the machine learning algorithms to gain the base of deep learning.

2.8 PAPER 8

Kabiraj, S.; Raihan, M.; Alvi, N.; Afrin, M.; Akter, L.; Sohagi, S.A.; Podder, E. Breast Cancer Risk Prediction using XGBoost and Random Forest Algorithm. In Proceedings of the 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020.

Breast cancer is taking the lives of women globally. It’s one of the most common malignancies in women, as well as the leading cause of cancer-related death. Even though there are no therapies for breast cancer, early detection and diagnosis are critical in determining survival chances. Machine learning for medical diagnosis and its accuracy is a key advancement and a predicted trend in the future medical model. Our goal is to use machine learning algorithms to diagnose breast cancer and perform performance analysis. Under the supervised machine learning approach, different classifiers are applied to the data sets to predict outcomes. We have used five such classifiers, K-Nearest Neighbors (KNN), Random Forest (RF), Decision Trees (DT), Logistic Regression (LR), and Support Vector Machines (SVM) on the Breast Cancer WISCONSIN (Diagnostic) data set to observe how accurately they predict the cancerous instances. The classifiers are treated with cross-validation approaches to get exact accuracies. The same is done with different partitions of the data set, and performance analysis is made based on every observation. Significance statement: we have done a breast cancer prediction using some machine learning models and made a performance analysis.

2.9 PAPER 9

Al-Azzam, N.; Shatnawi, I. Comparing supervised and semi-supervised Machine Learning Models on Diagnosing Breast Cancer. ***Ann. Med. Surg.*** **2021,** ***62*****, 53–64.**

Background Breast cancer disease is the most common cancer in US women and the second cause of cancer death among women. Objectives To compare and evaluate the performance and accuracy of the key supervised and semi-supervised machine learning algorithms for breast cancer prediction. Materials and methods We have used nine machine learning classification algorithms for supervised (SL) and semi-supervised learning (SSL): 1) Logistic regression; 2) Gaussian Naive Bayes; 3) Linear Support vector Machine; 4) RBF Support vector machine; 5) Decision Tree; 6) Random Forest; 7) Xgboost; 8) Gradient Boosting; 9) KNN. The Wisconsin Diagnosis Cancer data set was used to train and test these models. To ensure the robustness of the model, we have applied K-fold cross validation and optimized hyperparameters. We have evaluated and compared the models using accuracy, precision, recall, F1-score, and ROC curves. Results The results of all models are inspiring using both SL and SSL. The SSL have high accuracy (90%–98%) with just half of the training data. The KNN model for supervised and logistic regression for the SSL achieved the highest accuracy of 98% Conclusion The accuracies of SSL algorithms are very close to the SL algorithms. The accuracies of all models are in the range of 91–98%. SSL is a promising and competitive approach to solve the problem. Using a small sample of labeled and low computational power, the SSL is fully capable of replacing SL algorithms in diagnosing tumor type.

2.10 PAPER 10

Tang, X.; Cai, L.; Meng, Y.; Gu, C.; Yang, J.; Yang, J. A Novel Hybrid Feature Selection and Ensemble Learning Framework for Unbalanced Cancer Data Diagnosis with Transcriptome and Functional Proteomic. *IEEE Access* 2021, *9*, 51659–51668.

The high dimension, high redundancy and class imbalance of cancer multiple omics data are the main challenges for cancer diagnosis. Existing studies have neglected the role of functional proteomics in the occurrence and development of cancer. In this study, a novel hybrid feature selection and ensemble learning framework, referred to as the three-stage feature selection and twice-competitional ensemble learning method (TSFS-TCEM), is proposed for cancer diagnosis. Firstly, we combine the transcriptome and functional proteomics data to construct a multi-omics data on breast cancer, which is the first time to apply these combined biological data for diagnosing breast cancer. Secondly, the proposed method introduces multiple models during the feature selection and diagnostic model construction. The three-stage feature selections integrate the features from different types of data and the twice-competitional ensemble learning framework resolves the data imbalance problem suffer from a single classifier. The TSFS-TCEM achieves a diagnostic accuracy of 99.64%, outperforming all compared methods. In addition, the 5-fold cross-validation sensitivity, specificity and F-Measure of the method are above 99.63%.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 EXISTING SYSTEM

The existing systems for breast cancer prediction using machine learning (ML) encompass a range of approaches and methodologies. These systems leverage ML algorithms to analyze patient data and make predictions regarding the likelihood of breast cancer. Here are some common components and characteristics of existing systems:

Data Collection: Existing systems gather diverse datasets containing patient demographics, medical history, genetic information, mammography images, and histopathological reports. These datasets are crucial for training and validating ML models.

Feature Extraction and Selection: Features or biomarkers relevant to breast cancer prediction are extracted from the collected data. Feature selection techniques may be employed to identify the most informative features for model training.

Model Training: ML algorithms such as logistic regression, support vector machines (SVM), random forests, gradient boosting machines (GBM), and deep learning architectures like convolutional neural networks (CNNs) are trained on the data to build predictive models. Each algorithm has its strengths and weaknesses, and the choice depends on factors such as data characteristics and performance requirements.

Cross-Validation and Evaluation: Cross-validation techniques are used to assess the performance of ML models and ensure their generalization to unseen data. Evaluation metrics such as accuracy, sensitivity, specificity, area under the receiver operating characteristic curve (AUC-ROC), and precision-recall curve are commonly employed.

Integration with Clinical Workflow: Some systems aim to integrate predictive models into clinical workflows to assist healthcare professionals in decision-making processes. This integration may involve developing user-friendly interfaces or integrating models into existing electronic health record (EHR) systems.

Interpretability and Explainability: Efforts are made to enhance the interpretability and Explainability of ML models to facilitate clinical acceptance and trust. Techniques such as feature importance analysis, model visualization, and surrogate models are employed to interpret predictions.

Validation and Clinical Trials: Prospective validation studies and clinical trials are conducted to evaluate the real-world performance of ML models in diverse patient populations. Collaboration with clinicians and researchers is essential to ensure clinical relevance and validity.

Ethical and Regulatory Compliance: Considerations regarding patient privacy, data security, informed consent, and algorithm transparency are addressed to comply with ethical and regulatory requirements such as the Health Insurance Portability and Accountability Act (HIPAA) and the General Data Protection Regulation (GDPR).

Overall, existing systems for breast cancer prediction using ML aim to improve early detection, assist clinicians in decision-making, and ultimately improve patient outcomes through personalized medicine and targeted interventions.

3.2 PROPOSED SYSTEM

The system mainly consists of four processes. Once the image is acquitted, the system converts the image into gray scale image. By applying the suitable Image segmentation techniques, the system aims at extracting a meaningful object lying in the image. Clustering is one the powerful image segmentation technique involves grouping of data points to cluster, the system implements clustering technique using Fuzzy C-means (FCM).

The segmented region is completely analyze by using the Multi-level Discrete Wavelet Transform, Principal Component Analysis (PCA) along with Gray Level Co-occurrence Matrix (GLCM) features. Totally 13 features are extracted in the system and their pixel values in the form of matrix is stored in database that is in the db. mat file, some of the features extracted by the system are mean, variance, entropy etc. Then image undergoes classification process with respect to dataset in db. mat file and classifies the image into Benign, Malignant and Normal.

The system also performs morphological operations and calculates region properties of the image such as Area, Eccentricity and Euler number. If the image has cancer cells, then the tumor area is computed and displayed by the system along with the boundary detected image.

System Architecture of Breast Cancer Prediction and Tracking system the physician uploads the mammogram of the patient to the device that is subjected to the process of image segmentation. Using image processing technique, the segmented image is preprocessed.

Extraction methods to extract necessary features are applied to the image. The classifier model is given extracted features, then the test image classification process is performed with respect to the training data present in the database.

The test picture is marked as either cancerous or non-cancerous. Unless the test image is marked as cancerous, the tumor region will be measured, and the findings will be shown to the doctor along with the observed boundary image

Fuzzy c-means is a cluster function where data points are associated to several clusters with varying membership rates. One of the segmentation techniques applicable to gray level images is the FCM. In the proposed method, an initial location of the central cluster and the membership degree measure for every data point are determined with the aid of Fuzzy C-Means.

Classification after feature extraction relies on the shapes of cancer cells in the image and method often makes use of all features contained in the database for classification purposes. System should analyze similarity measures and use clustering techniques and feature extraction techniques such as Multi-level Discrete Wavelet Transform, Principal Component Analysis, GLCM to allow data set reduction for similarity measurement improvement. 15 The next step will be extraction of a function. These dimensionality reductions can be a helpful step in the visualization and analysis of high featured datasets, while maintaining as much variation as possible.

This technique is used to simplify the classification to predict a better output. The maximum variance must be found out in order to segregate the data into multiple clusters. For Example, if chosen 3 different centroids and then make 3 main elements in the 2D plane of high dimensionality we can see that the data is distributed and is consistent with the clusters and an find it apparent in the 2D diagram.

This reduces overfitting of the data and avoids the clusters to be far shortly spaced form each other or overlapping from each other. Likewise, the larger the number of illustrative variables permitted in regression study, the greater the risk that the model will be overfitted, resulting in the output to fail in terms of the generalization with the rest of the datasets. One approach is to lessen them to a few main components, particularly when there are well built association between different potential variables, and then execute the regression in opposition to them which is the Principal component regression abbreviated as PCA. PCA is a method which uses linear variation to relocate a group of examined correlated variables (entities each of which takes on different numerical values) into principal components.

PCA is a tool used mainly for illustrative data analysis and predictive model building [5]. Reduction of the dimensionality can also be sufficient when the illustrative variable points in a dataset are obstreperous. Using the PCA method the system can focus on most of the signals into the main components first and then the features can be captured by keeping the minimized dimensionality.

This process offers benefits because if the main component captures noise then it there is loss of features. 16 After the PCA extraction the next is segmentation. The segmentation is used because for the improvement of the delineation of image to a visible examination. Its insight is to see the condition of these uncertain areas to assign and help the irregularities of the classification between the cancerous and non-cancerous features.

3.3 PROPOSED TECHNIQUES

Analysing and visualizing COVID-19 data can provide valuable insights into the spread and impact of the virus. Here are some proposed techniques for such a project:

1. Data Collection: Gather COVID-19 data from reliable sources such as government health departments, the World Health Organization (WHO), or reputable datasets like Johns Hopkins University's COVID-19 Data Repository.

2. Data Cleaning: Clean the data by removing duplicates, handling missing values, and ensuring consistency in data formats. This step is crucial for accurate analysis.

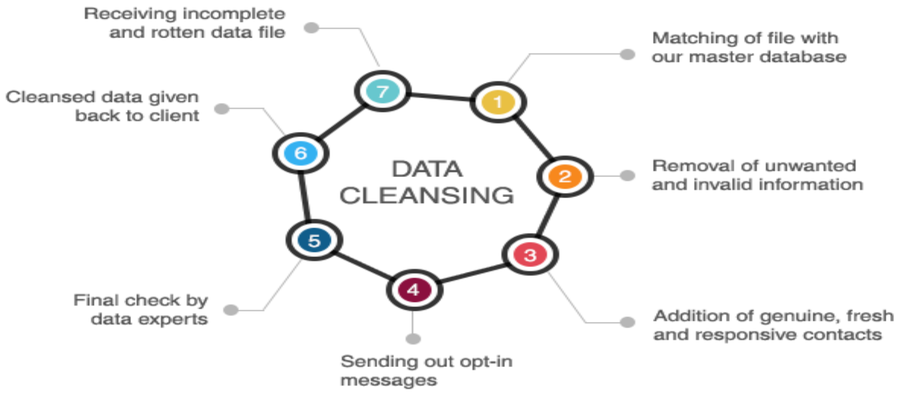

Receiving incomplete and rotten data file
Matching of file with our master database
Cleansed data given back to client
DATA CLEANSING
Removal of unwanted and invalid information
Final check by data experts
Addition of genuine, fresh and responsive contacts
Sending out opt-in messages
1
2
3
4
5
6
7

Fig 3.3.1 DATA CLEANING

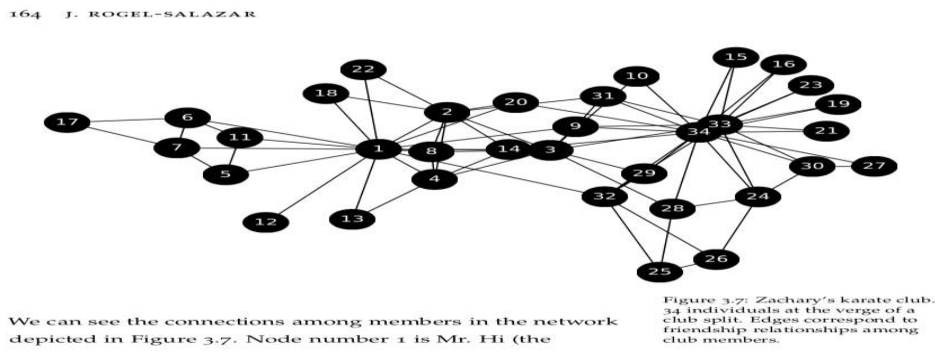
3. Exploratory Data Analysis (EDA): Conduct EDA to understand the characteristics of the data, such as trends over time, geographical distribution, and demographic patterns. Techniques like summary statistics, histograms, and time series analysis can be helpful.

4. Time Series Analysis: Analyze the temporal trends in COVID-19 cases, deaths, and other relevant metrics using time series techniques such as decomposition, autocorrelation, and forecasting models like ARIMA or Prophet.

5. Geospatial Analysis: Visualize the geographical spread of COVID-19 using maps and explore spatial patterns using techniques like choropleth maps, heatmaps, and spatial autocorrelation analysis.

6. Machine Learning Models: Develop machine learning models to predict COVID-19 outcomes, such as future case counts or mortality rates. Common algorithms include regression, random forests, and neural networks.

7. Network Analysis: Investigate the spread of COVID-19 through networks of interactions, such as social networks, transportation networks, or contact tracing data. Network analysis techniques like centrality measures and community detection can provide insights.



164 J. ROGEL-SALAZAR

We can see the connections among members in the network depicted in Figure 3.7. Node number 1 is Mr. Hi (the

Figure 3.7: Zachary's karate club. 34 individuals at the verge of a club split. Edges correspond to friendship relationships among club members.

FIG 3.3.2 NETWORK ANALYSIS

By employing these techniques, you can gain a comprehensive understanding of the COVID-19 pandemic and contribute to efforts in monitoring, mitigation, and decision-making.

3.4 PYTHON IN DATA SCIENCE

Python is one of the most popular programming languages for data science due to its simplicity, versatility, and extensive ecosystem of libraries. Here's how Python is commonly used in data science:

1. Data Manipulation: Python libraries like Pandas provide powerful tools for data manipulation, including reading and writing various file formats, handling missing data, reshaping datasets, and performing operations like filtering, sorting, and aggregation.

2. Data Visualization: Libraries like Matplotlib, Seaborn, and Plotly enable data visualization in Python, allowing you to create a wide range of plots, charts, and graphs to explore data distributions, relationships, and trends.

3. Machine Learning: Python offers rich libraries for machine learning, such as Scikit-learn, TensorFlow, and PyTorch. These libraries provide implementations of various machine learning algorithms, including classification, neural networking, regression, clustering, and dimensionality reduction, as well as tools for model evaluation and hyperparameter tuning.

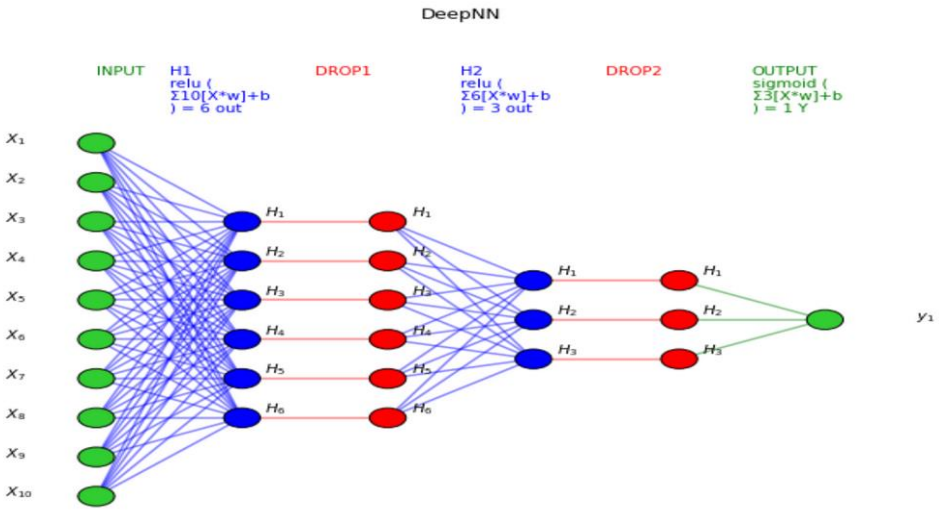


FIG 3.4.1 NEURAL NETWORK

4. Statistical Analysis: Python's stats models library provides tools for statistical modelling, hypothesis testing, and time series analysis, allowing data scientists to conduct rigorous statistical analyses and make data-driven decisions.

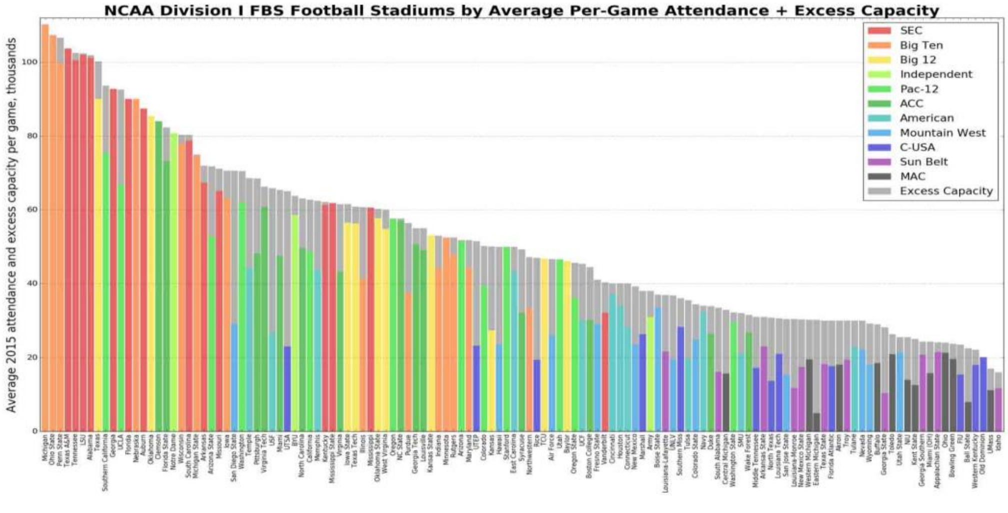


FIG 3.4.2 STATISTICAL ANALYSIS

5. Web Scraping: Python's Beautiful Soup and Scrapy libraries are commonly used for web scraping, enabling data scientists to extract data from websites and APIs for analysis and modelling.

6. Data Cleaning and Preprocessing: Python offers libraries like NumPy and SciPy for numerical computing and advanced mathematical functions, which are essential for data cleaning, Preprocessing, and transformation tasks.

7. Big Data Processing: Python interfaces with big data processing frameworks like Apache Spark and Disk, allowing data scientists to analyse large-scale datasets distributed across clusters of machines.

8. Deep Learning: Python libraries like TensorFlow and PyTorch are widely used for deep learning tasks, including neural network design, training, and deployment, enabling data scientists to build complex models for tasks like image recognition, natural language processing, and reinforcement learning.

9. Interactive Computing: Python's Jupyter Notebook and JupyterLab provide interactive computing environments that combine code, visualizations, and narrative text, making it easy for data scientists to explore data, experiment with algorithms, and communicate their findings.

10. Integration with Other Tools: Python seamlessly integrates with other tools and technologies commonly used in data science, such as databases (e.g., SQL databases, NoSQL databases), cloud services (e.g., AWS, Google Cloud), and visualization tools (e.g., Tableau, Power BI).

Overall, Python's rich ecosystem of libraries, combined with its simplicity and flexibility, makes it an ideal choice for data scientists to tackle a wide range of data science tasks effectively.

3.5 TRAINING

Training data is a large dataset used to teach a machine learning model how to recognize outcomes. It can include text, images, video, or audio, and can be structured in many ways.

For example, for sequential decision trees, the training data would be raw text or alphanumerical data. For supervised ML models, the training data is labeled, while the data used to train unsupervised ML models is not labeled. The quality of training data is important when creating reliable algorithms.

Training involves learning good values for all the weights and the bias from labeled examples. For example, a machine learning algorithm builds a model by examining many examples and attempting to find a model that minimizes loss, a process called empirical risk minimization.

Here are some skills that data science training programs can help fill out: Critical thinking, Coding for data engineering and analysis, Generating valuable insights from data, and Predictive analytics and data mining.

3.5.1 TRAINING DATASET

The training dataset is a fundamental component in the process of training a machine learning model. It's essentially the data that the model uses to learn patterns, associations, and relationships between input features and their corresponding target outputs. Here's a detailed breakdown of the training dataset:

1. Input Data (Features): The training dataset consists of a set of input data points, also known as features. These features are the variables or attributes that the model will use to make predictions or classifications. Features can be of various types, including numerical, categorical, or textual data, depending on the nature of the problem being solved. Each data point in the training dataset is represented by a set of features, where each feature provides specific information about the input.

2. Output Labels or Targets: Along with input features, the training dataset also includes corresponding output labels or targets. These labels represent the desired output or prediction that the model should learn to approximate based on the input features. In supervised learning tasks, where the model learns from labeled data, the training dataset contains both input features and their corresponding output labels.

3. Size and Quantity: The size and quantity of the training dataset can significantly impact the performance and generalization ability of the trained model. Typically, a larger training dataset provides more diverse examples for the model to learn from, potentially leading to better generalization on unseen data. However, collecting and labeling large datasets can be time-consuming and resource-intensive, so practitioners often strive to strike a balance between dataset size and model performance.

4. Data Preprocessing: Before feeding the data into the model for training, it often undergoes preprocessing steps to clean, normalize, and transform the features. Data preprocessing may involve tasks such as handling missing values, scaling numerical features, encoding categorical variables, and splitting the dataset into training and validation subsets.

5. Training Process: During the training process, the model iteratively adjusts its internal parameters to minimize the difference between its predictions and the actual target labels in the training dataset.

6. Evaluation: Once the model is trained using the training dataset, it is evaluated on a separate validation or test dataset to assess its performance and generalization ability.

In summary, the training dataset is a crucial component in the machine learning pipeline, providing the raw material from which the model learns to make predictions or classifications. Its quality, size, and diversity play a significant role in determining the performance and robustness of the trained model.

3.6 TESTING

The usage of the word testing in relation to data science projects is primarily used for testing the model performance in terms of accuracy of the model. It can be noted that the word, “Testing” means different for software development and data science projects developments.

3.6.1. TESTING DATASET

In COVID-19 data analysis projects, several datasets are commonly used for various analyses and modelling tasks. Some of the frequently tested datasets include:

1. Case Data: This dataset includes information about confirmed cases, deaths, and recoveries due to COVID-19. It usually contains attributes such as date, location (country, region), case counts, and demographic information.

2. Testing Data: Information about COVID-19 testing, including the number of tests conducted, test positivity rates, and testing methodologies. This dataset helps in understanding testing trends and assessing the spread of the virus.

3. Hospitalization Data: Data related to COVID-19 hospitalizations, including hospital admissions, ICU occupancy rates, ventilator usage, and hospital capacity. This dataset assists in evaluating healthcare system readiness and capacity planning.

4. Vaccination Data: Information about COVID-19 vaccination campaigns, including the number of doses administered, vaccination rates, vaccine types, and demographic distribution. This dataset helps in assessing vaccination progress and effectiveness.

5. Genomic Data: Genomic sequences of the SARS-CoV-2 virus, including variants of concern (VOCs) and their prevalence over time and geographic regions.

6. Mobility Data: Data on human mobility patterns, including travel, commuting, and social interactions. This dataset helps in studying the impact of mobility on virus transmission and predicting outbreaks.

7. Policy Data: Information about government interventions and public health measures implemented to control the spread of COVID-19, such as lockdowns, mask mandates, and social distancing regulations. This dataset aids in assessing the effectiveness of different policy interventions.

Testing these datasets involves various steps, including data cleaning, Preprocessing, validation, and verification to ensure data quality, consistency, and reliability for accurate analysis and decision-making in COVID-19 research and public health response efforts.

3.7 SPLITTING, IMPUTATION AND INTERPOLATION

- ▶ Splitting – pandas sample () is used to generate sample random row or column from the data frame.
- ▶ Imputation – The process of replacing the missing data with substituted values.
- ▶ Interpolation – A method of constructing new data points within the range of a discrete set of know points.

3.8 DATA FRAMES

In a COVID-19 data science project, data frames are commonly used to organize and analyse data. You can use libraries like pandas in Python to create and manipulate data frames.

These data frames can contain various information such as the number of cases, deaths, recoveries, and other relevant metrics, organized by different attributes like date, location, demographics, etc. They serve as a structured way to handle and process the large amounts of data typically involved in COVID-19 analysis. The main key as,

- It is the crucial components in Covid-19 data analysis project.
- They help organize and manipulate data efficiency.
- Python library packages like NumPy, pandas are used for this purpose.
- Data Frames accepts many different kinds of inputs.

CHAPTER 4

INTRODUCTION OF MACHINE LEARNING

4.1 GENERAL

AI is a mechanism which features algorithms and calculations based on a normal human intelligence to address a problem. The AI behaves and approaches a problem in a similar way that a normal human brain would. Its working mechanism is influenced by human thinking. A collection of expectation and result is achieved by AI by portraying information in a form termed as 'test information' without making use of any predetermined models or being trained in that particular domain. Problems catering to non-related dimensions such as email sifting, PC vision, location of system gate crashers is addressed. Thus, it is assertive that it is not possible to train an AI to address a particular domain, instead an AI trained with general problem-solving abilities, builds up its own algorithms for a set of problems.

An AI engine is allocated with responsibility of prediction or analysis using a PC framework and set of data. For this an AI engine is allocated with packages of scientific methods, logistic calculations, data sets and knowledge about the field of the problems for performing. Moreover, the entire operation of AI is carried based on unsupervised learning model which leaves a very less room for training a robust AI for only a problem specific solution. However, for business purposes modifications are performed before its application.

4.2 OVERVIEW OF MACHINE LEARNING

The name was authored in 1959 by Arthur Samuel Tom M. Mitchell gave a generally cited, increasingly formal meaning of the calculations contemplated in the AI field. This meaning of the assignments in which AI is concerned offers an in a general sense operational definition as opposed to characterizing the field in psychological terms. This pursues Alan Turing's proposition in his paper "Registering Machinery and Intelligence", in which the inquiry "Can machines believe?" is supplanted with the inquiry "Can machines do what we (as speculation elements) can do?" In Turing's proposition the different attributes that could be controlled by a reasoning machine and the different ramifications in building one is uncovered.

Before the introduction of machine learning a general assumption was that a robot needs to learn everything from a human brain to function appropriately. But as efforts were made to do so, it was realized that it is very difficult to make a robot to learn everything from a human brain as the human brain is very much sophisticated. An idea was then proposed that rather than teaching a robot everything we know, it is easier to make the robot learn on its own. The type of dataset we are working upon largely determines how we approach while training the model. Based on the dataset we will feed to the algorithm; the training model would vary. The size, type and dynamism of the dataset will decide what type of training model we would build. Finally, on deciding upon the training model, modifications need to be made to achieve the proper objective function to generate proper set of output that we wish to achieve. The stages of machine learning process are rather termed as ingredients than steps, because the machine learning is an iterative process. The iterative process is repeated each time to achieve maximum optimization and efficiency.

4.3 MACHINE LEARNING-BASED APPROCHES

The following is a concise outline of mainstream AI based systems for inconsistency identification.

4.3.1 DENSITY BASED DETECTION OF ANOMALY

It derives its working mechanism from KNN algorithm

Assumption - Relevant data locates themselves around a common point in close proximity whereas irregular data are placed at a distance. The data points are clustered at a closed proximity based on a density score, which may be derived using Euclidian distance or appropriate methods based on the data. Classification is made on two bases:

K closest neighbour: In this method the basic clustering mechanism is dependent on separation measurements of each data points which determines the clustering or similarities of each information considered.

Relative thickness of the information - Also known as Least Outlier Fraction (LOF).
Calculation is performed on the basis of separation metric.

4.3.2 CLUSTERING BASED DETECTION OF ANOMALY

Clustering is an exceptional algorithm known for its optimization and robust nature. For this reason, it is widely used in unsupervised learning

Assumption - Data points that are similar tends to get gather around specific points. The relative distance of each cluster is achieved by its shortest distance from the centroid of the space.

K means is widely used in data classification. It makes use of k means algorithm to cluster closely related data in close proximity forming clusters.

4.3.3 SVM BASED DETECTION OF ANOMALY

- A support vector machine is one of the most important algorithm used for classification purposes

- The SVM uses methods to determine a soft boundary to distinguish data clusters. Data closely related falls within the parameter of a closed boundary. This results in formation of multiple clusters. SVM is widely used for binary classifications also. Most of the SVM algorithms works based on unsupervised learning.
- The yield of an abnormality locator are mostly numeric scalar qualities for distinguishing areas of explicit edges.

In this Jupiter journal we are going to assume the acknowledgment card misrepresentation recognition as the contextual investigation for understanding this idea in detail utilizing the accompanying Anomaly Detection Techniques in particular

4.4 DATASET

A dataset corresponds to a collection of data which may or may not be related to each other. A dataset can consist of data related to a particular domain. It may consist information for a single member or a group of member. For ex personal and other relevant details of an employee can be termed as a dataset, whereas collection of the information of all the employees working for that company is also a dataset. Thus the purpose of the problem defines the size of the dataset. A dataset consists of multiple columns often termed as parameters and multiple rows known as tuples. Individual data pieces are also termed as datum. For example, in a data set consisting of employee details of a company.

CHAPTER 5

SYSTEM SPECIFICATION

5.1 GENERAL

The necessity for the most part dependent on two classes: they is practical portray every single required usefulness for framework administrations which are given by the customers. Non-useful necessities characterize the framework properties and compels. The equipment prerequisites indicate the equipment functionalities and required speed and limit of the fringe.

The product prerequisites incorporate programming expected to create and run the framework.

5.2 HARDWARE SPECIFICATION

• System	- Core i5
• Mobile	- Android
• Monitor	- RGB colour
• Hard Disk	- 2 TB
• Mouse	- Microsoft
• Ram	- 8GB

5.3 SPECIFICATION OF THE SOFTWARE

Operating system	- Win 10
Dataset	- csv
Language	- Python

5.4 SOFTWARES USED

- Python 3.5
- NumPy 1.11.3
- Matplotlib 1.5.3
- Pandas 0.19.1
- Seaborn 0.7.1
- SciPy
- Scikit-learn 0.18.1

5.5 PYTHON PACKAGES

Python is one of the most popular programming languages used across various tech disciplines, especially in data science and machine learning. Python offers an easy-to-code, object-oriented, high-level language with a broad collection of libraries for a multitude of use cases. It has over 137,000 libraries.

One of the reasons Python is so valuable to data science is its vast collection of data manipulation, data visualization, machine learning, and deep learning libraries.

5.5.1 NUMPY

NumPy, is one of the most broadly-used open-source Python libraries and is mainly used for scientific computation. Its built-in mathematical functions enable lightning-speed computation and can support multidimensional data and large matrices.

It is also used in linear algebra. NumPy Array is often used preferentially over lists as it uses less memory and is more convenient and efficient.

5.5.2 PANDAS

Pandas is an open-source library commonly used in data science. It is primarily used for data analysis, data manipulation, and data cleaning. Pandas allow for simple data modeling and data analysis operations without needing to write a lot of code.

As stated on their website, pandas is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool. Some key features of this library include:

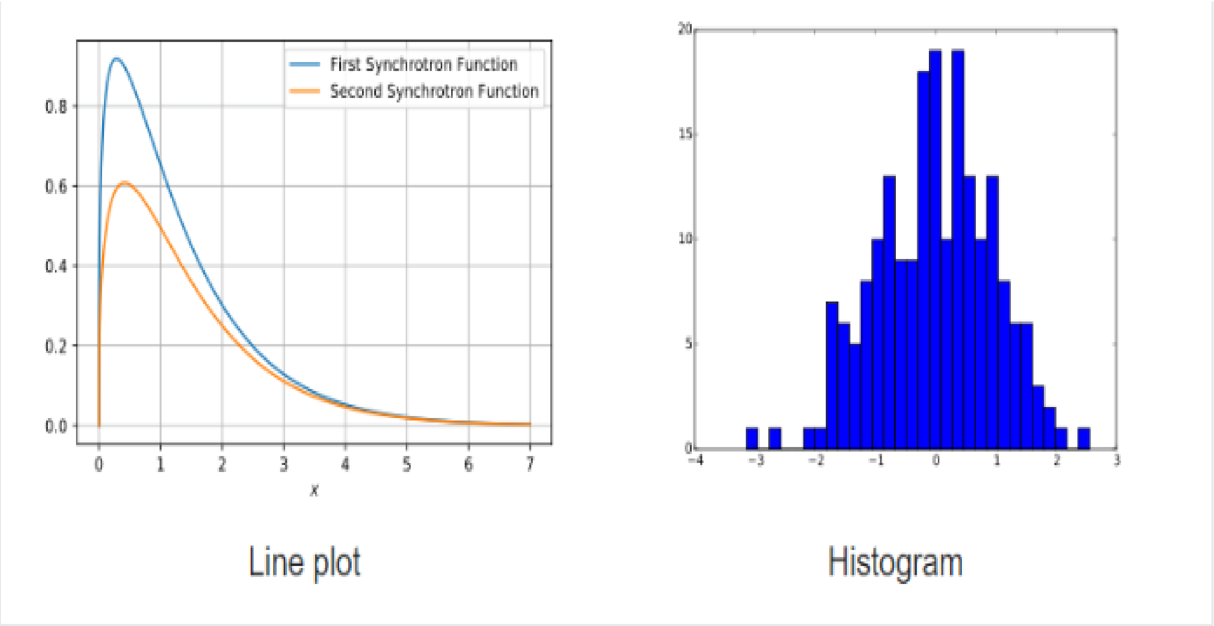
- Data Frames, which allow for quick, efficient data manipulation and include integrated indexing;
- Several tools which enable users to write and read data between in-memory data structures and diverse formats, including Excel files, text and CSV files, Microsoft, HDF5 formats, and SQL databases;
- Intelligent label-based slicing, fancy indexing, and sub setting of large data sets;
- High-performance merging and joining of data sets;

- A powerful group by engine which enables data aggregation or transformation, allowing users to perform split-apply-combine operations on data sets;
- Time series-functionality which enables date range generation and frequency conversion, moving window statistics, date shifting, and lagging. You'll even be able to join time series and create domain-specific time offsets without worrying you'll lose data;
- Ideal when working with critical code paths written in C or Python.

5.5.3 MATPLOTLIB

Matplotlib is an extensive library for creating fixed, interactive, and animated Python visualizations. A large number of third-party packages extend and build on Matplotlib's functionality, including several higher-level plotting interfaces (Seaborn, HoloViews, ggplot, etc.)

Matplotlib is designed to be as functional as MATLAB, with the additional benefit of being able to use Python. It also has the advantage of being free and open source. It allows the user to visualize data using a variety of different types of plots, including but not limited to scatterplots, histograms, bar charts, error charts, and boxplots. What's more, all visualizations can be implemented with just a few lines of code.



Line plot

Histogram

FIG 4.3.3.1 MATPLOTLIB

5.5.4 SEABORN

Another popular Matplotlib-based Python data visualization framework, seaborn is a high-level interface for creating aesthetically appealing and valuable statistical visuals which are crucial for studying and comprehending data.

This Python library is closely connected with both NumPy and pandas'data structures. The driving principle behind Seaborn is to make visualization an essential component of data analysis and exploration; thus, its plotting algorithms use data frames that encompass entire datasets.

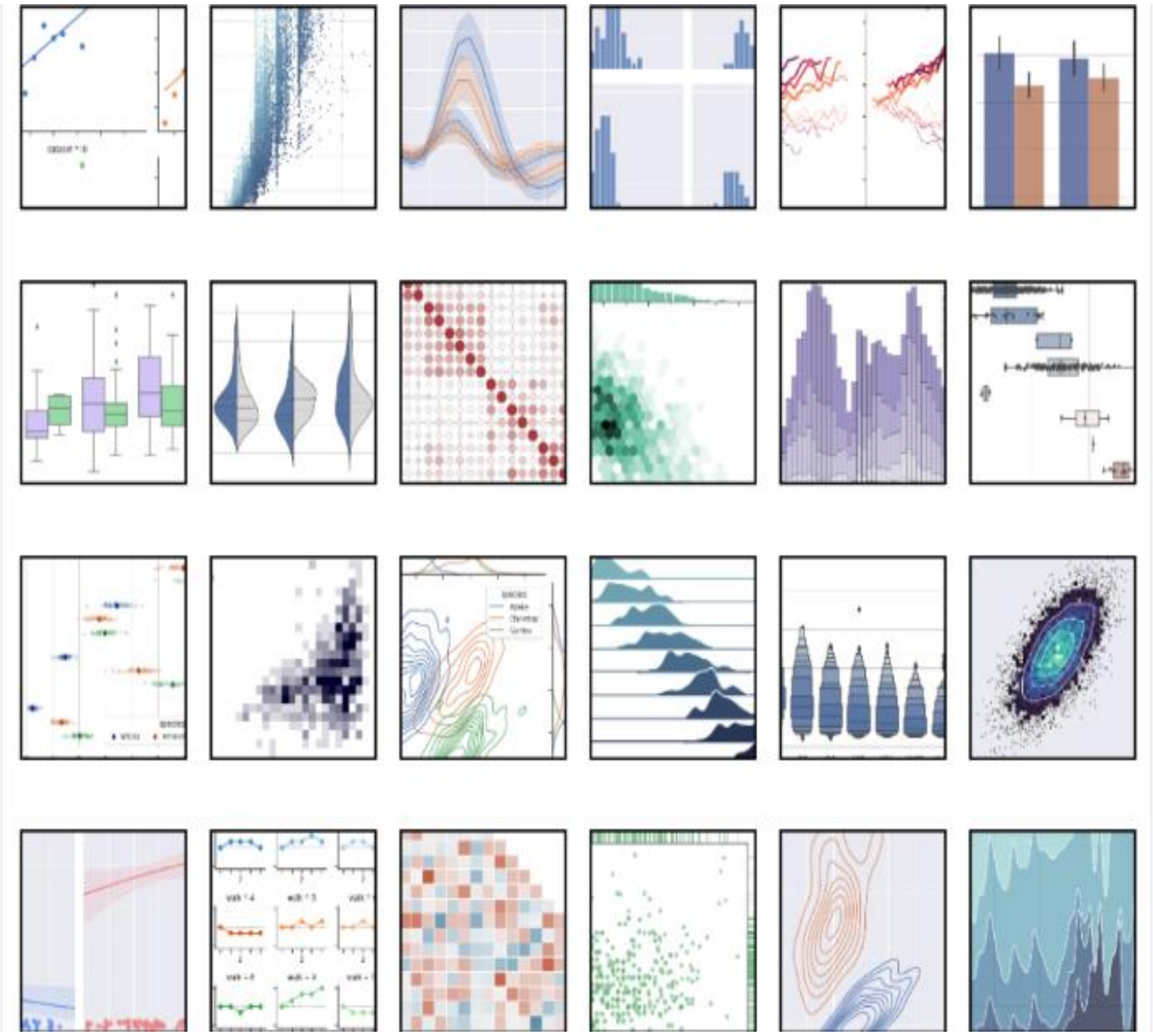


FIG 4.3.4.1 SEABORN

5.5.5 PLOTLY

The hugely popular open-source graphing library Plotly can be used to create interactive data visualizations. Plotly is built on top of the Plotly JavaScript library (plotly.js) and can be used to create web-based data visualizations that can be saved as HTML files or displayed in Jupyter notebooks and web applications using Dash.

It provides more than 40 unique chart types, such as scatter plots, histograms, line charts, bar charts, pie charts, error bars, box plots, multiple axes, sparklines, dendrograms, and 3-D charts. Plotly also offers contour plots, which are not that common in other data visualization libraries.

If you want interactive visualizations or dashboard-like graphics, Plotly is a good alternative to Matplotlib and Seaborn. It is currently available for use under the MIT license.

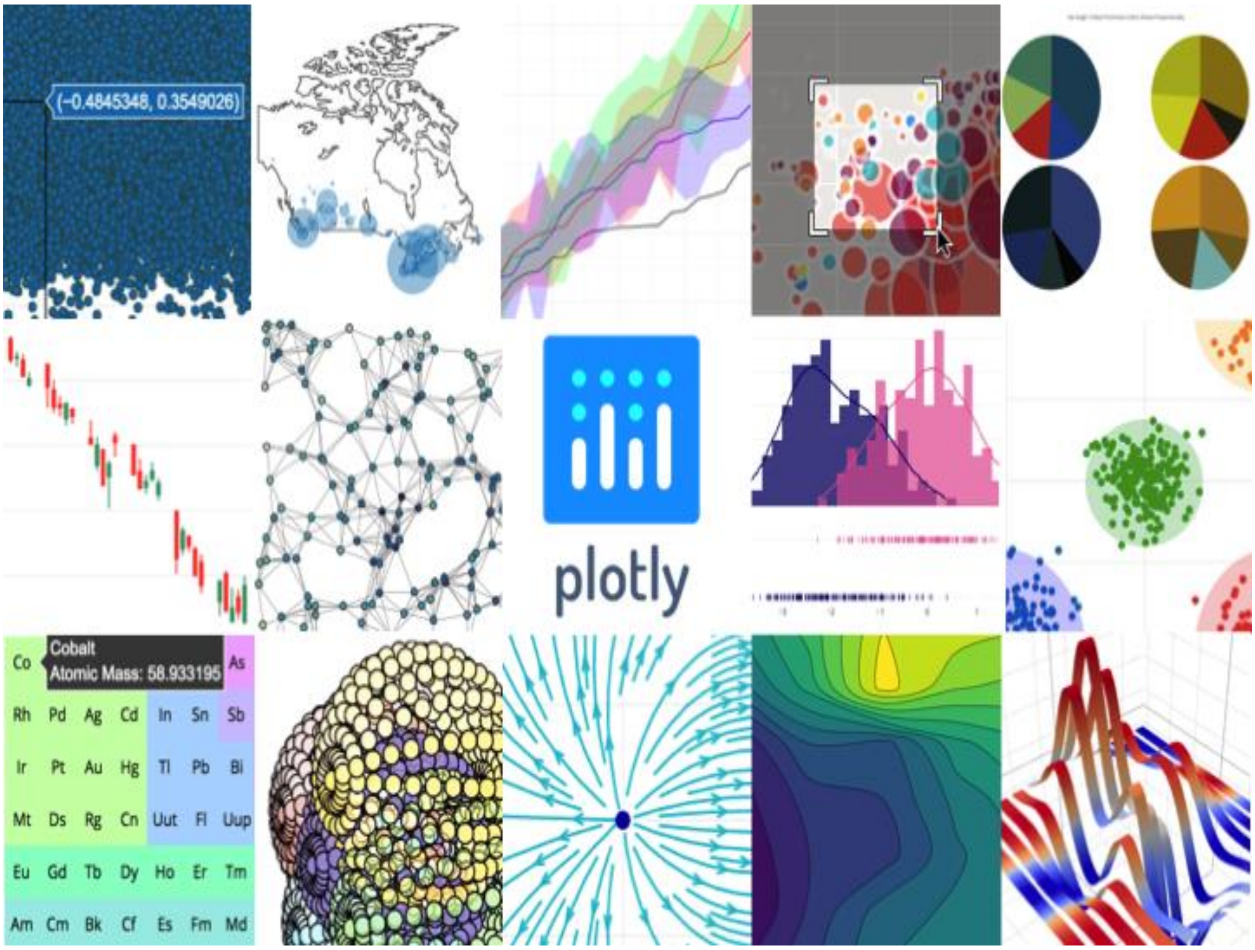


FIG 4.3.5.1 PLOTLY

5.6 JUPYTER NOTEBOOK

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. It supports various programming languages, including Python, R, Julia, and others, making it a versatile tool for data science projects.

1. Interactive Computing: Jupyter Notebooks provide an interactive computing environment where you can write and execute code in individual cells. This allows you to experiment with code, test hypotheses, and explore data interactively.

2.Integration of Code and Documentation: One of the key features of Jupyter Notebooks is the ability to include narrative text, equations, and visualizations alongside code cells. This integration of code and documentation makes it easy to create rich, self-explanatory documents that document your data analysis process step by step.

3. Data Exploration and Visualization: Jupyter Notebooks are well-suited for data exploration and visualization tasks. You can use libraries like Pandas, NumPy, Matplotlib, Seaborn, and Plotly to analyze and visualize data directly within the notebook environment. Interactive visualizations can be created using tools like Plotly or Bokeh, allowing for exploration of complex datasets.

4. Reproducibility: Jupyter Notebooks promote reproducibility in data science projects by capturing the entire data analysis workflow in a single document. By including code, data, visualizations, and explanations in one place, you make it easier for others to understand and reproduce your analysis.

5. Collaboration and Sharing: Jupyter Notebooks can be easily shared with colleagues or collaborators, either as static documents or interactive notebooks hosted on platforms like GitHub or Jupyter Hub. This facilitates collaboration and allows team members to review, comment, and contribute to the analysis.

CHAPTER 6

DESIGN ENGINEERING

6.1 ARCHITECTURE DIAGRAM

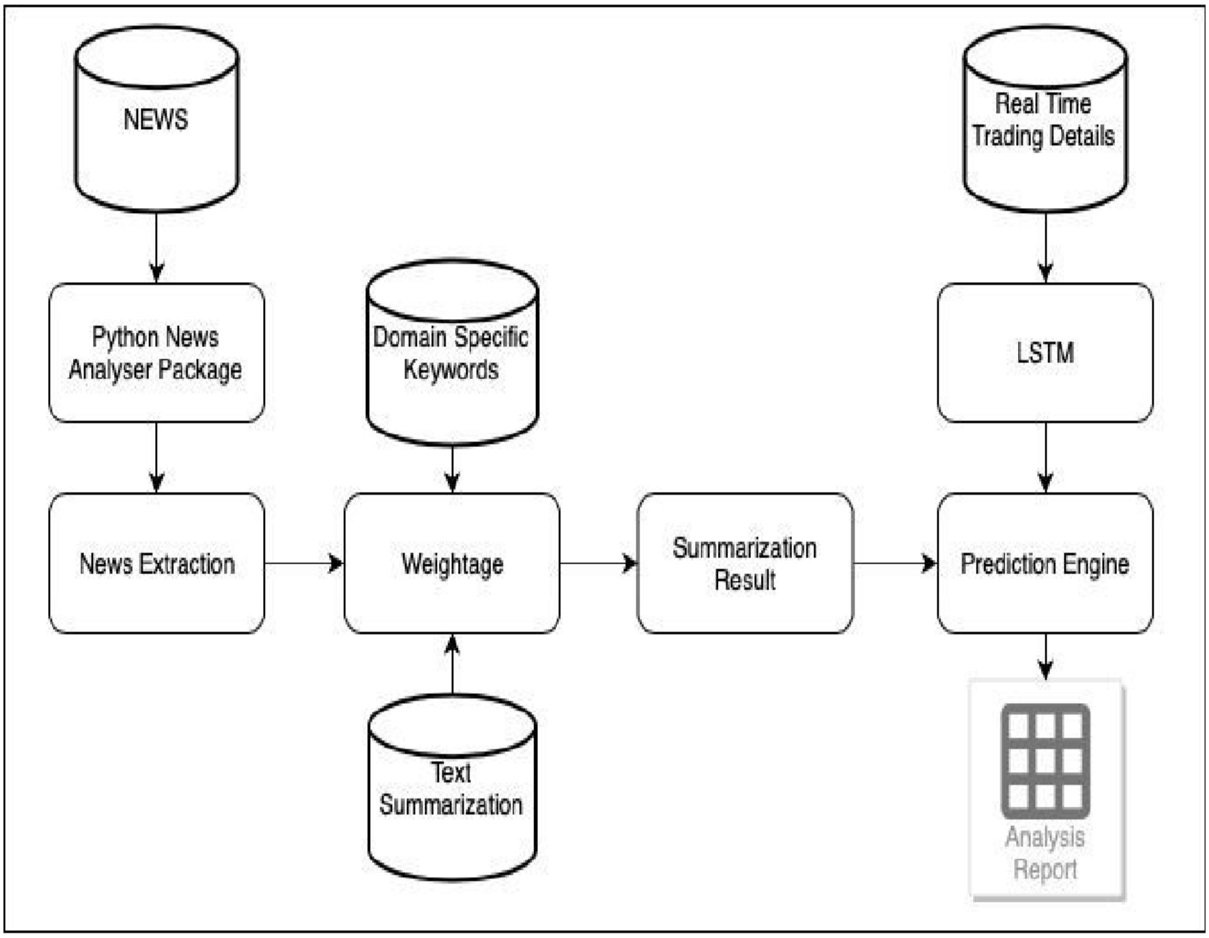


FIG 6.1 ARCHITECTURE DIAGRAM

6.2 FLOW CHART

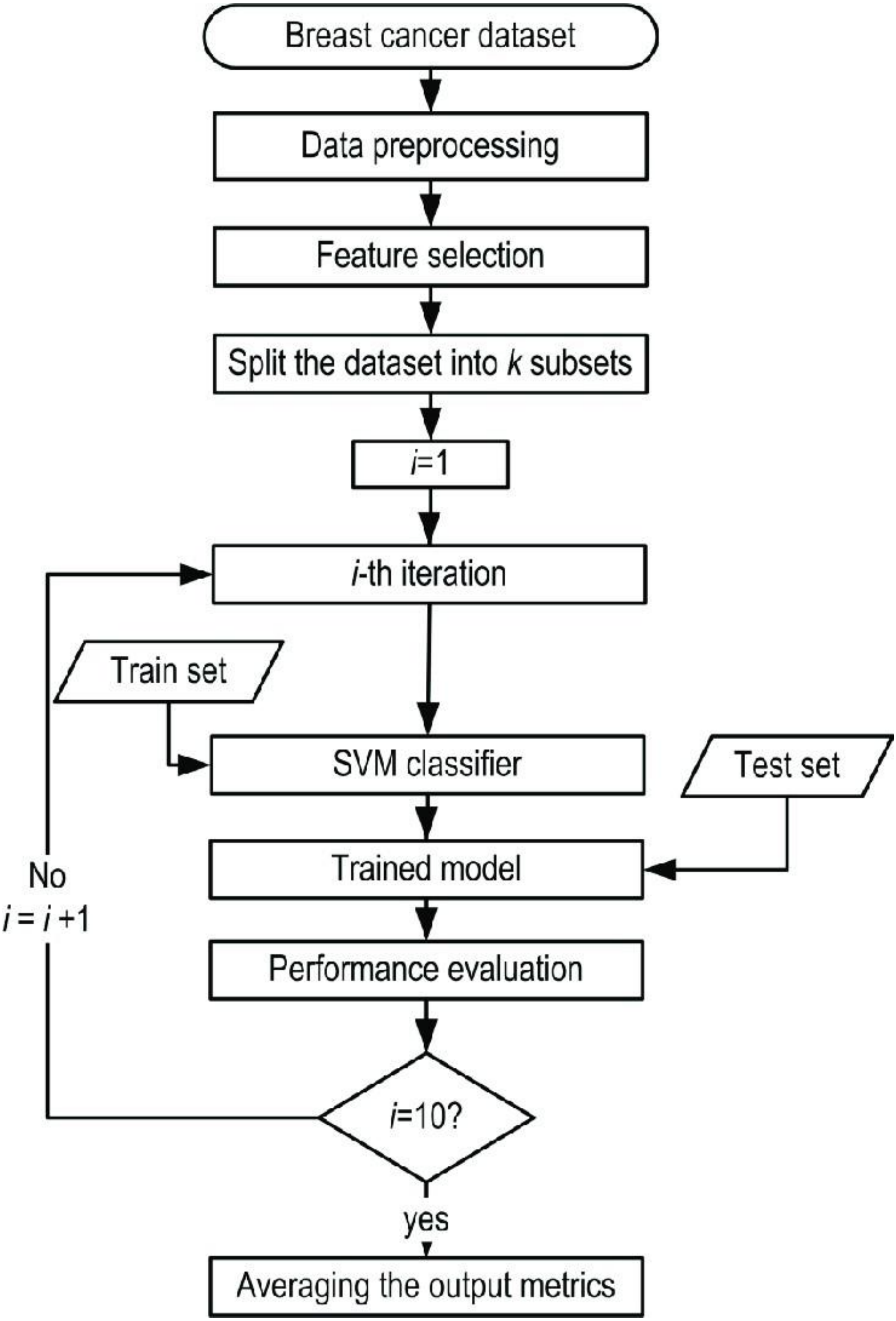

Breast cancer dataset
Data preprocessing
Feature selection
Split the dataset into *k* subsets
i=1
i-th iteration
Train set
SVM classifier
Test set
Trained model
Performance evaluation
No
i = *i* +1
i=10?
yes
Averaging the output metrics

FIG 6.2.1 FLOW CHART

6.3 SYSTEM FLOW DIAGRAM

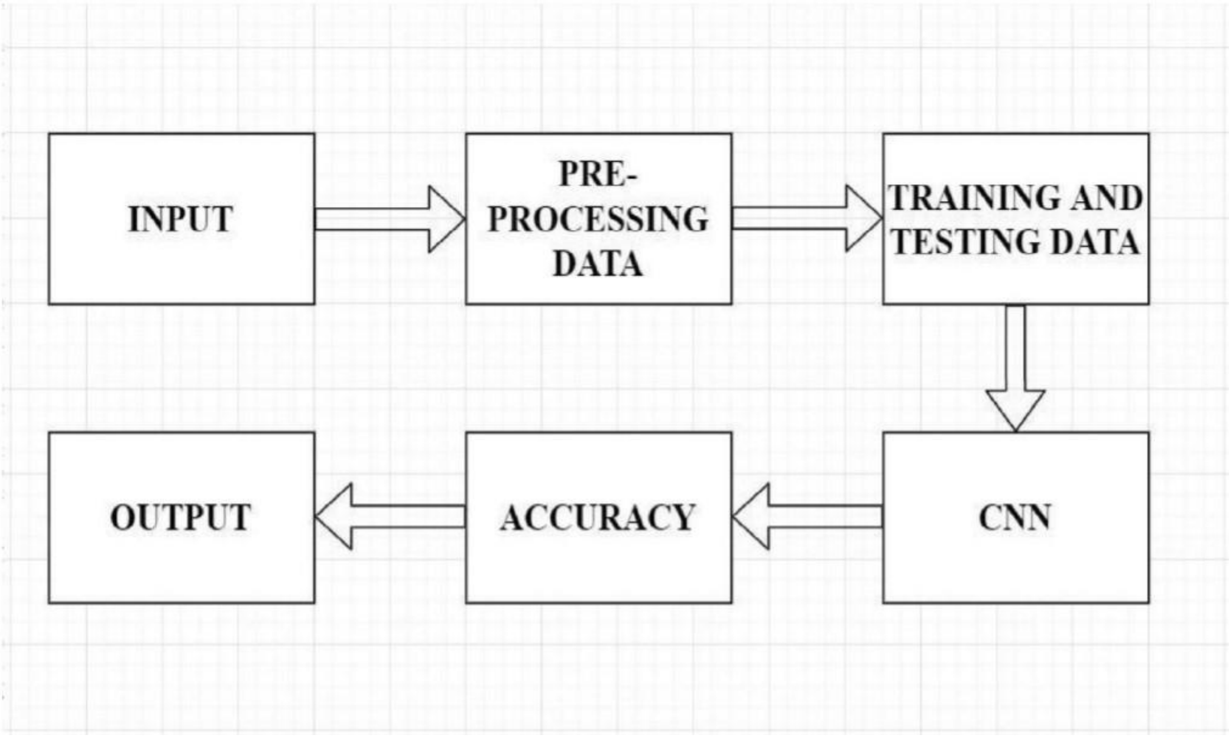


FIG 6.3.1 SYSTEM DIAGRAM

CHAPTER 7

ALGORITHM

7.1 Logistic Regression Logistic

Logistic Regression is a Classification model, which tries to classify the data based on the probability of it occurring.

This algorithm is used in multiple places where classification is required, we have used it to classify if the patient is susceptible to be infected by coved or not This is one of the classification methods which we have used. It used Sigmoid function to classify the data.

Logistic regression is a supervised machine learning algorithm that accomplishes binary classification tasks by predicting the probability of an outcome, event, or observation. The model delivers a binary or dichotomous outcome limited to two possible outcomes: yes/no, 0/1, or true/false.

Logical regression analyzes the relationship between one or more independent variables and classifies data into discrete classes. It is extensively used in predictive modeling, where the model estimates the mathematical probability of whether an instance belongs to a specific category or not.

For example, 0 – represents a negative class; 1 – represents a positive class. Logistic regression is commonly used in binary classification problems where the outcome variable reveals either of the two categories (0 and 1).

Some examples of such classifications and instances where the binary response is expected or implied are:

1. Determine the probability of heart attacks: With the help of a logistic model, medical practitioners can determine the relationship between variables such as the weight, exercise, etc., of an individual and use it to predict whether the person will suffer from a heart attack or any other medical complication.

2. Possibility of enrolling into a university: Application aggregators can determine the probability of a student getting accepted to a particular university or a degree course in a college by studying the relationship between the estimator variables, such as GRE, GMAT, or TOEFL scores.

3. Identifying spam emails: Email inboxes are filtered to determine if the email communication is promotional/spam by understanding the predictor variables and applying a logistic regression algorithm to check its authenticity.

7.1.1 ADVANTAGES OF LOGISTICS ALGORITHM

The logistic regression analysis has several advantages in the field of machine learning.

1. Easier to implement machine learning methods: A machine learning model can be effectively set up with the help of training and testing. The training identifies patterns in the input data (image) and associates them with some form of output (label). Training a logistic model with a regression algorithm does not demand higher computational power. As such, logistic regression is easier to implement, interpret, and train than other ML methods.

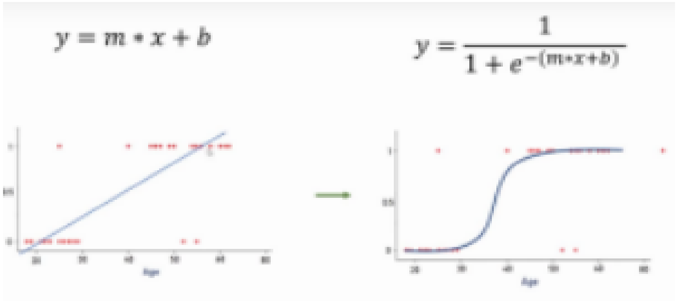
2. Suitable for linearly separable datasets: A linearly separable dataset refers to a graph where a straight line separates the two data classes. In logistic regression, the y variable takes only two values. Hence, one can effectively classify data into two separate classes if linearly separable data is used.

3. Provides valuable insights: Logistic regression measures how relevant or appropriate an independent/predictor variable is (coefficient size) and also reveals the direction of their relationship or association (positive or negative).

$$sigmoid(z) = \frac{1}{1 + e^{-z}}$$

e = Euler's number ~ 2.71828

Sigmoid function converts input into range 0 to 1



$y = m * x + b$

$y = \frac{1}{1 + e^{-(m*x+b)}}$

FIG 7.1.1 GRAPH FOR LOGISTIC REGRESSION

7.2 KNN

KNN is a supervised machine learning algorithm. KNN forms groups based on the criteria's and then decides for the incoming data where to put in which category It can be used for regression and for classification too, but mostly for the classification only its used.

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So, for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs' images and based on the most similar features it will put it in either cat or dog category.

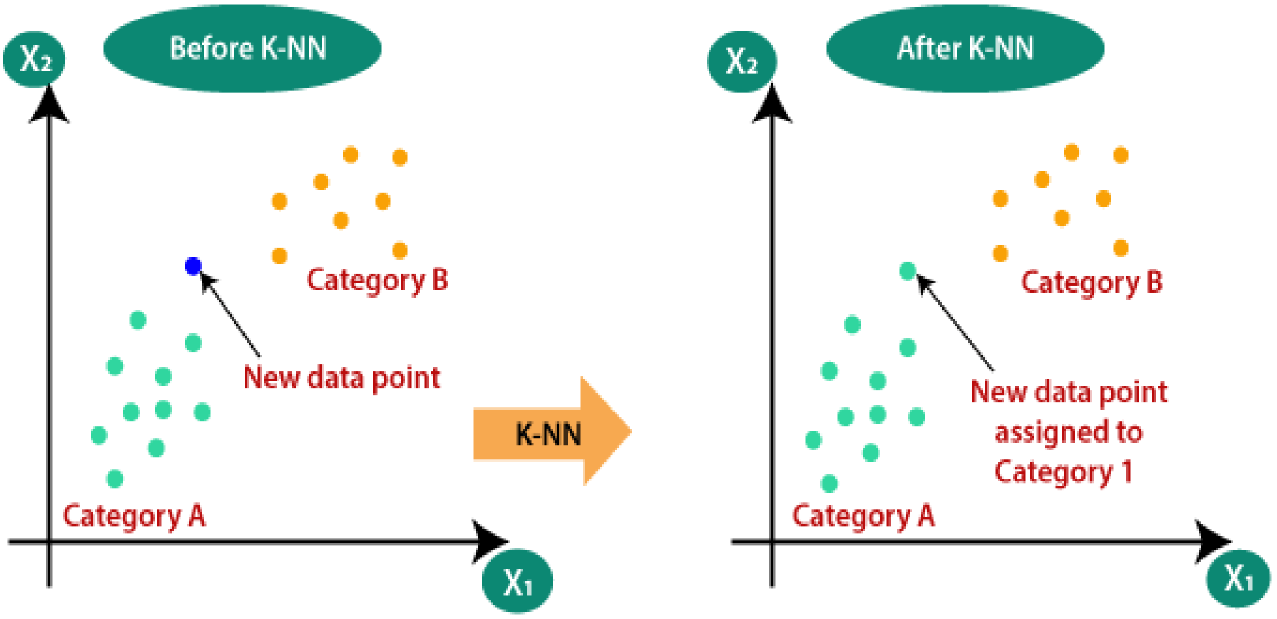


FIG 7.2.1 REPRESENTATION OF KNN ALGORITHM

7.3 RANDOM FOREST CLASSIFIER

Random forest is a supervised learning algorithm. The "forest" it builds is a group of decision trees, usually trained with the “bagging” system.

The general idea of the bagging system is that a combination of learning models increases the overall result.

Put simply: random forest builds multiple decision trees and combines them together to get a more accurate and stable prediction.

One big advantage of random forest is that it can be used for both classification and regression problems, which form the most of current machine learning systems.

A Random Forest Algorithm is a supervised machine learning algorithm that is extremely popular and is used for Classification and Regression problems in Machine Learning.

We know that a forest comprises numerous trees, and the more trees more it will be robust. Similarly, the greater the number of trees in a Random Forest Algorithm, the higher its accuracy and problem-solving ability.

Random Forest is a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

It is based on the concept of ensemble learning which is a process of combining multiple classifiers to solve a complex problem and improve the performance of the model.

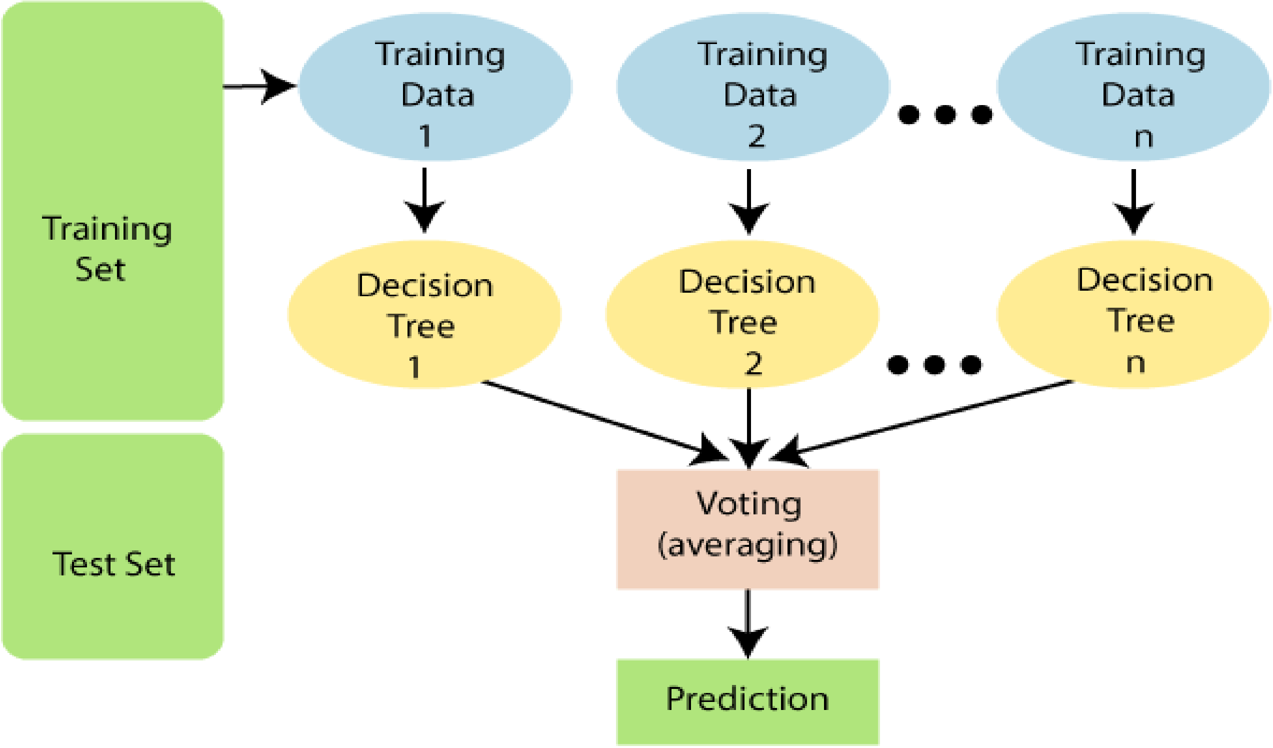


FIG 7.3.1. RANDOM FOREST CLASSIFIER

7.4 DECISION TREE ALGORITHM

A. Decision Tree is a supervised machine learning algorithm.

B. Two nodes which are decision node and leaf node are the ones making the decision.

C. Repeated if clauses are at work when deciding the classification for the algorithm.

A decision tree is **a non-parametric supervised learning algorithm for classification and regression tasks**. It has a hierarchical tree structure consisting of a root node, branches, internal nodes, and leaf nodes. Decision trees are used for classification and regression tasks, providing easy-to-understand models.

A decision tree is a hierarchical model used in decision support that depicts decisions and their potential outcomes, incorporating chance events, resource expenses, and utility.

This algorithmic model utilizes conditional control statements and is non-parametric, supervised learning, useful for both classification and regression tasks. The tree structure is comprised of a root node, branches, internal nodes, and leaf nodes, forming a hierarchical, tree-like structure.

It is a tool that has applications spanning several different areas. Decision trees can be used for classification as well as regression problems.

The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves.

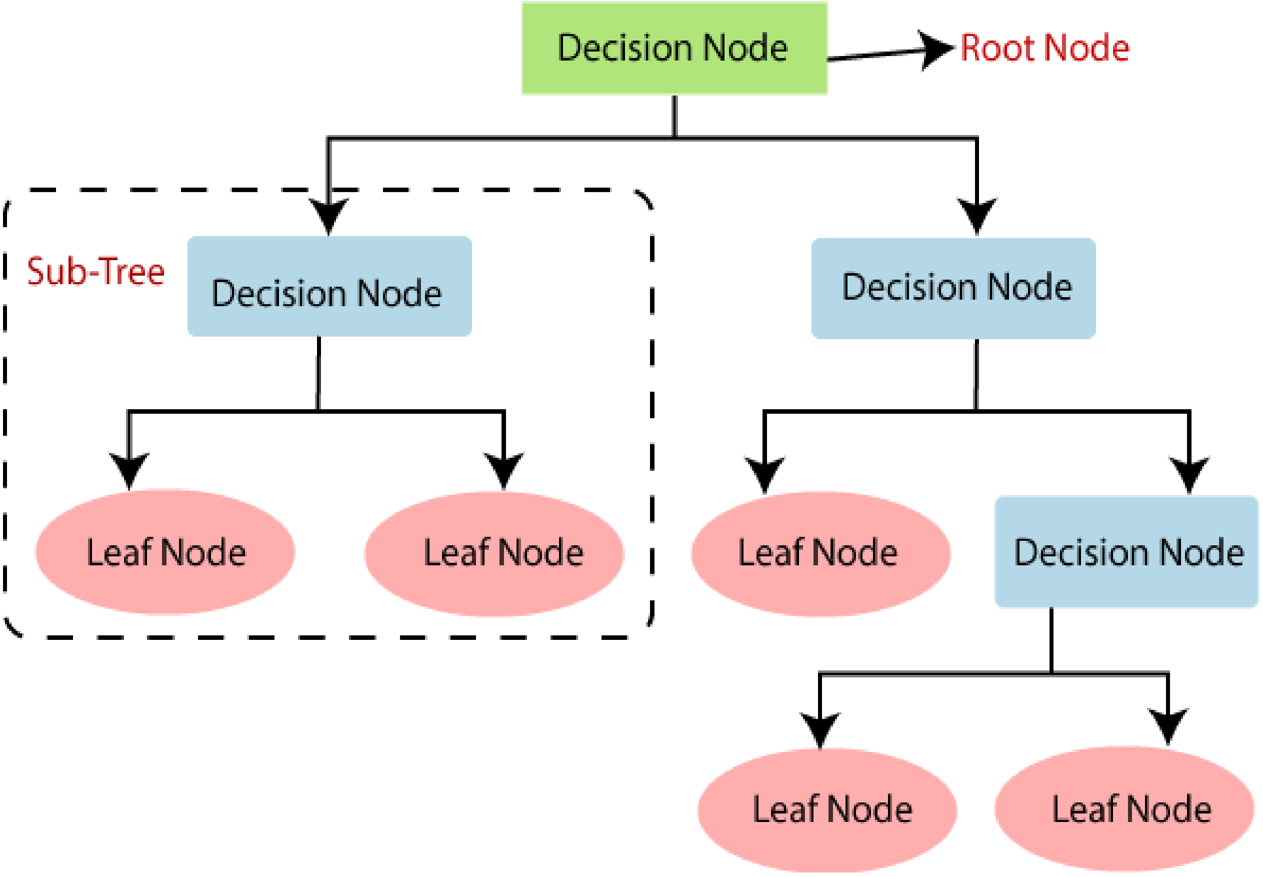


FIG 7.4.1 DECISION TREE ALGORITHM

CHAPTER 8

MODULES

Building a breast cancer prediction system using machine learning involves several steps, from data collection and Preprocessing to model selection and evaluation. Here's a list of modules you might consider for such a project:

Data Collection: Obtain a dataset containing features relevant to breast cancer prediction. You can use publicly available datasets like the Wisconsin Breast Cancer dataset (Wisconsin Diagnostic Breast Cancer (WDBC)) from the UCI Machine Learning Repository.

Data Preprocessing: Data Cleaning: Handle missing values, if any.

Data Transformation: Convert categorical variables into numerical format using techniques like one-hot encoding.

Feature Scaling: Normalize or standardize numerical features to ensure all features contribute equally to the analysis.

Exploratory Data Analysis (EDA): Visualize the distribution of features.

Explore relationships between features. Check for correlations between features and the target variable (presence or absence of breast cancer).

Feature Selection:

Identify the most relevant features for prediction using techniques like correlation analysis, feature importance, or dimensionality reduction methods such as PCA (Principal Component Analysis).

Model Selection:

Choose appropriate machine learning algorithms for classification, such as:

Logistic Regression Decision Trees, Random Forest, Support Vector Machines (SVM), Gradient Boosting algorithms, (XGBoost, LightGBM) Experiment with multiple algorithms and select the one with the best performance metrics.

Model Training: Split the dataset into training and testing sets. Train the selected machine learning models on the training data.

Model Evaluation: Evaluate model performance using appropriate metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Utilize techniques like cross-validation to ensure robustness of the model.

Hyperparameter Tuning:

Optimize the hyperparameters of the selected model using techniques like grid search or random search to improve performance further.

Model Deployment:

Once the model is trained and evaluated, deploy it in a production environment. Create a user-friendly interface for users to input their data for prediction.

Monitoring and Maintenance:

Monitor the deployed model's performance over time. Periodically retrain the model with new data to ensure its accuracy and relevance.

Documentation and Reporting:

Document the entire process, including data Preprocessing steps, model selection criteria, evaluation metrics, and deployment details. Prepare a report summarizing the findings and insights gained from the project.

By breaking down your breast cancer prediction project into these modules, you can systematically approach each stage and build a robust and accurate predictive model.

CHAPTER 9

IMPLEMENTAION

9.1 GENERAL

Implementation phase brings out the design tweaked out into a operational system. Hence this can be deliberated to be most precarious juncture in accomplishing the efficacious system and in convincing the user faith that system will operate and be effective. This phase encompasses vigilant planning & design, examination of prevailing system and constraints on execution, design & scheming of methods to change over.

9.2 PROCEDURE FOLLOWED DURING IMPLEMENTATION

The application – Credit Card Fraud Detection which is in itself the complete & full-fledged GUI enabled application to envisage/foresee the authenticity & legitimacy of a transaction has been implemented, as per the following steps:

- Install Anaconda from a reliable source.
- Import packages: pandas, Scipy, Matplotlib, Seaborn
- Load the dataset, a dataset is the pool of data for analytical/critical purpose, a (.CSV) file.

- Reconnoiter and get through the dataset through data. Shape, data. Describe.

- Determine the count of fraud cases by checking if class is 0 or 1.

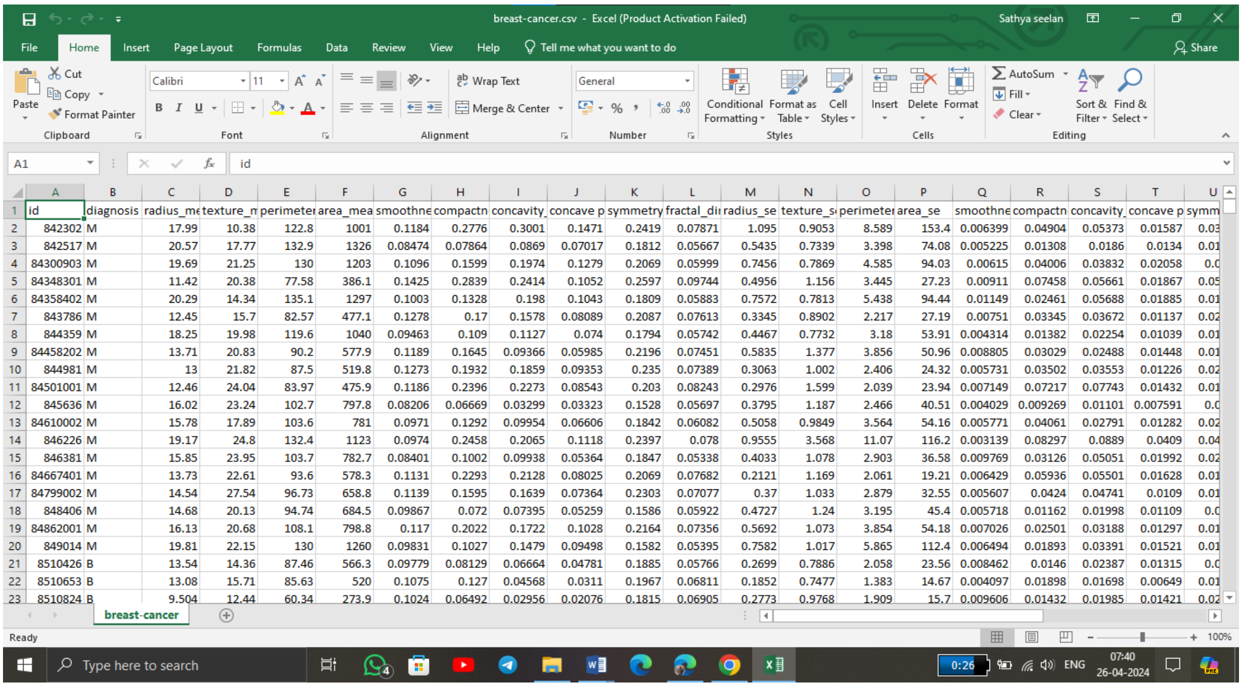
- In the similar procedure, get the correlation matrix.
- Next, there is a need to determine the local outlier factor.

- The GUI is developed using PyQt library.

- The PyQt library, provides tools to achieve a complete GUI enabled application, similar to swings in java environment.

- Define the constructor in the file.

9.3 DATASET DESIGN



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	id	diagnosis	radius_me	texture_m	perimeter	area_mea	smoothne	compactn	concavity_	concave p	symmetry	fractal_di	radius_se	texture_s	perimeter	area_se	smoothne	compactn	concavity_	concave p	symm
2	842302	M	17.99	10.38	122.8	1001	0.1184	0.2776	0.3001	0.1471	0.2419	0.07871	1.095	0.9053	8.589	153.4	0.006399	0.04904	0.05373	0.01587	0.03
3	842517	M	20.57	17.77	132.9	1326	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	0.5435	0.7339	3.398	74.08	0.005225	0.01308	0.0186	0.0134	0.01
4	84300903	M	19.69	21.25	130	1203	0.1096	0.1599	0.1974	0.1279	0.2069	0.05999	0.7456	0.7869	4.585	94.03	0.00615	0.04006	0.03832	0.02058	0.0
5	84348301	M	11.42	20.38	77.58	386.1	0.1425	0.2839	0.2414	0.1052	0.2597	0.09744	0.4956	1.156	3.445	27.23	0.00911	0.07458	0.05661	0.01867	0.05
6	84358402	M	20.29	14.34	135.1	1297	0.1003	0.1328	0.198	0.1043	0.1809	0.05883	0.7572	0.7813	5.438	94.44	0.01149	0.02461	0.05688	0.01885	0.01
7	843786	M	12.45	15.7	82.57	477.1	0.1278	0.17	0.1578	0.08089	0.2087	0.07613	0.3345	0.8902	2.217	27.19	0.00751	0.03345	0.03672	0.01137	0.02
8	844359	M	18.25	19.98	119.6	1040	0.09463	0.109	0.1127	0.074	0.1794	0.05742	0.4467	0.7732	3.18	53.91	0.004314	0.01382	0.02254	0.01039	0.01
9	84458202	M	13.71	20.83	90.2	577.9	0.1189	0.1645	0.09366	0.05985	0.2196	0.07451	0.5835	1.377	3.856	50.96	0.008805	0.03029	0.02488	0.01448	0.01
10	844981	M	13	21.82	87.5	519.8	0.1273	0.1932	0.1859	0.09353	0.235	0.07389	0.3063	1.002	2.406	24.32	0.005731	0.03502	0.03553	0.01226	0.02
11	84501001	M	12.46	24.04	83.97	475.9	0.1186	0.2396	0.2273	0.08543	0.203	0.08243	0.2976	1.599	2.039	23.94	0.007149	0.07217	0.07743	0.01432	0.01
12	845636	M	16.02	23.24	102.7	797.8	0.08206	0.06669	0.03299	0.03323	0.1528	0.05697	0.3795	1.187	2.466	40.51	0.004029	0.009269	0.01101	0.007591	0.0
13	84610002	M	15.78	17.89	103.6	781	0.0971	0.1292	0.09954	0.06606	0.1842	0.06082	0.5058	0.9849	3.564	54.16	0.005771	0.04061	0.02791	0.01282	0.02
14	846226	M	19.17	24.8	132.4	1123	0.0974	0.2458	0.2065	0.1118	0.2397	0.078	0.9555	3.568	11.07	116.2	0.003139	0.08297	0.0889	0.0409	0.04
15	846381	M	15.85	23.95	103.7	782.7	0.08401	0.1002	0.09938	0.05364	0.1847	0.05338	0.4033	1.078	2.903	36.58	0.009769	0.03126	0.05051	0.01992	0.02
16	84667401	M	13.73	22.61	93.6	578.3	0.1131	0.2293	0.2128	0.08025	0.2069	0.07682	0.2121	1.169	2.061	19.21	0.006429	0.05936	0.05501	0.01628	0.01
17	84799002	M	14.54	27.54	96.73	658.8	0.1139	0.1595	0.1639	0.07364	0.2303	0.07077	0.37	1.033	2.879	32.55	0.005607	0.0424	0.04741	0.0109	0.01
18	848406	M	14.68	20.13	94.74	684.5	0.09867	0.072	0.07395	0.05259	0.1586	0.05922	0.4727	1.24	3.195	45.4	0.005718	0.01162	0.01998	0.01109	0.0
19	84862001	M	16.13	20.68	108.1	798.8	0.117	0.2022	0.1722	0.1028	0.2164	0.07356	0.5692	1.073	3.854	54.18	0.007026	0.02501	0.03188	0.01297	0.01
20	849014	M	19.81	22.15	130	1260	0.09831	0.1027	0.1479	0.09498	0.1582	0.05395	0.7582	1.017	5.865	112.4	0.006494	0.01893	0.03391	0.01521	0.01
21	8510426	B	13.54	14.36	87.46	566.3	0.09779	0.08129	0.06664	0.04781	0.1885	0.05766	0.2699	0.7886	2.058	23.56	0.008462	0.0146	0.02387	0.01315	0.0
22	8510653	B	13.08	15.71	85.63	520	0.1075	0.127	0.04568	0.0311	0.1967	0.06811	0.1852	0.7477	1.383	14.67	0.004097	0.01898	0.01698	0.00649	0.01
23	8510824	B	9.504	12.44	60.34	273.9	0.1024	0.06492	0.02956	0.02076	0.1815	0.06905	0.2773	0.9768	1.909	15.7	0.009606	0.01432	0.01985	0.01421	0.02

FIG 9.3.1 DATASET

The dataset commonly used for breast cancer prediction in machine learning is the Wisconsin Breast Cancer dataset, also known as the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. This dataset is publicly available and widely used for research purposes. Here's an explanation of this dataset:

Attributes: The dataset contains features computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. These features describe various characteristics of cell nuclei present in the image. Some of the attributes include:

Radius: Mean, standard error, and worst (largest) radius of the nucleus.

Texture: Mean, standard error, and worst (standard deviation) of gray-scale values in the nucleus.

Perimeter: Mean, standard error, and worst (largest) perimeter of the nucleus.

Area: Mean, standard error, and worst (largest) area of the nucleus.

Smoothness: Mean, standard error, and worst (local variation in radius lengths) of the nucleus.

Compactness: Mean, standard error, and worst (perimeter^2 / area - 1.0) of the nucleus.

Concavity: Mean, standard error, and worst (severity of concave portions of the contour) of the nucleus.

Concave points: Mean, standard error, and worst (number of concave portions of the contour) of the nucleus.

Symmetry: Mean, standard error, and worst (symmetry of the nucleus).

Fractal dimension: Mean, standard error, and worst (coastline approximation) of the nucleus.

Target Variable: The dataset contains a binary target variable indicating the diagnosis of breast cancer. A diagnosis can be either "M" (malignant) or "B" (benign). This is the variable that machine learning models aim to predict based on the features provided.

Number of Instances: The dataset typically contains a few hundred instances, each representing a different patient and their corresponding breast mass characteristics.

Data Format: The dataset is usually provided in a comma-separated values (CSV) format or similar tabular format, where each row represents an instance (patient) and each column represents a feature or the target variable.

Availability: The Wisconsin Breast Cancer dataset is publicly available and can be accessed from various sources, including the UCI Machine Learning Repository and Kaggle.

This dataset is popular among researchers and practitioners in the field of machine learning for its relevance to real-world medical diagnosis tasks and its relatively small size, making it suitable for experimentation and prototyping of breast cancer prediction models.

9.4 PREPROCESSING

The data values have been plotted using histogram describing the numerical distribution of the data values.

After selecting the dataset, the first step is to pre-process the data to make it suitable for model training and testing.

Data Preprocessing is a crucial step in building machine learning models for breast cancer prediction. It involves transforming raw data into a format that is suitable for training machine learning algorithms. Here's how data Preprocessing can be carried out for breast cancer prediction using machine learning:

Handling Missing Values:

Check for missing values in the dataset.

If missing values are present, decide on an appropriate strategy to handle them. This might involve:

Imputing missing values using techniques such as mean, median, mode, or using advanced imputation methods like KNN imputation.

Removing instances with missing values if they are negligible in number.

Dropping features with a high percentage of missing values if they are not informative.

Handling Categorical Data:

If the dataset contains categorical variables, encode them into numerical format, as most machine learning algorithms require numerical input.

One-hot encoding: Convert categorical variables into binary vectors with one column for each category.

Label encoding: Convert categorical labels into numerical labels (e.g., "Malignant" -> 1, "Benign" -> 0). However, be cautious with label encoding as it might introduce unintended ordinal relationships.

Feature Scaling:

Scale numerical features to ensure they have similar ranges. This prevents features with larger scales from dominating the learning process.

Common scaling techniques include Min-Max scaling and Standardization (Z-score normalization).

Feature Selection/Dimensionality Reduction:

Identify and select relevant features that contribute most to the prediction task. This can help reduce noise and overfitting.

Techniques for feature selection include:

Univariate feature selection: Selecting features based on statistical tests like ANOVA or chi-square test.

Feature importance from tree-based models: Using algorithms like Random Forest or Gradient Boosting to rank features based on their importance.

Dimensionality reduction techniques like Principal Component Analysis (PCA) or Singular Value Decomposition (SVD) can be used to reduce the dimensionality of the feature space while retaining most of the variance.

Train-Test Split:

Split the dataset into training and testing sets. The training set is used to train the model, while the testing set is used to evaluate its performance.

Typical splits include 70-30 or 80-20 for training and testing, respectively. Cross-validation can also be used for more robust evaluation.

Normalization of Target Variable (Optional):

If the target variable is imbalanced (i.e., one class significantly outnumbers the other), consider techniques such as oversampling, under sampling, or using class weights during training to address class imbalance.

Data Transformation:

Perform any additional data transformations that might improve model performance, such as log transformations, polynomial transformations, or interactions between features.

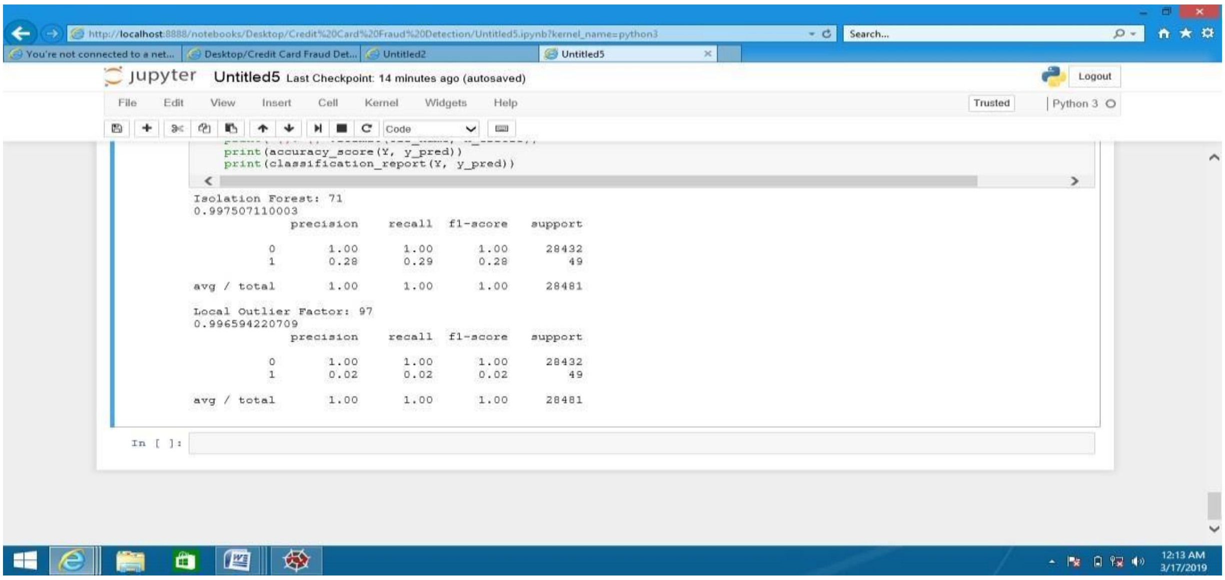
Pipeline Creation:

Build a preprocessing pipeline that encapsulates all the preprocessing steps, ensuring consistency and reproducibility in model training and evaluation.

By performing these data preprocessing steps, you can ensure that your dataset is well-prepared for training machine learning models for breast cancer prediction, leading to more accurate and reliable predictions

9.5 PREDICTION

The prediction that has been achieved using the Isolation Forest Algorithm and Local Outlier Factor Algorithm has been shown below


http://localhost:8888/notebooks/Desktop/Credit%20Card%20Fraud%20Detection/Untitled5.ipynb?kernel_name=python3
Search...
You're not connected to a net...
Desktop/Credit Card Fraud Det...
Untitled2
Untitled5
jupyter
Untitled5 Last Checkpoint: 14 minutes ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Code
12:13 AM
3/17/2019

```
    print(accuracy_score(Y, y_pred))
    print(classification_report(Y, y_pred))
```

```
Isolation Forest: 71
0.997507110003
             precision    recall  f1-score   support

          0       1.00      1.00      1.00     28432
          1       0.28      0.29      0.28        49

avg / total       1.00      1.00      1.00     28481

Local Outlier Factor: 97
0.996594220709
             precision    recall  f1-score   support

          0       1.00      1.00      1.00     28432
          1       0.02      0.02      0.02        49

avg / total       1.00      1.00      1.00     28481
```

```
In [ ]:
```

FIG 9.5.1 ACCURACY

CHAPTER 10

SOFTWARE TESTING

10.1 GENERAL

In a generalized way, we can say that the system testing is a type of testing in which the main aim is to make sure that system performs efficiently and seamlessly. The process of testing is applied to a program with the main aim to discover an unprecedented error, an error which otherwise could have damaged the future of the software. Test cases which brings up a high possibility of discovering and error is considered successful. This successful test helps to answer the still unknown

10.2 TESTING

Table 10.1.1: Tabulated Results

Test Case (sample split)	**Assumption**	**Description**	**Expected Output**	**Actual Output**		**Log Message**
				Isolation Forest Algorithm- Algorithm I Accuracy(%)	**Local Outlier Factor - Algorithm II Accuracy(%)**	
10:90	Algorithm-I will perform better	Check for accuracy at 10% training of data	99.70505	99.75071	99.65942	Success
15:85	Algorithm-II will perform better	Check for accuracy at 15% training of data	99.71675	99.75421	99.67931	Fail
20:80	Algorithm-II will perform better	Check for accuracy at	99.73485	99.69628	99.77352	Success

		20% training of data				
25:75	Algorithm-I will perform better	Check for accuracy at 25% training of data	99.73311	99.77107	99.69523	Success
30:70	Algorithm-I will perform better	Check for accuracy at 30% training of data	99.73425	99.77645	99.69218	Success

The test cases has been based on the following sample split (train: test) :- (10:90), (15:85), (20:80), (25:75) and (30:70).

Outlier Fraction: Describes the ratio of outlier values to the real values in the dataset

Data Shape: Describes the number of rows and columns in the training sample.

Isolation Forest Algorithm Accuracy: Describes the accuracy achieved on the test dataset using Isolation Forest Algorithm

Local Outlier Factor Accuracy: Describes the accuracy achieved on the test dataset using Local Outlier Factor

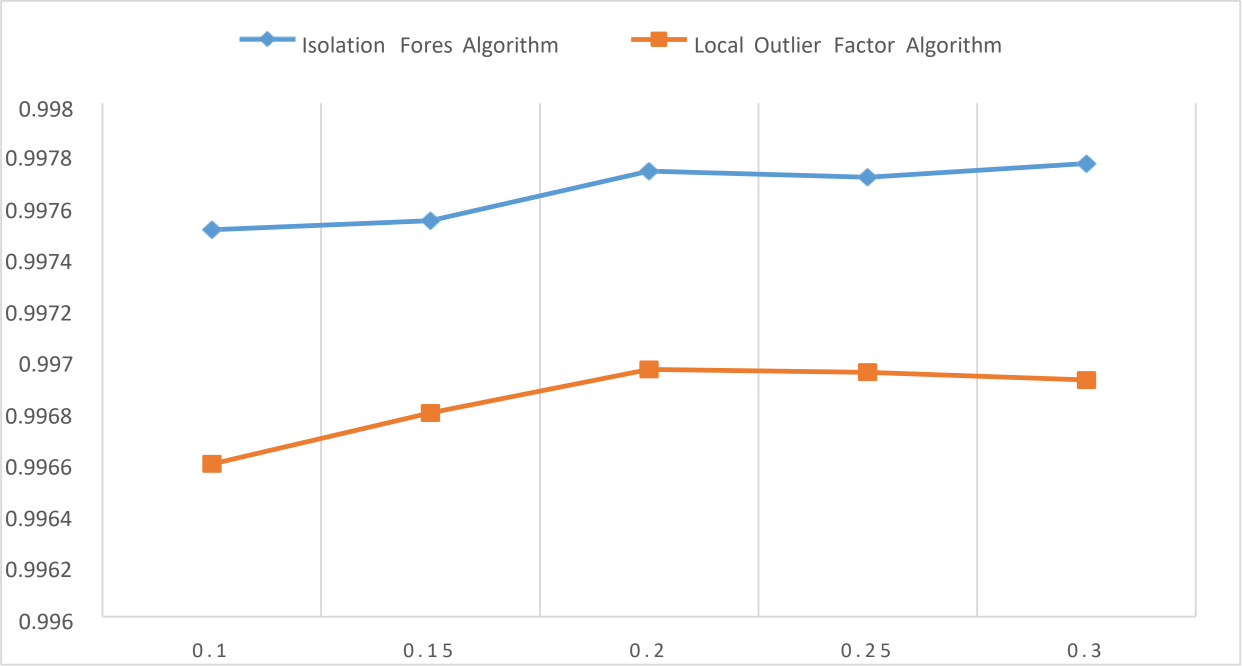


Figure 10.1.1: Comparison Chart

As we tested the application under different test conditions, the application gave appropriate results. The above chart depicts the accuracy based on two algorithms used, i.e. the Isolation Forest Algorithm and the Local Outlier Factor Algorithm.

CHAPTER 11

CODING

```
import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

import missingno as msno

import warnings

warnings.filterwarnings('ignore')

sns.set()

plt.style.use('ggplot')


df = pd.read_csv("C:/Users/lenovo/OneDrive/Desktop/archive/breast-cancer.csv")


df.head()


# Supervised-> target

# Unsupervised

df.describe()


msno.bar(df, color="red")
```

```
df['diagnosis'] = df['diagnosis'].apply(lambda val:1 if val=='M' else 0)

plt.hist(df['diagnosis'])

plt.title('Diagnosis(M=1, B=0)')

plt.show()


# each 5 row its having 6 columns

# density graph

plt.figure(figsize=(20,15))

plotnumber=1

for column in df:

    if plotnumber<=30:

        ax = plt.subplot(5,6, plotnumber)

        sns.distplot(df[column])

        plt.xlabel(column)

    plotnumber+=1

plt.tight_layout()

plt.show()


# heatmap

plt.figure(figsize=(20,12))

corr=df.corr()

mask = np.triu(np.ones_like(corr, dtype=bool))
```

```
sns.heatmap(corr, mask=mask, linewidths=1, annot=True, fmt = ".2f")
plt.show()


# feature selection
corr_matrix = df.corr().abs()
mask = np.triu(np.ones_like(corr_matrix, dtype=bool))
tri_df = corr_matrix.mask(mask)
to_drop = [x for x in tri_df.columns if any(tri_df[x]>0.92)]
df = df.drop(to_drop, axis=1)
print(df.shape[1])


X=df.drop('diagnosis', axis=1)
y=df['diagnosis']


# scaling data
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

```
from sklearn.linear_model import LogisticRegression
log_reg = LogisticRegression()
log_reg.fit(X_train, y_train)

from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
print(accuracy_score(y_train, log_reg.predict(X_train)))
log_reg_acc = accuracy_score(y_test, log_reg.predict(X_test))
print(log_reg_acc)
y_pred = log_reg.predict(X_test)
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))

# KNN
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(X_train, y_train)

from sklearn.metrics import accuracy_score, confusion_matrix,
classification_report
print(accuracy_score(y_train, knn.predict(X_train)))
knn_acc = accuracy_score(y_test, knn.predict(X_test))
```

```
print(knn_acc)

y_pred = knn.predict(X_test)

print(confusion_matrix(y_test, y_pred))

print(classification_report(y_test, y_pred))


# SVC

#Hyperparameter tuning

from sklearn.svm import SVC

from sklearn.model_selection import GridSearchCV

svc= SVC(probability=True)

parameters = {

    'gamma': [0.0001, 0.001, 0.01, 0.1],

    'C':[0.01, 0.05, 0.5, 0.1, 1,10, 15,20]

}

grid_search = GridSearchCV(svc, parameters)

grid_search.fit(X_train, y_train)


from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

print(accuracy_score(y_train, svc.predict(X_train)))

svc_acc = accuracy_score(y_test, svc.predict(X_test))

print(svc_acc)
```

```
y_pred = svc.predict(X_test)

print(confusion_matrix(y_test, y_pred))

print(classification_report(y_test, y_pred))


#diagnosis count

diagnosis_counts = df['diagnosis'].value_counts()

plt.figure(figsize=(19,9))

plt.bar(diagnosis_counts.index, diagnosis_counts.values, color=['pink', 'skyblue'])

plt.xlabel('')

plt.ylabel('')

plt.title('Diagnostic count', fontsize=20)

plt.xticks([0 ,1 ], ['M', 'B'])

plt.show()


corr_diagnosis = corr['diagnosis']

corr_diagnosis_sorted = corr_diagnosis.sort_values()

corr_diagnosis_sorted = corr_diagnosis_sorted.drop('diagnosis')

fig, ax = plt.subplots(figsize=(19, 9))

corr_diagnosis_sorted.plot(kind='barh', ax=ax)

plt.title('Correlation with the Diagnosis', fontsize=20)

plt.show()
```

```
model = LogisticRegression(max_iter=10000)
model.fit(X_train, y_train)
y_predict = model.predict(X_test)
conf_matrix = confusion_matrix(y_test, y_predict)
print("Confusion Matrix:")
print(conf_matrix)

class_report = classification_report(y_test, y_predict)
print("\nClassification Report:")
print(class_report)

#ACCURACY
accuracy = accuracy_score(y_test, y_predict)
print("\nModel Accuracy:")
print(accuracy)
```

CHAPTER 12

OUTPUT

12.1 PREDICTED DATA FROM BAR CHART

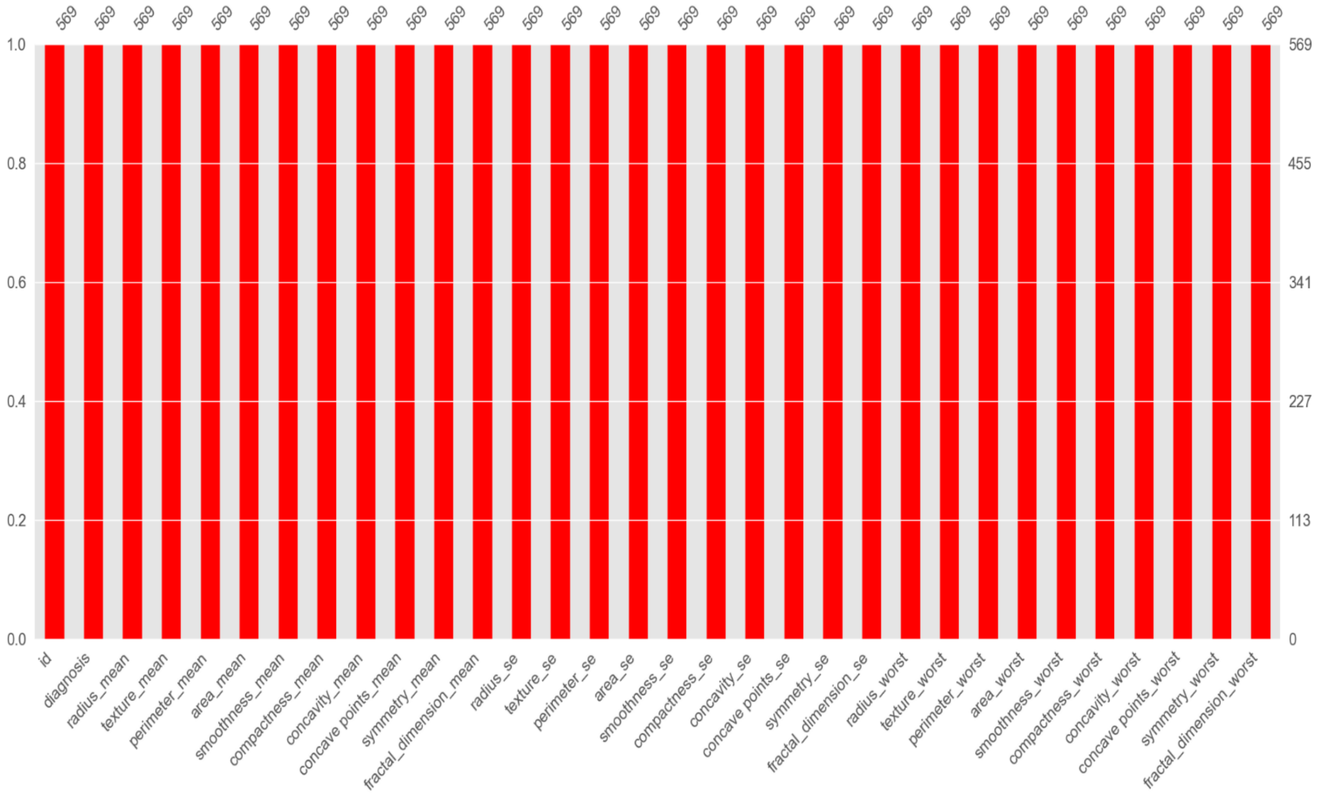


FIG 12.1.1 PREDICTED VALUES

12.2 DIAGNOSIS

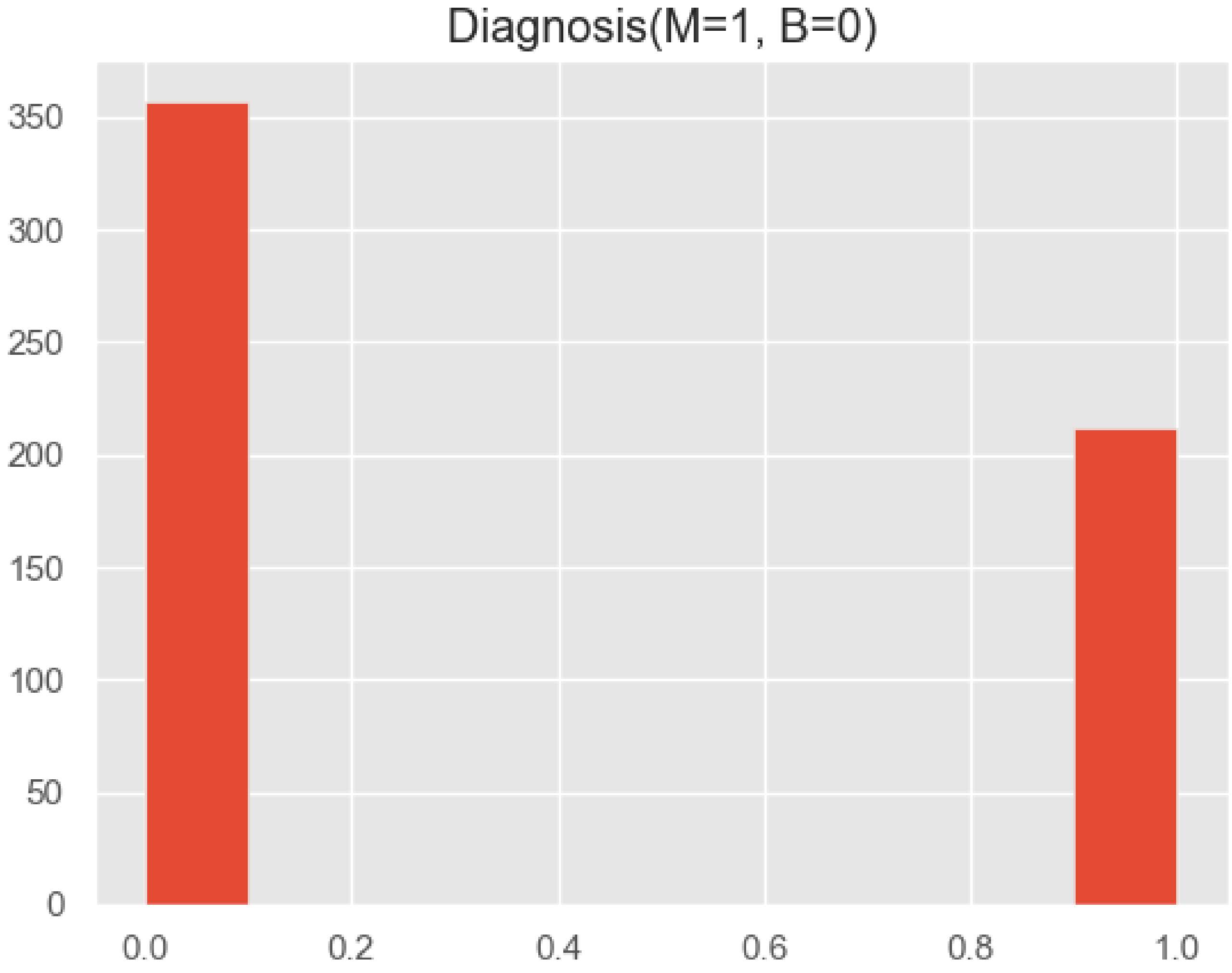


FIG 12.2.1 DIAGNOSIS

12.3 DATA PREPROCESSING

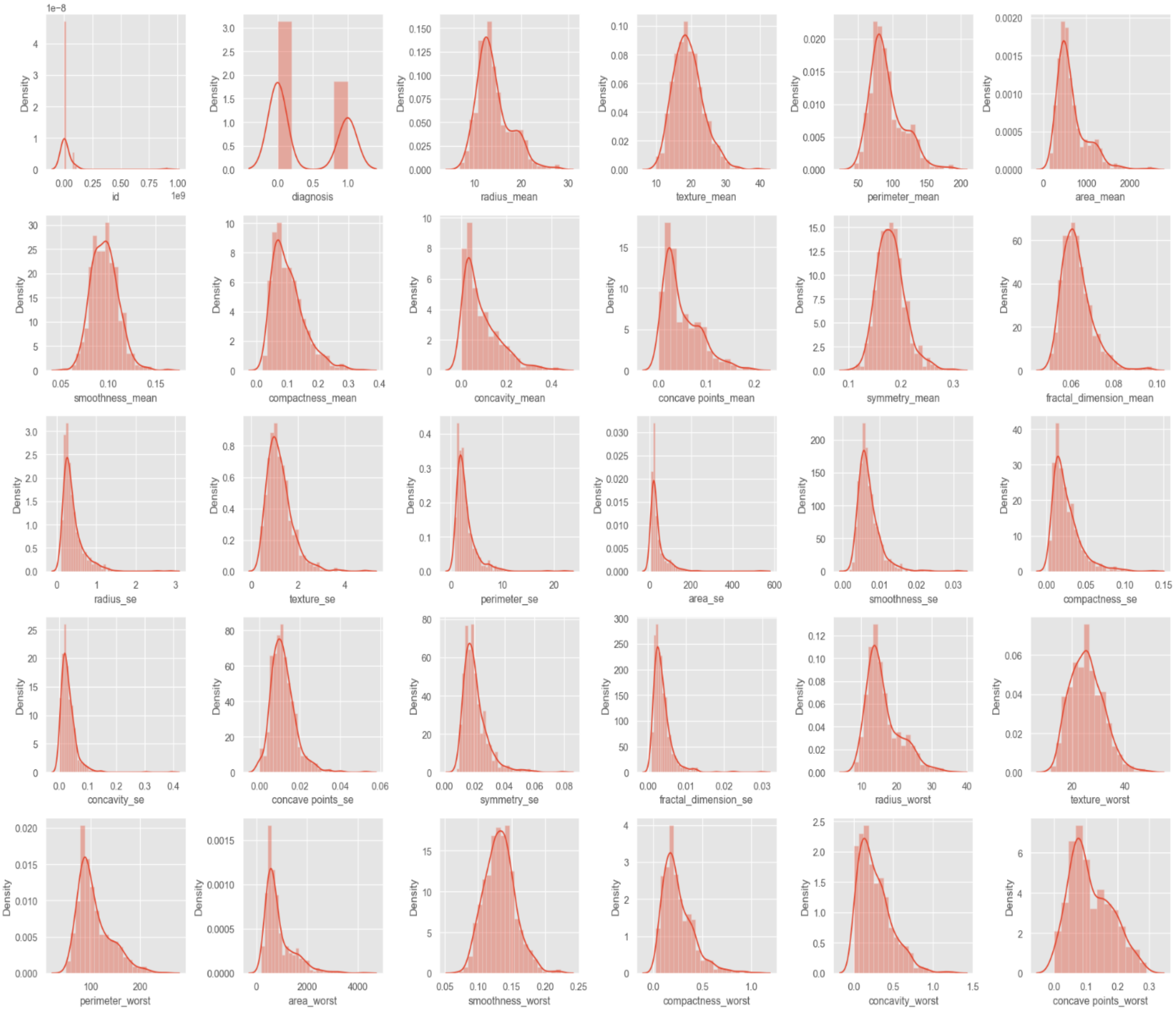


FIG 12.3.1 DATA PREPROCESSING

12.4 HEAT MAP

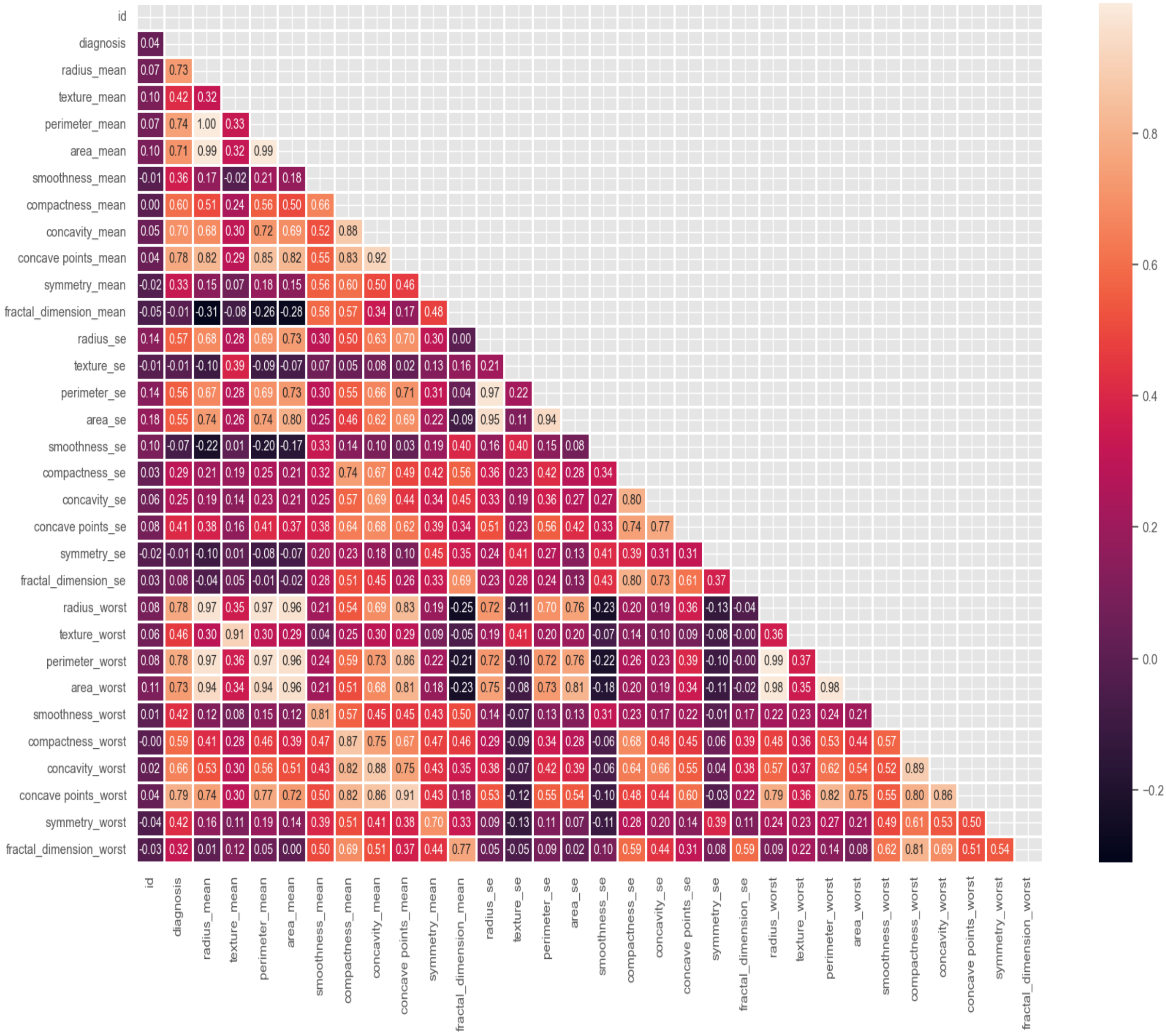


FIG 12.4.1 HEAT MAP

12.5 DIAGNOSIS COUNT

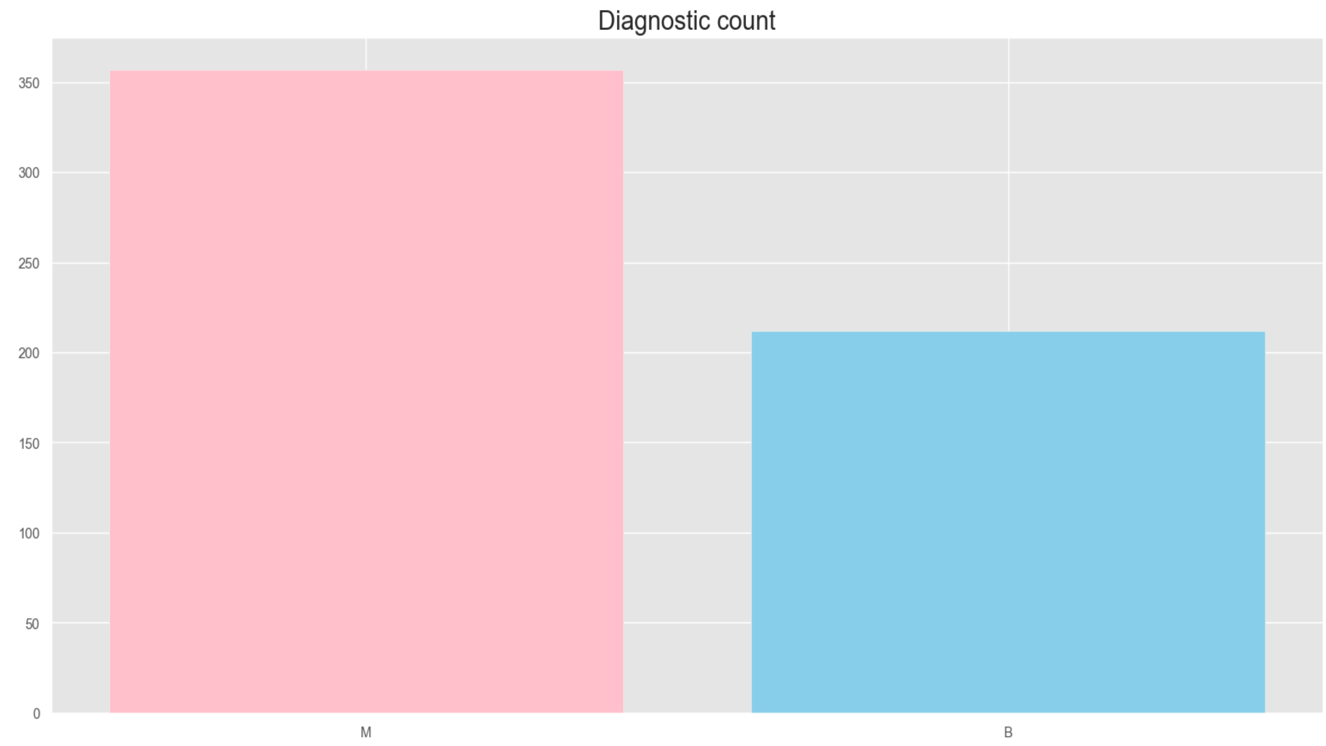


FIG 12.5.1 DIAGNOSIS COUNT

12.6 CORRELATION WITH THE DIAGNOSIS

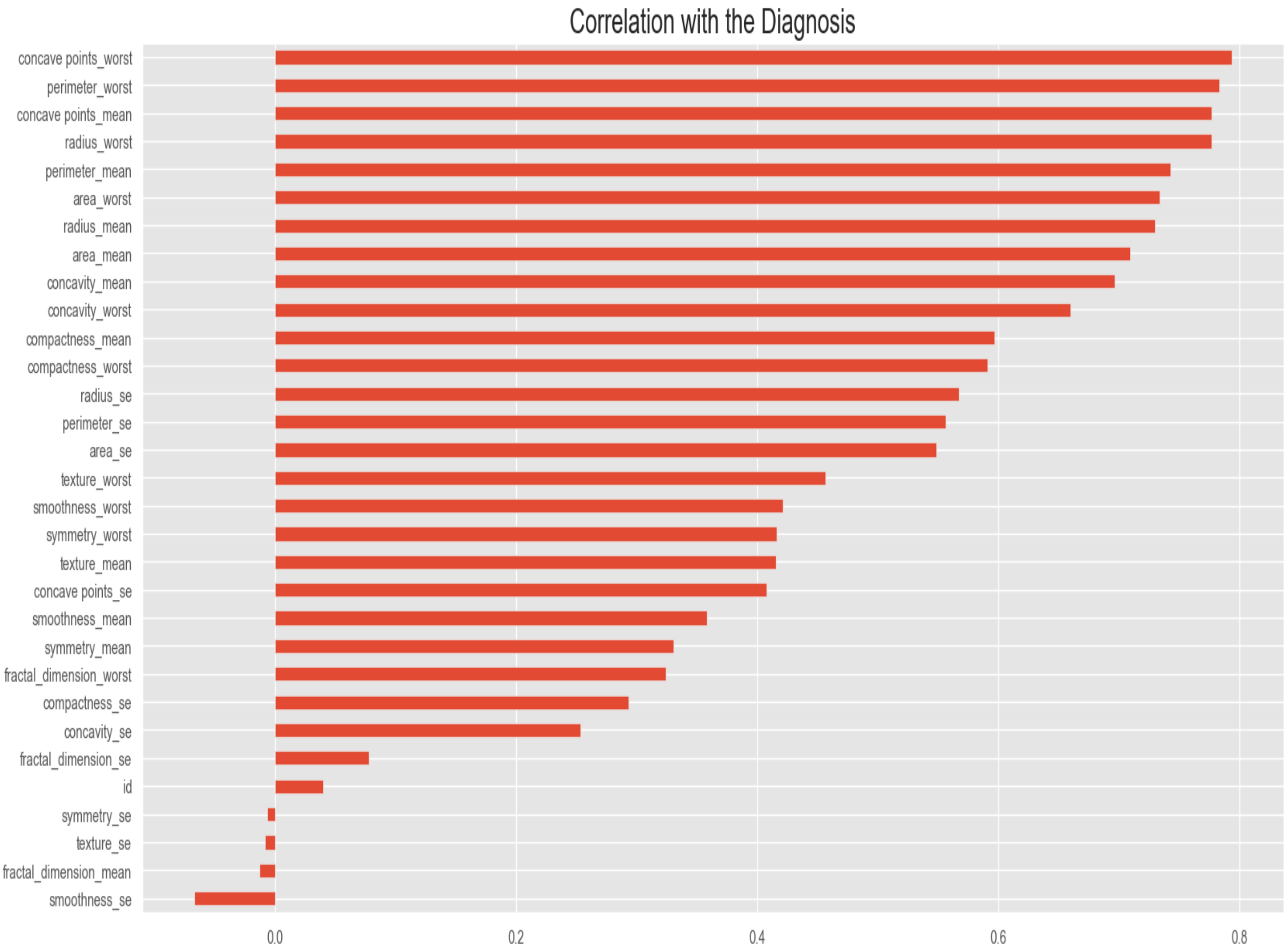
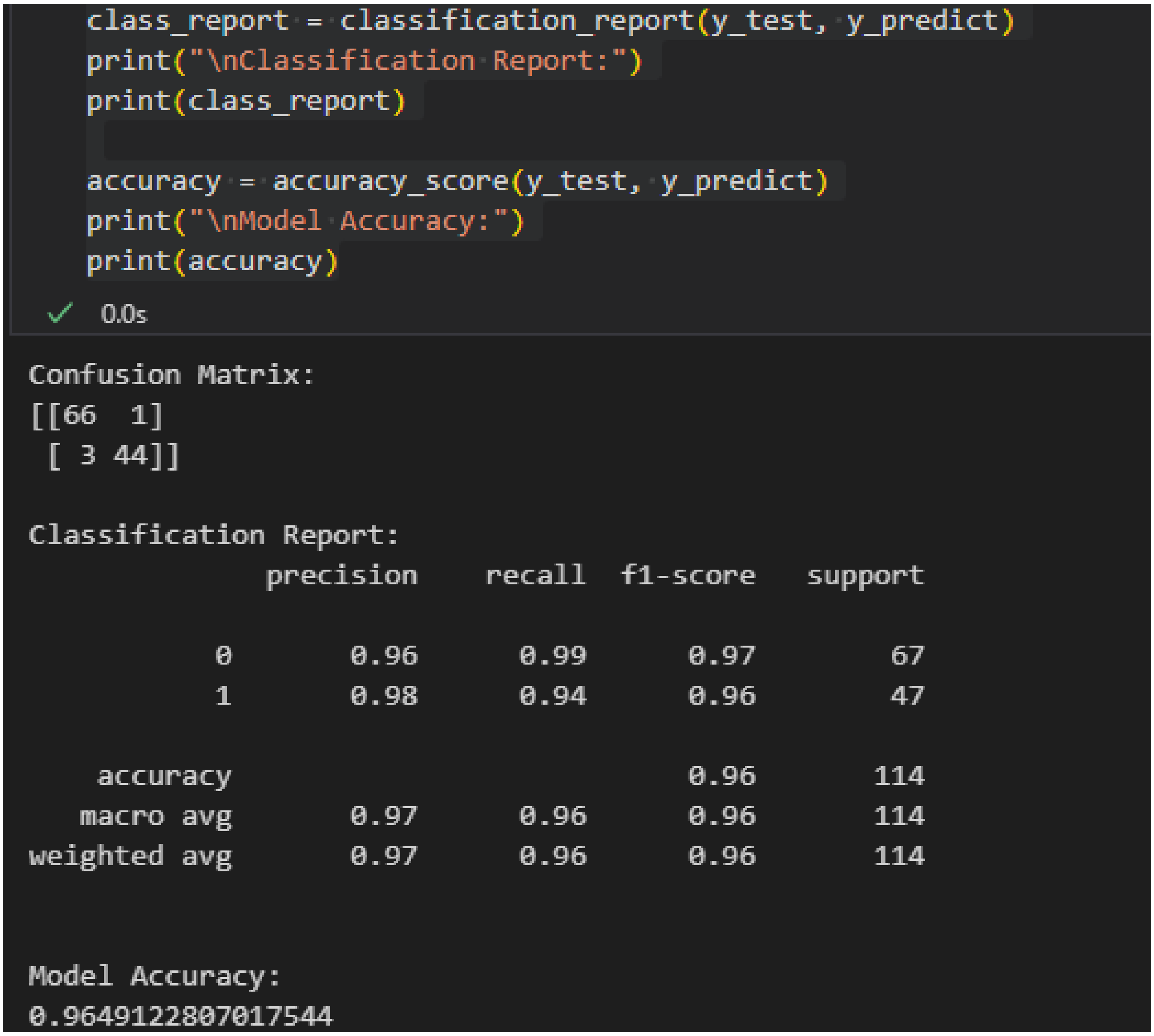


FIG 12.6.1 CORRELATION WITH THE DIAGNOSIS

12.7 ACCURACY



```
class_report = classification_report(y_test, y_predict)
print("\nClassification Report:")
print(class_report)

accuracy = accuracy_score(y_test, y_predict)
print("\nModel Accuracy:")
print(accuracy)
```

✓ 0.0s

```
Confusion Matrix:
[[66  1]
 [ 3 44]]

Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.99      0.97        67
           1       0.98      0.94      0.96        47

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


Model Accuracy:
0.9649122807017544
```

FIG 12.7.1 ACCURACY

CHAPTER 13

CONCLUSION AND FUTURE ENHANCEMENTS

13.1 CONCLUSION

In conclusion, leveraging machine learning (ML) for breast cancer prediction holds significant promise in advancing early detection and improving patient outcomes. Here's a summary of key points and potential future enhancements:

Accuracy and Reliability: ML models have demonstrated impressive accuracy in breast cancer prediction by analyzing various factors such as patient demographics, medical history, genetic markers, and imaging data. Continued efforts should focus on refining these models to enhance accuracy and reliability.

Early Detection: ML algorithms can help identify subtle patterns indicative of early-stage breast cancer, enabling timely intervention and treatment. Future research could explore integrating real-time data streams and wearable technology to enhance early detection capabilities.

Personalized Medicine: ML enables personalized risk assessment and treatment planning by considering individual patient characteristics and genetic profiles. Future enhancements may involve integrating multi-omics data (genomics, proteomics, metabolomics) to tailor treatments based on molecular subtypes and disease progression.

Interpretability and Explainability: Enhancing the interpretability and Explainability of ML models is crucial for gaining trust from healthcare providers and patients. Future research should focus on developing transparent and interpretable ML techniques that provide insights into model predictions.

Data Quality and Accessibility: Access to high-quality and diverse datasets is essential for training robust ML models. Future efforts should prioritize data collection initiatives, ensure data privacy and security, and promote data sharing collaborations among healthcare institutions.

Clinical Integration: Successful integration of ML-based breast cancer prediction tools into clinical practice requires collaboration between data scientists, clinicians, and regulatory bodies. Future enhancements should focus on developing user-friendly interfaces, integrating decision support systems into electronic health records, and validating models through rigorous clinical trials.

Ethical Considerations: Addressing ethical concerns related to data bias, privacy, and equity is crucial for the responsible deployment of ML in healthcare. Future research should prioritize ethical guidelines and frameworks to ensure fair and equitable access to predictive tools and minimize unintended consequences.

Continued Research and Innovation: Breast cancer is a complex disease with evolving challenges, necessitating ongoing research and innovation in ML-based prediction models. Future enhancements should explore novel techniques such as deep learning, federated learning, and transfer learning to further improve prediction accuracy and generalizability.

In summary, while ML-based breast cancer prediction shows great potential, continuous research, collaboration, and innovation are necessary to address existing challenges and realize the full benefits of these predictive models in clinical practice

13.2 FUTURE ENHANCEMENT

Enhancing breast cancer prediction using machine learning (ML) involves several potential avenues for improvement. Here are some ideas:

Feature Selection and Engineering: Identify and select the most relevant features for prediction. This could involve exploring genetic markers, imaging data, patient demographics, and lifestyle factors. Feature engineering might involve transforming or combining features to better capture predictive patterns.

Advanced Algorithms: Explore more sophisticated ML algorithms beyond traditional logistic regression or decision trees. Deep learning approaches, such as convolutional neural networks (CNNs) for image data or recurrent neural networks (RNNs) for sequential data, could uncover complex patterns in breast cancer data.

Ensemble Methods: Combine multiple models to improve prediction accuracy. Techniques like random forests, gradient boosting, or stacking can harness the strengths of different algorithms and mitigate their weaknesses.

Imbalanced Data Handling: Address the class imbalance problem inherent in medical datasets, where the number of cancer cases is often much smaller than non-cancer cases. Techniques like oversampling, under sampling, or using algorithms designed to handle imbalanced data (e.g., SMOTE) can improve model performance.

Regularization and Optimization: Apply regularization techniques to prevent overfitting and optimize hyperparameters to fine-tune model performance. This might involve techniques like cross-validation, grid search, or Bayesian optimization.

Integration of Multi-Modal Data: Combine different types of data sources, such as genetic data, histopathological images, and clinical records, to provide a more comprehensive picture for prediction. Fusion techniques can integrate information from diverse sources effectively.

Interpretability and Explainability: Ensure that the ML models provide interpretable results, especially in medical contexts where understanding the rationale behind predictions is crucial for acceptance and trust. Techniques like SHAP values or LIME can provide insights into model predictions.

Continuous Learning and Adaptation: Implement systems that can continuously learn from new data and adapt over time. This could involve updating models with new patient data or integrating feedback from healthcare professionals to refine predictions.

Privacy Preservation: Develop techniques to ensure patient privacy while still allowing for collaborative learning across multiple healthcare institutions. Federated learning or secure multiparty computation approaches can facilitate model training without sharing sensitive patient data.

Validation and Clinical Trials: Validate the performance of ML models in real-world clinical settings through rigorous testing and clinical trials. Collaboration with healthcare professionals is essential to ensure that predictive models meet the standards of clinical practice and provide meaningful benefits to patients.

By pursuing these avenues, researchers and practitioners can work towards more accurate, reliable, and clinically useful breast cancer prediction models using machine learning.

CHAPTER 14

REFERENCES

1. Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. Computational and Structural Biotechnology Journal, 13, 8-17.

2. Cruz-Roa, A., Gilmore, H., Basavanhally, A., Feldman, M., Ganesan, S., Shih, N., ... & Madabhushi, A. (2013). Accurate and reproducible invasive breast cancer detection in whole-slide images: A Deep Learning approach for quantifying tumor extent. Scientific reports, 7(1), 46450.

3. Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Ghosh, A., Mondal, S. K., & Sharma, N. (2020). Early Detection of Breast Cancer Using Machine Learning Techniques: A Review. SN Computer Science, 1(3), 161.

4. Hosny, A., Parmar, C., Quackenbush, J., Schwartz, L. H., Aerts, H. J., & Artificial Intelligence in Radiology (AIR) Study Group. (2018). Artificial intelligence in radiology. Nature Reviews Cancer, 18(8), 500-510.

5. Dhungel, N., Carneiro, G., Bradley, A. P., & Aneja, S. (2017). A deep learning approach for the analysis of masses in mammograms with minimal user intervention. Medical image analysis, 37, 114-128.

6. Becker, A. S., Marcon, M., Ghafoor, S., Wurnig, M. C., Frauenfelder, T., Boss, A., & Guckenberger, M. (2018). Deep learning in mammography: diagnostic accuracy of a multipurpose image analysis software in the detection of breast cancer. Investigative radiology, 53(11), 647-654.

7. Zhang, J., Zhang, W., Fan, Y., Li, L., & Yu, H. (2017). Automatic breast cancer detection and classification using deep learning techniques. Journal of Medical Systems, 41(8), 132.

8. Al-Antari, M. A., Al-Masni, M. A., Choi, M. T., Han, S. M., & Kim, T. S. (2017). A Fully integrated computer-aided diagnosis system for digital X-ray mammograms via deep learning detection, segmentation, and classification. International Journal of Medical Informatics, 107, 174-183.

9. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

10. Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018). Detecting and classifying lesions in mammograms with Deep Learning. Scientific reports, 8(1), 4165.

11. Zeng, N., Du, K. L., Liu, J., & Min, F. (2019). An explainable breast cancer histopathological image classification framework based on synergic deep learning. Knowledge-Based Systems, 179, 1-9.

12. Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... & Kim, R. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. Jama, 316(22), 2402-2410.

13. Dey, N., Ashour, A. S., Shi, F., Sifaki-Pistolla, D., & Zhang, Y. D. (2019). A Novel Hybrid Technique for the Prediction of Breast Cancer Survivability Using Supervised Machine Learning and Statistical Optimization. Journal of Personalized Medicine, 9(1), 10.

14. Aksu, H., Kaya, S., & Cömert, Z. (2021). A new approach based on machine learning algorithms for breast cancer detection. Health Information Science and Systems, 9(1), 1-8.

15. Parikh, R. B., Kakad, S., Bates, D. W., & Saria, S. (2019). Predictive analytics and artificial intelligence to improve health care. JAMA, 320(11), 1101-1102.

16. Park, S. H., & Han, K. (2018). Methodologic guide for evaluating clinical performance and effect of artificial intelligence technology for medical diagnosis and prediction. Radiology, 286(3), 800-809.

17. Han, S., Kang, H. K., & Jeong, J. Y. (2017). Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm. Healthcare informatics research, 23(1), 46-53.

18. Wang, S., Yang, D. M., Rong, R., Zhan, X., Xiao, G., & Liao, X. (2018). Artificial intelligence in lung cancer pathology image analysis. Cancers, 10(9), 403.

19. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115-118.

20. Samuel, J. A., & Chandrasekar, C. (2019). Breast cancer prediction using machine learning algorithms: a review. International Journal of Engineering Research in Computer Science and Engineering, 6(7), 112-115.

21. Nanni, L., Brahnam, S., & Lumini, A. (2017). Breast cancer diagnosis from biopsy images using texture features and SVM. Engineering Applications of Artificial Intelligence, 64, 38-44.

22. Al-Antari, M. A., Al-Masni, M. A., Choi, M. T., Han, S. M., & Kim, T. S. (2017). A Fully integrated computer-aided diagnosis system for digital X-ray mammograms via deep learning detection, segmentation, and classification. International Journal of Medical Informatics, 107, 174-183.