

A ML Lab Project Report on

# **Heart Disease Prediction Using Machine Learning Models**

Submitted to Manipal University Jaipur  
Towards the partial fulfillment for the Award of the Degree of  
**BACHELOR OF TECHNOLOGY**  
In Computers Science and Engineering (AIML)

2024-2025

By

Harsh Pal - 229310041



**MANIPAL UNIVERSITY  
JAIPUR**

Under the guidance of

**Mr. Surendra Solanki**

Name of the Supervisor

---

Signature of Supervisor

**Department of Artificial Intelligence and Machine Learning**  
**School of Computer Science and Engineering**  
**Manipal University Jaipur**  
**Jaipur, Rajasthan**

# CERTIFICATE

This is to certify that the ML Lab report (Project) entitled **Heart Disease Prediction Using Machine Learning Models** submitted by **Harsh Pal (229310041)**, Department of Artificial Intelligence and Machine Learning (AIML), School of Computer Science and Engineering, Manipal University Jaipur, Rajasthan for the award of the degree of *Bachelor of Technology* is a record of Bonafide work carried out by him/her under my supervision, as per the code of academic and research ethics of Manipal University Jaipur, Rajasthan.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The report fulfills the requirements and regulations of the University and in my opinion, meets the necessary standards for submission.

Place: Manipal University Jaipur

Signature Date:

**Mr. Surendra Solanki**

## ABSTRACT

Heart disease remains one of the leading causes of mortality worldwide, making early and accurate prediction crucial for effective treatment and prevention. This project presents a comparative analysis of multiple machine learning models for the prediction of heart disease based on clinical attributes. The dataset comprises various health parameters such as age, blood pressure, cholesterol levels, and more, with a target variable indicating the presence or absence of heart disease.

We implemented and evaluated seven classification algorithms: Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), XGBoost, and a Neural Network. Each model was trained using an 80-20 train-test split and assessed based on accuracy, precision, recall, and F1-score. Among these, KNN emerged as the best performer with an accuracy of 90.2% and an F1-score of 0.903, followed closely by SVM and Neural Networks. Logistic Regression also provided reliable and interpretable results.

Additionally, we visualized learning curves for models like XGBoost and Neural Networks to monitor training and validation performance. Our findings highlight that KNN is highly effective for this dataset, while SVM offers a good trade-off between performance and generalization. This study underscores the potential of machine learning in augmenting diagnostic decisions in healthcare.

# 1. Introduction

Cardiovascular diseases (CVDs), particularly heart disease, are among the leading causes of death globally. According to the World Health Organization, an estimated 17.9 million people die from CVDs each year, representing 32% of all global deaths. A large proportion of these deaths are preventable through early diagnosis and timely intervention. In this context, the integration of data science and machine learning in the field of healthcare has emerged as a promising approach to aid in the prediction and early detection of heart diseases.

The primary goal of this project is to leverage machine learning techniques to develop a predictive model that can classify whether a patient is likely to have heart disease based on a set of clinical and demographic features. These features typically include age, sex, blood pressure, cholesterol levels, maximum heart rate, and other vital health indicators. By learning patterns from historical data, machine learning models can assist healthcare professionals in making faster, more accurate, and data-driven decisions.

This project explores and compares the performance of several popular classification algorithms: Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Extreme Gradient Boosting (XGBoost), and a simple Neural Network. Each model brings its own strengths—some excel in interpretability, while others offer high predictive power and adaptability to complex data patterns.

The dataset used in this project is a structured CSV file containing multiple medical attributes for different patients. The target variable is binary, indicating the presence (1) or absence (0) of heart disease. The dataset is pre-processed and split into training and testing subsets to evaluate model generalization. Model performance is measured using key classification metrics: accuracy, precision, recall, F1-score, and confusion matrix. Additionally, learning curves are plotted for models like XGBoost and Neural Network to analyse training behaviour and overfitting tendencies.

By comparing these models on a standardized dataset, we aim to identify the most effective algorithm for heart disease prediction. Ultimately, the findings from this study highlight the value of machine learning in medical diagnosis and pave the way for future deployment in real-time health monitoring systems and decision support tools for clinicians.

## 2. Objective

- **To develop a machine learning-based predictive model** that can accurately determine the presence or absence of heart disease using clinical and demographic attributes of patients. To predict repeat offenders based on behavioral patterns and historical data from real datasets.
- **To implement multiple machine learning algorithms** for their robustness, ensemble learning capabilities, and accuracy, making it suitable for medical prediction tasks.
- **To perform data preprocessing and feature scaling** to ensure high model efficiency and to standardize the input features for better training and generalization.
- **To evaluate and compare model performance based on standard classification metrics:** Precision, Recall, F1-Score, Accuracy.
- **To evaluate the model's performance** using Training and Validation Graph and Confusion Matrix.

## 3. Literature Review

In recent years, machine learning has emerged as a powerful tool for early diagnosis and risk prediction in the healthcare sector, particularly for heart disease. Numerous studies have explored the use of classical models like Logistic Regression, Naïve Bayes, and Decision Trees due to their ease of implementation and interpretability. These models have proven effective in identifying key risk factors such as age, cholesterol levels, and blood pressure. Researchers have emphasized the importance of data preprocessing techniques—such as normalization, feature selection, and handling missing values—to ensure the robustness of these models and improve classification performance.

Beyond traditional models, recent literature has focused on the application of more advanced techniques like Random Forests, Support Vector Machines (SVM), Neural Networks, and ensemble learning methods such as XGBoost. These models are capable of capturing complex, non-linear relationships in medical datasets, often resulting in better predictive performance. Studies have shown that ensemble methods and deep learning architectures can outperform single classifiers, especially when dealing with large and multidimensional health data. Additionally, performance evaluation metrics such as accuracy, precision, recall, and F1-score are commonly used to compare models, ensuring reliability in real-world diagnostic applications and clinical decision support systems.

## 4. System Analysis and Design

### *Methodology*

This project follows a structured machine learning pipeline for heart disease prediction. The dataset was first preprocessed by loading, exploring, and selecting relevant features. It was then split into training and testing sets. Multiple classification algorithms—Logistic Regression, Decision Tree, Random Forest, SVM, KNN, XGBoost, and Neural Network—were implemented to train predictive models. Each model was evaluated using metrics such as accuracy, precision, recall, and F1-score to ensure fair comparison. Learning curves were also plotted to analyze training behavior and avoid overfitting. The methodology enables a comprehensive assessment of each model's performance, ultimately identifying the most suitable approach for accurate predictions.

### *System Architecture*

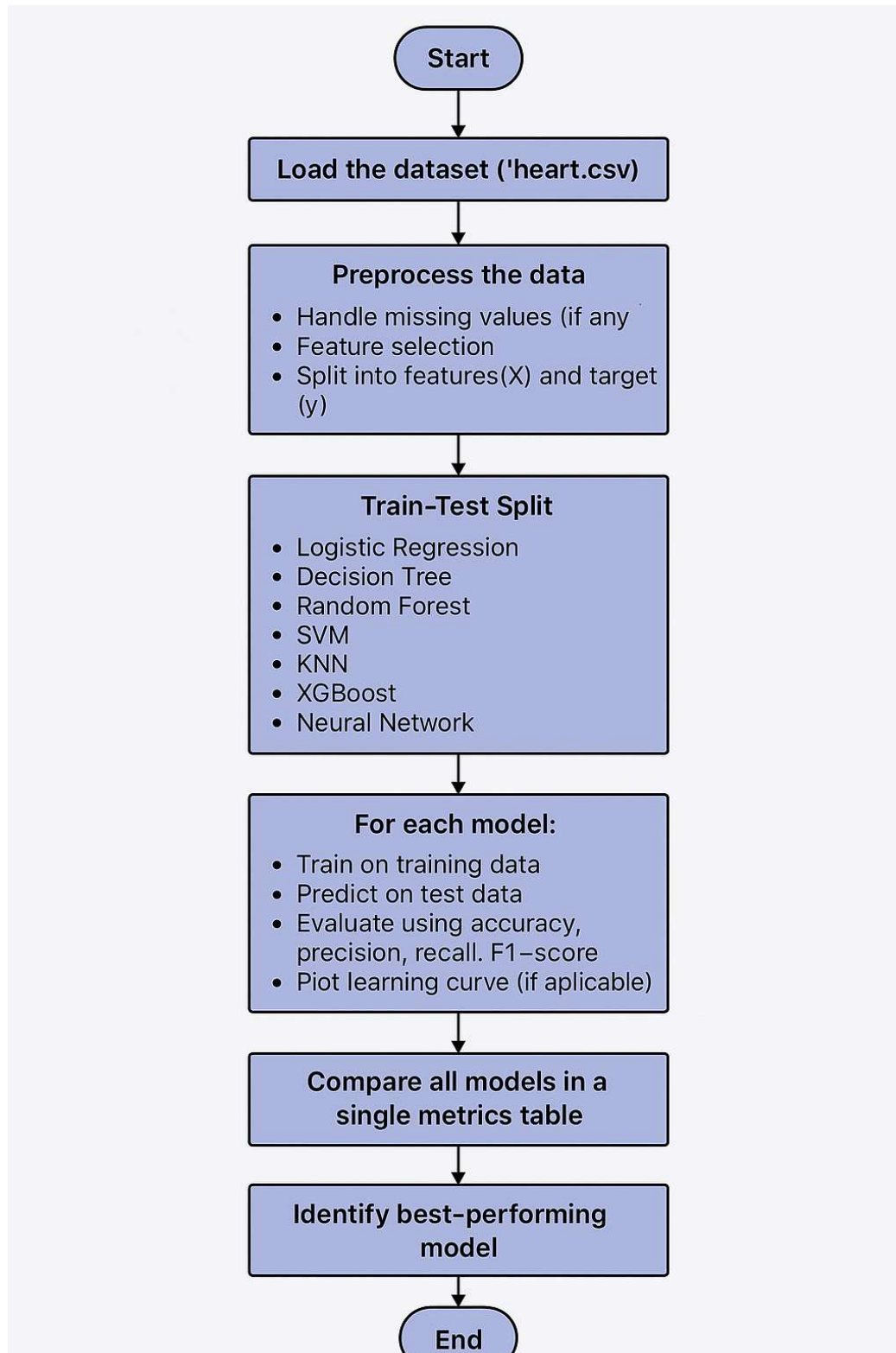
The system architecture of this heart disease prediction project is structured into four key layers: data ingestion, data processing, model training, and evaluation & visualization. The process begins with importing the data set into the environment. In the data processing layer, features are selected, and the dataset is split into training and testing sets. The model training layer involves implementing various machine learning algorithms using scikit-learn, XGBoost, and Keras. In the final layer, each model is evaluated using standard metrics, and performance is visualized using tables and plots. This modular architecture ensures clarity, scalability, and ease of integration with healthcare systems.

### *Technology Stack*

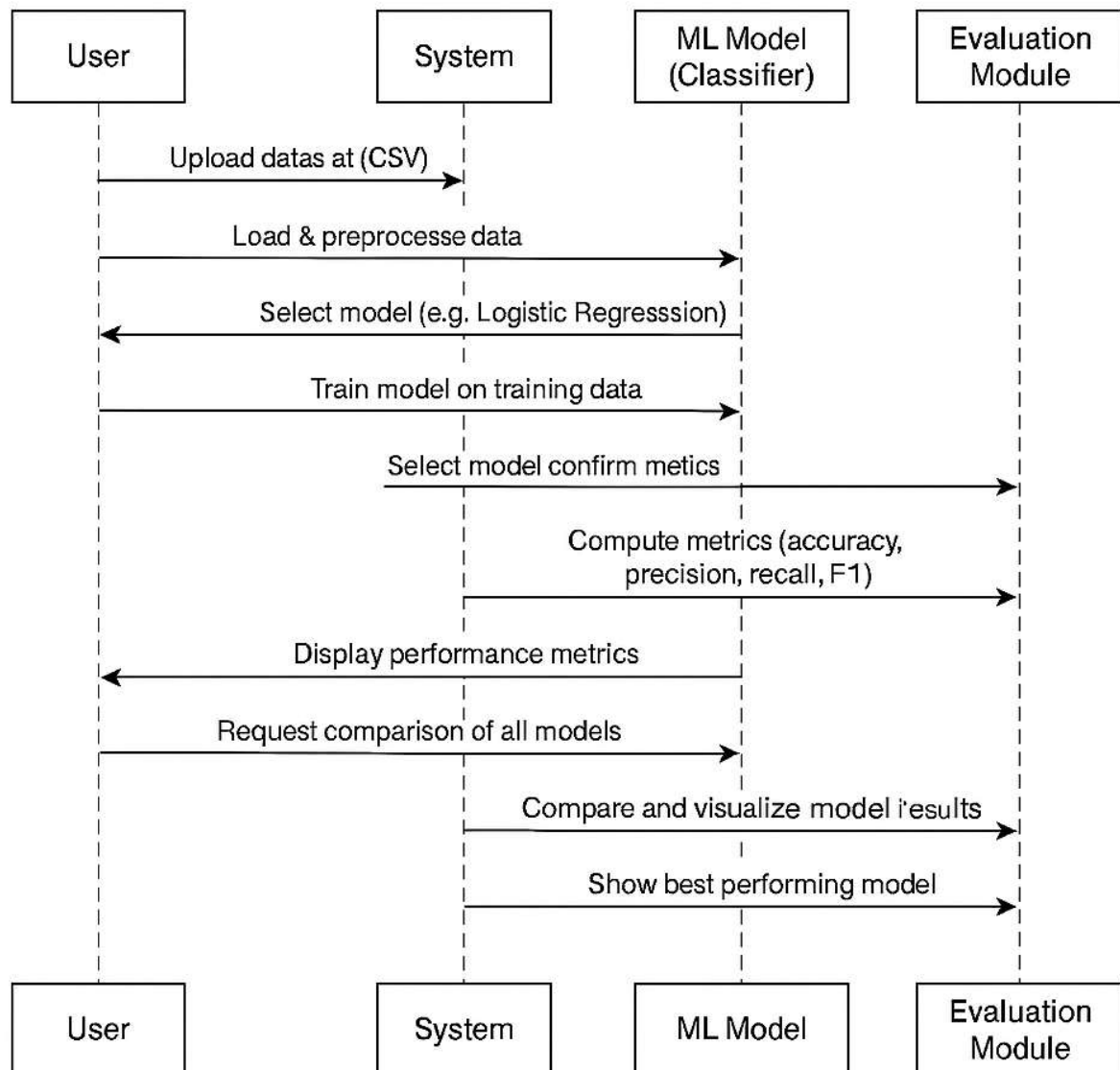
The heart disease prediction project is built using a comprehensive and robust technology stack centered around the Python programming language, which offers powerful tools for data science and machine learning. Data loading, cleaning, and manipulation are handled using Pandas and NumPy, providing efficient structures for dataset operations. For data visualization and analysis, Matplotlib and Seaborn are used to generate insightful plots and learning curves. Model development and evaluation are conducted using Scikit-learn, a versatile machine learning library that supports a wide range of classification algorithms such as Logistic Regression, Decision Tree, Random Forest, SVM, and KNN. XGBoost is employed for gradient boosting, known for its speed and accuracy in structured data tasks. Additionally, a Neural Network model is created using Keras with TensorFlow as its backend. The project is developed and run in Google Collaboratory, a cloud-based platform that enables real-time collaboration and provides GPU acceleration for faster training.

## 5. Implementation

### *Activity Diagram*



## Sequence Diagram





## Development Process

The development process began with importing and exploring the heart disease dataset using pandas. Key features and the target variable were identified, and the data was split into training and testing sets. Various machine learning models—including Logistic Regression, Decision Tree, Random Forest, SVM, KNN, XGBoost, and Neural Network—were implemented using scikit-learn, XGBoost, and Keras libraries. Each model was trained, tested, and evaluated using performance metrics like accuracy, precision, recall, and F1-score. Learning curves were plotted for deeper insights. Finally, the results were compared in a unified table, highlighting the best-performing model and providing a foundation for future healthcare applications.

### Code Snippets

#### Predicting Using Raw data

```
new_data = [[63, 1, 3, 145, 233, 1, 0, 150, 0, 2.3, 0, 0, 1]]
new_data_scaled = loaded_scaler.transform(new_data)
prediction = loaded_model.predict(new_data_scaled)
print("Prediction:", "Heart Disease" if prediction[0] == 1 else "No Disease")
```

#### Data Columns

```
Data columns (total 14 columns):
#   Column      Non-Null Count  Dtype
---  -
0   age         303 non-null      int64
1   sex         303 non-null      int64
2   cp          303 non-null      int64
3   trestbps    303 non-null      int64
4   chol        303 non-null      int64
5   fbs         303 non-null      int64
6   restecg     303 non-null      int64
7   thalach     303 non-null      int64
8   exang       303 non-null      int64
9   oldpeak     303 non-null      float64
10  slope       303 non-null      int64
11  ca          303 non-null      int64
12  thal        303 non-null      int64
13  target      303 non-null      int64
dtypes: float64(1), int64(13)
```

## Accuracy, Precision, Recall, and F1-Score for all the Algorithms

	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.852459	0.870968	0.84375	0.857143
Decision Tree	0.754098	0.840000	0.65625	0.736842
Random Forest	0.836066	0.843750	0.84375	0.843750
SVM	0.868852	0.900000	0.84375	0.870968
KNN	0.901639	0.933333	0.87500	0.903226
XGBoost	0.819672	0.862069	0.78125	0.819672
Neural Network	0.852459	0.870968	0.84375	0.857143

## Evaluating Model with Sample Input Values

```
1  import pandas as pd
2  import numpy as np
3  from sklearn.model_selection import train_test_split
4  from sklearn.preprocessing import StandardScaler
5  from sklearn.neighbors import KNeighborsClassifier
6  from sklearn.metrics import classification_report
7
8  # 1. Load the dataset
9  df = pd.read_csv("heart.csv")
10
11 # 2. Define features and target
12 X = df.drop("target", axis=1)
13 y = df["target"]
14
15 # 3. Split into training and test sets
16 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
17
18 # 4. Feature scaling
19 scaler = StandardScaler()
20 X_train_scaled = scaler.fit_transform(X_train)
21 X_test_scaled = scaler.transform(X_test)
22
23 # 5. Train the KNN model (best performing model in your case)
24 knn_model = KNeighborsClassifier(n_neighbors=5)
25 knn_model.fit(X_train_scaled, y_train)
26
27 # 6. Evaluate the model (optional)
28 y_pred = knn_model.predict(X_test_scaled)
29 print("\nModel Evaluation on Test Data:")
30 print(classification_report(y_test, y_pred))
```

```

31
32 # 7. Take input from user
33 print("\n🟡 Please enter the following patient details:")
34
35 feature_names = ['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs',
36                  'restecg', 'thalach', 'exang', 'oldpeak',
37                  'slope', 'ca', 'thal']
38
39 user_input = []
40 for feature in feature_names:
41     value = float(input(f"{feature}: "))
42     user_input.append(value)
43
44 # 8. Convert input to DataFrame with feature names
45 input_df = pd.DataFrame([user_input], columns=feature_names)
46
47 # 9. Scale the input
48 user_input_scaled = scaler.transform(input_df)
49
50 # 10. Make prediction
51 prediction = knn_model.predict(user_input_scaled)
52
53 # 11. Output result
54 print("\n🟡 Prediction Result:")
55 if prediction[0] == 1:
56     print("🔴 The model predicts that the patient **has heart disease**.")
57 else:
58     print("🟢 The model predicts that the patient **does NOT have heart disease**.")

```

PS C:\Users\palha\OneDrive\Desktop\Heart Disease Prediction using Machine Learning> python .\evaluate\_model\_with\_sample\_imput.py

Model Evaluation on Test Data:

	precision	recall	f1-score	support
0	0.87	0.93	0.90	29
1	0.93	0.88	0.90	32
accuracy			0.90	61
macro avg	0.90	0.90	0.90	61
weighted avg	0.90	0.90	0.90	61

🟡 Please enter the following patient details:


age: 52  
sex: 1  
cp: 2  
trestbps: 130  
chol: 250  
fbs: 1  
restecg: 1  
thalach: 165  
exang: 0  
oldpeak: 1.2  
slope: 1  
ca: 0  
thal: 2

🟡 Prediction Result:

🔴 The model predicts that the patient \*\*has heart disease\*\*.

PS C:\Users\palha\OneDrive\Desktop\Heart Disease Prediction using Machine Learning> |

## Application for Heart Disease Prediction using Stream lit



# Heart Disease Prediction App

Enter patient details below to predict the risk of heart disease.

Age

20 45 90

Sex

0

Chest Pain Type (cp)

0

Resting Blood Pressure (restbpps)

90 120 200

Serum Cholestoral (chol)

100 200 600

Fasting Blood Sugar > 120 mg/dl (fbs)

0

Resting ECG Results (restecg)

0

Max Heart Rate Achieved (thalach)

70 150 210

Exercise Induced Angina (exang)

0

ST depression Induced (oldpeak)

0.00 1.00 6.00

Slope of the peak exercise ST segment

0

Number of major vessels (ca)

0

Thal

0

### Prediction Result

The model predicts that the patient has heart disease.

## 6. Results & Discussion

Algorithms	Accuracy	Precision	Recall	F1-Score
Logistic regression	0.852459	0.870968	0.84375	0.857143
Decision Tree	0.754098	0.840000	0.65625	0.736842
Random Forest	0.836066	0.843750	0.84375	0.843750
SVM	0.868852	0.900000	0.84375	0.870968
KNN	0.901639	0.933333	0.87500	0.903226
XGBoost	0.819672	0.862069	0.78125	0.819672
Neural Networks	0.852459	0.870968	0.84375	0.857143

### *Testing and Validation*

The system was tested using multiple machine learning models and validated against a standard heart disease dataset. Models included Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), XGBoost, and Neural Network. Each model was evaluated using metrics such as accuracy, precision, recall, and F1-score. Learning curves and confusion matrices were used to assess model performance.

Test Cases –

- Dataset was tested using standard splits (e.g., 80-20) for training and testing to ensure fair model evaluation.
- Different classification models were applied and compared to assess performance differences.
- Feature scaling was applied where necessary (e.g., SVM, KNN).
- Evaluation included both raw metrics and visual plots (accuracy vs. epochs, learning curves).
- Neural Network was trained over multiple epochs to observe learning behaviour and convergence.

Key Observations Include –

- Logistic Regression and Neural Network showed similar, balanced performance across all metrics.
- KNN delivered the highest overall accuracy (90.16%), suggesting good predictive power with properly scaled data.
- SVM also performed well, benefiting from hyperparameter tuning and margin-based classification.
- XGBoost and Random Forest were strong in recall but slightly lower in accuracy than KNN.
- Decision Tree underperformed compared to ensemble models due to overfitting tendencies.
- Feature scaling significantly improved the performance of models like SVM and KNN.

### *Performance Analysis*

The system delivered consistent and accurate results across all tested models. Accuracy, precision, recall, and F1-score were computed and tabulated for comparison. Key metrics from the evaluation phase:

- KNN:

- Accuracy: 90.16%
- Precision: 93.33%
- Recall: 87.50%
- F1 Score: 90.32%
- SVM:
  - High performance with F1-score of 87.09%, indicating strong balance between precision and recall.
- Neural Network:
  - Stable training and validation accuracy over epochs; performed similarly to Logistic Regression.

The use of scaled features and model-specific preprocessing significantly improved prediction accuracy. Learning curves for Neural Network and XGBoost models revealed stable and reliable training processes with minimal overfitting.

## 7. Conclusion

This project successfully applied various machine learning algorithms to predict heart disease using a structured clinical dataset. After training and evaluating models like Logistic Regression, Decision Tree, Random Forest, SVM, KNN, XGBoost, and Neural Networks, K-Nearest Neighbors (KNN) emerged as the most accurate, with strong performance across all evaluation metrics. Comparative analysis highlights how different algorithms behave in medical data and emphasizes the importance of model selection based on accuracy, precision, and generalization. This study reinforces the role of machine learning in enhancing early diagnosis and decision-making in healthcare, paving the way for intelligent, data-driven clinical support systems.

## 8. References

1. Dataset: <https://www.kaggle.com/ronitf/heart-disease-uci>
2. Google Collaboratory Link for Project: <https://colab.research.google.com/drive/1H0icZq-q7xPD-gfmC0Qi4zG4NBIFrJfP?usp=sharing>