JAGRAN LAKECITY UNIVERSITY

igniting minds; changing lives

Heart Disease Prediction

Report

25/3/2023

Submitted to:
Mr.Shivendra Dubey

Submitted by:
Abhishek Shrivas

Table of Contents

1. Introduction

Millions of individuals around the world are afflicted by heart disease, a serious public health issue. For lowering the risk of complications and improving patient outcomes, early detection and prevention of heart disease are crucial. Based on a variety of risk indicators, machine learning (ML) algorithms have become an effective tool for heart disease prediction.

The most recent research on heart disease risk factors will be incorporated into the heart disease prediction system, which will be created utilising cutting-edge ML algorithms. The system will be made to be scalable and flexible enough to be used in hospitals, clinics, and private practises, among other types of healthcare facilities.

1.1 Purpose

The goal of this project is to develop a software application that, given a person's symptoms and medical history, can determine if they have heart disease or not. To effectively identify and forecast the presence of cardiac disease, the software will use a variety of machine learning modelling techniques.

The project will use different techniques of Machine Learning Models to analyze the input data and predict the likelihood of heart disease. These models will be trained on a large dataset of medical records, which will be used to identify patterns and correlations between different symptoms and risk factors.

Overall, the purpose of the Heart Disease Prediction project is to improve the accuracy and efficiency of heart disease diagnosis and prevention, ultimately leading to better patient outcomes and a healthier population.

1.2 Scope

The goal of this project is to create a user-friendly software programme that medical practitioners can utilise to assist patients in being diagnosed with heart disease. The software will evaluate the patient's signs, symptoms, and past health information to make a prognosis regarding the possibility of heart disease. To guarantee the precision of the forecasts, a variety of machine learning techniques will be used in the development of the programme.

The scope of the project is to develop a machine learning model that can accurately predict if a person is at risk of having heart disease based on their symptoms. The model will be trained using a dataset that contains information about patients who have been diagnosed with heart disease as well as patients who have not been diagnosed with heart disease. The model will be designed to take into account a wide range of factors, such as age, gender, blood pressure, cholesterol levels, and smoking status, among others.

1.3 Definitions, Acronyms, and Abbreviations

Definitions:

- Dataset: a collection of data that can be analyzed and used for research purposes

- Keywords: significant words or phrases that are used to indicate the main topics or themes in a conversation or text
- Links: clickable addresses or URLs that lead to other web pages or resources
- Charts: graphical representations of data that help to visualize trends or patterns
- Bar graphs: charts that use rectangular bars to represent data values

Acronyms and Abbreviations:

- SRS: Software Requirements Specification
- ML: Machine Learning
- GUI: Graphical User Interface

1.4 References

- Pandas Library. (2021). Retrieved from https://pandas.pydata.org/
- Python Software Foundation. (2021). Retrieved from https://www.python.org/
- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications (1998). IEEE Computer Society.
- Matplotlib Library. Retrieved from https://matplotlib.org/
- Sklearn Library . Retrieved from https://scikit-learn.org/stable/
- Cleveland Clinic Heart and Vascular Institute. (2021). Heart Disease. Retrieved from https://my.clevelandclinic.org/health/diseases/16814-heart-disease
- UCI Machine Learning Repository. (n.d.). Heart Disease Data Set. Retrieved from https://archive.ics.uci.edu/ml/datasets/heart+disease

1.5 Overview

In order to determine if a patient has heart disease or not, the programme will analyse their symptoms, including chest pain, blood pressure, and cholesterol levels. Medical professionals can enter patient data into the application and get a prognosis using its user-friendly GUI..

2. General Description

The given dataset contains various attributes related to individuals and their health parameters, which are used to predict the likelihood of heart disease.

The attributes include age, sex, chest pain type (cp), resting blood pressure (trestbps), cholesterol level (chol), fasting blood sugar (fbs), resting electrocardiogram results (restecg), maximum heart rate achieved during exercise (thalach), exercise induced angina (exang), ST depression induced by exercise relative to rest (oldpeak), slope of the peak exercise ST segment (slope), number of major vessels colored by fluoroscopy (ca), and thallium heart scan results (thal).

The target attribute indicates whether or not an individual is diagnosed with heart disease, with a value of 0 representing no heart disease and a value of 1 representing heart disease. These attributes are used as inputs to various machine learning models to predict the likelihood of heart disease in an individual based on their health parameters.

2.1 Product Perspective

The Heart Disease Prediction software will be an independent software application that can be used by medical professionals to assist in the diagnosis of heart disease. It will not be integrated with any other systems.

he product perspective involves the use of machine learning models to detect the presence of heart disease in patients based on their symptoms. The product will be designed for healthcare professionals, such as doctors and nurses, who will use it to aid in the diagnosis of heart disease.

2.2 Product Functions

The following tasks will be carried out by the programme:

- Let medical staff to enter patient symptoms and medical background
- To forecast the likelihood of cardiac disease, use machine learning algorithms.
- Provide a user-friendly Interface for simple data input and navigation
- Provide the medical practitioner with a forecast of the likelihood of heart disease.

1.Input of patient data: The system should allow the user to input relevant patient data such as age, sex, blood pressure, cholesterol levels, and other symptoms that could indicate a risk for heart disease.

2. Data preprocessing: The system should preprocess the input data by cleaning, normalizing, and scaling the data to ensure it is in a suitable format for analysis.

3. Heart disease prediction: The system should use various machine learning models to analyze the preprocessed data and predict whether the patient is at risk for heart disease or not.

4. Result visualization: The system should provide clear and concise visualization of the prediction results to the user. This could be in the form of a report, graph or other interactive visual representations.

2.3 User Characteristics

Medical professionals that are trained and knowledgeable in the diagnosis and treatment of heart disease, such as doctors and nurses, are the software's intended users.
User characteristics refer to the specific attributes of the users who will interact with the heart disease prediction system. Some potential user characteristics for this system could include:

1.**Patients**: Individuals who will provide their symptoms as input to the system and receive a prediction of their likelihood of having heart disease.

2.**Healthcare Professionals**: Medical professionals who will use the system to support their clinical decision-making and diagnosis of heart disease.

3.**Researchers**: Scientists who may use the system to conduct further research on heart disease and related medical conditions.

4.**Developers**: Software engineers or data scientists who will develop and maintain the heart disease prediction system.

5.**Technical Support Staff**: Personnel who will be responsible for providing technical support to users of the system.

It is important to consider the specific needs and requirements of each user group when designing and developing the heart disease prediction system, to ensure that it is user-friendly, effective, and meets the needs of its intended users.

2.4 General Constraints

The programme must be created within realistic time and financial constraints. The quality and quantity of the submitted data will determine how accurate the predictions are.
These constraints can include technical, operational, or environmental limitations that impact the system's design or implementation. Some examples of general constraints that may apply to this project include:

1.Technical constraints: The heart disease prediction system must be developed using specific programming languages, databases, or software frameworks. Technical constraints may also include limitations related to hardware or software compatibility, such as operating system requirements or memory constraints.

2.Operational constraints: The heart disease prediction system must be designed to meet the needs of specific users or stakeholders, such as medical professionals or patients. This may include requirements related to user interface design, accessibility, or ease of use.

3.Environmental constraints: The heart disease prediction system must be developed with specific environmental considerations in mind, such as security or privacy requirements. For example, the system may need to comply with specific data privacy regulations or security protocols to protect patient data.

4.Time constraints: The heart disease prediction system must be developed within a specific timeframe, which may impact the scope or complexity of the project. This may also include requirements related to project milestones, such as the completion of specific features or testing phases.

5.Budget constraints: The heart disease prediction system must be developed within a specific budget, which may impact the choice of development tools or outsourcing decisions. This may also include requirements related to cost savings or ROI (Return on Investment) projections.

It is important to consider all general constraints when developing the heart disease prediction system to ensure that it meets all requirements and limitations set forth by the stakeholders and to ensure the system's overall success.

2.5 Assumptions and Dependencies

This subsection of the SRS should list each of the factors that affect the requirements stated in the SRS. These factors are not design constraints on the software but are, rather, any changes to them that can affect the requirements in the SRS. For example, an assumption might be that a specific operating system will be available on the hardware designated for the software product. If, in fact, the operating system is not available, the SRS would then have to change accordingly.

3. Specific Requirements

Specific requirements are detailed and precise statements of what the system should do. These requirements provide a clear understanding of the system's functionality and performance expectations.

3.1 External Interface Requirements

3.1.1 User Interfaces

The software will have a user-friendly GUI that can be used by medical professionals to input patient symptoms and medical history. The GUI will include text fields, dropdown menus, and checkboxes for easy data input.

3.1.2 Hardware Interfaces

The software will require a computer with sufficient processing power and memory to run the machine learning algorithms and store patient data.

3.1.3 Software Interfaces

The software will utilize various machine learning libraries and frameworks, such as scikit-learn and TensorFlow, to implement the machine learning models.

3.1.4 Communications Interfaces

The software will not require any external communications interfaces.

3.2 Functional Requirements

The software must accurately predict the likelihood of heart disease based on the patient's symptoms and medical history. The prediction must be displayed to the medical professional..

3.2.1 <Functional Requirement or Feature #1>

Functional requirements are specific features that the software must provide to fulfill its purpose. For the heart disease prediction project:

- The ability for users to input a patient's symptoms and other relevant information.
- The use of machine learning models to analyze the input data and provide a prediction of whether or not the patient has heart disease.
- The ability for users to view the prediction results, including the likelihood of heart disease and any relevant data used to make the prediction.
- The ability for users to update or modify the patient's information and rerun the prediction.

…

3.3 Use Cases

3.3.1 Use Case #1

Use cases describe specific scenarios or interactions that a user might have with the software:

- A doctor uses the software to input a patient's symptoms and receive a prediction of whether or not they have heart disease.
- A patient uses the software to input their own symptoms and receive a prediction of whether or not they have heart disease.
- A researcher uses the software to analyze data from multiple patients and identify potential patterns or correlations related to heart disease

3.4 Classes / Objects

3.4.1 <Class / Object #1>

Classes and objects refer to the various components and entities within the software system:

- Patient: stores information about individual patients, including their symptoms and other relevant data.
- Machine learning model: performs the analysis of the patient data to provide a prediction of heart disease.
- User interface: allows users to interact with the software and input or view information.

…

3.5 Non-Functional Requirements

- Performance: the software should be able to provide predictions quickly and efficiently, even with large amounts of data.
- Reliability: the software should provide accurate predictions and avoid errors or crashes.
- Security: the software should protect patient data and prevent unauthorized access.
- Maintainability: the software should be designed in a way that makes it easy to update or modify in the future.
- Portability: the software should be able to run on multiple platforms and be easily accessible to users.

3.6 Inverse Requirements

Inverse requirements refer to things that the software should not do or should avoid doing:

- The software should not provide medical advice or replace the expertise of a qualified medical professional.
- The software should not store or share patient data without explicit consent.
- The software should not rely on incomplete or inaccurate data to provide predictions.

3.7 Design Constraints

Design constraints refer to limitations or requirements that must be considered during the development of the software. :

- Compliance with relevant laws and regulations related to healthcare data and patient privacy.
- Use of specific programming languages or technologies.
- Integration with other existing software or systems.

.

3.8 Logical Database Requirements

Logical database requirements refer to how the software will store and access data.:

- Use of a database to store patient information and prediction results.
- Use of encryption and other security measures to protect patient data.
- Integration with other existing databases or data sources.

.

3.9 Other Requirements

Other requirements refer to any additional considerations or needs that should be taken into account during the development of the software:

- User training and education on how to use the software effectively

4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS's requirements.

4.1 Sequence Diagrams

4.3 Data Flow Diagrams (DFD)

4.2 State-Transition Diagrams (STD)

5. Change Management Process

Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.

Result

Here are the sample, we are first showing the top five head of the dataset & printing the info we can see in fig 1.1

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0

(fig 1.1)

Here we are showing the different columns mean, count , min, max, of each of the following attribute so that we get the overall idea what data we have to deal with, we can see in figure 1.2

```
<bound method DataFrame.info of       age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  \
0      52    1   0       125   212    0        1      168      0      1.0
1      53    1   0       140   203    1        0      155      1      3.1
2      70    1   0       145   174    0        1      125      1      2.6
3      61    1   0       148   203    0        1      161      0      0.0
4      62    0   0       138   294    1        1      106      0      1.9
...   ...  ...  ..       ...   ...  ...      ...      ...    ...      ...
1020   59    1   1       140   221    0        1      164      1      0.0
1021   60    1   0       125   258    0        0      141      1      2.8
1022   47    1   0       110   275    0        0      118      1      1.0
1023   50    0   0       110   254    0        0      159      0      0.0
1024   54    1   0       120   188    0        1      113      0      1.4

      slope  ca  thal  target
0         2   2     3       0
1         0   0     3       0
2         0   0     3       0
3         2   1     3       0
4         1   3     2       0
...     ...  ..   ...     ...
1020      2   0     2       1
1021      1   1     3       0
1022      1   1     2       0
1023      2   0     2       1
1024      1   1     3       0

[1025 rows x 14 columns]>
```

Fig(1.2)

Now we have print the values of how many people are having heart disease by representing it by 1 and normal people by 0, we can see in this figure 1.3

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.336585	1.071512
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000000	0.000000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000000	0.800000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000000	1.800000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000000	6.200000

```
1    526
0    499
Name: target, dtype: int64
```

Fig (1.3)

Now we have sperate the target from the other attributes we can see int his figure 1.4

```
      age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  \
0      52    1   0       125   212    0        1      168      0      1.0
1      53    1   0       140   203    1        0      155      1      3.1
2      70    1   0       145   174    0        1      125      1      2.6
3      61    1   0       148   203    0        1      161      0      0.0
4      62    0   0       138   294    1        1      106      0      1.9
...   ...  ...  ..       ...   ...  ...      ...      ...    ...      ...
1020   59    1   1       140   221    0        1      164      1      0.0
1021   60    1   0       125   258    0        0      141      1      2.8
1022   47    1   0       110   275    0        0      118      1      1.0
1023   50    0   0       110   254    0        0      159      0      0.0
1024   54    1   0       120   188    0        1      113      0      1.4

      slope  ca  thal
0         2   2     3
1         0   0     3
2         0   0     3
3         2   1     3
4         1   3     2
...     ...  ..   ...
1020      2   0     2
1021      1   1     3
1022      1   1     2
1023      2   0     2
1024      1   1     3

[1025 rows x 13 columns]
```

Figure (1.4)

Printing the total values that are present in the target attribute we can see in this figure 1.5

```
0       0
1       0
2       0
3       0
4       0
       ..
1020    1
1021    0
1022    0
1023    1
1024    0
Name: target, Length: 1025, dtype: int64
```

Figure (1.5)

And diving the values into training set and the test set in this figure 1.6

```
(1025, 13) (820, 13) (205, 13)
```

Figure (1.6)

Now we are have prepare our Logistic Regression model we can see in this figure 1.7

```
▾ LogisticRegression
LogisticRegression()
```

Figure(1.7)

Now we our calculating our accuracy of training dataset & test dataset in this figure 1.8

```
Accuracy on training data: 0.8524390243902439
```

```
Accuracy on test data: 0.8048780487804879
```

Figure (1.8)

After all these step now we give an input an data and it will give us the following output weather The person having a heart disease or not see int this figure 1.9

```
[0]
the person does not have a heart disease
```

Figure (1.9)

…