INTRODUCTION

The most commonly occurring type of cancer is breast cancer. It is known to affect over two million women annually. For women diagnosed during 2010-14, five-year survival for breast cancer shows very heavy variation with changes in location. It is generally known to be above 50% in most places. There are no prevention techniques for breast cancer but early detection and diagnosis are critical in determining the chances of survival.

During the early stages of the disease, the symptoms are not presented well and hence diagnosis is delayed. It is recommended by the NBCF (National Breast Cancer Foundation) that women over the age of forty years of age should get a mammogram once a year. A mammogram is an X-ray of the breast. It is a medical technique used for the detection of breast cancer in women without any side effects deeming the procedure safe. Women who get regular mammograms have a higher survival rate as compared to women who do not. In 2018, over six hundred thousand fatalities were caused by breast cancer.

Breast cancer can occur in women and rarely in men. The ICMR (Indian Council of Medical Research) recently published a report which stated that in 2020 the total number of new cancer cases is expected to be about 17.3 lakhs. An Indian woman is diagnosed with breast cancer every four minutes. Breast cancer is a disease that occurs but when a woman or a man is aware of this symptom, it immediately goes beyond its original stage.

There are two types of breast cancer, Malignant and Benign. The first is classified as harmful, has the ability to infect other organs, and is cancerous, Benign is classified as non-cancerous. This disease infects the women's chest and specifically glands and milk ducts, the spread of breast cancer to other organs is frequent and could be through the bloodstream.

Cancer in women always has a huge incidence rate and mortality rate. Breast cancer alone is estimated to account for 25% of all new cancer diagnoses worldwide and 15% of cancer deaths in women worldwide, according to the latest cancer statistics. Every 1 in 8 Women in the USA develops breast cancer in her lifetime. In case of any sign or symptom, people usually visit a doctor immediately, who may refer you to an oncologist for help. An oncologist can diagnose breast cancer by Examining the patient's medical history thoroughly, examining both breasts and even checking for swelling or hardening of any lymph nodes in the armpits.

Here in this project, we have used the Wisconsin Breast Cancer Dataset (WBCD) and with the dataset, we have used machine learning algorithms to predict whether a patient has breast cancer.

RELATED WORKS

One of the early studies on using machine learning for breast cancer detection was conducted by Elter et al. (2007). They developed a neural network-based system that used mammographic features to differentiate between malignant and benign breast masses. The system achieved an accuracy of 92%, demonstrating the potential of machine learning in this area.

In 2014, Wang et al. proposed a deep learning-based system for breast cancer detection that used convolutional neural networks (CNNs). Their system achieved an accuracy of 89.2% in detecting breast cancer in mammography images, outperforming traditional machine learning algorithms.

Another notable study was conducted by Arevalo et al. (2016), who developed a system that used a combination of CNNs and support vector machines (SVMs) to detect breast cancer in ultrasound images. Their system achieved an accuracy of 95.9%, demonstrating the potential of combining multiple machine-learning algorithms for better performance.

Breast cancer detection using Relevance Vector Machine, obtained an accuracy of 97% using the Wisconsin original dataset which has 699 instances and 11 attributes, while allots distinct weights to different attributes with regard to their capabilities of prediction and yielded an accuracy of 92% working with the weighted naïve Bayes method. A hybrid classifier of Support Vector Machines and decision trees in WEKA obtained an accuracy of 91%.

Linear Discriminant Analysis for feature selection and training the dataset by using one of the fuzzy inference methods called the Mamdani Fuzzy inference model and obtained an accuracy of 93%. Various differentiation between multiple techniques has been provided like Bayes Network, Pruned Tree, and kNN algorithm using WEKA on breast cancer dataset.

In conclusion, machine learning has shown great promise in the field of breast cancer detection and prognosis. The studies reviewed in this statement demonstrate the potential of various machine learning algorithms, including neural networks and SVMs, in detecting breast cancer in mammography and ultrasound images with high accuracy. As we move forward, continued research in this area will likely lead to even more accurate and efficient systems for breast cancer detection and diagnosis.

MOTIVATION

Breast cancer is considered a multifactorial disease and the most common cancer in women worldwide with approximately 30% of all female cancers (i.e. 1.5 million women are diagnosed with breast cancer each year, and 500,000 women die from this disease in the world). Over the past 30 years, this disease has increased, while the death rate has decreased. However, the reduction in mortality due to mammography screening is estimated at 20%, and improvement in cancer treatment is estimated at 60%.

Diagnostic mammography can assess abnormal breast cancer tissue in patients with subtle and inconspicuous malignancy signs. Due to a large number of images, this method cannot effectively be used in assessing cancer-suspected areas. According to a report, approximately 50% of breast cancers were not detected in screenings of women with very dense breast tissue. However, about a quarter of women with breast cancer are diagnosed negatively within two years of screening. Therefore, the early and timely diagnosis of breast cancer is crucial.

Machine learning, as a modeling approach, represents the process of extracting knowledge from data and discovering hidden relationships, widely used in healthcare in recent years to predict different diseases. Breast cancer occurs in every country of the world in women at any age after puberty but with increasing rates in later life. Breast cancer mortality changed little from the 1930s through to the 1970s.

Improvements in survival began in the 1980s in countries with early detection programs combined with different modes of treatment to eradicate invasive diseases. Approximately half of breast cancers develop in women who have no identifiable breast cancer risk factor other than gender (female) and age (over 40 years). Certain factors increase the risk of breast cancer including increasing age, obesity, harmful use of alcohol, family history of breast cancer, history of radiation exposure, reproductive history (such as age that menstrual periods began and age at first pregnancy), tobacco use, and postmenopausal hormone therapy.

Breast cancer most commonly presents as a painless lump or thickening in the breast. It is important that women finding an abnormal lump in the breast consult a health practitioner without a delay of more than 1-2 months even when there is no pain associated with it. Seeking medical attention at the first sign of a potential symptom allows for more successful treatment.

OBJECTIVE

The objective is to predict breast cancer using different machine-learning approaches applying demographic, laboratory, and mammographic data at an early stage so that the patient can take treatments according to his or her condition.

- Improving the availability and accessibility of screening services, particularly to underprivileged and remote areas with little access to healthcare services.

- Increasing the knowledge of the public on the benefits of early detection, changing attitudes and behaviors to seek early detection services, and raising awareness around breast cancer and cancer prevention in general.

- Establishing national unified protocols and guidelines that cover all processes of a comprehensive early detection and screening program.

- Developing collaborations and partnerships among different national, regional, and international entities and stakeholders to maintain sustainability.

- Monitoring and evaluating the implementation and progress of different JBCP work, and creating an enabling environment for the availability of data for sound decision-making around breast cancer early detection and screening.

- Early detection: Machine learning algorithms can detect breast cancer at an earlier stage, potentially improving outcomes for patients.

- Cost-effective: Machine learning algorithms can provide a cost-effective alternative to traditional methods of breast cancer detection, reducing the need for expensive and time-consuming tests.

Overall, the use of machine learning algorithms in breast cancer detection systems has the potential to improve accuracy, reduce costs, and increase accessibility, ultimately leading to improved outcomes for patients.

DATASET

A dataset is a collection of data. In the case of tabular data, a data set corresponds to one or more database tables, where every column of a table represents a particular variable, and each row corresponds to a given record of the data set in question.

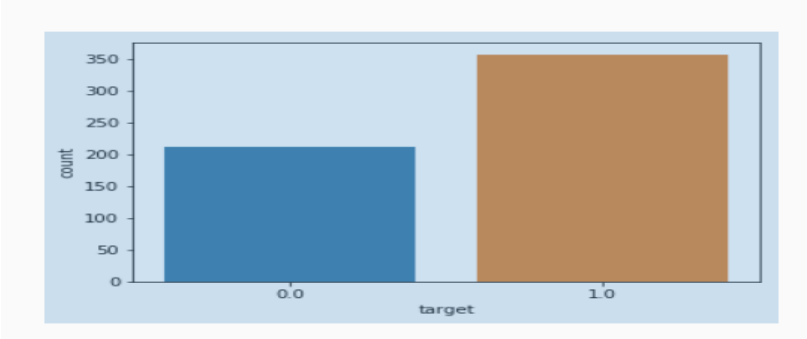
The project is based on the WISCONSIN DIAGNOSIS BREAST CANCER data set. The data set has been obtained from the package sklearn.datasets but it is also available on Kaggle.

It has 569 instances and 32 attributes and there are no missing values. The output variable is either benign(357 observations) or malignant (212 observations). The variables are namely diagnosis, radius mean, texture mean, perimeter mean, area mean, smoothness means, compactness mean, concavity mean, concave points mean, symmetry mean, smoothness, compactness, concavity, concave points, symmetry.

The negative class (0) means malignant tumor and the positive class (1) means benign tumor.

Dataset	No. of Attributes	No,. of Instances	No. of classes
Wisconsin Diagnosis Breast Cancer (WDBC)	32	569	2

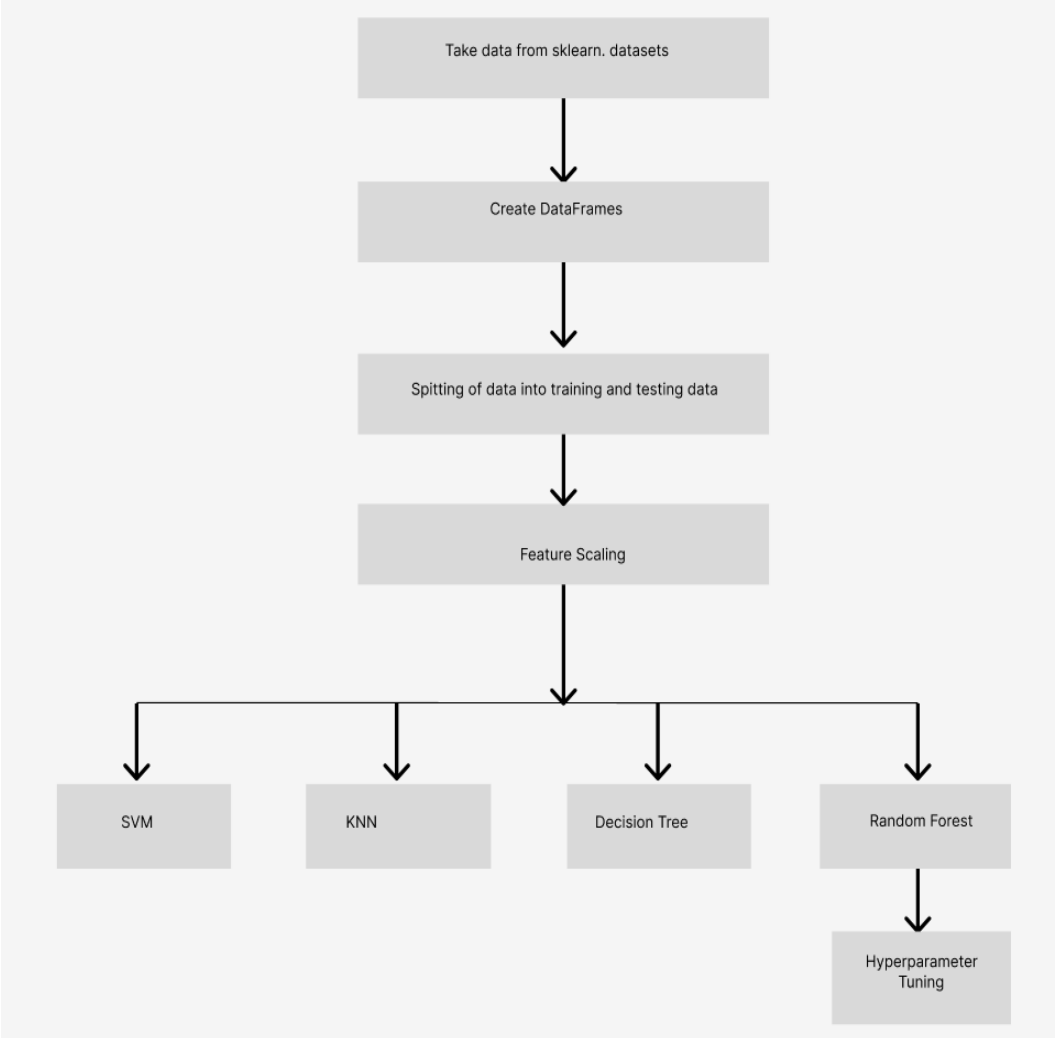
Description of WDBC Dataset



Graphical representation of output class

IMPLEMENTATION

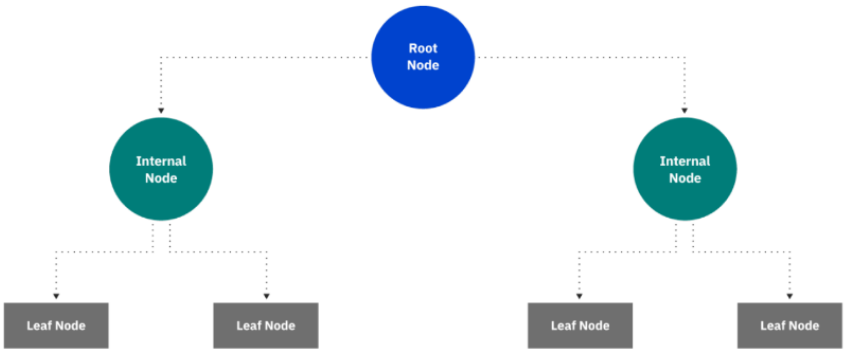
1. Project Architecture



Flowchart

2. Machine learning algorithms:

Decision tree algorithms:- Decision tree algorithms are successful machine learning classification techniques. They are the supervised learning methods that use information gained and pruned to improve results. Moreover, decision tree algorithms are commonly used for classification in many research, for example, in the medicine area and health issues.



How decision tree works

K-nearest-neighbours (kNN) algorithm:- It is a simple supervised learning algorithm in pattern recognition. It is one of the most popular neighborhood classifiers due to its simplicity and efficiency in the field of machine learning. KNN algorithm stores all cases and classifies new cases based on similarity measures; it searches the pattern space for the k training tuples that are closest to the unknown tuples. The performance depends on the optimal number of neighbours (k) chosen, which is different from one data sample to another.

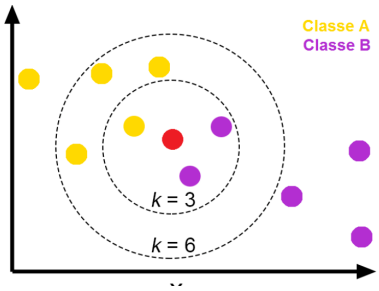
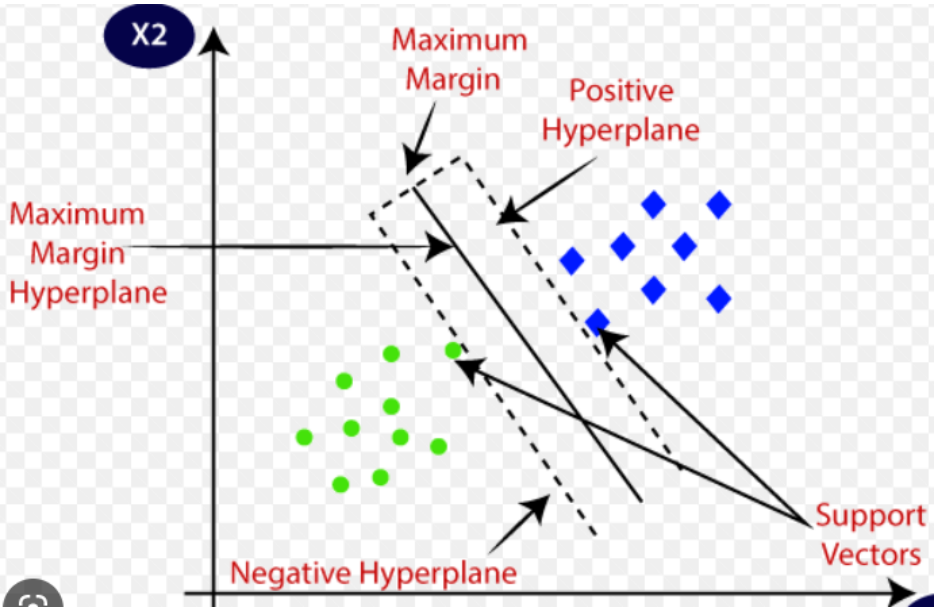


Illustration of KNN

Support Vector Machine (SVM):-It is a supervised learning method derived from statistical learning theory for the classification of both linear and nonlinear data. SVM classifies data into two classes over a hyperplane at the same time avoiding overfitting the data by maximizing the margin of hyperplane separation.



SVM Illustration

Random Forest :-It is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

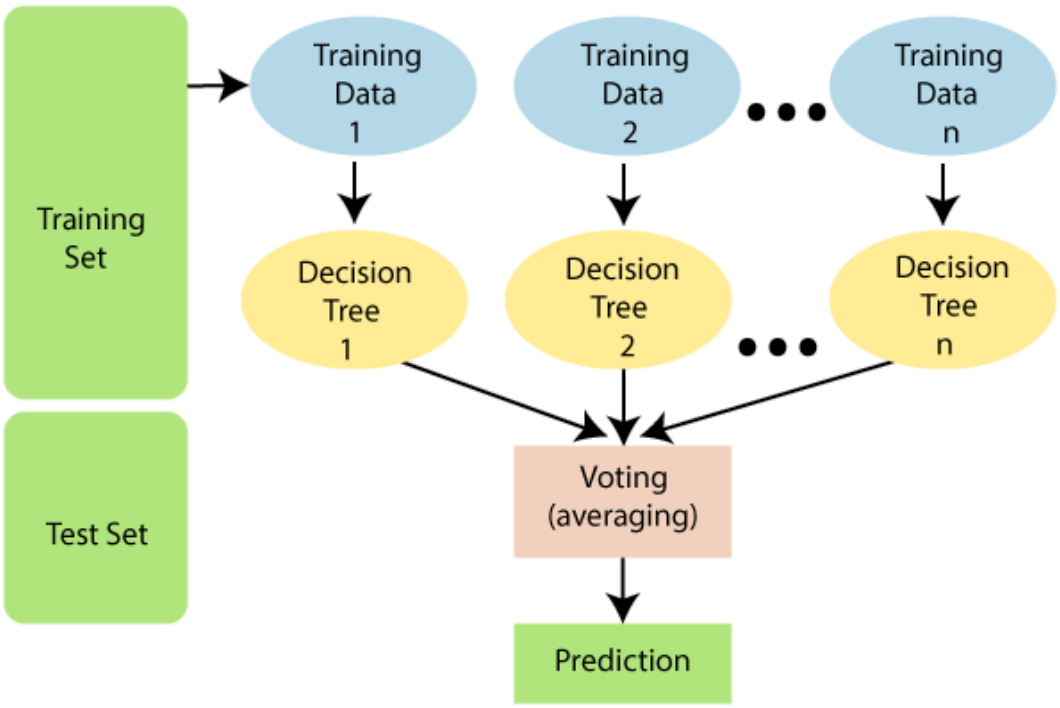


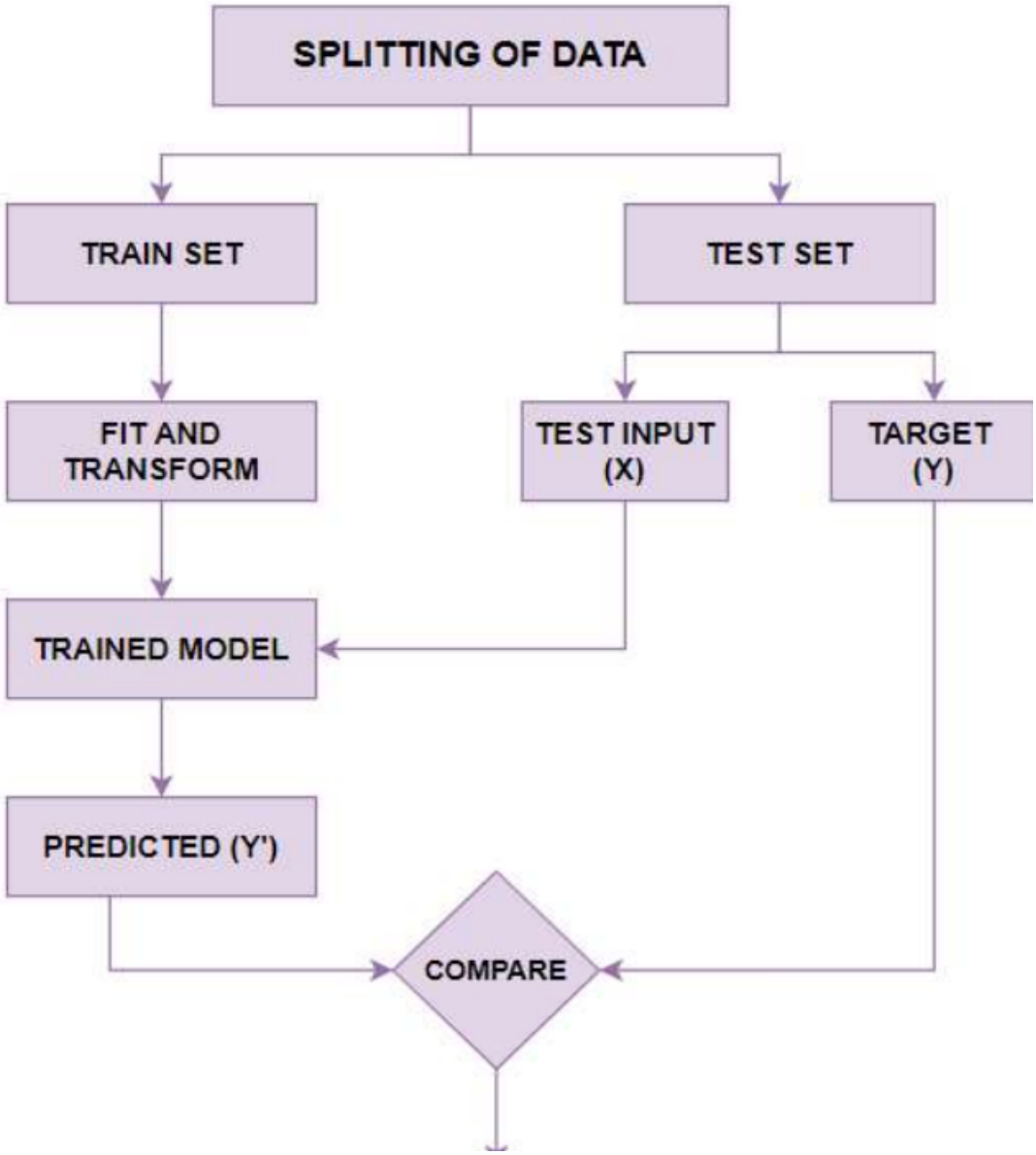
Illustration of Random forest

3. Tech details:

- Google Colab - Colab allows anybody to write and execute arbitrary Python code through the browser, and is especially well suited to machine learning, and data analysis
- Language used:
  Python
  Libraries used:
  - Pandas - for data manipulation or analysis
  - Matplotlib - for numeric calculation
  - Seaborn - for data visualization
  - Numpy - for calculation of data
  - Sklearn - Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, and regression.

4. Classification flow

```
[ ]  # import libraries
     import pandas as pd # for data manupulation or analysis
     import numpy as np # for numeric calculation
     import matplotlib.pyplot as plt # for data visualization
     import seaborn as sns # for data visualization
```



5. Code Explanation

Step 1. Importing Essential Libraries

```
[ ] # import libraries
    import pandas as pd # for data manupulation or analysis
    import numpy as np # for numeric calculation
    import matplotlib.pyplot as plt # for data visualization
    import seaborn as sns # for data visualization
```

Step 2. Loading of Data from sklearn.datasets

```
#Load breast cancer dataset
from sklearn.datasets import load_breast_cancer
cancer_dataset = load_breast_cancer()
```

Step 3. Creation of DataFrames using pandas

```
[ ] # create datafrmae
    cancer_df = pd.DataFrame(np.c_[cancer_dataset['data'],cancer_dataset['target']],
                columns = np.append(cancer_dataset['feature_names'], ['target']))
```

```
[ ] # DataFrame to CSV file
    cancer_df.to_csv('breast_cancer_dataframe.csv')
```

Step 4.Data Visualization using seaborn library

```
# Paiplot of cancer dataframe
sns.pairplot(cancer_df, hue = 'target')
```

```
[ ] # pair plot of sample feature
    sns.pairplot(cancer_df, hue = 'target',
                 vars = ['mean radius', 'mean texture', 'mean perimeter', 'mean area', 'mean smoothness'] )
```

```
[ ] # heatmap of DataFrame
    plt.figure(figsize=(16,9))
    sns.heatmap(cancer_df)
```

Step 5. Splitting of DataFrame

```
# split dataset into train and test
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= 5)
```

Step 5. Feature scaling

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train_sc = sc.fit_transform(X_train)
X_test_sc = sc.transform(X_test)
```

Step 6. Machine Learning Model Building

6. a) Support Vector Classifier

```
[ ]  # Support vector classifier
     from sklearn.svm import SVC
     svc_classifier = SVC()
     svc_classifier.fit(X_train, y_train)
     y_pred_scv = svc_classifier.predict(X_test)
     accuracy_score(y_test, y_pred_scv)
```

```
0.9385964912280702
```

```
[ ]  # Train with Standard scaled Data
     svc_classifier2 = SVC()
     svc_classifier2.fit(X_train_sc, y_train)
     y_pred_svc_sc = svc_classifier2.predict(X_test_sc)
     accuracy_score(y_test, y_pred_svc_sc)
```

```
0.9649122807017544
```

6.b) K-nearest Neighbour Classifier

+ Code

```
[ ] # K - Nearest Neighbor Classifier
    from sklearn.neighbors import KNeighborsClassifier
    knn_classifier = KNeighborsClassifier(n_neighbors = 5, metric = 'minkowski', p = 2)
    knn_classifier.fit(X_train, y_train)
    y_pred_knn = knn_classifier.predict(X_test)
    accuracy_score(y_test, y_pred_knn)
```

```
0.9385964912280702
```

```
[ ] # Train with Standard scaled Data
    knn_classifier2 = KNeighborsClassifier(n_neighbors = 5, metric = 'minkowski', p = 2)
    knn_classifier2.fit(X_train_sc, y_train)
    y_pred_knn_sc = knn_classifier.predict(X_test_sc)
    accuracy_score(y_test, y_pred_knn_sc)
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have
```

6.c) Decision tree

```
# Decision Tree Classifier
from sklearn.tree import DecisionTreeClassifier
dt_classifier = DecisionTreeClassifier(criterion = 'entropy', random_state = 51)
dt_classifier.fit(X_train, y_train)
y_pred_dt = dt_classifier.predict(X_test)
accuracy_score(y_test, y_pred_dt)
```

```
0.9473684210526315
```

```
[ ] # Train with Standard scaled Data
    dt_classifier2 = DecisionTreeClassifier(criterion = 'entropy', random_state = 51)
    dt_classifier2.fit(X_train_sc, y_train)
    y_pred_dt_sc = dt_classifier.predict(X_test_sc)
    accuracy_score(y_test, y_pred_dt_sc)
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not hav
  warnings.warn(
0.7543859649122807
```

6.d) Random Forest Classifier

```
# Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
rf_classifier = RandomForestClassifier(n_estimators = 20, criterion = 'entropy', random_state = 51)
rf_classifier.fit(X_train, y_train)
y_pred_rf = rf_classifier.predict(X_test)
accuracy_score(y_test, y_pred_rf)
```

```
0.9736842105263158
```

```
# Train with Standard scaled Data
rf_classifier2 = RandomForestClassifier(n_estimators = 20, criterion = 'entropy', random_state = 51)
rf_classifier2.fit(X_train_sc, y_train)
y_pred_rf_sc = rf_classifier.predict(X_test_sc)
accuracy_score(y_test, y_pred_rf_sc)
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, b
  warnings.warn(
0.7543859649122807
```

Step 7. Classification Report

7. a) Classification Report of SVM

```
print(classification_report(y_test, y_pred_scv))
```

```
              precision    recall  f1-score   support

         0.0       1.00      0.85      0.92        48
         1.0       0.90      1.00      0.95        66

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

7. b) Classification Report of KNN

```
print(classification_report(y_test, y_pred_knn))
```

```
              precision    recall  f1-score   support

         0.0       1.00      0.85      0.92        48
         1.0       0.90      1.00      0.95        66

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

7. c)Classification report of Decision tree

```
print(classification_report(y_test, y_pred_dt))
```

```
              precision    recall  f1-score   support

         0.0       0.96      0.92      0.94        48
         1.0       0.94      0.97      0.96        66

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

7.d) Classification report of Random Forest

```
[ ] print(classification_report(y_test, y_pred_rf))
```

```
              precision    recall  f1-score   support

         0.0       1.00      0.94      0.97        48
         1.0       0.96      1.00      0.98        66

    accuracy                           0.97       114
   macro avg       0.98      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```

Step 8. Hyperparameter tuning on Random Forest

```
# Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
rf_classifier = RandomForestClassifier(n_estimators = 30, criterion = 'gini', random_state = 51)
rf_classifier.fit(X_train, y_train)
y_pred_rf = rf_classifier.predict(X_test)
accuracy_score(y_test, y_pred_rf)
```

```
0.9649122807017544
```

```
[ ] # Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
rf_classifier = RandomForestClassifier(n_estimators = 15, criterion = 'gini', random_state = 43)
rf_classifier.fit(X_train, y_train)
y_pred_rf = rf_classifier.predict(X_test)
accuracy_score(y_test, y_pred_rf)
```

```
0.956140350877193
```

```
[ ] # Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
rf_classifier = RandomForestClassifier(n_estimators = 30, criterion = 'log_loss', random_state = 51)
rf_classifier.fit(X_train, y_train)
y_pred_rf = rf_classifier.predict(X_test)
accuracy_score(y_test, y_pred_rf)
```

```
0.9736842105263158
```

RESULT ANALYSIS

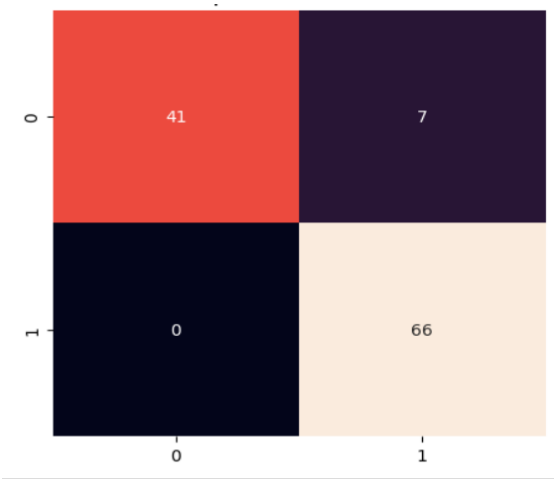
1. Comparison between Techniques

Techniques	Accuracy without Standard scale	Accuracy with Standard scale
SVM	93.85 %	96.49%
KNN	93.85%	57.89%
Decision Tree	94.73%	75.43%
Random Forest	97.36%	75.43

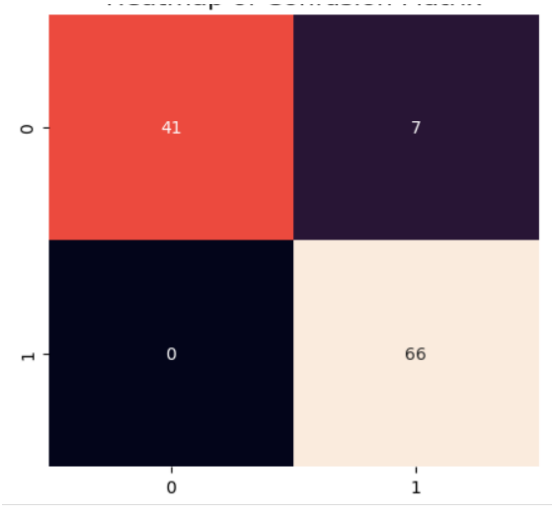
2. Confusion Matrix

Confusion Matrix is used for evaluating the performance of a classification model. The Matrix compares the actual target values with predicted values by the machine learning model. It shows the ways in which your classification model gets confused when it makes predictions.

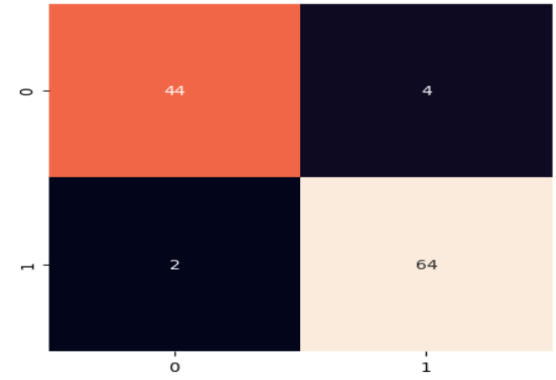
2. a) Confusion Matrix of SVM



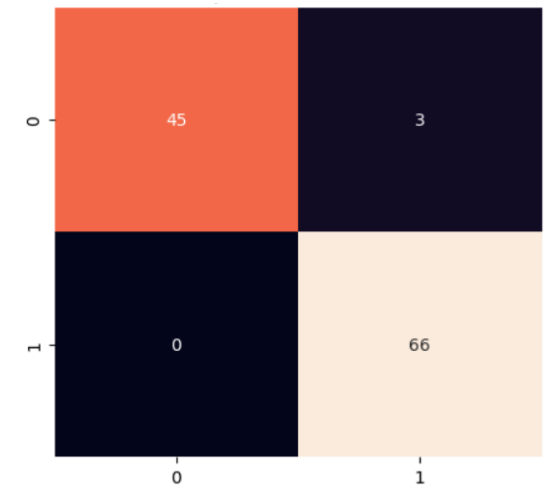
2. b) Confusion Matrix of KNN



2. c) Confusion Matrix of Decision Tree



2. d) Confusion Matrix of Random Forest



3. Accuracy

Accuracy is a good predictor for the degree of correctness in the training of the model and how it may perform generally. It may be defined as the measure of the correct prediction in correspondence to the wrong ones. Accuracy = (TruePositive + TrueNegative) /(TruePositive + FalsePositive + TrueNegative + False Negative)

4. F1-score

The F_1 score is the harmonic mean of the precision and recall. It thus symmetrically represents both precision and recall in one metric.

The different approaches are compared with recall metrics, F1 score, precision, and accuracy metrics and is mentioned below in table.

Algorithm	Cancer	Precision	Recall	F1 score	Support	Accuracy
SVM	Malignant(0) Benign(1)	1.00 0.90	0.85 1.00	0.92 0.95	48 66	93.85%
KNN	Malignant(0) Benign(1)	1.00 0.90	0.85 1.00	0.92 0.95	48 66	93.85%
Decision Tree	Malignant(0) Benign(1)	0.96 0.94	0.92 0.97	0.94 0.96	48 66	94.73
Random Forest	Malignant(0) Benign(1)	1.00 0.96	0.94 1.00	0.97 0.98	48 66	97.36%

CONCLUSION

We learned to build a breast cancer tumor predictor on the Wisconsin dataset and created graphs and results for the same. It has been observed that a good dataset provides better accuracy. The selection of appropriate algorithms with a good home dataset will lead to the development of prediction systems.

These systems can assist in proper treatment methods for a patient diagnosed with breast cancer. There are many treatments for a patient based on breast cancer stage; data mining and machine learning can be a very good help in deciding the line of treatment to be followed by extracting knowledge from such suitable databases.

For reference, we have used **“BREAST CANCER DETECTION SYSTEM USING MACHINE LEARNING ALGORITHMS**” by **Shubham Sharma, Archit Aggarwal, Tanupriya Choudhury.**

Future Aspects

Feature Extraction

Feature Selection is the method of reducing the input variable to your model by using only relevant data and eliminating noise in data.

It is the process of automatically choosing relevant features for your machine learning model based on the type of problem you are trying to solve. We do this by including or excluding important features without changing them. It helps in cutting down the noise in our data and reducing the size of our input data.

Classification

We have used machine learning algorithms to classify between benign and malignant tumors where we achieve an accuracy of up to 97%.
We can try different machine learning algorithms to achieve greater accuracy.

Suggesting surgery

We can identify the stage of cancer based on the value of different features and suggest the surgery a person has to go through making the process easier and what precautions they have to go take.