FINAL PROJECT REPORT

Healthcare Predictive Analytics — Heart Disease Risk Prediction

1. Executive Summary

This project focuses on developing a complete healthcare predictive analytics system designed to assess

and predict a patient’s likelihood of having heart disease. Using a dataset of 3,069 patients, we performed

thorough data exploration, preprocessing, modeling, optimization, and deployment.

The early steps included examining the dataset structure, identifying feature distributions, handling

missing values, transforming categorical variables, and analyzing the relationships between medical

features and heart disease outcomes. We implemented several machine learning models, including SVM,

Random Forest, XGBoost, and MLP neural networks. Each model was trained, evaluated, and optimized to

ensure the best possible predictive accuracy.

Our final system incorporates MLOps best practices, including MLflow tracking and experiment logging.

A Flask API and Streamlit application were developed to make the prediction model accessible for real-time

patient evaluation, providing value to healthcare professionals for early diagnosis and informed clinical

decision-making.

2. Introduction

Heart disease is a leading global cause of death, and the ability to predict its onset can significantly

improve patient care. Predictive analytics allows healthcare systems to extract value from patient data

and identify patterns associated with disease progression.

The goal of this project is to develop a machine learning system that predicts whether a patient is likely to

have heart disease based on clinical and demographic characteristics. This includes building a data

pipeline, analyzing health metrics, evaluating machine learning algorithms, optimizing model performance,

and deploying a real-time predictive application.

The project follows a structured pipeline similar to professional data science workflows:

- **Data collection**

- **Data preprocessing**

- **Exploratory data analysis (EDA)**

- **Model development and optimization**

- **Deployment and monitoring**

By the end of the project, we successfully deployed an optimized heart disease prediction model and built a

user-friendly interface for real-world usage.

3. Dataset Description

The dataset contains 3,069 patient entries, each described by multiple clinical attributes. These include:

• Demographics: age, sex

• Vital signs: resting blood pressure, cholesterol, maximum heart rate (thalach)

• Medical test results: ECG readings (restecg), ST depression (oldpeak), slope of ST segment

• Diagnostic features: number of major vessels colored by fluoroscopy (ca), thalassemia indicators (thal)

• Lifestyle and comorbidities: smoking status, diabetes status, BMI

• Target variable: heart_disease (1 = disease, 0 = no disease)

The dataset was consistent with no missing values, and no duplicate rows were found. Each feature carried

medical significance, allowing the models to learn non-linear relationships commonly present in medical

diagnosis.

During initial exploration:

• Age distribution was skewed toward middle-aged patients.

• Cholesterol and blood pressure displayed wide variability.

• Oldpeak, chest pain type, and number of vessels (ca) showed strong relationships with disease presence.

4. Data Preprocessing

Several essential preprocessing steps were completed to clean and prepare the data for modeling:

1. Encoding Categorical Features:

- Sex was mapped to numerical values (Male=1, Female=0).

- Chest pain type and ST slope categories were numerically encoded.

- Thalassemia and exercise-induced angina were converted into binary or ordinal values.

2. Scaling Numerical Features:

StandardScaler was used to normalize continuous variables such as:

- Resting blood pressure

- Cholesterol

- Oldpeak

- Maximum heart rate (thalach)

This ensures all models using distance-based calculations perform correctly.

3. Handling Imbalanced Data:

The dataset had more non-disease patients than disease patients. To correct this,

we used SMOTE to synthetically create samples of the minority class and balance the dataset.

4. Dimensionality Reduction Using PCA:

To reduce noise and improve model performance, PCA was applied to keep 95% of feature variance.

This step reduced the model complexity and improved training efficiency.

These preprocessing steps ensured a clean, consistent dataset ready for accurate predictive modeling.

5. Exploratory Data Analysis

Extensive EDA was performed using histograms, scatterplots, boxplots, countplots, violin plots, heatmaps,

and pairplots. The purpose was to identify meaningful patterns and relationships between variables.

Key insights from EDA:

• Chest Pain: Asymptomatic chest pain (type 3) had the strongest association with heart disease.

• Oldpeak: Higher ST depression values indicated elevated risk.

• Slope: Flat ST slopes (slope=2) corresponded to higher disease occurrences.

• Colored Vessels: Patients with two or more colored vessels had a significantly higher chance of disease.

• Thalach: Lower maximum heart rate was linked to disease.

• Gender: Both men and women showed disease cases, though males were slightly more represented.

Additionally, the correlation heatmap showed that most features had weak linear correlations with the target, suggesting that non-linear models might perform better for prediction.

6. Model Development

We implemented multiple machine learning models to compare performance:

1. Support Vector Machine (SVM):

- RBF kernel used due to non-linear relationships in medical data.
- Sensitive to scaling, so StandardScaler was applied.

2. XGBoost Classifier:

- Handles complex relationships well.
- Capable of capturing non-linear patterns.
- Strong baseline performance.

3. Random Forest Classifier:

- Multiple decision trees combined for robust predictions.
- Good for handling noisy data.

4. MLP Neural Network:

- Two hidden layers (100, 50 neurons).
- Uses the Adam optimizer and ReLU activation.

All models used training data balanced with SMOTE and transformed PCA components.

Train-test split of 80/20 ensured fair evaluation.

7. Model Optimization & Selection

Hyperparameter tuning was done using GridSearchCV with 5-fold cross-validation.

Each model's parameters were optimized to improve predictive performance.

The SVM model emerged as the best final model after optimization, achieving:

- High accuracy (~72%)
- Strong recall for both classes
- Balanced F1-score
- ROC-AUC close to 0.79

The confusion matrix showed effective detection of positive cases, which is crucial in medical prediction where false negatives can be dangerous.

Therefore, the SVM RBF optimized classifier was selected as the final deployment model.

8. MLOps, Deployment & Monitoring

We adopted MLOps practices to ensure model reliability, versioning, and monitoring.

1. MLflow Tracking:

- Logged parameters, metrics, and artifacts.
- Saved the final optimized model.
- Ensured reproducibility of experiments.

2. Flask API Deployment:

- Created a /predict endpoint.
- Accepts patient features and returns prediction + probability.
- Supports real-time integration into hospital systems.

3. Streamlit Web Application:

- Designed an interactive interface for clinicians.
- Collects patient inputs and displays predictions clearly.
- User-friendly and accessible.

4. Monitoring:

- All predictions logged automatically in CSV.
- Drift detection enabled by tracking prediction patterns over time.
- Useful for scheduling retraining and ensuring reliability.

9. Challenges & Limitations

Challenges:

- Dataset imbalance — required SMOTE for correction.
- Weak predictive features — cholesterol and resting blood pressure had poor predictive value.
- PCA reduces interpretability — original feature contributions become abstract.

Limitations:

- Dataset is not time-series.
- No imaging, ECG waveform, or multi-modal data.
- Predictions cannot replace clinical diagnosis; only assist clinicians.

10. Recommendations & Future Work

Future improvements can strengthen the system:

1. Collect more diverse datasets:

- Include hospitals from multiple regions.
- Add time-series vitals such as heart rate trends.

2. Enhance model sophistication:

- Explore LightGBM, CatBoost, or deep learning models.
- Use Bayesian optimization for hyperparameter tuning.

3. Improve deployment:

- Use Docker for containerization.
- Deploy on cloud platforms for scalability.
- Add automatic retraining pipelines using Airflow.

4. Add clinician-focused dashboard:

- Visualize prediction history.
- Add explanations using SHAP for model interpretability.

11. Conclusion

This project successfully implemented an end-to-end predictive analytics pipeline for heart disease risk.

The optimized SVM model performed reliably, and the system was deployed using modern MLOps and

web technologies.

With additional data and enhancements, this predictive system can become a powerful clinical decision

support tool to improve early detection and patient care.