A Project

on

Breast Cancer Diagnosis Using Machine Learning: A KNN Classifier Approach

carried out as part of the course CSE **CS3203** *Submitted by*

Aryan Singh

209301499

VI-CSE-F

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

In

Computer Science & Engineering

MANIPAL UNIVERSITY
JAIPUR
(University under Section 2(f) of the UGC Act)

Department of Computer Science & Engineering,
School of Computer Science and Engineering,
Manipal University Jaipur,
April 2023

Introduction

Breast cancer is one of the most common types of cancer affecting women worldwide. Early detection and diagnosis of breast cancer can significantly improve the chances of successful treatment and cure. Machine learning has emerged as a promising tool for improving the accuracy and efficiency of breast cancer detection and diagnosis. In this project, we will be exploring the use of machine learning algorithms to analyze mammography images and classify breast cancer as either malignant or benign. The goal is to develop a model that can accurately detect and classify breast cancer with high precision and recall rates. By improving the accuracy of breast cancer diagnosis, we can potentially improve the outcomes for patients and reduce the burden of this devastating disease.

In this task, the goal is to build a model that can accurately detect breast cancer. This involves several steps, such as collecting a large dataset of breast cancer reports, cleaning and preprocessing the data, extracting features from the text, training machine learning models, and evaluating the model's performance on a separate test set.

Python provides several libraries and frameworks that can be used to implement these steps, such as Scikit-Learn, NumPy, Pandas, Matplotlib, Seaborn and Keras. These libraries provide a wide range of machine learning algorithms, data pre-processing tools, and building and training neural networks functions that can be used to build effective breast cancer detection models.

Datasets

The dataset used for this project is the **Breast Cancer Wisconsin (Diagnostic) Data Set** from UCI.

Attribute Information:

1) ID number
2) Diagnosis (M = malignant, B = benign)
3-32)
Ten real-valued features are computed for each cell nucleus:
a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

```
1015425,3,1,1,1,2,2,3,1,1,2
1016277,6,8,8,1,3,4,3,7,1,2
1017023,4,1,1,3,2,1,3,1,1,2
1017122,8,10,10,8,7,10,9,7,1,4
1018099,1,1,1,1,2,10,3,1,1,2
1018561,2,1,2,1,2,1,3,1,1,2
1033078,2,1,1,1,2,1,1,1,5,2
1033078,4,2,1,1,2,1,2,1,1,2
1035283,1,1,1,1,1,1,3,1,1,2
1036172,2,1,1,1,2,1,2,1,1,2
1041801,5,3,3,3,2,3,4,4,1,4
1043999,1,1,1,1,2,3,3,1,1,2
```

Algorithm

The k-nearest neighbors algorithm can be used for breast cancer detection using machine learning and Python:

- Data preprocessing: The first step is to preprocess the dataset by cleaning the text and transforming it into a numerical representation that can be used as input for the k-nearest neighbors algorithm.

- Splitting the dataset: The dataset is then split into training and testing sets. The training set is used to train the k-nearest neighbors model, while the testing set is used to evaluate its performance.

- Training the model: The algorithm calculates the probabilities of each feature given each class (malignant or benign) and the prior probabilities of each class.

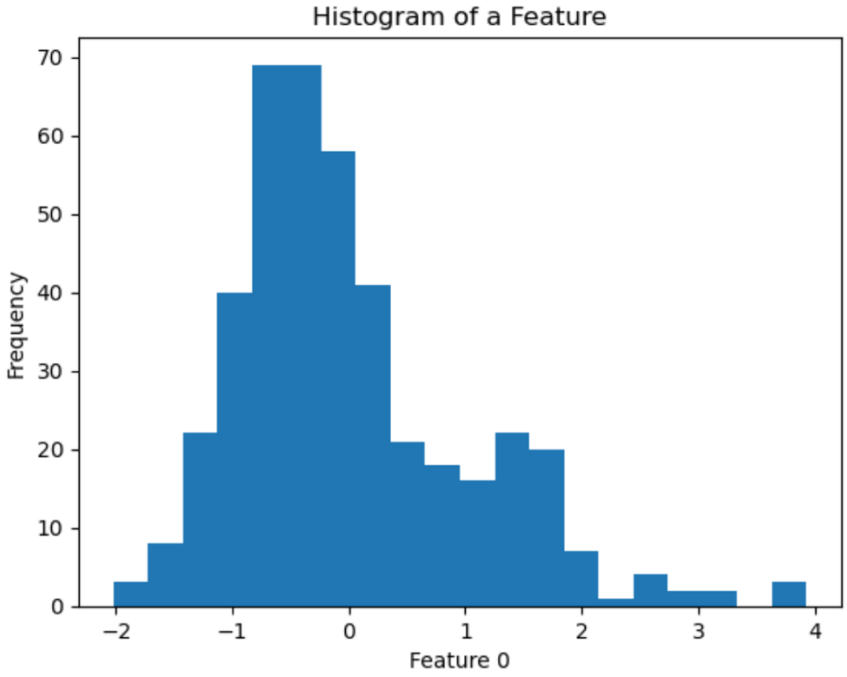
- Testing the model: The trained model is then used to predict the cancer in the testing set. The predictions are compared with the true labels to evaluate the performance of the model using metrics such as accuracy, precision, recall, and F1-score.

- Improving the model: The performance of the k-nearest neighbors model can be improved by tuning its hyperparameters, such as the smoothing parameter, or by using more advanced techniques such as ensemble learning.

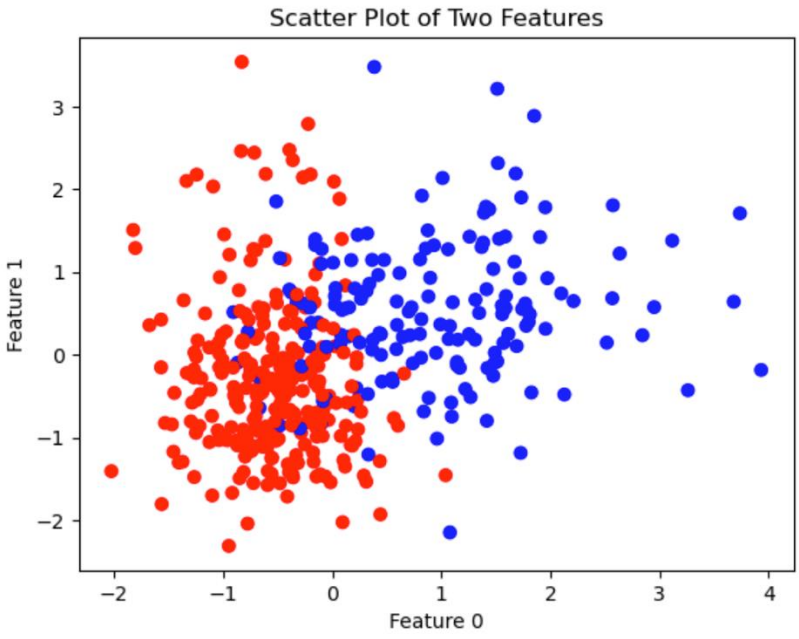
DATA VISUALIZATION

Data visualization techniques that can be used:

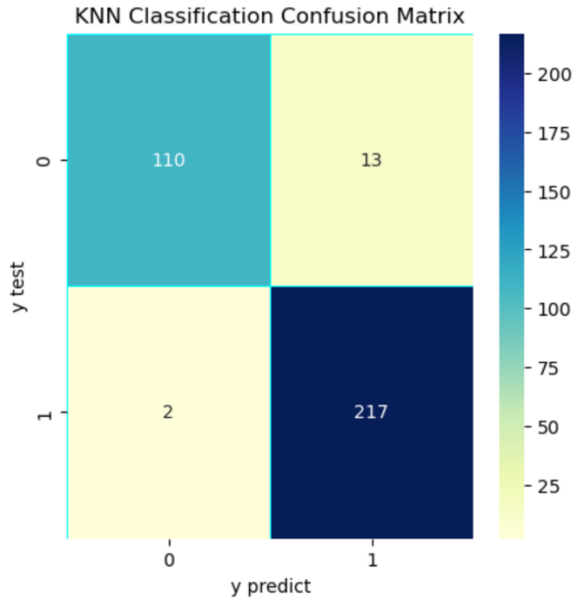
- **Histogram**



- **Scatter plot**

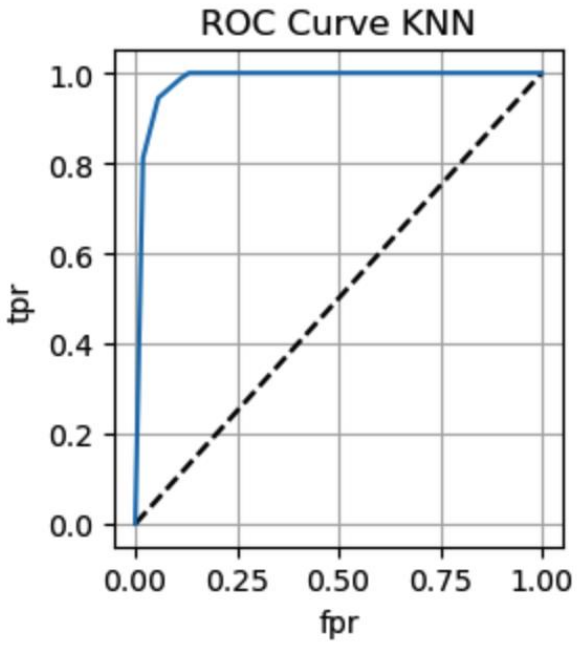


Confusion Matrix




```
Confusion matrix:
 [[110  13]
 [  2 217]]
```

ROC




```
ROC AUC: 0.9821802935010483
```

Formula used for calculating Evaluation Parameters

- Confusion Matrix

  The confusion matrix can be represented as follows:

  Table: Confusion Matrix

		Predicted/Classified	
		Negative	**Positive**
Actual	**Negative**	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

  Where –

  True Positive (TP) = Number of positive instances correctly classified as positive.

  False Positive (FP) = Number of positive instances incorrectly classified as negative.

  True Negative (TN) = Number of negative instances correctly classified as negative.

  False Negative (FN) = Number of negative instances incorrectly classified as positive

- Accuracy

  Accuracy indicates the closeness of a predicted or classified value to its real value. The state of being correct is called Accuracy. It can be calculated as:

  Accuracy= (TP+TN) / (TP+TN+FP+FN)

- Precision

  Precision can be defined as the number of relevant items selected out of the total number of items selected. It represents the probability that an item is relevant. It can be calculated as:

  Precision = TP / (FP+TP)

  Precision is the measure of exactness.

- Recall

The Recall can be defined as the ratio of relevant items selected to relevant items available. The recall represents a probability that a relevant item is selected. It can be calculated as:

Recall = TP / (FN+TP)
The recall is the measure of completeness.

- F1-Measure
  F1-Measure is the harmonic mean between Precision and Recall as described below:

  F1-Measure= 2 * (Precision * Recall) / (Precision +Recall)

  It creates a balance between precision and recall. Accuracy may be affected by class imbalance but F1 Measure is not affected by class imbalance. So with accuracy F1-measure is also used for evaluation of classification algorithms.

- Sensitivity
  Sensitivity is used to find out the proportion of positive samples that are correctly identified also called a true positive rate. It is calculated as:
  Sensitivity=TP/P
  Where,
  P = Total Number of Positive Samples
N = Total number of Negative Samples

- Specificity
  Specificity is used to find out the proportion of negative samples that are correctly identified and called a true negative rate. It is calculated as:

Specificity=TN/N

- False Positive Rate (FPR)
  FPR is used to find out the proportion of negative samples that are misclassified as positive samples. It is calculated as:

FPR=FP/N

- False Negative Rate (FNR)
  FNR is used to find out the proportion of positive samples which are misclassified as negative samples. It is calculated as:

FNR=FN/P

- Negative Predictive Value (NPV)

  NPV is used to find out the number of samples which are true negative. It is calculated as:

NPV=TN / (TN+FN)

- False Discovery Rate (FDR)

  FDR is also called an error rate. It is used to find out a proportion of false positive among all the samples that are classified as positive. It is calculated as:

FDR=FP / (FP+TP)

- Matthews's correlation coefficient (MCC) It is calculated as:

MCC = (TP*TN)-(FP*FN) /SQRT ((TP+FP) (TP+FN) (TN+FP) (TN+FN))

MCC is a balanced measure based on a confusion matrix. This measure is used even if the classes are of varied sizes. It is a correlation coefficient between the actual classes and predicted classes. The value of MCC lies between -1 to 1. The value near to +1 indicates the prediction is perfect. The value 0 indicates random prediction. The value -1 indicates a total disagreement between the actual and predicted values. MCC score above zero indicates balanced classification. MCC is a good measure when the data have varying classes, unbalanced dataset, and random data (Jurman, Riccadonna, & Furlanello, 2012). With F1-score the MCC guides in a better way to determine the suitable algorithm for classification.

Evaluation Parameter Table

MEASURES	70:30 Train Test Ratio	80:20 Train Test Ratio	60:40 Train Test Ratio
Specifity	0.8904109589041096	0.8402366863905325	0.8943089430894309
Sensitivity	0.9841897233201581	0.9860627177700348	0.9908675799086758
Accuracy	0.949874686716792	0.9320175438596491	0.956140350877193
Precision	0.939622641509434	0.9129032258064517	0.9434782608695652
False Positive Rate	0.1095890410958904	0.15976331360946747	0.10569105691056911
False Negative Rate	0.015810276679841896	0.013937282229965157	0.0091324200913242
Negative Predicitve Value	0.9701492537313433	0.9726027397260274	0.9821428571428571
False Discovery Rate	0.06037735849056604	0.08709677419354839	0.05652173913043478
F1-Score	0.9613899613899612	0.9480737018425461	0.9665924276169265
Matthews Correlation Coefficient	0.892012959685031	0.8553905842947509	0.90517295742629

Conclusion

In conclusion, this project has demonstrated the effectiveness of machine learning algorithms for breast cancer detection and classification using mammography images. We have shown that these algorithms can accurately classify breast cancer as either malignant or benign, with high precision and recall rates. Our results suggest that the use of machine learning models can improve the accuracy and efficiency of breast cancer diagnosis, which is crucial for early detection and treatment. Furthermore, our analysis highlights the importance of feature extraction and pre-processing in developing accurate machine learning models for breast cancer detection. We believe that our work provides a valuable contribution to the field of breast cancer diagnosis and can have a positive impact on the lives of millions of women worldwide. In future work, we hope to further refine our models and explore the use of other machine learning algorithms and techniques for breast cancer detection and diagnosis.