Department of Computer Science and Engineering

University of Barishal

Lab Report on

Comparative Analysis of Machine Learning Algorithms on the Breast Cancer Dataset

Submitted By

Rupa Samodder

Computer Science and Engineering

Roll: 19CSE011

Submitted To

Dr. Tania Islam

Assistant professor

Department of Computer Science and Engineering

University of Barisal

October 02, 2024

Contents

1 Introduction 1

2 Data Description and Preprocessing 1

3 Machine Learning Algorithms 2

3.1 Support Vector Machine (SVM) 2

3.1.1 Algorithm Methodology 2

3.1.2 Result of SVM 3

3.2 K-Nearest Neighbors (KNN) 4

3.2.1 Algorithm Methodology 4

3.2.2 Result of KNN 5

3.3 K-Means Clustering 5

3.3.1 Algorithm Methodology 6

3.3.2 Result of K-Means 6

3.4 Naive Bayes (NB) 6

3.4.1 Algorithm Methodology 6

3.4.2 Result of NB 7

3.5 Decision Tree (DT) 7

3.5.1 Algorithm Methodology 8

3.5.2 Result of DT 8

4 Results and Discussion 9

4.1 Evaluation Matrix 9

4.2 Result and Comparison of Classifiers 10

5 Conclusions 11

List of Figures

1 SVM classifier . . . 2

2 Result of SVM classifier on Breast Cancer Dataset . . . 3

3 KNN classifier . . . 4

4 Result of KNN classifier on Breast Cancer Dataset . . . 5

5 Steps of K-means clustering . . . 5

6 Result of K-Means Clustering on Breast Cancer Dataset . . . 6

7 Naive Bayes classifier . . . 7

8 Result of Naive Bayes on Breast Cancer Dataset . . . 7

9 Decision Tree . . . 8

10 Result of Naive Bayes on Breast Cancer Dataset . . . 8

11 A comparison of the applied ML models . . . 10

12 Comparison of the accuracy on applied ML models . . . 11

List of Tables

1 Description of Breast Cancer Dataset . . . 1

1 Introduction

Machine Learning has become an integral part of data analysis for making predictions and automate decision-making processes. Among the wide range of available algorithms **Support Vector Machine**, **Decision Tree**, **Naive Bayes**, **K-Nearest Neighbors**, and **K-means clustering** represent some of the most commonly used in both supervised and unsupervised learning tasks. This report explores the implementation and results of these five algorithms.

2 Data Description and Preprocessing

The Machine Learning algorithms are tested on **Breast Cancer Dataset** by **UCI Machine Learning** Repository. The dataset was created by Dr. William H. Wolberg at the University of Wisconsin for distinguishing between benign and malignant tumors.

Variable	**Role**	**Type**	**Description**	**Missing Values**
Sample code number	ID	Categorical	ID number	no
Clump Thickness	Feature	Integer	1-10	no
Uniformity of Cell Size	Feature	Integer	1-10	no
Uniformity of Cell Shape	Feature	Integer	1 - 10	no
Marginal Adhesion	Feature	Integer	1 - 10	no
Single Epithelial Cell Size	Feature	Integer	1 - 10	no
Bare Nuclei	Feature	Integer	1 - 10	yes
Bland Chromatin	Feature	Integer	1 - 10	no
Normal Nucleoli	Feature	Integer	1 - 10	no
Mitoses	Feature	Integer	1 - 10	no

Table 1: Description of Breast Cancer Dataset

The dataset contains 699 instances and 11 attributes, which includes features such as clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, and mitoses. The target variable indicates whether the tumor is **benign** (coded as 2) or **malignant** (coded as 4).

For this study, the dataset was preprocessed by handling missing values (as some entries in the "**bare nuclei**" feature are missing) before applying ML algorithms, scaling the features to ensure proper convergence, and splitting it into training and testing sets for model evaluation.

3 Machine Learning Algorithms

3.1 Support Vector Machine (SVM)

SVM is a linear supervised machine learning approach that is used for classification and regression. It does not cause the problem of overfitting [1]. SVM separates the classes by defining a decision boundary aiming to maximize the margin between the decision hyperplane and the nearest data point [2].

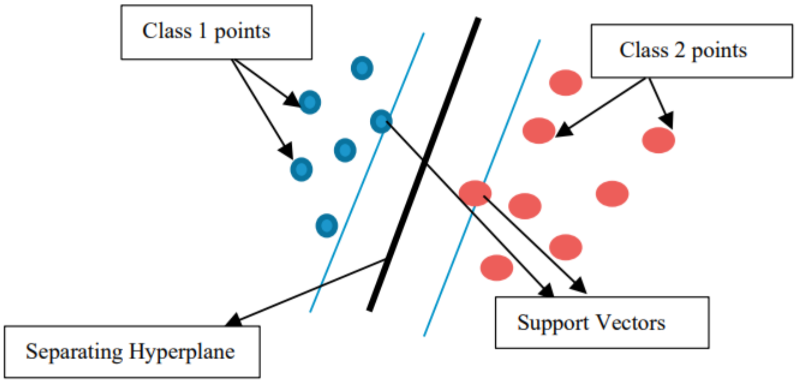


Figure 1: SVM classifier

3.1.1 Algorithm Methodology

The SVM algorithm starts by representing the dataset in an n-dimensional space, where n is the number of features in the dataset. In the case of binary classification, SVM seeks to identify a hyperplane that best separates the two classes. The margin is the distance between the hyperplane and the closest data points from either class. These closest data points are known as **support vectors**.

SVM selects the hyperplane that maximizes the margin, as this helps ensure the most robust separation between classes. A larger margin leads to better generalization and reduces the likelihood of overfitting.

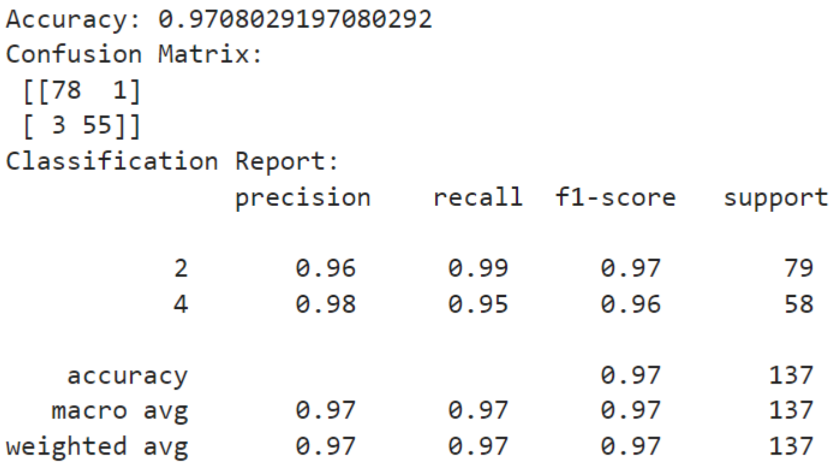
If the data is linearly separable, a straight hyperplane can effectively divide the two classes. In this case, the SVM uses a linear kernel to compute the hyperplane. If the data is not linearly separable, SVM uses a technique known as the kernel trick to project the data into a higher-dimensional space where it becomes linearly separable.

3.1.2 Result of SVM

Here, 80% of the data is used for training the model, while 20% is reserved for testing.

- 20% of 699 instances $\approx$ 137 instances were set aside for testing.
- The remaining 70%, or approximately 559 instances, were used for training the model.

Output:



```
Accuracy: 0.9708029197080292
Confusion Matrix:
 [[78  1]
 [ 3 55]]
Classification Report:
              precision    recall  f1-score   support

           2       0.96      0.99      0.97        79
           4       0.98      0.95      0.96        58

    accuracy                           0.97       137
   macro avg       0.97      0.97      0.97       137
weighted avg       0.97      0.97      0.97       137
```

Figure 2: Result of SVM classifier on Breast Cancer Dataset

3.2 K-Nearest Neighbors (KNN)

The K-Nearest Neighbors algorithm is a supervised machine learning method used for classification and regression [2].

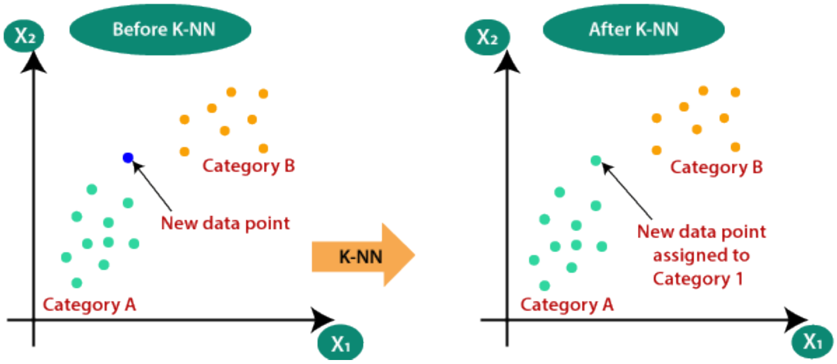


Figure 3: KNN classifier

3.2.1 Algorithm Methodology

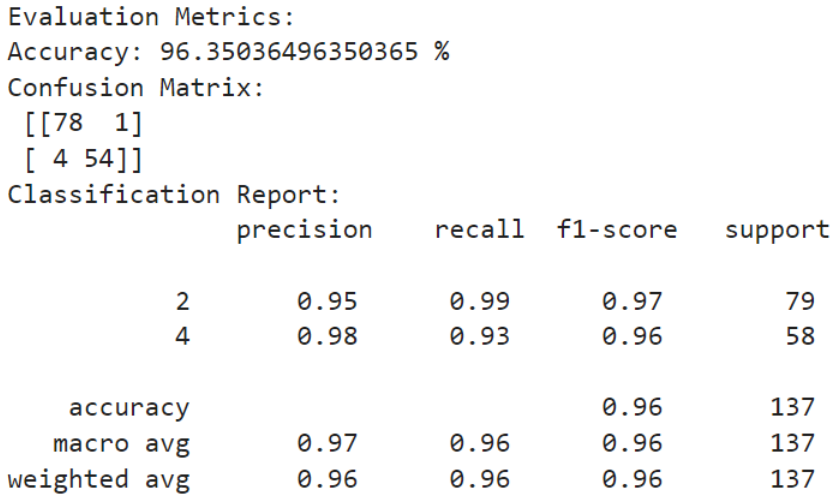
KNN operates under the assumption that similar data points are located in closely, where 'K' signifies the number of seed points to be chosen, requiring a careful selection to minimize errors [1]. It relies on the concept of similarity through factors like distance or nearest neighbor identification[2]. By finding the k data points nearest to a new data point x using the Euclidean distance metric, KNN employs majority voting to assign a label to x, and values of k (k=1 to k=10) yielded the highest accuracy [3].

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

KNN is a lazy learner, meaning it doesn't build a model during training. Instead, it stores the entire training dataset and makes predictions only when a new data point is presented. For classification task, the majority class should be determine among the K nearest neighbors. For regression, the average or weighted average is calculated of the target values of the K nearest neighbors.

3.2.2 Result of KNN

Output:



```
Evaluation Metrics:
Accuracy: 96.35036496350365 %
Confusion Matrix:
 [[78  1]
 [ 4 54]]
Classification Report:
               precision    recall  f1-score   support

           2       0.95      0.99      0.97        79
           4       0.98      0.93      0.96        58

    accuracy                           0.96       137
   macro avg       0.97      0.96      0.96       137
weighted avg       0.96      0.96      0.96       137
```

Figure 4: Result of KNN classifier on Breast Cancer Dataset

3.3 K-Means Clustering

K-means clustering is an unsupervised machine learning algorithm used to group data into distinct clusters. The goal is to group similar data points together while maximizing the distance between clusters.

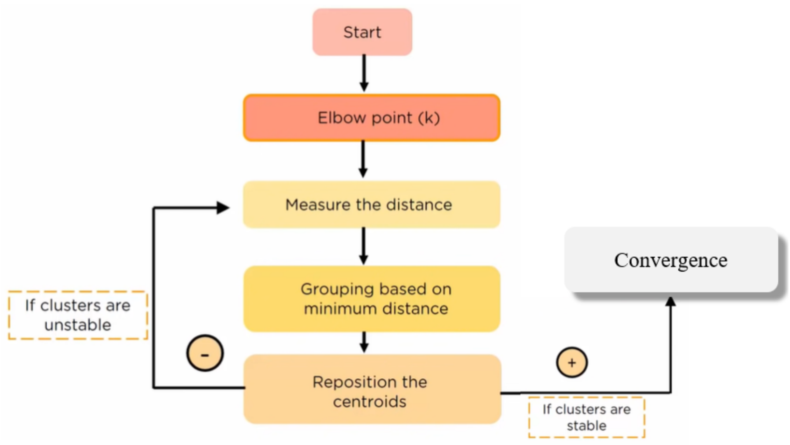


Figure 5: Steps of K-means clustering

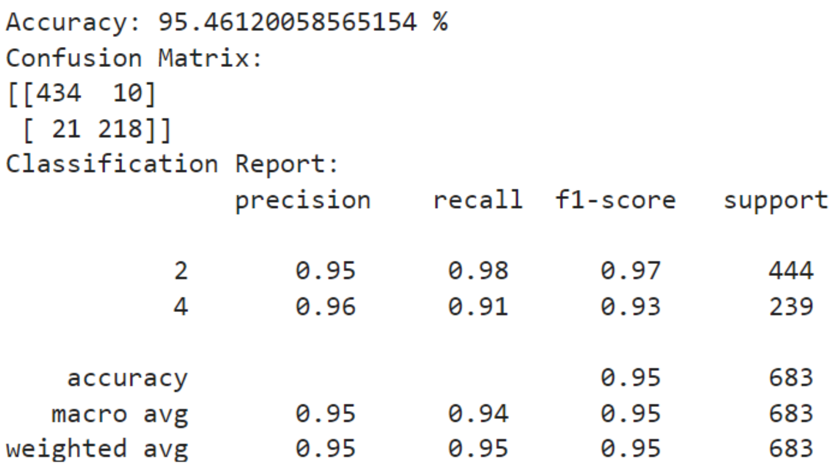
3.3.1 Algorithm Methodology

K-means works by iteratively assigning each data point to the nearest cluster centroid, which is the average of all points in that cluster.

Initially, a random set of centroids is chosen or can be chosen by different such as Elbow, and then the algorithm repeatedly assigns data points to clusters, recalculates centroids, and repeats until the clusters stabilize. The goal is to minimize the sum of distances between data points and their respective cluster centroids, effectively grouping similar data together.

3.3.2 Result of K-Means

Output:



```
Accuracy: 95.46120058565154 %
Confusion Matrix:
[[434  10]
 [ 21 218]]
Classification Report:
              precision    recall  f1-score   support

           2       0.95      0.98      0.97       444
           4       0.96      0.91      0.93       239

    accuracy                           0.95       683
   macro avg       0.95      0.94      0.95       683
weighted avg       0.95      0.95      0.95       683
```

Figure 6: Result of K-Means Clustering on Breast Cancer Dataset

3.4 Naive Bayes (NB)

Naïve bayes is one of the supervised machine learning approaches that are mainly known as Bayesian algorithms with a simple conditional probability distribution [2].

3.4.1 Algorithm Methodology

The main principle of naïve bayes is focused on the expectations of freedom, which indicates less training time to be compared to the SVM approach [2]. The algorithm

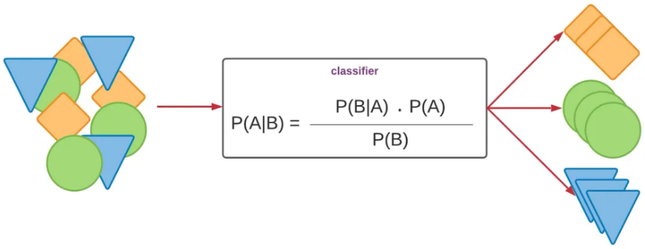
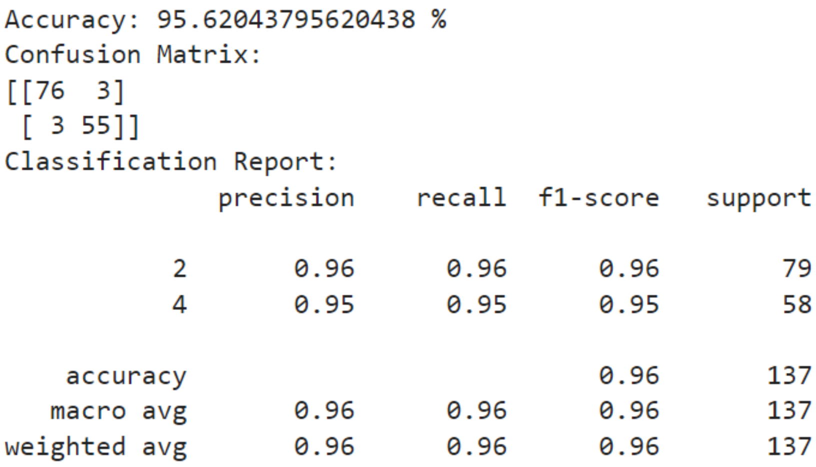


Figure 7: Naive Bayes classifier

assumes that features are independent of each other given the class, a simplification known as the ”naive” assumption. It calculates the posterior probability for a dataset using the prior probability and likelihood. [1]. Then the class with the highest calculated probability is assigned to the new data point.

3.4.2 Result of NB

Output:



```
Accuracy: 95.62043795620438 %
Confusion Matrix:
[[76  3]
 [ 3 55]]
Classification Report:
              precision    recall  f1-score   support

           2       0.96      0.96      0.96        79
           4       0.95      0.95      0.95        58

    accuracy                           0.96       137
   macro avg       0.96      0.96      0.96       137
weighted avg       0.96      0.96      0.96       137
```

Figure 8: Result of Naive Bayes on Breast Cancer Dataset

3.5 Decision Tree (DT)

Decision trees are supervised machine learning algorithms used for both classification and regression tasks. Decision trees work by creating a series of decision nodes and leaf nodes. Each decision node represents a test on an attribute, while leaf nodes represent the final decision or prediction.

3.5.1 Algorithm Methodology

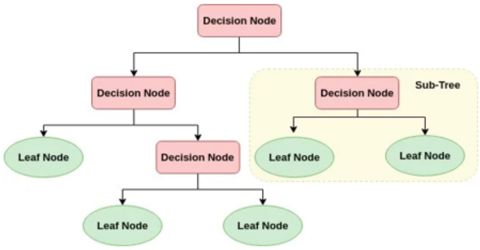
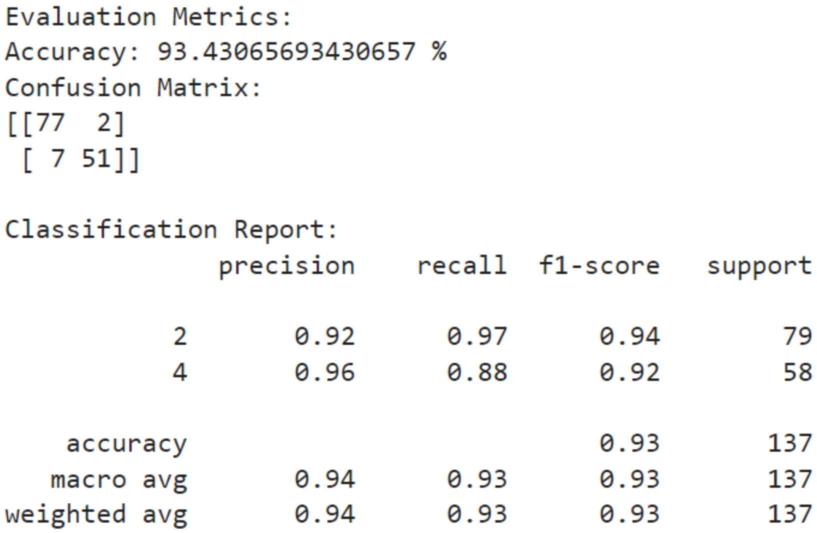


Figure 9: Decision Tree

The algorithm recursively splits the data based on the attribute that results in the most homogeneous subsets until a stopping criterion is met. This process creates a tree-like structure where each branch represents a possible combination of attribute values and the corresponding outcome. To make a prediction for a new data point, it is simply passed through the tree, following the branches based on its attribute values until it reaches a leaf node, which provides the prediction.

3.5.2 Result of DT

Output:



```
Evaluation Metrics:
Accuracy: 93.43065693430657 %
Confusion Matrix:
[[77  2]
 [ 7 51]]

Classification Report:
              precision    recall  f1-score   support

           2       0.92      0.97      0.94        79
           4       0.96      0.88      0.92        58

    accuracy                           0.93       137
   macro avg       0.94      0.93      0.93       137
weighted avg       0.94      0.93      0.93       137
```

Figure 10: Result of Naive Bayes on Breast Cancer Dataset

4 Results and Discussion

4.1 Evaluation Matrix

- **True positive (TP):** The individual has Cancer which is malignant and we predicted correctly that the individual has Cancer.

- **True negative (TN):** The individual does not have Cancer and we predicted correctly that the individual does not have Cancer.

- **False positive (FP):** The individual does not have Cancer, but we predicted incorrectly that the individual has Cancer.

- **False negative (FN):** The individual has Cancer, but we predicted incorrectly that the individual does not have Cancer.

The above four categories when put together in the form of matrix produce the confusion matrix [3].The performance generally measured by accuracy, precision, recall, and F1 score.

- **Accuracy** is the simplest metric, and it measures the percentage of predictions that the model makes correctly [3]. For example, if a model predicts that 100 examples are positive and 90 of those predictions are correct, then the model has an accuracy of 90% .

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}.$$

- **Precision** measures the percentage of positive predictions that are actually correct [3]. For example, if a model predicts that 100 examples are positive and 90 of those predictions are correct, then the model has a precision of 90%.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

- **Recall** measures the percentage of actual positives that the model correctly predicts [2]. For example, if there are 100 actual positive examples and the model predicts that 90 of them are positive, then the model has a recall of 90%.

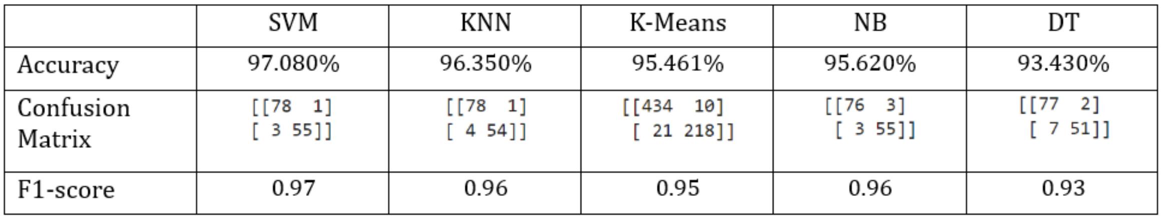
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

- **F1 score** is a harmonic mean of precision and recall, and it is often used to evaluate the performance of machine learning models when there is a trade-off between precision and recall [2].

$$F1 = \frac{2 \times precision \times recall}{precision + recall}$$

4.2 Result and Comparison of Classifiers

The evaluation on accuracy, confusion matrix, and F1-score key metrics provide a comprehensive view of the classification performance and the ability of each algorithm to generalize on unseen data.



	SVM	KNN	K-Means	NB	DT
Accuracy	97.080%	96.350%	95.461%	95.620%	93.430%
Confusion Matrix	[[78 1] [3 55]]	[[78 1] [4 54]]	[[434 10] [21 218]]	[[76 3] [3 55]]	[[77 2] [7 51]]
F1-score	0.97	0.96	0.95	0.96	0.93

Figure 11: A comparison of the applied ML models

- **Accuracy**

The highest accuracy of 97.08% was achieved by the SVM model, closely followed by KNN with an accuracy of 96.35%. K-Means, an unsupervised algorithm, also showed competitive accuracy at 95.46%, which was similar to the performance of the Naive Bayes (NB) classifier at 95.62%. The Decision Tree (DT) classifier, while still achieving a respectable accuracy of 93.43%, lagged behind the other models.

- **Confusion Matrix**

The confusion matrices reveal how well each model distinguishes between benign and malignant cases. For both SVM and KNN, the true positive and true negative counts were very close, with SVM producing only 1 false positive and 3 false negatives, while KNN

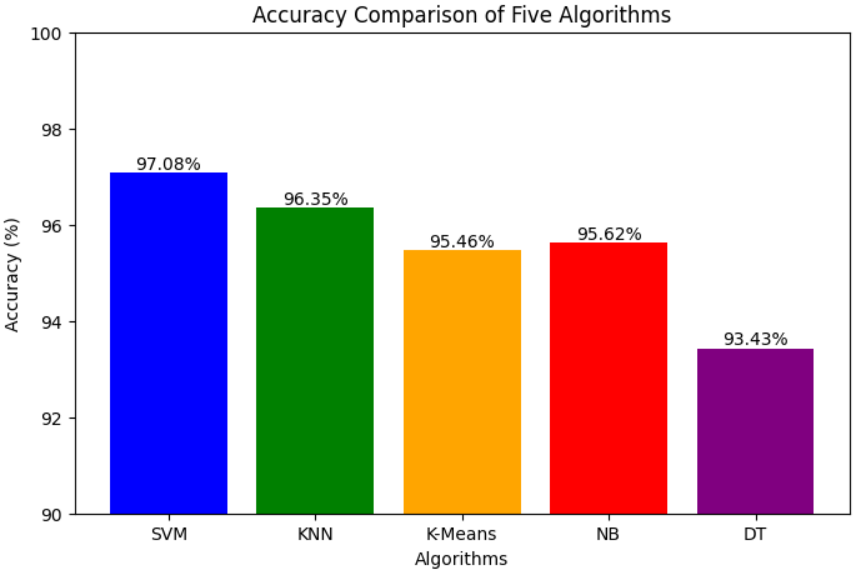


Figure 12: Comparison of the accuracy on applied ML models

had 1 false positive and 4 false negatives. K-Means showed higher error rates compared to the other models, though this is expected as it is an unsupervised algorithm that clusters without prior knowledge of the class labels. NB and DT had similar confusion matrices.

- **F1-Score**

The F1-score, which balances precision and recall, was highest for SVM at 0.97, indicating its overall strong performance across both metrics. K-Means, despite being an unsupervised method, achieved an F1-score of 0.95. NB matched KNN with an F1-score of 0.96 which is consistent with its high accuracy and confusion matrix results, while DT scored slightly lower at 0.93, reflecting its relatively lower accuracy and higher error rates.

5 Conclusions

The performance of five machine learning algorithms **Support Vector Machine (SVM), K-Nearest Neighbors (KNN), K-Means, Naive Bayes (NB), and Decision Tree (DT)** is evaluated on the **Breast Cancer Wisconsin (Original)** dataset. These Machine Learning algorithms on the dataset give the overall understanding of how efficiently the accuracy can be found.

References

[1] Suman Raj and Sarfaraz Masood. Analysis and detection of autism spectrum disorder using machine learning techniques. *Procedia Computer Science*, 167:994–1004, 2020.

[2] Nurul Amirah Mashudi, Norulhusna Ahmad, and Norliza Mohd Noor. Classification of adult autistic spectrum disorder using machine learning approach. *IAES International Journal of Artificial Intelligence*, 10(3):743, 2021.

[3] Kaushik Vakadkar, Diya Purkayastha, and Deepa Krishnan. Detection of autism spectrum disorder in children using machine learning techniques. *SN Computer Science*, 2:1–9, 2021.