Contents

Requirement Title

Cardio vascular determination using Machine Learning algorithms.

Document History

Version	Author	Reason for change	Date
1.0	Ramani Nagarajan	Initial	26/10/2025

Contacts

Role	Name	Email	Tel no
AI – Business Analyst	Logeshwari Parthiban		
AI Solution Architect	Ramani Nagarajan		

About the requirement:

Business wants an AI solution for identifying the Cardiovascular Disease based on the datasets shared

About the dataset:

This dataset is related to the **cardiovascular disease.**

Such data is typically colle cted through **medical studies**, **healthcare systems**, or **public health surveys**. Specifically, for a dataset like this (the *Cardiovascular Disease Dataset*), the data usually comes from:

1. **Hospitals or Clinics** –
   Doctors and nurses record patient details (age, height, weight, blood pressure, cholesterol, etc.) during routine checkups or medical exams.
2. **Government Health Programs** –
   Public health agencies (like the Ministry of Health or WHO projects) collect anonymized data from clinics for research on chronic diseases such as heart disease or diabetes.
3. **Medical Research Studies** –
   Universities or research institutions conduct clinical studies where participants volunteer to provide their medical measurements and lifestyle information (like smoking, alcohol, activity).
4. **Health Insurance or Wellness Programs** –
   In some cases, health insurance or corporate wellness programs collect similar data to assess health risks (with user consent).

About the columns in the Datacollection:

There are around 13 columns in the data collection. Below are the details about every column.

Column	Meaning	Type / Unit	Details
id	Record ID	Integer	Unique identifier for each person/record.
age	Age in days	Integer	Person's age (e.g., 18393 days ≈ 50.4 years). To convert to years: age / 365.
gender	Gender	Categorical (1 or 2)	1 = Female, 2 = Male (in this dataset convention).
height	Height	Centimeters	Person's height in cm.
weight	Weight	Kilograms	Person's weight in kg.
ap_hi	Systolic blood pressure	mmHg	Higher value of blood pressure (normal around 120).
ap_lo	Diastolic blood pressure	mmHg	Lower value of blood pressure (normal around 80).
cholesterol	Cholesterol level	Ordinal (1–3)	1 = normal, 2 = above normal, 3 = well above normal.
gluc	Glucose level	Ordinal (1–3)	1 = normal, 2 = above normal, 3 = well above normal.
smoke	Smoking status	Binary (0 or 1)	1 = currently smokes, 0 = does not smoke.
alco	Alcohol intake	Binary (0 or 1)	1 = drinks alcohol, 0 = does not drink.
active	Physical activity	Binary (0 or 1)	1 = physically active, 0 = inactive.
cardio	Cardiovascular disease	Target variable (0 or 1)	1 = has cardiovascular disease, 0 = healthy.

Data processing

The categorical data needs to be converted to binary data before applying the AI solutions on it.

Below are the process:

Column "gender"

All rows against the column "gender" are filled with categorical data such as "Male" or "Female". This needs to be converted to binary format using python using below code:

```
dataset=pd.get_dummies(data,drop_first=True,dtype=int)
```

Column value with "NaN"

Columns such as smoke, alco,active,cardio are filled with NaN values. This is to be replaced with the value '0'

```
dataset["smoke"].fillna(0,inplace=True)
dataset["alco"].fillna(0,inplace=True)
dataset["active"].fillna(0,inplace=True)
dataset["cardio"].fillna(0,inplace=True)
```

Descriptive Summary Statistics

This code calculates and stores key descriptive statistical measures (such as mean, median, mode, quartiles, percentiles, and max) for each column, which is the essence of summary or descriptive statistics in data analysis.

Univariate

In univariate analysis, we describe, summarize, and find patterns in just one variable without considering relationships or interactions with other variables. Typical univariate techniques include calculating summary statistics like mean, median, mode, standard deviation, and creating plots such as histograms or box plots to understand the distribution of that one variable.

Following are calculated using the Univariate Analysis:

Mean, Median, Mode, Q1,Q2,Q3, IQR based on the Quantitative columns:

```
for columnName in quan:
  descriptive[columnName]["Mean"]=dataset[columnName].mean()
  descriptive[columnName]["Median"]=dataset[columnName].median()
  descriptive[columnName]["Mode"]=dataset[columnName].mode()[0]
  descriptive[columnName]["Q1:25%"]=dataset.describe()[columnName]["25%"]
  descriptive[columnName]["Q2:50%"]=dataset.describe()[columnName]["50%"]
  descriptive[columnName]["Q3:75%"]=dataset.describe()[columnName]["75%"]
  descriptive[columnName]["99%"]=np.percentile(dataset[columnName],99)
  descriptive[columnName]["Q4:100%"]=dataset.describe()[columnName]["max"]
  descriptive[columnName]["IQR"]= Q3 – Q1
  descriptive[columnName]["1.5rule"]=1.5*descriptive[columnName]["IQR"]
  descriptive[columnName]["Lesser"]=descriptive[columnName]["Q1:25%"]-
descriptive[columnName]["1.5rule"]
  descriptive[columnName]["Greater"]=descriptive[columnName]["Q3:75%"]+descriptive[
columnName]["1.5rule"]
  descriptive[columnName]["Min"]=dataset[columnName].min()
  descriptive[columnName]["Max"]=dataset[columnName].max()
```

Finding Best Model

Finding the best model is done using Advanced Feature Selection and Dimensionality Reduction. Below given models are used and calculation done accordingly.

The models such as:

1. **SelectKBest**: A feature selection method that selects the top k features based on univariate statistical tests, helping to pick the most relevant features for the model.
2. **Recursive Feature Elimination (RFE):** A wrapper-type feature selection method that recursively removes least important features based on a model's weights or importance scores to improve model performance and reduce overfitting.
3. **Principal Component Analysis (PCA)**: A dimensionality reduction technique that transforms correlated features into a smaller set of uncorrelated components, preserving most of the variance in the data.
4. **Logistic Regression:** A linear model used for binary classification problems that predicts the probability of the target class based on input features.
5. **Support Vector Machine (SVM):** A powerful supervised learning model used for classification and regression tasks by finding the optimal hyperplane that separates classes or fits the data.
6. **K-Nearest Neighbors (KNN):** A simple, instance-based learning algorithm that classifies a data point based on the majority class among its k nearest neighbors.
7. **Naive Bayes:** A probabilistic classifier based on Bayes' theorem assuming feature independence; often used in text classification and spam detection.
8. **Decision Tree:** A tree-structured model used for classification and regression that splits data based on feature values to make predictions.
9. **Random Forest:** An ensemble learning method that builds multiple decision trees (a forest) and combines their predictions to improve accuracy and reduce overfitting.

These are common machine learning methods and techniques for feature selection, dimensionality reduction, and classification or regression modeling in data science.

Model Selection:

The performance scores values obtained from different machine learning algorithms or feature selection/dimensionality reduction methods obtained. And the best model is selected based on the score value.

Unit Test Report
