CSE449: High Performance Computing

Leveraging High-Performance Computing for Heart Disease Prediction

Submitted by:
Shouvik Banerjee Argha (20301118)
Arian Wazed (20301039)

Table of Contents

1.Introduction:

Heart is one of the most extensive and vital organs of the human body so the care of the heart is essential. Most diseases are related to the heart so the prediction about heart diseases is necessary and for this purpose comparative study needed in this field, today most patients die because their diseases are recognized at the last stage due to lack of accuracy of the instrument so there is need to know about more efficient algorithms for disease prediction. Machine Learning is one of the efficient technologies for testing, which is based on training and testing. It is the branch of Artificial Intelligence(AI) which is one of broad areas of learning where machines emulate human abilities, machine learning is a specific branch of AI. On the other hand machines learning systems are trained to learn how to process and make use of data hence the combination of both technologies is also called Machine Intelligence. As the definition of machine learning, it learns from the natural phenomenon, natural things so in this project we uses the biological parameter as testing data such as cholesterol, Blood pressure, sex, age, etc. and on the basis of these, comparison is done in the terms of accuracy of algorithms such as in this project we have used four algorithms which are Decision tree, Logistic Regression, KNN, Naive Bayes.In this paper, we calculate the accuracy of four different machine learning approaches and on the basis of calculation we conclude that which one is best among them.

2.Dataset Description:

The dataset comprises 303 rows and 14 columns, where each row represents a data point and each column corresponds to a feature or attribute. Among these features, there are 13 predictors and one target variable, making it a classification problem with a binary outcome. The features encompass both quantitative aspects like age, resting blood pressure, serum cholesterol, maximum heart rate achieved, and ST depression induced by exercise relative to rest, as well as categorical attributes such as sex, chest pain type, fasting blood sugar, resting electrocardiographic results, exercise-induced angina, slope of the peak exercise ST segment, number of major vessels colored by fluoroscopy, and thalassemia. Understanding correlations between these features is facilitated by a correlation heatmap matrix, where darker points signify stronger correlations and lighter points indicate weaker or no correlations. Additionally, the dataset exhibits class imbalance, meaning that certain classes within the target variable have significantly fewer samples compared to others. Addressing this imbalance involves imputing missing values with column means and employing algorithms tailored for imbalanced data to enhance model performance and ensure fair representation of all classes, measured through metrics like F1-score, precision, and recall.

3. Dataset Preprocessing:

Faults: In the dataset there are some null values which we imputed later.
Solutions: Identified the null values and removed the row which contained Null values

```
[31]  1 df.isnull().sum()

age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

```
1 # making new data frame with dropped NA values
2 data = df
3 new_data = data.dropna(axis=0, how='any')
4
5 # comparing sizes of data frames
6 print("Old data frame length:", len(data),
7       "\nNew data frame length:",
8       len(new_data),
9       "\nNumber of rows with at least")
```

```
Old data frame length: 303
New data frame length: 303
Number of rows with at least
```

4. Feature Scaling :

MinMaxScaler is useful when the data has a bounded range.Scaling these values using MinMaxScaler ensures that the values are within a fixed range and contributes equally to the analysis. In the dataset, some of the values were outliers that's why it's needed.

```
[10]
from sklearn.preprocessing import MinMaxScaler

# Define the feature columns
feature_cols = ['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach', 'exang', 'oldpeak', 'ca', 'thal']

# Select the features and target variable
X = df[feature_cols]
y = df.target

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=80)

# Initialize MinMaxScaler
scaler = MinMaxScaler()

# Fit and transform the training data
X_train_scaled = scaler.fit_transform(X_train)

# Transform the testing data
X_test_scaled = scaler.transform(X_test)
```

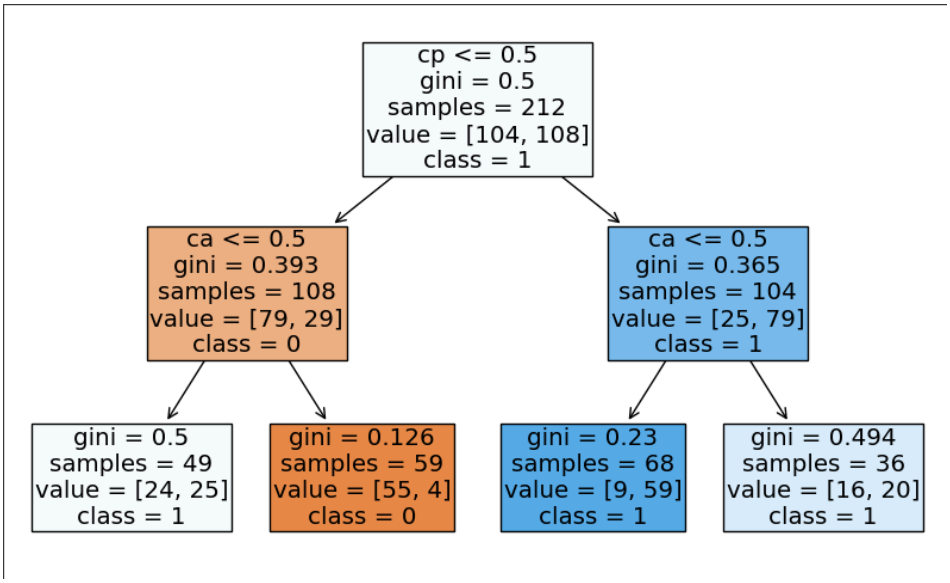
5. Dataset Splitting:

The dataset which has been used, the test size is 0.3 So, the training set ratio is 70%, testing set ratio 30% and random_state is 80.

6.Model training & testing:

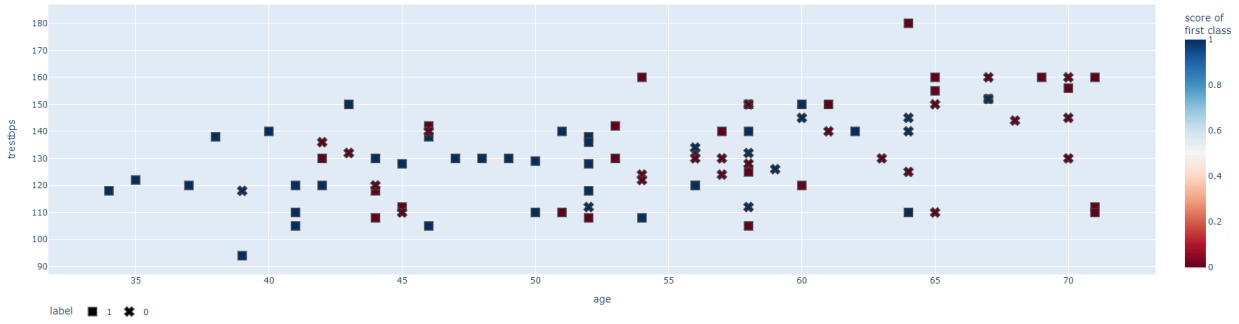
Decision Tree:

The Decision Tree classifier, trained with a maximum depth of 2 and random state 0, is visualized for its decision-making. Its accuracy on the test set showcases its performance. The ROC curve and AUC score highlight its ability to discriminate between classes.



KNN:

The K-Nearest Neighbors (KNN) classifier with k=15 is trained on the provided training data (X_train, y_train). The model's predicted probabilities for the test set (X_test) are visualized using Plotly Express, where markers represent instances, color intensity denotes the probability of belonging to the first class, and shapes indicate the true class labels.



Logistic regression:

The Logistic Regression model, initialized with a random state of 2 and using the 'lbfgs' solver with maximum iterations set to 1000, is trained on the provided training data (X_train, y_train). After prediction on the test set (X_test), the classification report is generated, detailing precision, recall, and F1-score for each class.

	precision	recall	f1-score	support
without heart disease	0.79	0.76	0.78	34
with heart disease	0.86	0.88	0.87	57
accuracy			0.84	91
macro avg	0.82	0.82	0.82	91
weighted avg	0.83	0.84	0.83	91

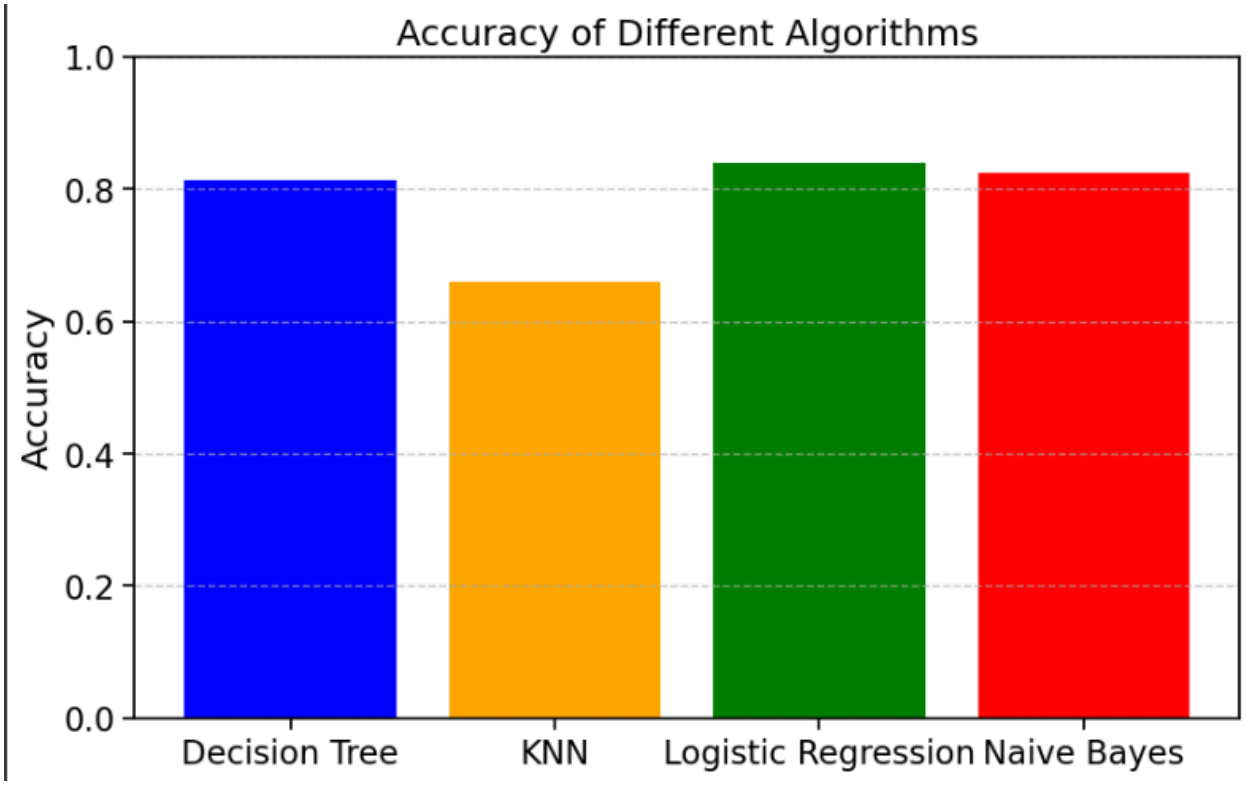
Naive Bayes:

The Gaussian Naive Bayes (GNB) classifier is trained on the provided training data (X_train, y_train). Following prediction on the test set (X_test), the accuracy of the model is evaluated using metrics.accuracy_score, demonstrating its performance in correctly classifying instances. Additionally, a scatter plot is created using make_blobs to visualize synthetic data with two classes, showcasing the separation achieved by the GNB classifier.

Fig: Naive Bayes Scatter Diagram

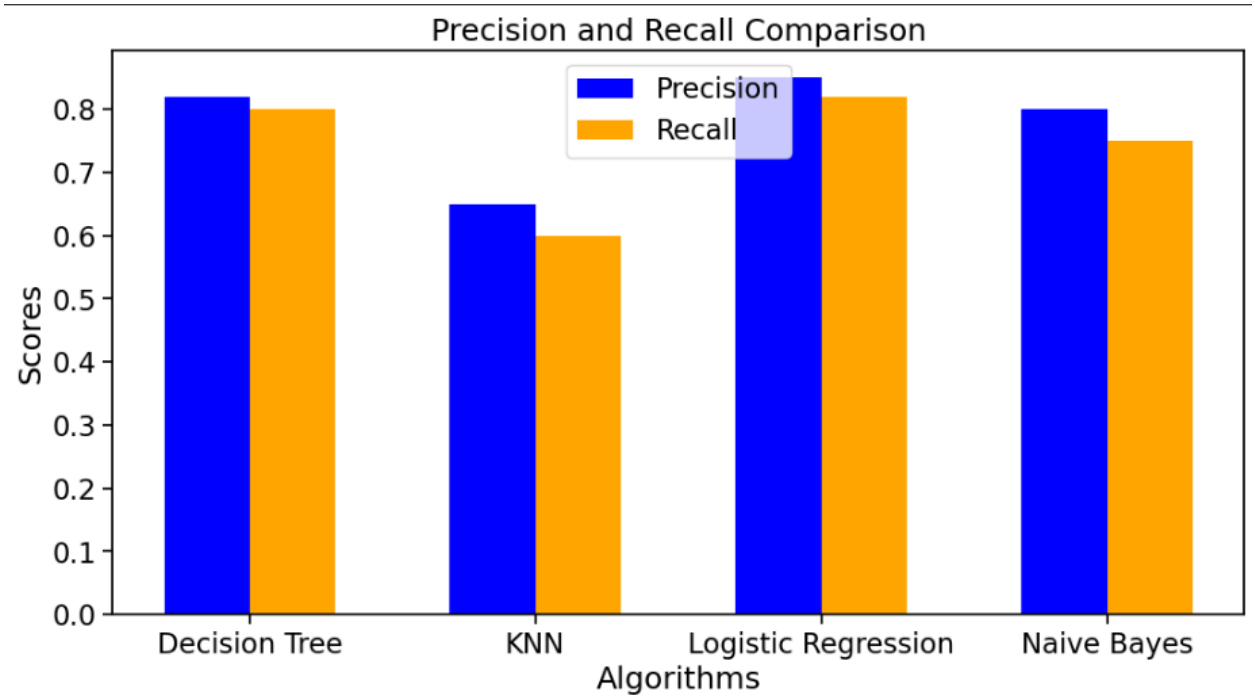
7.Model Selection/Comparison Analysis:

i) Bar chart showcasing prediction accuracy of all models:



The bar chart displays the prediction accuracy of four different machine learning models: Decision Tree, KNN (K-Nearest Neighbors), Logistic Regression, and Naive Bayes. Each bar represents the accuracy score achieved by a specific model on a given dataset. Among the models, Logistic Regression exhibits the highest accuracy at 84%, closely followed by Naive Bayes with an accuracy of approximately 82.42%. Decision Tree achieves an accuracy of around 81.32%, while KNN performs slightly lower with an accuracy of approximately 65.93%. The chart provides a clear visual comparison of the prediction performance of these models, highlighting Logistic Regression and Naive Bayes as the top-performing algorithms in terms of accuracy.

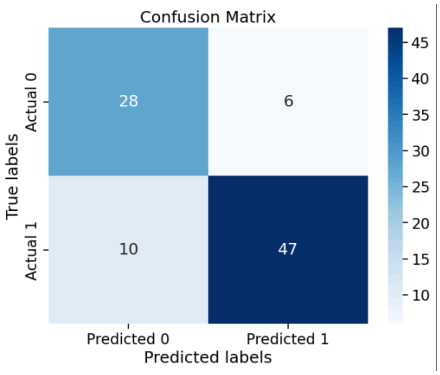
ii) Precision, recall comparison of each model:



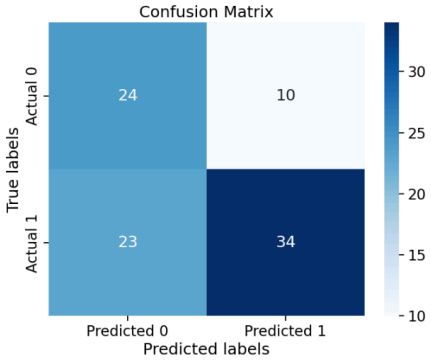
This F1-Score chart compares the precision and recall scores of four different machine learning algorithms: Decision Tree, KNN (K-Nearest Neighbors), Logistic Regression, and Naive Bayes. Precision measures the accuracy of positive predictions made by the model, while recall measures the proportion of actual positives that were correctly identified by the model. In this comparison, Logistic Regression demonstrates the highest precision score of 0.85, indicating that it is relatively adept at correctly identifying positive cases. However, Decision Tree and Naive Bayes closely follow with precision scores of 0.82 and 0.80, respectively. Regarding recall, Logistic Regression again exhibits the highest score of 0.82, suggesting its effectiveness in capturing a larger proportion of actual positive cases. Meanwhile, Decision Tree and Naive Bayes maintain slightly lower recall scores of 0.80 and 0.75, respectively, implying that they might miss identifying some positive instances. KNN lags behind with the lowest scores for both precision (0.65) and recall (0.60), indicating a comparatively poorer performance in accurately identifying positive cases and capturing true positives. Overall, this

comparison offers insights into the strengths and weaknesses of each algorithm in terms of their predictive accuracy and ability to detect positive instances.

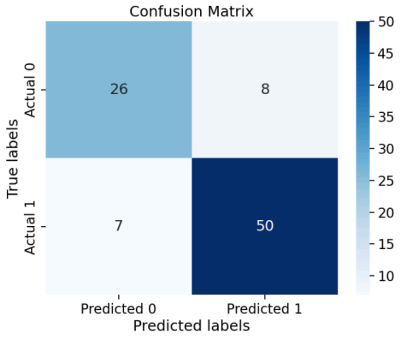
iii) Confusion Matrix:



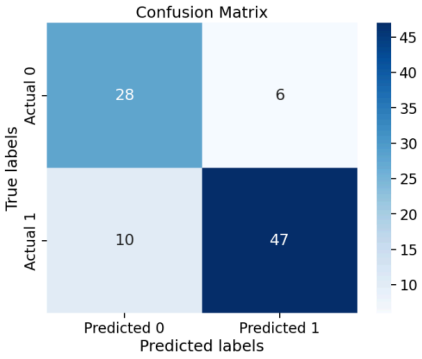
Confusion matrix for decision tree



Confusion matrix for KNN



Confusion matrix for Logistic Regression



Confusion matrix for Naive Bayes

Accuracy of the algorithms depends on four values namely true positive(TP), false positive(FP), true negative(TN) and false negative(FN).

Accuracy= (FN+TP) / (TP+FP+TN+FN)
The numerical value of TP, FP, TN, FN defines as:

TP = Number of person with heart diseases
TN = Number of person with heart diseases and no heart diseases
FP = Number of person with no heart diseases
FN = Number of person with no heart diseases and with heart Diseases

For Recall = TP/ (TP+FN)
For Precision= TP/ (TP+FP)

8.Conclusion:

Heart is one of the essential and vital organs of the human body and prediction about heart diseases is also an important concern for the human beings so that the accuracy of the algorithm is one of the parameters for analysis of performance of algorithms. Accuracy of the algorithms in machine learning depends upon the dataset that is used for training and testing purposes. When we perform the analysis of algorithms on the basis of a dataset and on the basis of the confusion matrix, we find Logistic Regression is the best one. For the Future Scope more machine learning approach will be used for best analysis of the heart diseases and for earlier prediction of diseases so that the rate of the death cases can be minimized by the awareness about the diseases.