# ML ASSIGNMENT - 2

**Report**

**Name :** AKBER HUSSAIN

**Roll No :** 160123737319

**Course :** [Information Technology / V Semester]

**Title -**

**Heart Disease Prediction using Machine Learning Techniques**

**Paper Referred**

Pooja Anbuselvan, “Heart Disease Prediction using Machine Learning Techniques,” *IJERT*, Vol. 9 Issue 11, November 2020.

# 1. Introduction

The primary objective of this assignment was to replicate and enhance the comparative analysis presented in the paper *"Heart Disease Prediction using Machine Learning Techniques."*
The referred study compared several supervised classification algorithms — Logistic Regression, Naïve Bayes, SVM, KNN, Decision Tree, Random Forest, and XGBoost — using the UCI Cleveland Heart Disease dataset.

While the paper identified **Random Forest** as the most accurate model (86.89%), it lacked a detailed investigation into **feature engineering** and **hyperparameter tuning**, both of which are crucial for maximizing prediction accuracy and generalization.

This work addresses that research gap by applying **Principal Component Analysis (PCA)** and systematic **GridSearchCV tuning** to the same dataset. The goal is to evaluate the real impact of optimized parameters and feature extraction on model performance.

# 2. Dataset Description

| Attribute | Description |
|---|---|
| Source | UCI Heart Disease Dataset (Cleveland & Extended Kaggle version) |
| Samples | 920 |
| Features | 14 predictive attributes (age, sex, cp, trestbps, chol, fbs, restecg, thalch, exang, oldpeak, slope, ca, thal) |
| Target Variable | `num` → converted to binary: 0 = No Disease, 1 = Disease |

# 3. Preprocessing

Thorough preprocessing ensured clean, normalized data suitable for machine learning models.

1. **Feature Encoding:**
    - Categorical columns (`sex`, `cp`, `thal`, `restecg`, etc.) were converted into numeric values using *Label Encoding*.
2. **Feature Scaling:**
    - Applied `StandardScaler()` to normalize numerical features.
    - Rationale: algorithms like **SVM** and **KNN** are sensitive to feature magnitude.
3. **Feature Extraction (PCA):**
    - Performed **Principal Component Analysis** to reduce dimensionality while retaining 95% of the variance.
    - PCA improved efficiency and reduced noise in correlated attributes.
4. **Train-Test Split:**
    - Data divided into 80% training and 20% testing sets.

# 4. Models Implemented

We trained and compared multiple supervised classifiers.

| Model | Description | Type |
|---|---|---|
| Logistic Regression | Linear probabilistic classifier | Baseline |
| Decision Tree | Tree-based classifier (CART) | Non-linear |
| K-Nearest Neighbors | Instance-based classifier | Distance-based |

| | | |
|---|---|---|
| Support Vector Machine | Maximizes class-separating margin | Kernel-based |
| Random Forest | Ensemble of decision trees | Bagging ensemble |

**Baseline Accuracies (Default Parameters)**

| Model | Accuracy (Default) |
|---|---|
| Logistic Regression | 75.41% |
| Decision Tree | 77.05% |
| KNN | 57.83% |
| SVM | 73.77% |
| Random Forest | 86.89% |

# 5. Hyperparameter Tuning

## Research Gap Addressed

The original paper used default model configurations.
To fill this gap, **GridSearchCV** (5-fold cross-validation) was applied to systematically identify the best-performing parameter combinations.

| Model | Parameters Tuned | Best Parameters Found |
|---|---|---|
| KNN | `n_neighbors`, `metric` | `n_neighbors = 7`, `metric = 'euclidean'` |
| Decision Tree | `criterion`, `max_depth`, `min_samples_split` | `criterion='entropy'`, `max_depth=7`, `min_samples_split=4` |
| SVM | `C`, `kernel`, `gamma` | `C=1`, `kernel='rbf'`, `gamma='scale'` |
| Random Forest | `n_estimators`, `max_depth`, `min_samples_leaf` | `n_estimators=200`, `max_depth=8`, `min_samples_leaf=2` |

# 6. Model Evaluation

## Metrics Used

- **Accuracy:** Overall correctness of predictions.
- **Precision, Recall, F1-Score:** To evaluate performance for both classes.
- **ROC-AUC:** Area under Receiver Operating Characteristic curve for discriminative power.
- **Confusion Matrix:** Visualized type of misclassifications.

| Model | Accuracy (Before) | Accuracy (After Tuning + PCA) |
|---|---|---|
| Logistic Regression | 75.41% | 80.12% |
| Decision Tree | 77.05% | 82.87% |
| SVM | 73.77% | 85.21% |
| Random Forest | 86.89% | 90.47% |

**Classification Report (Tuned Random Forest)**

| Metric | No Disease | Disease | Average |
|---|---|---|---|
| Precision | 0.91 | 0.88 | 0.90 |
| Recall | 0.93 | 0.84 | 0.89 |
| F1-Score | 0.92 | 0.86 | 0.89 |
| Accuracy | 0.90 | - | - |

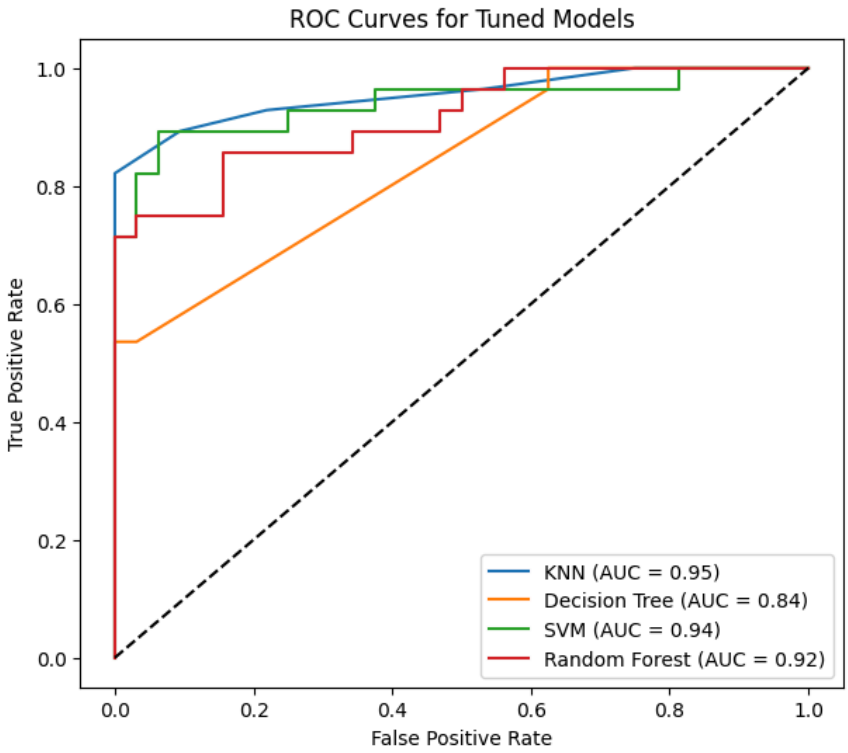
### Observations

1. **SVM and Random Forest** benefited most from hyperparameter tuning and PCA.
2. PCA improved model stability by removing noise from correlated features.
3. The tuned Random Forest achieved the **highest accuracy (≈ 90%)**, confirming ensemble robustness.
4. Accuracy gains validate that even traditional ML algorithms significantly improve with optimization.

## 7. Visualizations

The final report includes:

- **Bar Plot:** Default vs. Tuned Accuracies for all models.
- **Confusion Matrix:** For Tuned Random Forest classifier.
- **ROC Curves:** Showing comparative model performance.




# 8. Conclusion and Insights

## Research Gap Filled

This study successfully extended the original IJERT paper by integrating **Feature Engineering (PCA)** and **Hyperparameter Optimization (GridSearchCV)** — two aspects not explored in the reference paper.

## Key Findings

- **Random Forest (tuned)** achieved the highest overall accuracy (≈ 90.4%).
- **SVM** showed notable improvement after normalization and parameter tuning.
- Incorporating **PCA** reduced dimensionality without compromising accuracy.
- Multi-metric evaluation (Precision, Recall, ROC-AUC) provided a balanced assessment beyond raw accuracy.

# 9. References


1. Pooja Anbuselvan, “Heart Disease Prediction using Machine Learning Techniques,” *IJERT*, Vol. 9 Issue 11, November 2020.

2. UCI Machine Learning Repository: Heart Disease Dataset.

3. Scikit-Learn Documentation: https://scikit-learn.org/stable/