**CHAPTER THREE**

**SYSTEM ANALYSIS AND DESIGN**

**3.1 SYSTEM ANALYSIS**

In our proposed method, heart disease can be detected more efficiently and less costly within a short time. This paper worked with preprocessed data to train and test using machine learning algorithms. In the first stage, preprocessed data are divided into two parts. Most of those are used in the training phase (80%), and the rest (20%) are used in the testing phase. In the training and testing phase, the proposed system has trained our dataset using machine learning algorithms like Support vector machine. Using the Python platform, we have trained and finally predicted the result of a patient. As shown in Fig.1, our trained system will indicate the presence of cardiovascular disease in a patient. The primary intention of this raised system is to detect heart disease more efficiently and accurately.

**3.2 PREPROCESSING**

Firstly, we have preprocessed the collected dataset to reduce complexity and enhance user accessibility. There are many datasets in Kaggle, UCI, Cleveland, etc. In those entire data centers, more than 76 attributes are presented. We have selected the Kaggle dataset, where 13 parameters are assigned, and those factors are more responsible for heart disease. The selected Kaggle dataset contains almost equally condition and non-disease patient data, whereas the UCI dataset is not similarly divided for disease and non-disease patients. That diminishes the number of inputs to the network and helps it learn more accurately and efficiently. The most appropriate top 13 attributes are age, sex, chest pain, blood pressure, serum cholesterol, fasting blood sugar, resting ECG, thalassemia, max heart rate achieved, ST depression induced by exercise relative to rest, significant vessels and using those factors the system predicts the heart condition. The following Table1 represents in detail our input attributes.

Medical Dataset

Test Data

Train Data

User Input

Data Splitting

Data Processing

Heart Disease Prediction System

Classification

Performance Evaluation

Heart Risk: No

Heart Risk: Yes

**Fig. 1** Proposed Heart Prediction System

Thalach

The max heart rate achieved by an individual.

Exercise induced from angina.

Binary value

Exang

Binary value

Old peak

ST depression induced by exercise that represents the state of rest.

Floating

Value

Exercise induced angina: 1 = yes 0 = no

Exercise induced angina: 1 = yes and 0 = no

Any continuous value

Slope

Slope value during exercise that is

measured from ST segment.

Nominal

Value

Ca

Thal

Number of major vessels 0 to 3 that is colored by fluoroscopy.

Represent the heart rate of patient in three distinct values.

Numerical value

Nominal

Value

Peak exercise ST segment: 1 = up sloping; 2 = flat

3 = down sloping

Number of major vessels from 0 to 3.

For normal=3, fixed defect=6,

reversible defect=7

Result

Predicted outcome from system.

Binary value Absence of heart disease =0, Present of heart

disease=1

**Table 1** Representation of input attributes.

**3.3 SPLITTING**

The accurate classification result of the dataset depends on the training and testing phase. To get a better result, we divided our whole dataset into two parts: the majority percent of the dataset (80%) for training, and the rest of those are for testing (20%).

**3.4 CLASSIFICATION MODELS**

The training data uses machine-learning algorithm, i.e. SVM. The algorithm is explained in detail below.

**3.4.1 SUPPORT VECTOR MACHINE**

Support Vector Machines can manipulate multiple continuous and categorical variables. SVM constructs a hyper plane in multidimensional space to separate diverse classes. SVM iteratively generates an optimal hyper plane, used to minimize an error. The main idea of SVM is to find a maximum marginal hyper plane (MMH) that best divides the dataset into classes.

f(X) = wT + b

Where w is a dimensional co-efficient vector and b is an offset. A subsequent optimization problem can solve that.

**3.5 SYSTEM IMPLEMENTATION**

This process involves realizing a usable system using available software technologies and hardware tools to meet the requirement specification. From the analysis and design stage requirement, specification tools were selected to implement the system. These tools are classified as follows:

**3.5.1 PROGRAMMING LANGUAGES**

1. CSS 3
2. PYTHON
3. FLASK (WEB FRAMEWORK)

**3.5.2 SOFTWARE TOOLS**

1. 64 bit windows operating system
2. Edge browser

**3.5.3 DEVELOPMENT TOOLS & DATABASE**

1. Visual Studio Code
2. My Sqlite3

**3.5.4 OTHER LIBRARIES**

1. Csv Files