A

Preliminary Project Report (Project-I)

On

Coronary Heart Disease Predictor(CHDP)

By

Jaydeep Kishor Sonawane(21517220181124510044)
Yadnesh Vijay Kalal(21517220181124510054)

Department of Computer Engineering

The Shirpur Education Society's

R. C. Patel Institute of Technology, Shirpur

Maharashtra State, India

2021-22

A
Preliminary Project Report(Project-I)
On
Coronary Heart Disease Predictor (CHDP)

In partial fulfillment of requirement for the degree of

Bachelor of Technology

in

Computer Engineering

Submitted By

Jaydeep Kishor Sonawane(21517220181124510044)
Yadnesh Vijay Kalal(21517220181124510054)

Under the Guidance of

Prof. P. D. Saraf

Department of Computer Engineering

The Shirpur Education Society's

R. C. Patel Institute of Technology, Shirpur

Maharashtra State, India

2021-22

Department of Computer Engineering

SES's R. C. Patel Institute of Technology, Shirpur

Maharashtra State, India

CERTIFICATE

This is to certify that the preliminary project (Project-I) entitled **"Coronary Heart Disease Predictor (CHDP)"** has been carried out by team:

Jaydeep Kishor Sonawane(21517220181124510044)

Yadnesh Vijay Kalal(21517220181124510054)

under the guidance of Prof. P. D. Saraf in partial fulfillment of the requirement for the degree of Bachelor of Engineering in Computer Engineering of Dr. Babasaheb Ambedkar Technological University, Lonere during the academic year 2021-22.

Date:

Place: Shirpur

Prof. P. D. Saraf
Guide

Prof. Dr. R. B. Wagh
Project Coordinator

Prof. Dr. Nitin N. Patil
Head

Prof. Dr. J. B. Patil
Principal

Acknowledgment

No volume of words is enough to express my gratitude towards my guide, (guide name), Associate Professor in Computer Engineering Department, who has been very concerned and have aided for all the material essential for the preparation of this work. He has helped me to explore this vast topic in an organized manner and provided me with all the ideas on how to work towards a research oriented venture.

we wish to express our sincere gratitude towards Project Coordinator Prof. Dr. R. B. Wagh for his timely suggestions and instructions.

we are also thankful to Prof. Dr. Nitin N. Patil, Head of Department, Computer Engineering, for the motivation and inspiration that triggered me for the project work.

we are thankful to Prof. Dr. J. B. Patil, Principal, R. C. P. I. T., Shirpur for the support and encouragement.

Jaydeep Kishor Sonawane

Yadnesh Vijay Kalal

ABSTRACT

Coronary Heart Disease Predictor (CHDP)

Cardiovascular diseases are the most common cause of death worldwide over the last few decades in the developed as well as underdeveloped and developing countries. Early detection of cardiac diseases and continuous supervision of clinicians can reduce the mortality rate. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. In this project, we have developed and researched about models for heart disease prediction through the various heart attributes of patient and detect impending heart disease using Machine learning techniques like backward elimination algorithm, logistic regression and REFCV on the dataset available publicly in Kaggle Website, further evaluating the results using confusion matrix and cross validation. The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. Keywords: Machine Learning, Logistic regression, Cross-Validation, Backward Elimination, REFCV, Cardiovascular Diseases.

Contents

List of Abbreviations

CHDP	: Coronary Heart Disease Predictor
WEKA	: Waikato Environment for Knowledge Analysis
CSV	: Comma Separated Value
RFECV	: Recursive Feature Elimination using Cross-Validation
TP	: True Positive
FN	: False Negative
TN	: True Negative
FP	: False Positive

List of Figures

List of Tables

Chapter 1

INTRODUCTION

According to the World Health Organization, every year 12 million deaths occur worldwide due to Heart Disease. The load of cardiovascular disease is rapidly increasing all over the world from the past few years. Many researches have been conducted in attempt to pinpoint the most influential factors of heart disease as well as accurately predict the overall risk. Heart Disease is even highlighted as a silent killer which leads to the death of the person without obvious symptoms. The early diagnosis of heart disease plays a vital role in making decisions on lifestyle changes in high-risk patients and in turn reduce the complications. This project aims to predict future Heart Disease by analyzing data of patients which classifies whether they have heart disease or not using machine-learning algorithms.

1.1 Problem Definition

The major challenge in heart disease is its detection. There are instruments available which can predict heart disease but either they are expensive or are not efficient to calculate chance of heart disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall complications. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. Since we have a good amount of data in today's world, we can use various machine learning algorithms to analyze the data for hidden patterns. The hidden patterns can

be used for health diagnosis in medicinal data.

1.2 Objectives

The main objective of developing this project are:

1. To develop machine learning model to predict future possibility of heart disease by implementing Logistic Regression.
2. To determine significant risk factors based on medical dataset which may lead to heart disease.
3. To analyze feature selection methods and understand their working principle.

Chapter 2

RELATED WORKS

With growing development in the field of medical science alongside machine learning various experiments and researches has been carried out in these recent years releasing the relevant significant papers. The paper [1] propose heart disease prediction using KStar, J48, SMO, and Bayes Net and Multilayer perceptron using WEKA software. Based on performance from different factor SMO (89% of accuracy) and Bayes Net (87% of accuracy) achieve optimum performance than KStar, Multilayer perceptron and J48 techniques using k-fold cross validation. The accuracy performance achieved by those algorithms are still not satisfactory. So that if the performance of accuracy is improved more to give batter decision to diagnosis disease. [2]In a research conducted using Cleveland dataset for heart diseases which contains 303 instances and used 10-fold Cross Validation, considering 13 attributes, implementing 4 different algorithms, they concluded Gaussian Naïve Bayes and Random Forest gave the maximum accuracy of 91.2 percent. [3]Using the similar dataset of Framingham, Massachusetts, the experiments were carried out using 4 models and were trained and tested with maximum accuracy K Neighbors Classifier: 87%, Support Vector Classifier: 83%, Decision Tree Classifier: 79% and Random Forest Classifier: 84%.

Chapter 3

DATASETS

The dataset is publicly available on the Kaggle Website at [4] which is from an ongoing cardiovascular study on residents of the town of Framingham, Massachusetts. It provides patient information which includes over 4000 records and 14 attributes. The attributes include: age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting, sugar blood, resting electrocardiographic results, maximum heart rate, exercise induced angina, ST depression induced by exercise, slope of the peak exercise, number of major vessels, and target ranging from 0 to 2, where 0 is absence of heart disease. The data set is in csv (Comma Separated Value) format which is further prepared to data frame as supported by pandas library in python.

	male	age	education	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD
0	1	39	4.0	0	0.0	0.0	0	0	0	195.0	106.0	70.0	26.97	80.0	77.0	0
1	0	46	2.0	0	0.0	0.0	0	0	0	250.0	121.0	81.0	28.73	95.0	76.0	0
2	1	48	1.0	1	20.0	0.0	0	0	0	245.0	127.5	80.0	25.34	75.0	70.0	0
3	0	61	3.0	1	30.0	0.0	0	1	0	225.0	150.0	95.0	28.58	65.0	103.0	1
4	0	46	3.0	1	23.0	0.0	0	0	0	285.0	130.0	84.0	23.10	85.0	85.0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4235	0	48	2.0	1	20.0	NaN	0	0	0	248.0	131.0	72.0	22.00	84.0	86.0	0
4236	0	44	1.0	1	15.0	0.0	0	0	0	210.0	126.5	87.0	19.16	86.0	NaN	0
4237	0	52	2.0	0	0.0	0.0	0	0	0	269.0	133.5	83.0	21.47	80.0	107.0	0
4238	1	40	3.0	0	0.0	0.0	0	1	0	185.0	141.0	98.0	25.60	67.0	72.0	0
4239	0	39	3.0	1	30.0	0.0	0	0	0	196.0	133.0	86.0	20.91	85.0	80.0	0

4240 rows × 16 columns

Figure 3.1: Original Dataset Snapshot

Chapter 4

METHODS AND ALGORITHMS

The main purpose of designing this system is to predict the ten-year risk of future heart disease. We have used Logistic regression as a machine-learning algorithm to train our system and various feature selection algorithms like Backward elimination and Recursive feature elimination. These algorithms are discussed below in detail.

4.1 Logistic Regression

Logistic Regression is a supervised classification algorithm. It is a predictive analysis algorithm based on the concept of probability. It measures the relationship between the dependent variable (TenyearCHD) and the one or more independent variables (risk factors) by estimating probabilities using underlying logistic function (sigmoid function). Sigmoid function is used as a cost function to limit the hypothesis of logistic regression between 0 and 1 (squashing) i.e. $0 \leq h\theta(x) \leq 1. In logistic regression cost function is defined as$:

$\text{Cos}(\text{h}\theta(x), y) = (-log(\theta(x)) if y = 1 - log(1 - \theta(x)) if y = 0)$

Logistic Regression relies highly on the proper presentation of data. So, to make the model more powerful, important features from the available data set are selected using Backward elimination and recursive elimination techniques.

4.2 Backward Elimination Method

While building a machine learning model only the features which have a significant influence on the target variable should be selected. In the backward elimination method for feature selection, the first step is selecting a significance level or P-value. For our model, we have chosen a 5% significance level or P- value of 0.05. The feature with high P-value is identified, and if its P-value is greater than the significance level it is removed from the dataset. The model is fit again with a new dataset, and the process is repeated till all remaining features in dataset is less than the significance level. In this model, factors male, age, cigsPerDay, prevalentStroke, diabetes, and sysBP were chosen as significant ones after using the backward elimination algorithm

4.3 Recursive Feature Elimination using Cross-Validation (RFECV)

RFECV is greedy optimization algorithm which aims to find the best performing feature subset. Recursive Feature Elimination (RFE) fits a model repeatedly and removes the weakest feature until specified number of features is reached. The optimal number of features is used with RFE to score different feature subsets and select the best scoring collection of features which is RFECV. The main issue of this algorithm is that it can be expensive to run. So, it is better to reduce the number of features beforehand. Since correlated features provide the same information, such features can be eliminated prior to RFECV. To address this, correlation matrix is plotted and the correlated features are removed. The arguments for instance of RFECV are: a. estimator - model instance (RandomForestClassifier) b. step - number of features removed on each iteration (1) c. cv – Cross-Validation (StratifiedKFold) d. scoring – scoring metric (accuracy). Once RFECV is run and execution is finished, the features that are least important can be extracted and dropped from the dataset. Top 10 features ranked by the RFECV technique in our model listed below from least importance to highest importance.

1. prevalentStroke
2. diabetes
3. BPMeds
4. currentSmoker
5. prevalentHyp
6. male
7. cigsPerDay
8. heartrate
9. glucose
10. diaBP

Chapter 5

EXPERIMENTS

5.1 Data Preparation

Since the dataset consists of 4240 observations with 388 missing data and 644 observations to be risked for heart disease, two different experiments were performed for data preparation. First, we checked by dropping the missing data, leaving with only 3751 data and only 572 observations risked for heart disease.

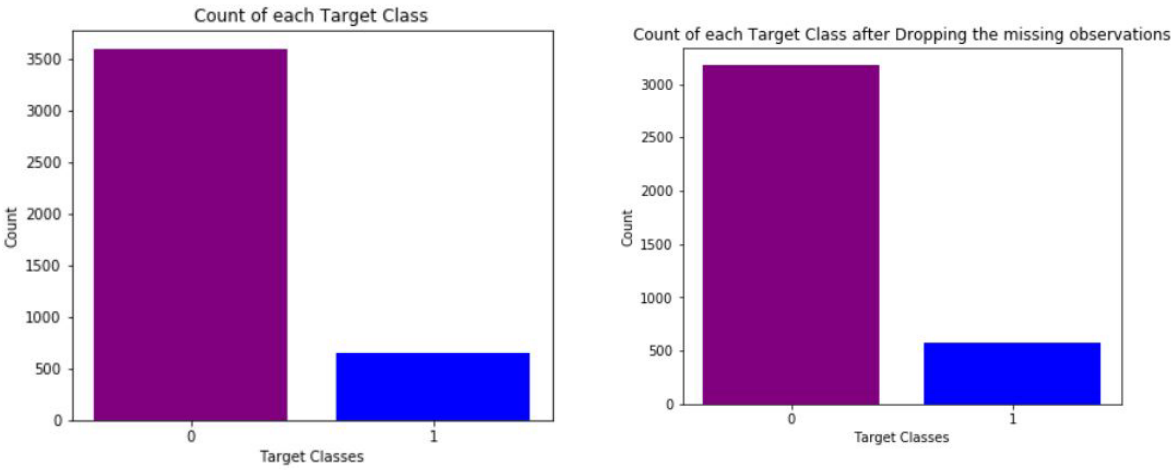


Figure 5.1: Bar Graph of the Target Classes Before Dropping and After Dropping

This leads to reduced number of the observations providing irrelevant training to

our model. So, we progressed with imputation of data with the mean value of the observations and scaling them using SimpleImputer and StandardScaler modules of Sklearn.

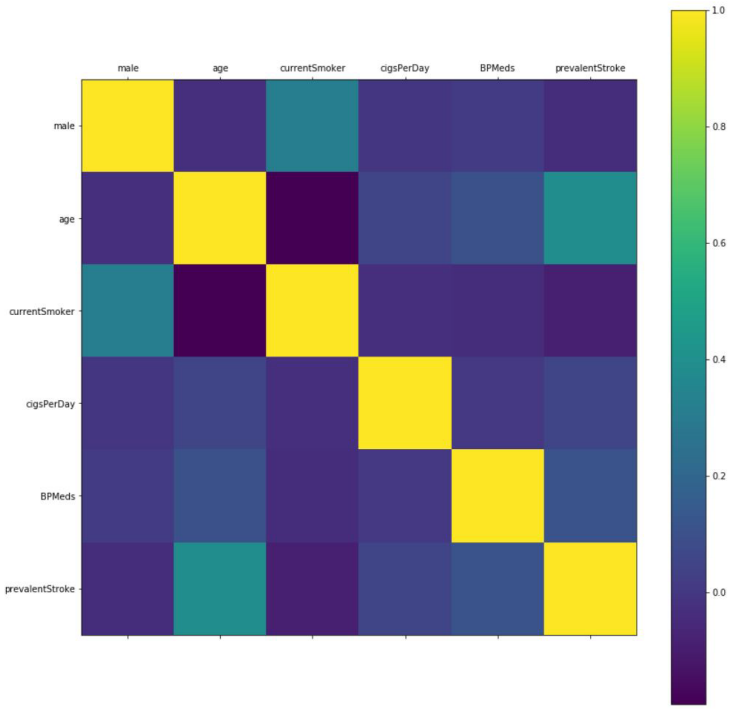
	male	age	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose
0	1.153113	-1.234283	-0.988276	-0.758062	-1.758000e-01	-0.077014	-0.671241	-0.162437	-0.940825	-1.196267	-1.083027	0.287258	0.342775	-2.174271e-01
1	-0.867217	-0.417664	-0.988276	-0.758062	-1.758000e-01	-0.077014	-0.671241	-0.162437	0.300085	-0.515399	-0.159355	0.719668	1.590435	-2.612309e-01
2	1.153113	-0.184345	1.011863	0.925410	-1.758000e-01	-0.077014	-0.671241	-0.162437	0.187275	-0.220356	-0.243325	-0.113213	-0.073111	-5.240539e-01
3	-0.867217	1.332233	1.011863	1.767146	-1.758000e-01	-0.077014	1.489778	-0.162437	-0.263965	0.800946	1.016227	0.682815	-0.904884	9.214724e-01
4	-0.867217	-0.417664	1.011863	1.177931	-1.758000e-01	-0.077014	-0.671241	-0.162437	1.089756	-0.106878	0.092555	-0.663554	0.758662	1.330035e-01
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4235	-0.867217	-0.184345	1.011863	0.925410	2.059493e-17	-0.077014	-0.671241	-0.162437	0.254961	-0.061487	-0.915087	-0.933810	0.675484	1.768073e-01
4236	-0.867217	-0.650984	1.011863	0.504542	-1.758000e-01	-0.077014	-0.671241	-0.162437	-0.602395	-0.265747	0.344466	-1.631564	0.841839	-6.224898e-16
4237	-0.867217	0.282295	-0.988276	-0.758062	-1.758000e-01	-0.077014	-0.671241	-0.162437	0.728764	0.051991	0.008585	-1.064025	0.342775	1.096688e+00
4238	1.153113	-1.117623	-0.988276	-0.758062	-1.758000e-01	-0.077014	1.489778	-0.162437	-1.166445	0.392425	1.268138	-0.049334	-0.738530	-4.364462e-01
4239	-0.867217	-1.234283	1.011863	1.767146	-1.758000e-01	-0.077014	-0.671241	-0.162437	-0.918263	0.029296	0.260496	-1.201610	0.758662	-8.601561e-02

4240 rows × 14 columns

Figure 5.2: Dataset after Scaling and Imputing

5.2 Exploratory Analysis

Correlation Matrix visualization Before Feature Selection shows



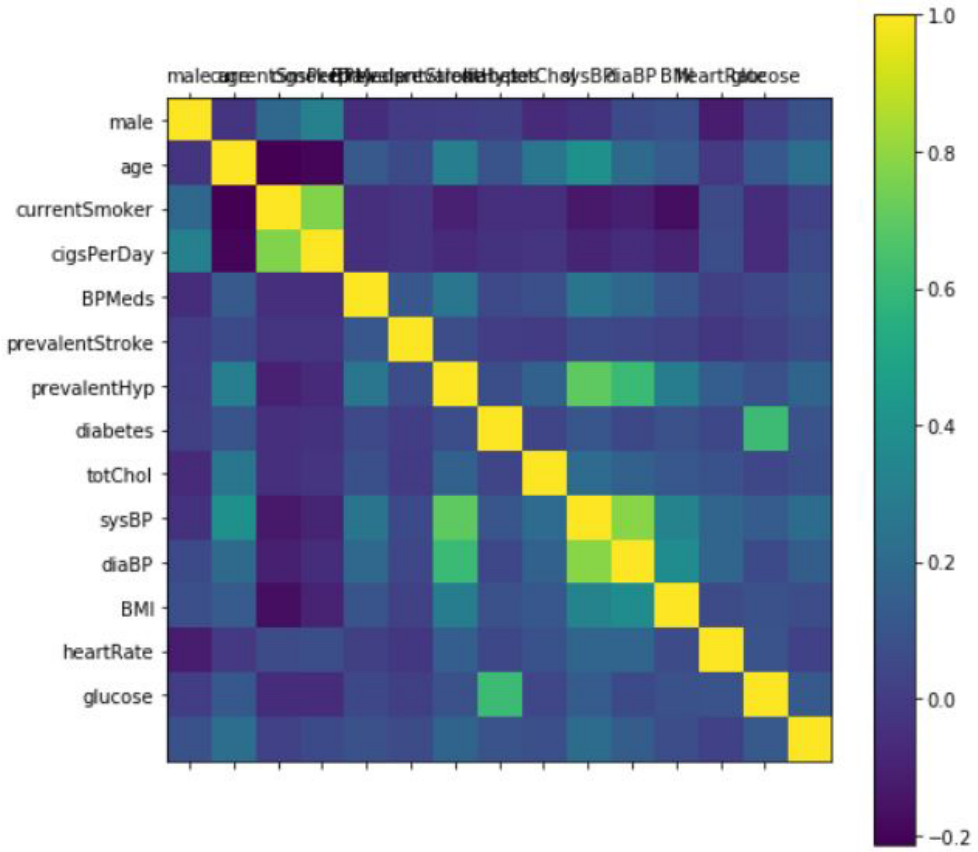


Figure 5.3: Correlation Matrix Visualization

It shows that there is no single feature that has a very high correlation with our target value. Also, some of the features have a negative correlation with the target value and some have positive. The data was also visualized through plots and bar graphs.

5.3 Feature Selection

Feature Selection using Backward Elimination (P-value) algorithm: Further the data was passed through the backward elimination function to select the most relevant features which gave following result:

Dep. Variable:	TenYearCHD	**No. Observations:**	4240
Model:	Logit	**Df Residuals:**	4234
Method:	MLE	**Df Model:**	5
Date:	Mon, 09 Mar 2020	**Pseudo R-squ.:**	-0.5700
Time:	10:32:30	**Log-Likelihood:**	-2835.5
converged:	True	**LL-Null:**	-1806.1
Covariance Type:	nonrobust	**LLR p-value:**	1.000

	coef	std err	z	P>|z|	[0.025	0.975]
male	0.1053	0.033	3.178	0.001	0.040	0.170
age	0.2626	0.035	7.505	0.000	0.194	0.331
cigsPerDay	0.1294	0.034	3.812	0.000	0.063	0.196
prevalentStroke	0.0813	0.038	2.124	0.034	0.006	0.156
diabetes	0.1055	0.035	3.046	0.002	0.038	0.173
sysBP	0.2244	0.035	6.370	0.000	0.155	0.293

Figure 5.4: Result from Feature Selection using Backward Elimination Method

According the result above the columns were dropped.

	male	age	cigsPerDay	prevalentStroke	diabetes	sysBP
0	1.153113	-1.234283	-0.758062	-0.077014	-0.162437	-1.196267
1	-0.867217	-0.417664	-0.758062	-0.077014	-0.162437	-0.515399
2	1.153113	-0.184345	0.925410	-0.077014	-0.162437	-0.220356
3	-0.867217	1.332233	1.767146	-0.077014	-0.162437	0.800946
4	-0.867217	-0.417664	1.177931	-0.077014	-0.162437	-0.106878
...	...	...	...	...	...	...
4235	-0.867217	-0.184345	0.925410	-0.077014	-0.162437	-0.061487
4236	-0.867217	-0.650984	0.504542	-0.077014	-0.162437	-0.265747
4237	-0.867217	0.282295	-0.758062	-0.077014	-0.162437	0.051991
4238	1.153113	-1.117623	-0.758062	-0.077014	-0.162437	0.392425
4239	-0.867217	-1.234283	1.767146	-0.077014	-0.162437	0.029296

4240 rows × 6 columns

Figure 5.5: Dataset After Dropping Columns after Feature Selection

Feature Selection using Recursive Feature Elimination and Cross- Validated selection method:

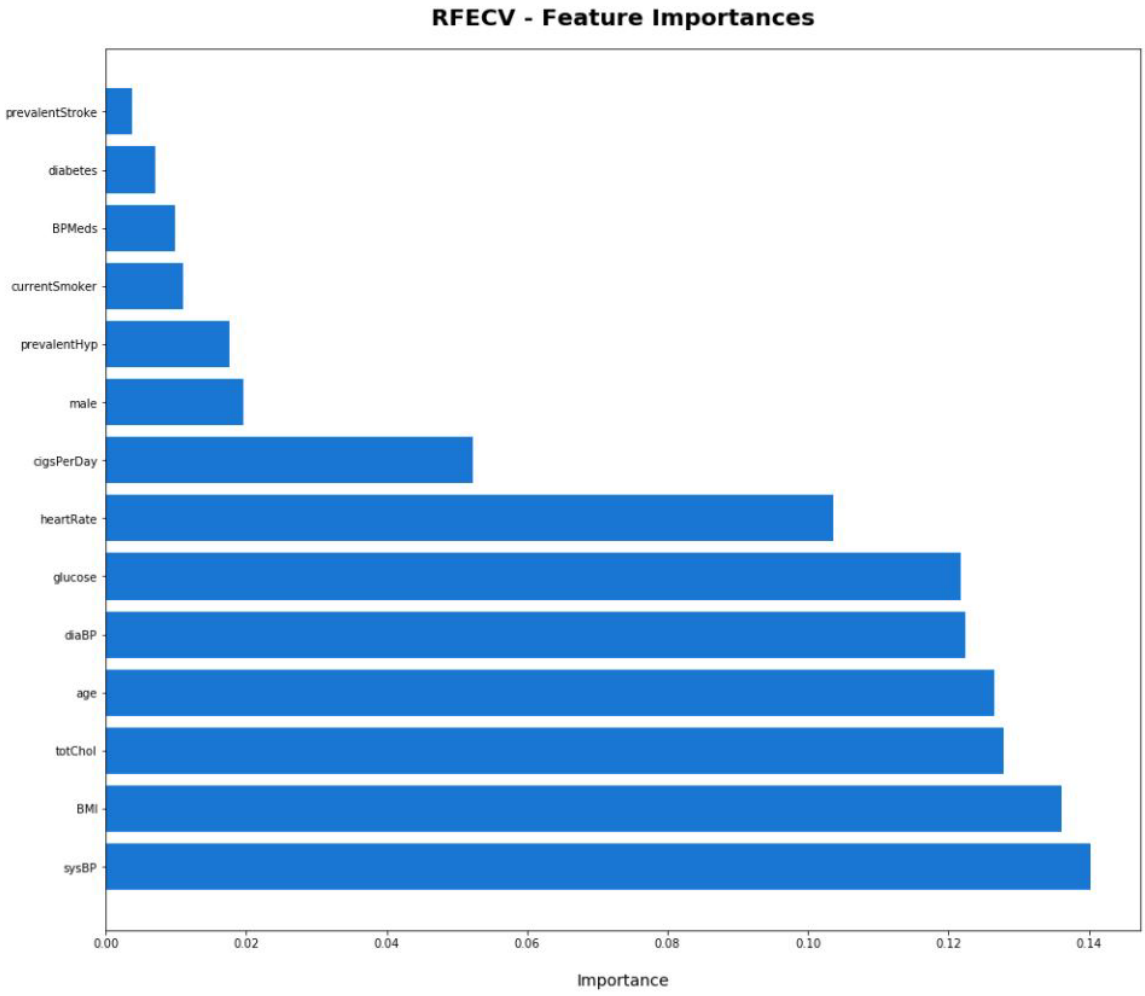


Figure 5.6: Top 10 important features supported by RFECV

5.4 Training and testing

Finally, this resulting data split into 80% train and 20% test data, which was further passed to the Logistic Regression model to fit, predict and score the model.

Chapter 6

EVALUATION METRICS

For the evaluation of our output from our training the data, the accuracy was analyzed "Confusion matrix".

6.1 Confusion Matrix

A confusion matrix, also known as an error matrix, is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. The key to the confusion matrix is the number of correct and incorrect predictions are summarized with count values and broken down by each class not just the number of errors made.

TP=3569	FP=27
FN=599	TN=45

Table 6.1: feature selection by backward elimination

6.2 Accuracy

The accuracy is calculated as: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

TP=3582	FP=14
FN=600	TN=44

Table 6.2: feature selection by RFECV method

Where,

- True Positive (TP) = Observation is positive, and is predicted to be positive.
- False Negative (FN) = Observation is positive, but is predicted negative.
- True Negative (TN) = Observation is negative, and is predicted to be negative.
- False Positive (FP) = Observation is negative, but is predicted positive

The obtained accuracy during training the data after feature selection using backward elimination was 86% and during testing was 83%. The obtained accuracy during training the data after feature selection using REFCV method was 86% and during testing was 85%.

6.3 Recall

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (a small number of FN). Recall is calculated as:

Recall = TP / TP+FN

The obtained recall during training the data after feature selection using backward elimination was and during testing was 0.99. The obtained recall during training the data after feature selection using REFCV method was 1.00 and during testing was 0.99.

6.4 Precision

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labelled as positive is indeed positive (a small number of FP). Precision is calculated as:

Precision = TP / TP+FP

The obtained precision during training the data after feature selection using backward elimination was 0.86 and during testing was 0.84. The obtained precision during training the data after feature selection using REFCV method and during testing was 0.86.

Chapter 7

DISCUSSION ON RESULTS

When performing various methods of feature selection, testing it was found that backward elimination gave us the best results among others. The various methods tried were Backward Elimination with and without KFold, Recursive Feature Elimination with Cross Validation. The accuracy that was seen in them ranged around 85% with 85.5% being maximum. Though both methods gave similar accuracy but it was seen that in Backward Elimination we found that the number of misclassifications of True Negative was more and it was observed that the accuracy had more variance compared to RFEV. The precision of Backward Elimination and RFEV are 84% and 86% respectively. And the recalls are 0.99 and 1 respectively. The precision and recall also shows that the number of misclassifications is less in RFECV than in Backward Elimination.

Evaluation Metrics	Backward Elimination	RFECV
Accuracy	83%	85%
Recall	0.99	0.99
Precision	0.84	0.86

Table 7.1: Comparison between the feature selection models after training and testing through Logistic Regression Model

Chapter 8

CODE

The coding portion were carried out to prepare the data, visualize it, pre-process it, building the model and then evaluating it. The code has been written in Python programming language using Jupyter Notebook as IDE. The experiments and all the models building are done based on python libraries. The code is available in the Git repository given in following link:

https://tinyurl.com/CoronaryHDP

8.1 Libraries used

1. NumPy : A library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

2. SciPy : Scipy is a free and open-source Python library used for scientific computing and technical computing. SciPy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

3. Matplotlib (pyplot, rcparams, matshow) : Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

4. Statsmodels : Statsmodels is a Python package that allows users to explore data, estimate statistical models, and perform statistical tests. An extensive list of descriptive statistics, statistical tests, plotting functions, and result statistics are available for different types of data and each estimator. It complements SciPy's stats module.

5. Pandas : pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license

6. Tkinter : Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Microsoft Windows and macOS installs of Python.

7. Sklearn : Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

Modules Used	Imported Class From Respective Modules
Sklearn.Impute	SimpleImputer
Sklearn.preprocessing	StandardScaler
Sklearn.pipeline	Pipeline
Sklearn.feature_selection	RFECV
Sklearn.ensemble	RandomForestClassifier
Sklearn.model_selection	Train_test_split, StratifiedKFold
Sklearn.linear_model	LogisticRegression
Sklearn.utils	Shuffle
Sklearn.metrics	Accuracy_score, Confusion_matrix

Table 8.1: Major Modules and Classes used from Sklearn

CONCLUSIONS

The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. This project resolved the feature selection i.e. backward elimination and RFECV behind the models and successfully predict the heart disease, with 85% accuracy. The model used was Logistic Regression. Further for its enhancement, we can train on models and predict the types of cardiovascular diseases providing recommendations to the users, and also use more enhanced models.

Bibliography

[1] A. H. M. S. U. Marjia Sultana, "Analysis of Data Mining Techniques for Heart Disease Prediction," 2018.

[2] M. I. K., A. I., S. Musfiq Ali, "Heart Disease Prediction Using Machine Learning Algorithms".

[3] K. Bhanot, "towarddatascience.com," 13 Feb 2019. [Online]. Available: $https : //towardsdatascience.com/predicting - presence - of - heart - diseases - usingmachinelearning - 36f00f3edb2c.$

[4] [Online]. Available: $https : //www.kaggle.com/ronitf/heart - disease - uciheart.csv.$

[5] M. A. K. S. H. K. M. a. V. P. M Marimuthu, "A Review on Heart Disease Prediction using Machine Learning and Data Analytics Approach".

[6] [Online] Prediction of heart disease and classifiers' sensitivity analysis: $https : //bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859 - 020 - 03626 - y.$