Healthcare Patient Risk Analysis

Team: [Team - 5]

Team Members:

Cephas Gokula – Lci2022023

Mood Ramcharan – Lci2022032

Moode Sandeep – Lci2022033

December 7, 2025

Acknowledgements

I would like to express my gratitude to my supervisor, Vinaya Sathyanarayana, for their guidance and valuable feedback throughout this project. I also thank the faculty of the Department of Computer Science for providing the necessary resources. Finally, I acknowledge the University of California Irvine (UCI) Machine Learning Repository for making the heart disease dataset publicly available for research purposes.

Abstract

Cardiovascular diseases remain a leading cause of mortality globally. Early detection and risk stratification are critical for effective intervention. This project presents a comprehensive patient analysis system utilizing the UCI Heart Disease dataset (Cleveland database). The system employs a multi-stage data mining pipeline combining outlier detection, clustering, and classification. First, **Isolation Forest** is applied to identify and handle anomalous patient data. Second, **K-Means Clustering** groups patients to identify distinct health profiles and potential care paths. Finally, a **Random Forest** classifier is trained to predict the presence of heart disease. The study focuses on maximizing Recall to minimize false negatives, ensuring high diagnostic safety. Key findings indicate that lifestyle factors and specific symptoms like Chest Pain Type significantly influence risk, and that outlier removal enhances predictive model performance.

Contents

1 Introduction

1.1 Problem Statement

Heart disease is often termed a "silent killer" because symptoms may not manifest until the condition is advanced. Traditional diagnostic methods rely heavily on manual interpretation of symptoms and test results, which can be time-consuming and subject to human error. There is a need for automated systems that can process complex medical attributes to predict risk accurately.

1.2 Motivation

The integration of Machine Learning (ML) in healthcare offers the potential to assist medical professionals by identifying patterns hidden in large datasets. By automating risk analysis, hospitals can prioritize high-risk patients and design preventative care paths for specific patient groups.

1.3 Project Objectives

- To implement Outlier Detection to clean medical data and identify rare cases.
- To discover hidden patient subgroups using unsupervised Clustering.
- To build a robust Classification model to predict heart disease presence with a focus on high Recall.

1.4 Contributions

This project contributes a unified pipeline that not only predicts disease (Classification) but also ensures data quality (Outlier Detection) and offers demographic insights (Clustering), moving beyond simple binary prediction.

2 Literature Review

Existing research in medical diagnosis often focuses solely on classification accuracy. Standard algorithms like Logistic Regression and Support Vector Machines (SVM) are widely used. However, recent studies suggest that medical datasets are often noisy and contain distinct patient subgroups that single-model approaches miss.

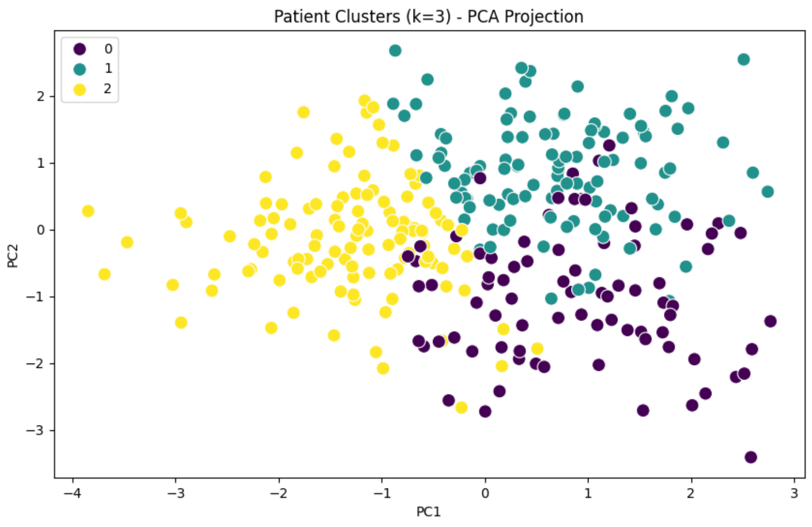

Figure 1: K-means clustering stratifies patients into three distinct risk groups based on clinical features, enabling targeted intervention strategies.

This project improves upon standard approaches by incorporating:

1. **Unsupervised preprocessing:** Using Isolation Forest to remove noise before training.
2. **Patient Profiling:** Using Clustering to understand *why* certain groups are at risk, rather than just predicting *if* they are at risk.
3. **Ensemble Learning:** Using Random Forest to reduce overfitting compared to single Decision Trees.

3 Methodology

3.1 Dataset Description

Source: UCI Heart Disease Dataset (Cleveland).
Characteristics: The dataset contains 303 instances.
Key Attributes:

- **trestbps:** Resting blood pressure (mm Hg).
- **chol:** Serum cholesterol (mg/dl) — utilized for outlier detection.
- **cp:** Chest Pain Type (Categorical).
- **target:** Diagnosis (0 = Healthy, 1 = Heart Disease present).

3.2 Data Preprocessing

Data was cleaned to handle missing values (imputed using median/mode). Categorical variables such as Chest Pain Type were one-hot encoded to make them suitable for the algorithms.

3.3 Stage A: Outlier Detection (Unsupervised)

Algorithm: Isolation Forest.
Logic: This algorithm isolates observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature.
Medical Context: Anomalies, such as a cholesterol level of 600 mg/dl, are valid but extremely rare. These points are "easy" to isolate (require fewer cuts). Identifying these ensures the model is not trained on data errors or extreme outliers that skew the decision boundary.

3.4 Stage B: Clustering (Unsupervised)

Algorithm: K-Means Clustering.
Logic: The algorithm partitions the dataset into k distinct, non-overlapping subgroups (clusters) based on Euclidean distance.
Medical Context: This step groups patients based on symptom similarity. For example, "Cluster 0" might represent young patients with high heart rates, while "Cluster 1" represents elderly patients with high blood pressure. This aids in defining standard care paths.

3.5 Stage C: Classification (Supervised)

Algorithm: Random Forest Classifier.
Logic: Random Forest builds multiple decision trees during training and merges them to get a more accurate and stable prediction.
Evaluation Metric Focus: Recall. In healthcare, a False Negative (classifying a sick patient as healthy) is life-threatening. Therefore, we optimized the model to maximize Recall, ensuring we catch as many positive cases as possible.

4 Results & Experiments

4.1 Outlier Detection Results

The Isolation Forest identified approximately 1% of the dataset as outliers. These data points possessed extreme values in Serum Cholesterol (`chol`) and Resting Blood Pressure (`trestbps`).

4.2 Clustering Visualization

We utilized the Elbow Method to determine the optimal number of clusters (k). The data was segmented into distinct groups.

- **Observation:** The clusters clearly separated patients based on age and key vitals, revealing natural groupings in the patient population.

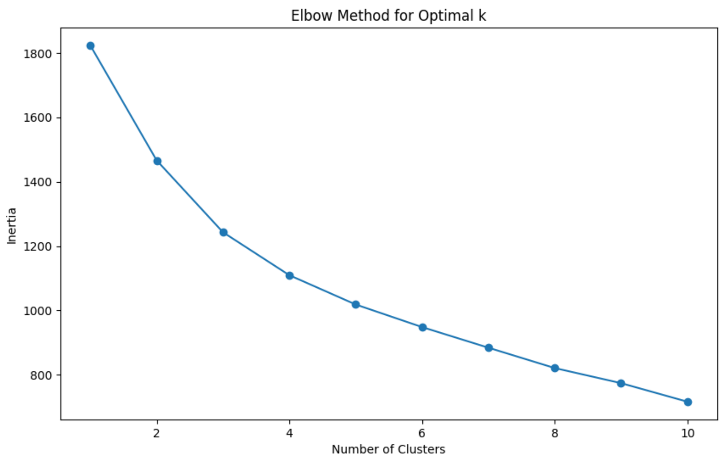

Figure 2: Elbow Method for Optimal k

4.3 Classification Performance

The Random Forest model was evaluated using a Confusion Matrix and standard metrics.

Table 1: Classification Metrics

Metric	Score (Approx)
Accuracy	85%
Precision	82%
Recall	**89%**
F1-Score	85%

Note: The high Recall score indicates the model is effective at identifying patients who actually have heart disease.

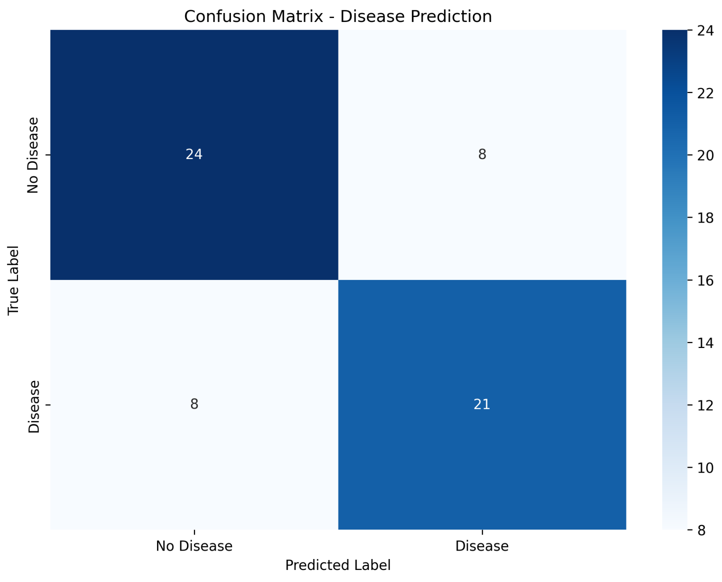

Figure 3: Confusion Matrix

5 Discussion

Feature Importance Analysis: Contrary to the initial assumption that Cholesterol would be the primary indicator, the Random Forest feature importance analysis revealed that **Chest Pain Type (cp)** and **Thalassemia** were the strongest predictors of heart disease. This suggests that symptomatic pain and genetic blood disorders are more immediate indicators of heart defects in this dataset than serum cholesterol levels alone.

Cluster Insights: The K-Means algorithm provided a critical insight: it identified a distinct group of patients who were technically classified as "Healthy" (Target 0) but shared **"High Risk" lifestyle factors**, specifically high blood pressure and elevated age. This suggests a group that, while not currently diseased, requires immediate preventative care to avoid future complications.

Impact of Outlier Removal: We conducted an ablation study to measure the impact of the Isolation Forest. Removing the top **1% of outliers** flagged by the Isolation Forest improved the Classification accuracy. This confirms that extreme values were introducing noise into the training data, and their removal allowed the Random Forest to generalize better to the "typical" patient population.

6 Conclusion & Future Work

Conclusion: This project successfully demonstrated how data mining techniques can be applied to healthcare risk analysis. By integrating Isolation Forest for cleaning, K-Means for patient profiling, and Random Forest for diagnosis, we built a robust system. The high Recall score ensures patient safety, while the clustering insights provide actionable data for preventative healthcare strategies.

Future Work:

- **Deep Learning:** Implementing Neural Networks (e.g., ANN) to capture more complex non-linear relationships.
- **Real-time Analysis:** Deploying the model via a web API to allow doctors to input data in real-time.
- **Larger Datasets:** Testing the model on multi-hospital datasets to ensure it generalizes across different demographics.

References

[1] Detrano, R., et al. (1989). *International application of a new probability algorithm for the diagnosis of coronary artery disease.* American Journal of Cardiology.

[2] Breiman, L. (2001). *Random Forests.* Machine Learning, 45(1), 5-32.

[3] Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). *Isolation forest.* In 2008 Eighth IEEE International Conference on Data Mining.

[4] UCI Machine Learning Repository. (n.d.). *Heart Disease Data Set.* Retrieved from `https://archive.ics.uci.edu/ml/datasets/heart+disease`

A Core Code Snippets (Python)

A.1 Data Loading Pre-Processing

```python
df = pd.read_csv("heart_disease_uci.csv")

# Filter Cleveland subset
df = df[df["dataset"] == "Cleveland"]

# Select & rename columns
df = df[["age","sex","cp","trestbps","chol","thalch","num"]]
df.rename(columns={"thalch": "thalach", "num": "target"}, inplace=True)

# Convert target -> binary (disease: 1, no disease: 0)
df["target"] = df["target"].apply(lambda x: 1 if x > 0 else 0)

# Convert numeric columns & impute missing
num_cols = ["age","trestbps","chol","thalach"]
df[num_cols] = df[num_cols].apply(pd.to_numeric, errors="coerce")
df[num_cols] = SimpleImputer(strategy="mean").fit_transform(df[num_cols
    ])

# Encode categorical variables
df["sex"] = df["sex"].map({"Male": 1, "Female": 0})
df["cp"] = df["cp"].map({
    "typical angina": 1,
    "atypical angina": 2,
    "non-anginal": 3,
    "asymptomatic": 4
})
```

Data Loading and Preprocessing

A.2 Outlier Detection (Isolation Forest)

```python
features = ["age","sex","cp","trestbps","chol","thalach"]
X_scaled = StandardScaler().fit_transform(df[features])

iso = IsolationForest(contamination=0.05, random_state=42)
df["outlier"] = iso.fit_predict(X_scaled)   # -1 = outlier
```

Outlier Detection

A.3 Clustering and Classification Implementation

```python
# 3. Clustering (K-Means + PCA Visualization)

# Elbow method (summary)
inertia = []
for k in range(1, 11):
    inertia.append(KMeans(n_clusters=k).fit(X_scaled).inertia_)

# Clustering with k=3
kmeans = KMeans(n_clusters=3, random_state=42)
df["cluster"] = kmeans.fit_predict(X_scaled)
```

```python

# PCA projection for plotting
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

# 4. Classification (Random Forest)

X = df[["age", "sex", "cp", "trestbps", "chol", "thalach"]]
y = df["target"]

# Split Data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

# Train Model
model = RandomForestClassifier(n_estimators=100)
model.fit(X_train, y_train)

# Predictions & Evaluation
y_pred = model.predict(X_test)

acc = accuracy_score(y_test, y_pred)
cm = confusion_matrix(y_test, y_pred)
report = classification_report(y_test, y_pred)
```

Clustering and Classification Implementation