VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

G. Jayalakshmi

Reg. No:20BCS0145.

Csc3005

Fundamentals of data analytics

Faculty: Dr. Rajesh kalori

PREDICTION OF BREAST CANCER BASED ONDYNAMIC CLASSIFICATION MACHINE LEARNING TECHNIQUE

ABSTRACT

Traditional methods of determining breast cancer resulted in minimum accuracy and performance were not satisfactory. After the Advancements in technology of Al and Robotics in Healthcare were used. The aim of this project is to develop a breast cancer prediction model using various machine learning techniques These models can predict any deviations in the normal pattern. The models analyzed are KNN, Naive Bayes, Decision Tree, SVC and Random Forest. The different performance analyzed for all techniques are accuracy, precision and recall Random Forest results in superior accuracy in comparison to other technique

INTRODUCTION

INTRODUCTION

Machine learning is one of the application of Artificial Intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmer. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves. As it is evident from the name it gives the computer that which makes it more similar to humans. The ability to learn. Using this advanced data science we can find any deviation in normal patterns and can get clinical consequences for health care Like Breast cancer, brain tumour, Diabetes analysis, etc

OBJECTIVES OF THE PROJECT

In this project I am going to apply supervised ML. classification techniques using 5 classifiers like Random Forest, Naive Bayes, Decision Tree, SVM and KNN to find the performance of the each clarifiers for prediction. The performance metrics analyzed are classification report. precision, all, accuracy score and confusion matrix

REAL TIME APPLICATIONS OF MACHINE LEARNING

The following are the real time applications of machine learning in our day-to day life.

- VPAS
- Recommendations on social media
- Predictions
- Image Recognition, Speech Recognition
- Medical Diagnosis
- Intelligent Gaming
- Self-Driving Cars

MACHINE LEARNING WITH PYTHON

- Python is a widely used high-level programming language for general purposeprogramming.
- Apart from being open source programming language, python is a great object-oriented,interpreted, and interactive programming language.

- Python combines remarkable power with very clear syntax. It has modules, classes,exceptions, very high level dynamic data types, and dynamic typing.
- It's simple to learn, Open source, data handling capacity, and in-built libraries

MACHINE LEARNING IN CANCER PREDICTION

- Machine learning offers way to find patterns and examine unstructured data in healthcare.
- Prediction of breast cancer cells using classification techniques
- Classifying Malignant or Benign cell based on the properties of the tissue
- Analysis of diabetes, tumor, cancer prediction can be best done using supervised learningalgorithms.

MACHINE LEARNING IN HEALTHCARE

- Breast cancer detection and prediction
- In diabetes Research
- In heart-disease prediction analysis
- Brain tumor diagnosis
- Cancer prediction
- Skin cancer image detection

1.2 LITERATURE REVIEWS

LITERATURE REVIEW 1

YEAR: 2019

• The Concept of classification and learning will suit well to medical applications, especially

Those that need complex diagnostic measurements.

• From the available studies it is evident that classification and learning methods can be used

Effectively to improve the accuracy of predicting a disease and its recurrence. In the present

Work classification techniques namely Support Vector Machine (SVM) and Random Forest

[RF] are used [1]

LITERATURE REVIEW 2

YEAR: 2018

• During their life, among 8% of women are diagnosed with Breast cancer (BC), after lung

Cancer, BC in the second popular cause of death in both developed and undeveloped

Worlds. BC is characterized by the mutation of genes, constant pain, changes in the size,

Color, tiredness), skin texture of breasts. Classification of breast cancer leads pathologists

To find a systematic and objective prognostic generally the most frequent classification is

Binary (benign cancer/malign cancer).

• In this paper, we present two different classifiers: Naïve Bayes (NB) classifier and knee

Rest neighbor (KNN) for breast cancer classification. We propose a comparison between

The two new implementations and evaluate their accuracy using cross validation Results

Show that KNN gives the highest accuracy (97.51%) with lowest error rate then NB

Classifier (96.19 %).

LITERATURE REVIEW 3

YEAR: 2017

• This study presents a system with textural features for classifying benign and malignant

Breast tumors on medical ultrasound systems.[3]

• A series of pathologically proven breast tumors were evaluated using the support vector

Machine (SVM) in the differential diagnosis of breast tumors.

• The main advantage of the proposed system is that the training and diagnosis procedure of

SVM are faster and more stable than that of multilayer perception neural networks

• The SVM is a reliable choice for the proposed system because it is fast and

excellent in

Ultrasound image classification.

LITERATURE REVIEW 4

YEAR: 2016

- In this paper[4], we compare two state-of-the-art classification techniques characterizing masses as either benign or malignant, using a dataset consisting of 271 cases (131 benign and 140 malignant), containing both a MLO and CC view.
- For suspect regions in a digitized mammogram, 12 out of 81 calculated image features have been selected for investigating the classification accuracy of support vector machines (SVMs) and Bayesian networks (BNS)
- Classifiers used are bayesian and svm for classifying the size of the tumor.

LITERATURE REVIEW 5

YEAR: 2015

- Sumalatha & Archana [32] studied different data mining techniques for early diagnosis and prediction of breast cancer. The research work analyses the J48 and ZeroR algorithms to predict breast cancer.
- These two algorithms were applied using WEKA. Total instances of ZeroR analysis were699. The three major steps used in this research, the collection of datasets, data preprocessing and classification.

LITERATURE REVIEW 6

YEAR: 2014

Deviatal. [33] investigated automated diagnosis of breast cancer based on a machine learning algorithm. The proposed approach was a three steps process. In the first step, the data were grouped into a number of clusters using the Farthest First clustering algorithm. Due to shrinking the size of the dataset, the computation time reduced greatly. In the second step, outliers are detected in breast cancer dataset using ODA (Outlier Detection Algorithm). The thirdstep identifies whether the cancer is benign or malignant in the pre-processed data set using J48classification algorithm. Wisconsin Breast Cancer Dataset (WBCD) and Wisconsin Diagnosis Breast Cancer (WDBC) was used to test the efficacy of the proposed system. The experiments were performed using WEKA (Waikato Environment for Knowledge Analysis) version 3.7.13.

Experimental results proved that the two steps proposed approach serves to be the best comparedto the existing research for the same data set. The highest accuracy was 99.9% for WBCD data set and 99.6% for WDBC data set. This research will help the doctors to diagnose breast cancer and thereby helping the patients in recovery.

LITERATURE REVIEW 7

YEAR: 2013

Chidambaranathan [36] used a hybrid algorithm of k-means and ELM to predict breast cancer. The k-means algorithm is responsible for clustering tumors based on the extracted features. Each cluster represents a specific tumor pattern. ELM was extended to the generalized SLFNs which effectively classifies with greater detection accuracy in a lesser amount of time. A hybrid algorithm of k-means and ELM is retained the extracted features as input after that the image is classified with SVM as normal, benign or malignant. The specificity, sensitivity, j accord distance, and accuracy are calculated. Results show that the proposed system works better
than the others to predict breast cancer.

LITERATURE REVIEW 8

YEAR: 2012

Lavanya et al. [37] presented breast cancer prediction system based on a hybrid approach; classification and regression trees (CART) classifier with feature selection and bagging technique for higher classification accuracy and improved diagnosis. They used the hybrid approach to enhance the classification accuracy of breast cancer and Feature Selection to remove irrelevant attributes that do not play any role in the classification task. The

Bagging means Bootstrap aggregation was used to classify the data with good accuracy. Data were
collected from machine learning repository of UCI where experiment three breast cancer datasets (Breast Cancer, Breast Cancer Wisconsin (original), Breast Cancer Wisconsin (diagnostic). The Breast Cancer Dataset contained 286 Instances and 10 Attributes; the Original Dataset contained 699 Instances and 11 Attributes. While the Diagnostic Dataset contained 569Instances and 32 Attributes, all previews dataset with two classes.

LITERATURE REVIEW 9

YEAR: 2011

Majal etal. [40] presented a system for diagnosis and prognosis of cancer using Classification and Association approach in Data Mining. The FP algorithm was used association Rule Mining approach to find the frequent patterns for the diagnosis of breast cancer type (benign and malignant). The researchers also used the Decision Tree algorithm in the classification approach was used to predict the prognosis of breast cancer based on three predictor attributes. The three attributes were age, gender, and intensity of symptoms to achieve a goal attribute (disease) which can be predicted from symptoms. Wisconsin data set was used; it contained 699
records and nine attributes. The researchers found that the accuracy of the diagnosis analysis is highly acceptable and can help the medical professionals in decision making to predict early diagnosis and avoid a biopsy.

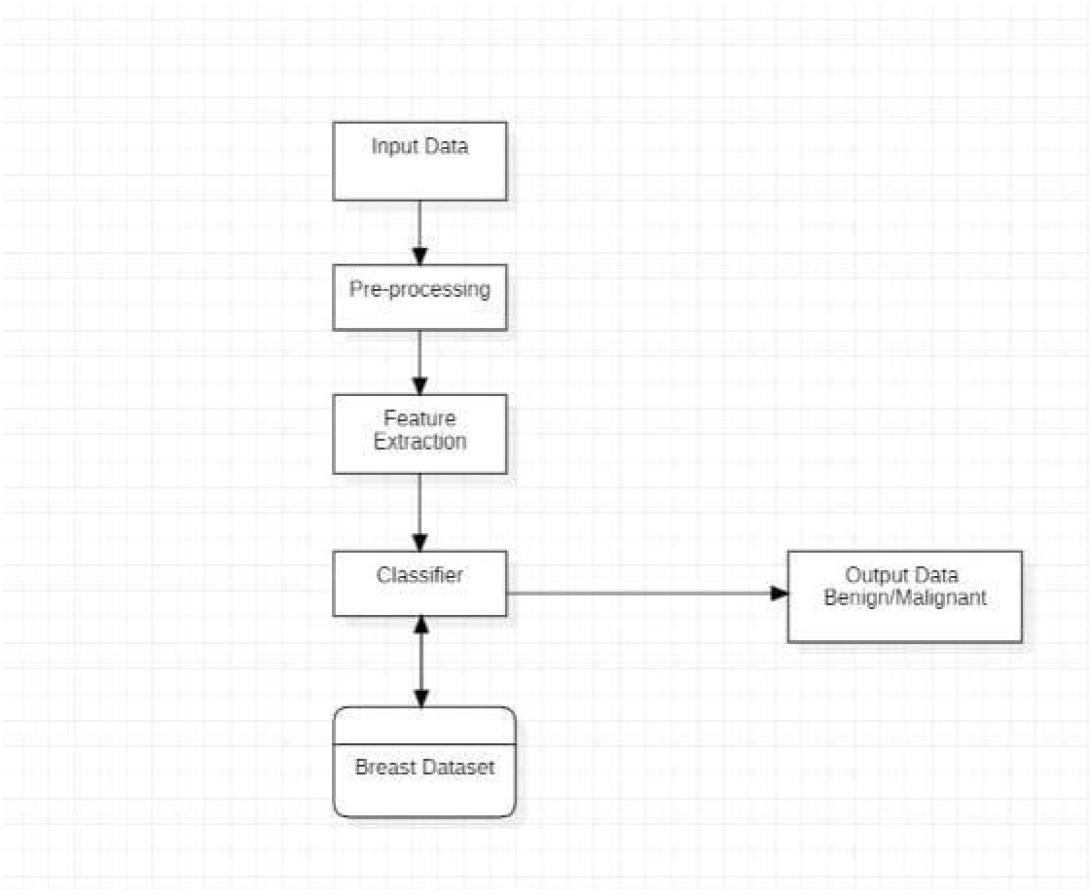
LITERATURE REVIEW 10

YEAR: 2010

Chandrasekar et al. [44] studied breast cancer prediction using data mining techniques. The study aimed to develop accurate prediction models for breast cancer with a neural network classification technique. An ensemble approach was used for possible improvements. The classification

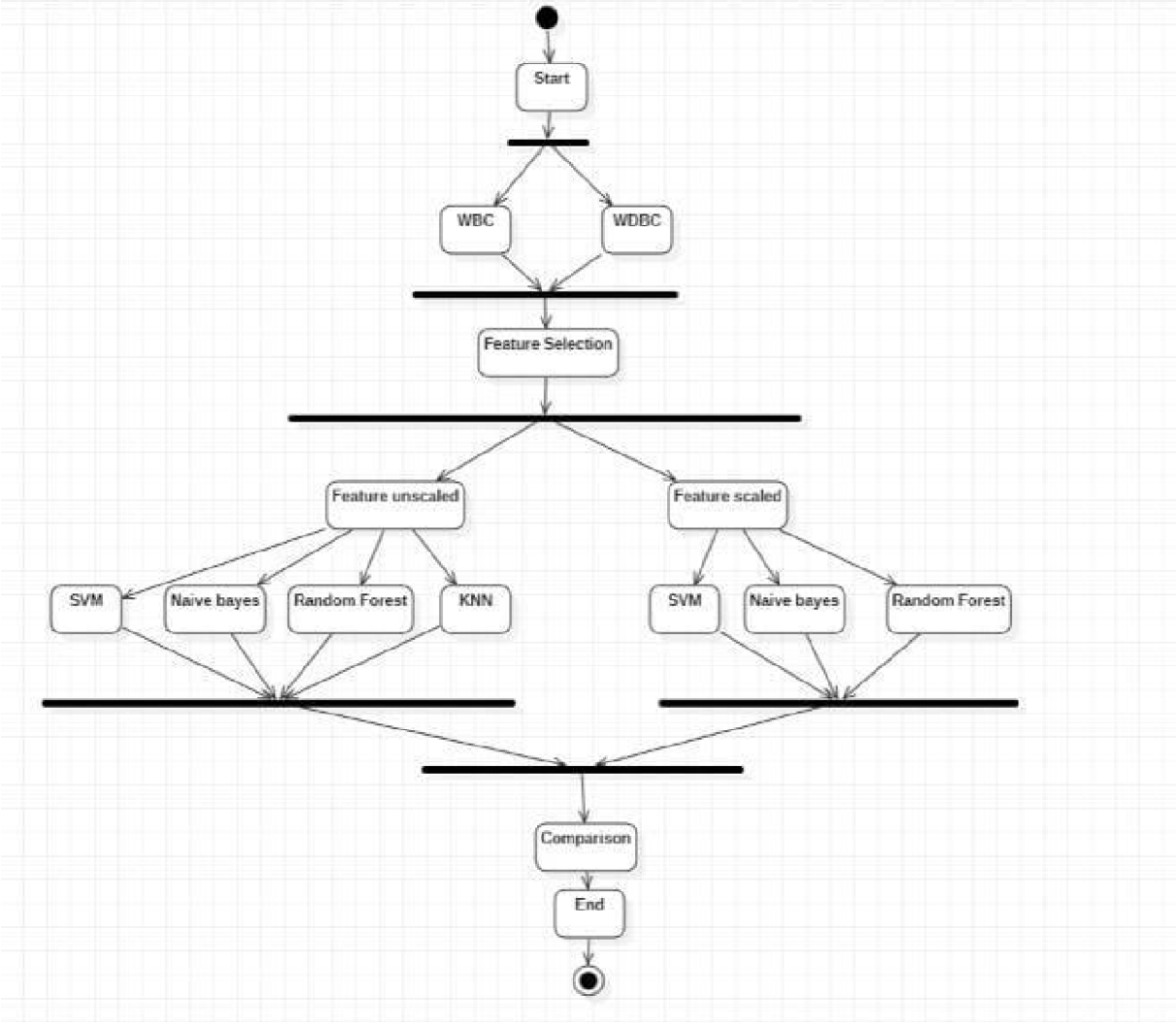
techniques included Lazy IBK, Tree Random Forest, Lazy K Star classifier, and Rules NNge were applied. Data were collected from the WBCD dataset. The experiment was analyzed by WEKA software. The dataset contained 286 instances which 201 of them were benign and 85 were malignant. These instances were described by 10 Attributes such as age,
tumor size, and class. In conclusion, Tree Random classifier achieved a classification accuracy of 98%. The researchers proposed using to analyze Ensemble classifier for 100% accuracy

SYSTEM DESIGN and SYSTEM ARCHITECTURE



UML/ER DIAGRAM

ACTIVITY DIAGRAM



MODULE DESCRIPTION

HARDWARE REQUIREMENTS

Windows RAM: 1GB or more memory

Hard-disk drive: 250 GB Hard-disk drive: 500 GB

SOFTWARE REQUIREMENTS

- IDLE PYTHON 3.7 (32-bit)
- Python packages for machine learning
- Python packages for GUI Programming

3.2 PYTHON PACKAGES DESCRIPTION

- Pandas - csv operation
- Scikit-learn - for ml problems
- Matplotlib - data visualization
- Seaborn - data visualization
- Numpy - scientific computing
- Tkinter - GUI Programming

DATASET DESCRIPTION

- The name of the dataset is Breast Cancer Wisconsin (Diagnostic) Dataset.
- The dataset has 569 instances
- Multivariate dataset
- Applicable for classification
- Output labels: Malignant or Benign
- Attributes: There are 32 attributes. Few of them are radius, texture, smoothness,concavity etc.

IMPLEMENTATION

STEPS IN MACHINE LEARNING

- Data Collection
- Data Preparation
- Choose the model
- Train the model
- Evaluate the model
- Parameter tuning
- Make predictions

IMPLEMENTATION MODULES

- **Preprocessing** - Data preprocessing is a data mining technique that involves transforming raw data into an understandable format. Real world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues.

- **Feature selection** - Feature Selection is the process where you automatically or manually select those features which contribute most to your prediction variable or output in which you are interested in. Having irrelevant features in your data can decrease the accuracy

ofthe models and make your model learn based on irrelevant features.

- **Classification** is a technique where we categorize data into a given number of classes. The main goal of a classification problem is to identify the category/class to which a new data bydeployingclassifierslike

ALGORITHM used

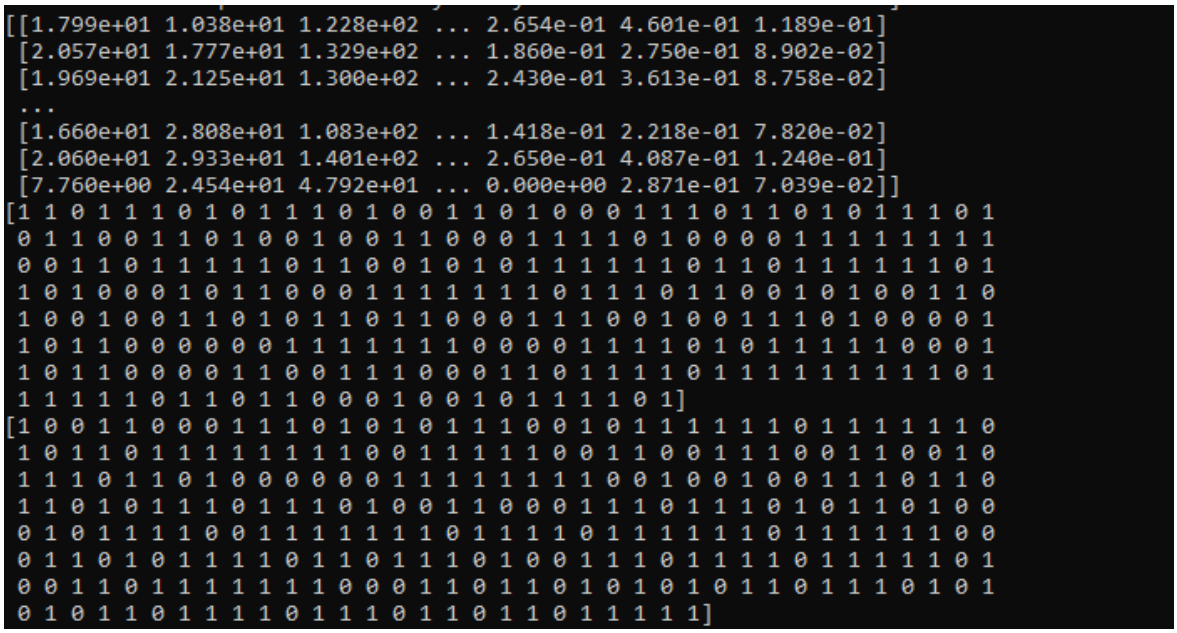
- Decision tree classifier.
- Random forest classifier.

IMPLEMENTATION

(GUI Window)

```
Dataset Selected
['malignant' 'benign']
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 1 0 0 0 0 0 0 0 0 1 0 1 1 1 1 1 0 0 1 0 0 1 1 1 1 0 1 0 0 1 1 1 1 0 1 0 0
 1 0 1 0 0 1 1 1 0 0 1 0 0 0 1 1 1 0 1 1 0 0 1 1 1 0 0 1 1 1 1 0 1 1 0 1 1
 1 1 1 1 1 1 0 0 0 1 0 0 1 1 1 0 0 1 0 1 0 0 1 0 0 1 1 0 1 1 0 1 1 1 1 0 1
 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 1 0 1 1 0 0 1 1 0 0 1 1 1 1 1 0 1 1 0 0 0 1 0
 1 0 1 1 1 0 1 1 0 0 1 0 0 0 0 1 0 0 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 0 0 1 1
 1 0 1 1 1 1 1 0 0 1 1 0 1 1 0 0 1 0 1 1 1 1 0 1 1 1 1 1 1 0 1 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 1 0 1 1 0 1 1 0 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1
 1 0 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 1 0 1 1 1 1 0 0 0 1 1
 1 1 0 1 0 1 0 1 1 1 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0
 0 1 0 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 1 1 0 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1
 1 0 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 0 1 1 1 1 1 0 1 1
 0 1 0 1 1 0 1 0 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 0 1
 1 1 1 1 1 1 0 1 0 1 1 0 1 1 1 1 1 1 0 0 1 0 1 0 1 1 1 1 1 0 1 1 0 1 0 1 0 0
 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 0 0 0 0 0 0 1]
['mean radius' 'mean texture' 'mean perimeter' 'mean area'
 'mean smoothness' 'mean compactness' 'mean concavity'
 'mean concave points' 'mean symmetry' 'mean fractal dimension'
 'radius error' 'texture error' 'perimeter error' 'area error'
 'smoothness error' 'compactness error' 'concavity error'
 'concave points error' 'symmetry error' 'fractal dimension error'
 'worst radius' 'worst texture' 'worst perimeter' 'worst area'
 'worst smoothness' 'worst compactness' 'worst concavity'
 'worst concave points' 'worst symmetry' 'worst fractal dimension']
[[1.799e+01 1.038e+01 1.228e+02 ... 2.654e-01 4.601e-01 1.189e-01]
 [2.057e+01 1.777e+01 1.329e+02 ... 1.860e-01 2.750e-01 8.902e-02]
 [1.969e+01 2.125e+01 1.300e+02 ... 2.430e-01 3.613e-01 8.758e-02]
 ...
 [1.660e+01 2.808e+01 1.083e+02 ... 1.418e-01 2.218e-01 7.820e-02]
 [2.060e+01 2.933e+01 1.401e+02 ... 2.650e-01 4.087e-01 1.240e-01]
 [7.760e+00 2.454e+01 4.792e+01 ... 0.000e+00 2.871e-01 7.039e-02]]
[1 1 0 1 1 1 0 1 0 1 1 1 0 1 0 0 1 1 0 1 0 0 0 1 1 1 0 1 1 0 1 0 1 1 1 0 1
 0 1 1 0 0 1 1 0 1 0 0 1 0 0 1 1 0 0 0 1 1 1 1 0 1 0 0 0 0 1 1 1 1 1 1 1 1
 0 0 1 1 0 1 1 1 1 1 0 1 1 0 0 1 0 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 0 1
 1 0 1 0 0 0 1 0 1 1 0 0 0 1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 0 0 1 0 1 0 0 1 1 0
 1 0 0 1 0 0 1 1 0 1 0 1 1 0 1 1 0 0 0 1 1 1 0 0 1 0 0 1 1 1 0 1 0 0 0 0 1
 1 0 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 1 1 1 1 0 1 0 1 1 1 1 1 0 0 0 1
 1 0 1 1 0 0 0 0 1 1 0 0 1 1 1 0 0 0 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1
 1 1 1 1 1 0 1 1 0 1 1 0 0 0 1 0 0 1 0 1 1 1 1 1 0 1]
```

(Dataset loading and classification 1)



(Dataset loading and classification 2)

Decision tree classifier

```
Decision Tree Classifier

accuracy =  0.9192982456140351
[[ 92   6]
 [ 17 170]]
              precision    recall  f1-score   support

           0       0.84      0.94      0.89        98
           1       0.97      0.91      0.94       187

    accuracy                           0.92       285
   macro avg       0.90      0.92      0.91       285
weighted avg       0.92      0.92      0.92       285

precision =  0.9659090909090909
recall =  0.9090909090909091
```

Random forest classifier

```
Random Forest Classifications

accuracy =  0.9578947368421052
[[ 91   7]
 [  5 182]]
              precision    recall  f1-score   support

           0       0.95      0.93      0.94        98
           1       0.96      0.97      0.97       187

    accuracy                           0.96       285
   macro avg       0.96      0.95      0.95       285
weighted avg       0.96      0.96      0.96       285

precision =  0.9629629629629629
recall =  0.9732620320855615
```

Non feature scaling:

```
   mean radius  mean texture  mean perimeter  mean area  mean smoothness  ...  worst concavity  worst concave points  worst symmetry  worst fractal dimension  target
0        17.99         10.38          122.80     1001.0          0.11840  ...           0.7119                0.2654          0.4601                  0.11890     0.0
1        20.57         17.77          132.90     1326.0          0.08474  ...           0.2416                0.1860          0.2750                  0.08902     0.0
2        19.69         21.25          130.00     1203.0          0.10960  ...           0.4504                0.2430          0.3613                  0.08758     0.0
3        11.42         20.38           77.58      386.1          0.14250  ...           0.6869                0.2575          0.6638                  0.17300     0.0
4        20.29         14.34          135.10     1297.0          0.10030  ...           0.4000                0.1625          0.2364                  0.07678     0.0

[5 rows x 31 columns]
   mean radius  mean texture  mean perimeter  mean area  ...  worst concavity  worst concave points  worst symmetry  worst fractal dimension
0        17.99         10.38          122.80     1001.0  ...           0.7119                0.2654          0.4601                  0.11890
1        20.57         17.77          132.90     1326.0  ...           0.2416                0.1860          0.2750                  0.08902
2        19.69         21.25          130.00     1203.0  ...           0.4504                0.2430          0.3613                  0.08758
3        11.42         20.38           77.58      386.1  ...           0.6869                0.2575          0.6638                  0.17300
4        20.29         14.34          135.10     1297.0  ...           0.4000                0.1625          0.2364                  0.07678

[5 rows x 30 columns]
0    0.0
1    0.0
2    0.0
3    0.0
4    0.0
Name: target, dtype: float64
Training X input feature:  (455, 30)
Testing X input feature:  (114, 30)
Training Y input feature:  (455,)
Testing Y input feature:  (114,)
             predicted_cancer  predicted_healthy
is_cancer                  66                  0
is_healthy                  8                 40
              precision    recall  f1-score   support

         0.0       1.00      0.83      0.91        48
         1.0       0.89      1.00      0.94        66

    accuracy                           0.93       114
   macro avg       0.95      0.92      0.93       114
weighted avg       0.94      0.93      0.93       114
```

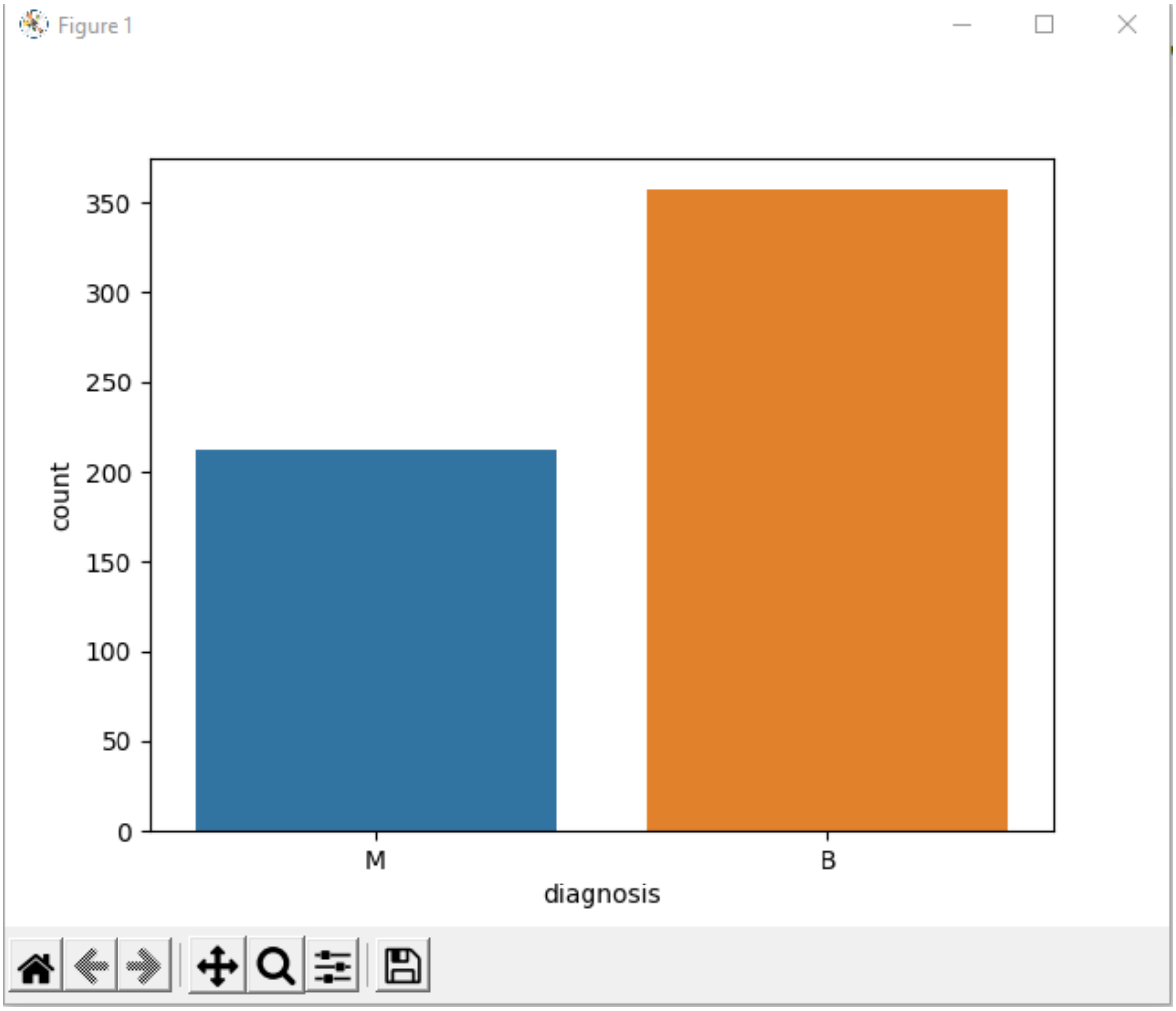
```
mean radius                6.981000
mean texture              10.380000
mean perimeter            43.790000
mean area                143.500000
mean smoothness            0.052630
mean compactness           0.019380
mean concavity             0.000000
mean concave points        0.000000
mean symmetry              0.106000
mean fractal dimension     0.049960
radius error               0.111500
texture error              0.360200
perimeter error            0.757000
area error                 6.802000
smoothness error           0.001713
compactness error          0.002252
concavity error            0.000000
concave points error       0.000000
symmetry error             0.007882
fractal dimension error    0.000895
worst radius               7.930000
worst texture             12.490000
worst perimeter           50.410000
worst area               185.200000
worst smoothness           0.071170
worst compactness          0.027290
worst concavity            0.000000
worst concave points       0.000000
worst symmetry             0.156500
worst fractal dimension    0.055040
dtype: float64
mean radius               28.11000
mean texture              39.28000
mean perimeter           188.50000
mean area               2501.00000
mean smoothness            0.14470
mean compactness           0.34540
mean concavity             0.42680
mean concave points        0.20120
mean symmetry              0.30400
mean fractal dimension     0.09296
radius error               2.87300
texture error              4.88500
```

After scaling:

```
worst radius                28.110000
worst texture               37.050000
worst perimeter            200.790000
worst area                4068.800000
worst smoothness             0.151430
worst compactness            1.030710
worst concavity              1.252000
worst concave points         0.291000
worst symmetry               0.420900
worst fractal dimension      0.152460
dtype: float64
     mean radius  mean texture  mean perimeter  mean area  ...  worst concavity  worst c
412     0.114345      0.391003        0.110290   0.053150  ...         0.149201
461     0.967343      0.549827        0.988943   1.000000  ...         0.545767
532     0.317052      0.205882        0.303849   0.183245  ...         0.096326
495     0.373373      0.340138        0.361620   0.227953  ...         0.135783
13      0.419755      0.469550        0.414000   0.271135  ...         0.185463
..           ...           ...             ...        ...  ...              ...
218     0.606702      0.386851        0.593670   0.460870  ...         0.288898
223     0.415022      0.341522        0.406399   0.262057  ...         0.317572
271     0.203938      0.092042        0.196531   0.103712  ...         0.101837
474     0.184533      0.181315        0.183954   0.091368  ...         0.268770
355     0.264045      0.300692        0.263493   0.145196  ...         0.190735

[455 rows x 30 columns]
            predicted_cancer  predicted_healthy
is_cancer                 61                  5
is_healthy                 0                 48
              precision    recall  f1-score   support

         0.0       0.91      1.00      0.95        48
         1.0       1.00      0.92      0.96        66

    accuracy                           0.96       114
   macro avg       0.95      0.96      0.96       114
weighted avg       0.96      0.96      0.96       114

accuracy Aftere scaling = 0.956140350877193
```

Before scaling:

```
   mean radius  mean texture  mean perimeter  mean area  mean smoothness  ...  worst concavity  worst concave points  worst symmetry  worst fractal dimension  target
0        17.99         10.38          122.80     1001.0          0.11840  ...           0.7119                0.2654          0.4601                  0.11890     0.0
1        20.57         17.77          132.90     1326.0          0.08474  ...           0.2416                0.1860          0.2750                  0.08902     0.0
2        19.69         21.25          130.00     1203.0          0.10960  ...           0.4504                0.2430          0.3613                  0.08758     0.0
3        11.42         20.38           77.58      386.1          0.14250  ...           0.6869                0.2575          0.6638                  0.17300     0.0
4        20.29         14.34          135.10     1297.0          0.10030  ...           0.4000                0.1625          0.2364                  0.07678     0.0

[5 rows x 31 columns]
   mean radius  mean texture  mean perimeter  mean area  ...  worst concavity  worst concave points  worst symmetry  worst fractal dimension
0        17.99         10.38          122.80     1001.0  ...           0.7119                0.2654          0.4601                  0.11890
1        20.57         17.77          132.90     1326.0  ...           0.2416                0.1860          0.2750                  0.08902
2        19.69         21.25          130.00     1203.0  ...           0.4504                0.2430          0.3613                  0.08758
3        11.42         20.38           77.58      386.1  ...           0.6869                0.2575          0.6638                  0.17300
4        20.29         14.34          135.10     1297.0  ...           0.4000                0.1625          0.2364                  0.07678

[5 rows x 30 columns]
0    0.0
1    0.0
2    0.0
3    0.0
4    0.0
Name: target, dtype: float64
Training X input feature:  (455, 30)
Testing X input feature:  (114, 30)
Training Y input feature:  (455,)
Testing Y input feature:  (114,)
            predicted_cancer  predicted_healthy
is_cancer                 66                  0
is_healthy                 8                 40
              precision    recall  f1-score   support

         0.0       1.00      0.83      0.91        48
         1.0       0.89      1.00      0.94        66

    accuracy                           0.93       114
   macro avg       0.95      0.92      0.93       114
weighted avg       0.94      0.93      0.93       114

accuracy before scaling = 0.9298245614035088
```



(FIG 6.13 Label comparison)

CONCLUSION & FUTURE WORKS

7.1 CONCLUSION & FUTURE WORKS

The breast cancer prediction model is developed using various machine learning techniques. Thesemodels can predict any deviations in the normal pattern. The classifiers deployed on Decision Tree and Random Forest. The different performance analyzed for all techniques are accuracy, precision and recall. Random Forest results in superior accuracy in comparison to other techniques. Feature selection is performed on all classifiers to compare the performance. Random Forest outperforms other classifiers.

In the future, we can analyze other classifiers and develop a hybrid model for breast cancer prediction.

IMPLEMENTATION CODING

GUI

```
from tkinter

import *root =

Tk()

root.geometry("1000x600+0+0")

root.title("BREAST CANCER

CLASSIFICATION")

top=Frame(root,width=160,height=40,relief="solid",b

g="hot pink")top.pack()

titleinfo=Label(top,font=('Times New
Roman',20,'bold'),text="BREAST CANCER
CLASSIFICATION",fg="dark
green",bd=10,anchor='w',padx=5,pady=5)

titleinfo.grid(row=0,colu

mn=0) from PIL import

ImageTk,Imageimport os

img =

ImageTk.PhotoImage(Image.open("C:\\Users\\sanza\\Download
```

```
s\\canc.jpg"))panel=Label(top,image=img,width=500,height=200)
panel.gri
d()
print("")
root.title("Breast
Cancer")def mm():
c=s.get()
```

```
if(c==1):

print("Dataset

Selected")

elif(c==2):

print("Dataset

Selected")print("")

s=IntVa

r()i

s.set(3)

Label(root,font=('Times New
Roman',15,'bold'),text="SELECT
DATASET",padx=25,justify=LEFT).pack(anchor=
W)

print("")

Radiobutton(root,text="BREAST CANCER WISCONSIN
DIAGNOSTIC",padx=25,variable=s,value=1,command=mm).pack(anchor=W)

Radiobutton(root,text="BREAST CANCER WISCONSIN
ORGINAL",padx=25,variable=s,value=2,command=mm).pack(anchor=W)

def

mmmm():

b=n.get()

if(b==1):

import

Random_forest
```

```
elif(b==2):
import
decision
elif(b==3):
import svc
elif(b==4):
import knn
elif(b==6):
import
naive
```

```
n=IntVar

()

n.set(3)

Label(root,font=('Times New
Roman',15,'bold'),text="CHOOSE THE
CLASSIFIER",padx=25,justify=LEFT).pack(anchor=W)

print("")

Radiobutton(root,text="Random_forest",padx=25,variable=n,value=1,comman
d=mmmm).pack( anchor=W)

Radiobutton(root,text="Decision_tree",padx=25,variable=n,value=2,command
=mmmm).pack(an chor=W)

Radiobutton(root,text="SVC",padx=25,variable=n,value=3,command=mmmm)
.pack(anchor=W)

Radiobutton(root,text="K_Nearest",padx=25,variable=n,value=4,command=m
mmm).pack(anch or=W)

Radiobutton(root,text="Naive_Bayes",padx=25,variable=n,value=6,command=
mmmm).pack(an chor=W)

def mmm():

a=v.get()

if(a==1):

import

feature_scaling

elif(a==2):

import

non_feature_scaling
```

```
elif(a==3):
import
selection
elif(a==4):
import
feature_imp
print("")
v=IntVar()
v.set(3)
```

```
Label(root,font=('Times New
Roman',15,'bold'),text="SCALING
MODE",padx=25,justify=LEFT).pack(anchor=W)

print("")

Radiobutton(root,text="After_Scaling",padx=25,variable=v,value=1,command
=mmm).pack(anc hor=W)

Radiobutton(root,text="Before_Scaling",padx=25,variable=v,value=2,comman
d=mmm).pack(an chor=W)

Radiobutton(root,text="Feature_Selection",padx=25,variable=v,value=3,comm
and=mmm).pack( anchor=W)

Radiobutton(root,text="Feature_Importance",padx=25,variable=v,value=4,com
mand=mmm).pac k(anchor=W)

root.mainloop()
```

Decision Tree Classifier

```
import sklearn
from sklearn.datasets import
load_breast_cancerfrom
sklearn.metrics import
accuracy_score from sklearn.metrics
import classification_reportfrom
sklearn.metrics import
confusion_matrix from
sklearn.metrics import
precision_score from sklearn.metrics
import recall_score
data = load_breast_cancer()
label_names =
data['target_names']label =
data['target']
feature_name =
data['feature_names']feature
= data['data']
print(label_names)
```

```
print(label)
print(feature_n
ame)
print(feature)
from sklearn.model_selection import train_test_split
train,test,train_label,test_label =
train_test_split(feature,label,test_size=0.5,random_state=42)
print(train_label)
print(test_label)
print("Decision Tree
Classifier\n")from sklearn
import tree
dt = tree.DecisionTreeClassifier()
```

```
dt.fit(train,train_label)
predicitons =
dt.predict(test)
print("accuracy =
",accuracy_score(test_label,predicitons))
print(confusion_matrix(test_label,predicitons))
print(classification_report(test_label,predicito
ns)) print("precision =
",precision_score(test_label,predicitons))
print("recall =
",recall_score(test_label,predicitons))
```

Random forest

```
import sklearn
from sklearn.datasets import load_breast_cancer
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
```

```
data = load_breast_cancer()
label_names = data['target_names']
label = data['target']
feature_name = data['feature_names']
feature = data['data']
print(label_names)
print(label)
print(feature_name)
print(feature)
from sklearn.model_selection import train_test_split
train,test,train_label,test_label =
train_test_split(feature,label,test_size=0.5,random_state=42)
print(train_label)
print(test_label)
print("Random Forest Classifications\n")
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier()
rf.fit(train,train_label)
predictions = rf.predict(test)
print("accuracy = ",accuracy_score(test_label,predictions))
print(confusion_matrix(test_label,predictions))
print(classification_report(test_label,predictions))
```

```
print("precision = ",precision_score(test_label,predictions))
print("recall = ",recall_score(test_label,predictions))
```

With Feature Scaling

```
import
pandas as pd
import numpy as
np
from sklearn.datasets import
load_breast_cancerfrom
sklearn.metrics import
accuracy_score cancer =
load_breast_cancer()
df_cancer =
pd.DataFrame(np.c_[cancer['data'],cancer['target']],
columns =np.append(cancer['feature_names'],['target']))
print(df_cancer.head())
X =
df_cancer.drop(['target'],axis
= 1)print(X.head())
Y =
df_cancer['target']
print(Y.head())
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size =
0.2,random_state = 20)print("Training X input feature:
```

```
",x_train.shape)
print("Testing X input feature:
",x_test.shape) print("Training Y input
feature: ",y_train.shape)print("Testing
Y input feature: ",y_test.shape) from
sklearn.svm import SVC
svc_model = SVC()
svc_model.fit(x_train,y_train)
y_predict =
svc_model.predict(x_test)
from sklearn.metrics import
classification_report,confusion_matrixcm =
np.array(confusion_matrix(y_test,y_predict,labels=[1,
0]))
```

```
confusion =
pd.DataFrame(cm,index=['is_cancer','is_healthy'],columns=['predicted_cance
r','predicted_health y'])

print(confusion)

print(classification_report(y_test,y_predic

t))

print("accuracy before scaling

=",accuracy_score(y_test,y_predict))x_train_min =

x_train.min()

print(x_train_min)

x_train_max =

x_train.max()

print(x_train_max)

x_train_range = (x_train_max-

x_train_min)print(x_train_range)

x_train_scaled = (x_train-

x_train_min)/(x_train_range)

print(x_train_scaled)

x_test_min = x_test.min()

x_test_range = (x_test-

x_test_min).max() x_test_scaled =

(x_test-x_test_min)/x_test_range

svc_model = SVC()
```

```
svc_model.fit(x_train_scaled,y_train)

y_predict = svc_model.predict(x_test_scaled)

cm =

np.array(confusion_matrix(y_test,y_predict,labels

=[1,0]))confusion =
pd.DataFrame(cm,index=['is_cancer','is_healthy'],columns=['predicted_cance
r','predicted_health y'])

print(confusion)

print(classification_report(y_test,y_predic

t))

print("accuracy Aftere scaling =",accuracy_score(y_test,y_predict))
```

Without Feature Scaling

```
import pandas

as pdimport

numpy as np

from sklearn.datasets import

load_breast_cancerfrom

sklearn.metrics import

accuracy_score cancer =

load_breast_cancer()

df_cancer =
pd.DataFrame(np.c_[cancer['data'],cancer['target']],
columns =np.append(cancer['feature_names'],['target']))

print(df_cancer.head())

X =

df_cancer.drop(['target'],axis

= 1)print(X.head())

Y =

df_cancer['target']

print(Y.head())

from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size =
```

```
0.2,random_state = 20)print("Training X input feature:
",x_train.shape)
print("Testing X input feature:
",x_test.shape) print("Training Y input
feature: ",y_train.shape)print("Testing
Y input feature: ",y_test.shape) from
sklearn.svm import SVC
svc_model = SVC()
svc_model.fit(x_train,y_train)
y_predict =
svc_model.predict(x_test)
from sklearn.metrics import classification_report,confusion_matrix
```

```
cm = np.array(confusion_matrix(y_test,y_predict,labels=[1,0]))

confusion =
pd.DataFrame(cm,index=['is_cancer','is_healthy'],columns=['predicted_cance
r','predicted_health y'])

print(confusion)

print(classification_report(y_test,y_predic

t))

print("accuracy before scaling =",accuracy_score(y_test,y_predict))
```

Scaling

```
import pandas

as pdimport

numpy as np

from sklearn.datasets import

load_breast_cancerfrom

sklearn.metrics import

accuracy_score cancer =

load_breast_cancer()

df_cancer =
pd.DataFrame(np.c_[cancer['data'],cancer['target']],
columns =np.append(cancer['feature_names'],['target']))

print(df_cancer.head())

X =

df_cancer.drop(['target'],axis

= 1)print(X.head())

Y =

df_cancer['target']

print(Y.head())

from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size =
```

```
0.2,random_state = 20)print("Training X input feature:
",x_train.shape)
print("Testing X input feature:
",x_test.shape) print("Training Y input
feature: ",y_train.shape)print("Testing
Y input feature: ",y_test.shape) from
sklearn.svm import SVC
svc_model = SVC()
svc_model.fit(x_train,y_train)
y_predict =
svc_model.predict(x_test)
from sklearn.metrics import classification_report,confusion_matrix
```

```
cm = np.array(confusion_matrix(y_test,y_predict,labels=[1,0]))

confusion =
pd.DataFrame(cm,index=['is_cancer','is_healthy'],columns=['predicted_cance
r','predicted_health y'])

print(confusion)

print(classification_report(y_test,y_predic

t))

print("accuracy before scaling =",accuracy_score(y_test,y_predict))
```

Feature Selection

```
import pandas
as pd import
seaborn as sns
import matplotlib.pyplot as plt
from sklearn.datasets import
load_breast_cancerimport sklearn
from sklearn.metrics import
accuracy_score from sklearn.metrics
import classification_reportfrom
sklearn.metrics import
confusion_matrix from
sklearn.metrics import
precision_score from sklearn.metrics
import recall_score
old_dataset=load_breast_cancer()
label_names=old_dataset['target_nam
es'] labels=old_dataset['target']
features=old_dataset['data']
feature_name=old_dataset['feature_n
ames']
```

```
from sklearn.model_selection import train_test_split
train,test,train_labels,test_labels=train_test_split(features,labels,r
andom_state=0) print("random forest classifier")
from sklearn.ensemble import
RandomForestClassifier
rfc=RandomForestClassifier()
rfc.fit(train,train_label
s)
predictions=rfc.predi
ct(test)
print("accurancy=",accuracy_score(test_labels,predictions))
```

```
print("*"*50)
dataset=pd.read_csv("C:/Users/Smartiee/Downlo
ads/data.csv")print(dataset.head())
print(dataset['diagnosis'].unique)
print(dataset.groupby('diagnosis').size())
sns.countplot(dataset['diagnosis'],label="
count")plt.show()
```

REFERENCE

S

[1] Ahmed. K. Ashfaq, et al. "Cancer discase prediction with support vector machine andrandom forest classification techniques Computational Intelligence and Cybernetics (CyberneticsComi. 2012 IEEE International Conference on. IEEE 2012.

[2] Amrane, Meriem, et al "Breast cancer classification using machinelearning 2018 ElectricElectronics, Computer Science. Biomedical Engineerings Meeting (EB87). IEEE, 2018

[3] Huang, Yu-Len, Kao-Lun Wang, and Dar-Ren Chen. "Diagnosis of breast tumors with ultrasonic texture analysis using support vector machines Neural Computing & Applications 15.2(2006): 164-169

[4] Samulski, Maurice, et al. "Classification of mammographic masses using support vector machines and Bayesian networks." Medical Imaging 2007: Computer-Aided Diagnosis Vol. 6514 International Society for Optics and Photonics, 2007.

[5] Jabbar MA, Deekshatulu BL, Chandra P. “Classification of heart disease using k-nearestneighbor and genetic algorithm”.
International Conference on Computational Intelligence: Modeling Techniques and Applications(CIMTA).2013; 10:85-94.

[6] Hassanat AB, Abbadi MA, Altarawneh GA, Alhasanat AA. “Solving the problem of theK parameter in the KNN classifier

using an ensemble learning approach". (IJCSIS) International Journal of ComputerScience and Information Securit. 2014; 12(8):33-9

[7] Srinivas R." Managing Large Data Sets Using Support Vector Machines". Computer Science and Engineering. M.Sc. Thesis, University of Nebraska - Lincoln, 2010

[8] Banu B, Thirumalaikolundusubramanian P.” Comparison of Bayes Classifiers for BreastCancer Classification”. Asian Pacific journal of cancer prevention (APJCP). 2018; 19(10):2917-20.DOI: 10.22034/APJCP.2018.19.10.2917

[9] Naik A, Samant L. “Correlation review of classification algorithm using data mining tool:WEKA, Rapidminer, Tanagra, Orangeand Knime”. International Conference on Computational Modeling and Security (CMS 2016). 2016; 85:662-8.

[10] Jovic A, Brkic K, Bogunovic N. “An overview of free software tools for general datamining”. 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). 2014: 1112-7.

[11] Slater S, Joksimović S, Kovanovic V, Baker RS, Gasevic D. “Tools for educational datamining: A review”. Journal of Educational and Behavioral Statistics. 2017; 42(1):85-106.

[12] Wang, D. Zhang and Y. H. Huang “Breast Cancer Prediction Using Machine Learning”(2018), Vol. 66, NO. 7.

[13] B. Akbugday, "Classification of Breast Cancer Data Using Machine Learning Algorithms,"2019 Medical Technologies Congress (TIPTEKNO), Izmir, Turkey, 2019, pp. 1-4.

[14] Keles, M. Kaya, "Breast Cancer Prediction and Detection Using Data Mining Classification Algorithms: A Comparative Study." Tehnicki Vjesnik - Technical Gazette, vol. 26, no. 1, 2019, p.149+.

[15] V. Chaurasia and S. Pal, “Data Mining Techniques: To Predict and Resolve Breast Cancer Survivability”, IJCSMC, Vol. 3, Issue. 1, January 2014, pg.10 – 22.

[16] Delen, D.; Walker, G.; Kadam, A. Predicting breast cancer survivability: A comparison of three data mining methods. Artif. Intell. Med. 2005, 34, 113–127.

[17] R. K. Kavitha1, D. D. Rangasamy, “Breast Cancer Survivability Using Adaptive Voting Ensemble Machine Learning Algorithm Adaboost and CART Algorithm” Volume 3, Special Issue1, February 2014

[18] P. Sinthia, R. Devi, S. Gayathri and R. Sivasankari, “Breast Cancer detection using PCPCET and ADEWNN”, CIEEE’ 17, p.63-65

[19] Vikas Chaurasia and S.Pal, “Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis” (FAMS 2016) 83 (2016) 1064 – 1069

[20] N. Khuriwal, N. Mishra. “A Review on Breast Cancer Diagnosis in Mammography Images Using Deep Learning Techniques”, (2018), Vol. 1, No. 1

DATASET

https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

PYTHON DATASCIENCE INTRO:

https://freelearningapp.com/course/machine-learning-a-z-hands-on-python-r-in-data-science- updated

GUI TUTORIALS

https://www.tutorialspoint.com/python/python_gui_programming.html