**A MAJOR PROJECT REPORT**

On

**DISEASE PREDICTION USING MACHINE LEARNING**

Submitted to

Amity University Uttar Pradesh



In partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

**TUSHAR SHANDILYA**

**A2305215037**

AND

**ANVI PURI**

**A2305215062**

Under the guidance of

**DR. ANKUR CHOUDHARY**

**Associate Professor**

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY

AMITY UNIVERSITY UTTAR PRADESH

NOIDA (U.P)

**DECLARATION**

We, **Anvi Puri and Tushar Shandilya,** students of B.Tech Computer Science & Engineering, hereby declare that the project titled “**Disease Prediction using Machine Learning**” which is submitted by us to Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Noida, in partial fulfillment of requirement for the award of the degree of Bachelor of Technology in 2019, has not been previously formed the basis for the award of any degree, diploma or other similar title or recognition.

|  |  |
| --- | --- |
| Noida |  |
|  |  |
| Date | Anvi Puri |
|  | Tushar Shandilya |

**CERTIFICATE**

On the basis of declaration submitted by **Anvi Puri and Tushar Shandilya**, students of B.Tech Computer Science & Engineering, I hereby certify that the project titled “**Disease Prediction using Machine Learning**”, submitted to Department of Computer Science & Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Noida, in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in 2019, is an original contribution with existing knowledge and faithful record of work carried out by them under my guidance and supervision.

To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

|  |  |
| --- | --- |
| Noida |  |
|  |  |
| Date | Dr Ankur Choudhary,  Department of Computer Science and Engineering,  Amity School of Engineering and Technology,  Amity University, Noida, Uttar Pradesh |

**ACKNOWLEDGEMENT**

We would like to thank our Head of Department Prof (Dr.) Abhay Bansal for his direction in our project as an important aspect of the curriculum and constantly guiding us with the same.

Secondly, we would like to thank our project guide Dr. Ankur Choudhary under whose guidance we were able to complete our project. We are highly thankful to him for giving us him valuable time and attention and or providing us various research papers which helped much in the research.

Last but not the least, our friends, parents, and colleagues helped us throughout this wonderful journey. We wish to acknowledge all of them with gratitude.

|  |  |
| --- | --- |
|  | Anvi Puri |
|  | Tushar Shandilya |

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| Declaration | i |
| Certificate | ii |
| Acknowledgement | iii |
| Table of Contents | iv |
| List of Figures | v |
| List of Tables | vi |
| List of Formulae | vii |
| **ABSTRACT** | 1 |
| **CHAPTER 1: INTRODUCTION** | 2 |
| 1.1 Challenges | 5 |
| 1.2 Purpose of Plan | 6 |
| 1.3 Project Goals and Objectives | 7 |
| **CHAPTER 2: LITERATURE REVIEW** | 8 |
| **CHAPTER 3: METHODOLOGY** | 13 |
| 3.1 System Design | 13 |
| 3.2 Software and Libraries used | 13 |
| 3.3 Data Source | 17 |
| 3.4 Graphical User Interface | 20 |
| 3.5 Workflow | 28 |
| **CHAPTER 4: RESULT** | 44 |
| **CHAPTER 5: ANALYSIS AND DISCUSSION OF RESULTS** | 48 |
| **CHAPTER 6: CONCLUSION** | 51 |
| **CHAPTER 7: FUTURE WORK** | 53 |
| 7.1 Future Scope | 53 |
| 7.2 Limitations | 54 |
| **REFERENCES** | 55 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| Figure 1. | PyCharm IDE | 14 |
| Figure 2. | Spyder IDE | 15 |
| Figure 3. | Framingham Dataset | 19-20 |
| Figure 4. | The Graphical User Interface | 21 |
| Figure 5. | Examples of error messages | 22 |
| Figure 6. | ‘Check Accuracy’ message | 23 |
| Figure 7. | Examples of ‘Show Graph’ when an algorithm is selected | 24 |
| Figure 8. | ‘Patience’ message | 25 |
| Figure 9. | ‘Low risk’ message | 26 |
| Figure 10. | ‘High risk’ message | 26 |
| Figure 11. | An example of generated pdf | 27 |
| Figure 12. | Workflow Diagram | 28 |
| Figure 13. | Support Vector Machine | 30 |
| Figure 14. | Implementing Kernel SVM | 32 |
| Figure 15. | Logistic Regression Vs Linear Regression | 33 |
| Figure 16. | Implementing Logistic Regression | 33 |
| Figure 17. | Implementing Naïve Bayes | 34 |
| Figure 18. | Different cases of K- Nearest Neighbour | 35 |
| Figure 19. | Implementing K-Nearest Neighbour | 36 |
| Figure 20. | Decision Tree- Dividing a dataset | 37 |
| Figure 21. | Implementing Decision Tree | 37 |
| Figure 22. | Random Forest | 38 |
| Figure 23. | Implementing Random Forest | 39 |
| Figure 24. | Artificial Neural Networks | 39 |
| Figure 25. | Rectifier Function | 40 |
| Figure 26. | Implementing Artificial Neural Networks | 42 |
| Figure 27. | Applying K-Fold Cross Validation and Grid Search for Kernel SVM | 43 |
| Figure 28a. | Accuracy of Logistic Regression model | 44 |
| Figure 28b. | Accuracy of Logistic Regression model after k fold cross validation and Grid Search | 45 |
| Figure 29a. | Accuracy of Kernel SVM model | 45 |
| Figure 29b. | Accuracy of Kernel SVM model after k fold cross validation and Grid Search | 45 |
| Figure 30a. | Accuracy of Random Forest model | 45 |
| Figure 30b. | Accuracy of Random Forest model after k fold cross validation and Grid Search | 45 |
| Figure 31a. | Accuracy of K-Nearest Neighbour model | 45 |
| Figure 31b. | Accuracy of K-Nearest Neighbour model after k fold cross validation and Grid Search | 46 |
| Figure 32. | Accuracy of Naïve Bayes model | 46 |
| Figure 33a. | Accuracy of Decision Tree model | 46 |
| Figure 33b. | Accuracy of Decision Tree model after k fold cross validation and Grid Search | 46 |
| Figure 34. | Accuracy of Artificial Neural Network model | 46 |
| Figure 35. | Accuracy Comparison Chart | 47 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| Table 1. | Dataset Attributes | 17-18 |
| Table 2. | Comparison of Accuracy | 44 |

**LIST OF FORMULAE**

|  |  |  |
| --- | --- | --- |
| Formula 1. | RBF Kernel Formula | 31 |
| Formula 2. | Sigmoid function formula for Logistic Regression | 32 |
| Formula 3. | Bayes Theorem Formula | 33 |
| Formula 4. | Posterior Probability Formula | 34 |
| Formula 5. | Sigmoid Function formula for ANN | 40 |
| Formula 6. | Rectifier Function Formula | 41 |

**ABSTRACT**

Patient health history plays an important role for prediction and cure of the disease. In today’s parlance almost every hospital maintains patient’s data using hospital information systems etc. This medical history data plays a vital role for prediction as well as decision making in treatment of disease.

There exist various approaches to analyze this data, but the selection of appropriate approach is still a challenge. In India, Heart related diseases are the 5th leading cause for death and increasing day by day. This project aims to study and provide effective machine learning technique that can be utilized to provide a pre-emptive medical diagnosis for heart related diseases. It also aims to provide an interactive user interface to ease of use and onsite prediction of likeliness of a heart disease to occur.

We have utilized various classification algorithms such as Decision Tree, Naïve Bayes, Logistic Regression, Kernel Support Vector Machine, Random Forest, K-Nearest Neighbor and Deep Learning algorithm- Artificial Neural Networks to predict the likeliness of a heart disease. Various classification algorithms are optimized using K-Fold Cross Validation technique followed by comparison of the improved accuracies.

**CHAPTER 1. INTRODUCTION**

According to the World Health Organization[1], millions of lives are taken away every year all over the world due to heart related diseases and the rate of people dying due to this particular cause only rises. Keeping this in mind it has become necessary to understand the cause and find ways to prevent these diseases before they occur. So, for the very same reason this project is based on predicting the likeliness of a heart disease to occur for a patient based on various traits and habits of the patient. It is an effort to understand and contribute to the cause as it is not only life threatening but also one of the main reasons for deaths happening across the globe.

The terminology “heart disease” or “cardiovascular disease” includes extensive variety of conditions that can affect the functioning of a heart. These can be of various types of heart diseases such as Atherosclerosis, valvular heart disease, cardiomyopathy, heart infections etc. Broken down to simplest definitions heart is a muscle and it needs a suitable amount of blood to provide oxygen so that it can deliver blood to the body and also to itself via coronary arteries. This is an important function for the human body to work properly.

When one or more arteries narrow down or get obstructed in any form it makes it difficult for the right amount of blood to flow in the arteries, causing the heart muscle to ache. This might cause symptoms of a cardiovascular disease. Should one the arteries get fully congested by some obstruction, a blood clot will happen causing the blood supply to the heart will be cut off, thereby causing a heart attack.

The closing of arteries and other symptoms leading up to cardiovascular diseases are caused by a variety of factors which range from our daily habits (such as a smoking habit, exercise activity etc.) to other passive conditions we might have developed overtime (obesity, etc.) and previous or ongoing medical conditions (such as strokes, blood pressure, diabetes, cholesterol, etc.).

However, this is only the problem in hand which not only needs a solution but also a way to predict the particular problem before it occurs using the various causes that lead to the problem in the first place.

A proper diagnosis by a doctor takes years of training and experience which require enormous amounts of time and effort. Even if a clinic is available an individual has to go through a health check-up regularly by themselves for an on-time prediction. There is no basis upon which the prediction might occur except the doctor’s insight and experience and not scientific methods which could prove that the disease is very likely to occur in the future.

In addition to this, a massive amount of medical data has been collected for the purpose of medical records and thus is widely available. Although some hospitals use expert systems for decision making, they are rare and usually answer simple questions like the number of people suffering from a heart disease. When it comes down to answering complex questions like the prediction of heart diseases, these systems fail. On the other hand, medical records containing data rich information can be analyzed, preprocessed and further utilized to predict the probability of a disease (heart disease specifically in case of this project) to occur in patients using the concept of machine learning.

The database of medical records can thus be integrated with various tools which clean the data by performing data preprocessing on it and then apply various machine learning algorithms on the current data to predict results of the future.

Machine Learning[2] for medical diagnosis and prediction has a great scope for discovering hidden patterns in the raw data of medical records. Thus, by using machine learning algorithms, we can cut down the effort and time required for prediction and diagnosis. Machine learning algorithms can be used to build prediction systems. In this project we have collected an appropriate dataset required to predict the likeliness of a heart disease to occur. Further the dataset has been preprocessed to make it suitable for analysis and prediction and the accuracy of each algorithm is calculated and compared to find the most efficient and best performing algorithm. Further a graphical user interface has been developed for the patient to predict the likeliness of a heart disease to occur, which can be printed as a report as well.

In the project we have compared various supervised algorithms as well as a deep learning algorithm to generate a result[3]. The system is built based upon the algorithms and can predict heart disease at significantly lower cost and a higher availability.

Machine learning is a subfield of Artificial Intelligence and it focusses on predicting accurate outcomes without being programmed towards that notion but by learning themselves. The main idea of machine learning is to form models based upon algorithms and the give dataset and then use the models to predict an output. It allows a machine or a computer to learn automatically without external aid or intervention. Machine learning algorithms are divided into various kinds of algorithms namely algorithms based on regression, algorithms based on classification, algorithms based on clustering, algorithms based on deep learning, natural language processing, etc. This project uses classification and deep learning algorithms to build models for prediction.

Classification algorithms are based on the concept of classification which classifies or divides the data into the most appropriate class among the available classes. Applications of classification can include disease prediction, handwriting recognition, document classification, etc. Classifiers can be of two types i.e. binary classifiers and multi-class classifiers. Binary classifiers classify data into two distinct classes and multi class classifiers classify data into more than two distinct classes. Classification algorithms include Logistic Regression, Support Vector Machine, Naïve Bayes, Decision Tree, Random Forest and K-Nearest Neighbour and all of them have been implemented in the project.

Deep Learning algorithms are used run the data in the dataset through the different layers of a neural network where the output of each layer acts as an input to the next layer. Neural network algorithms are based on the concept of the human brain and help in clustering and classification. They are used to group the data which is unlabeled based on the similarities between them or in the case of a labeled dataset, they can be used to classify the data into distinct classes. Artificial Neural Networks which is a deep learning algorithm has been implemented in the project.

**1.1 CHALLENGES**

However, while building a system that accurately predicts the likeliness of a heart disease to occur, many basic problems can occur which can hinder or reduce the accuracy of the algorithms or the system in general. Some of the problems that can be encountered are:

* Insufficient and incomplete Datasets

Making an accurate system using machine learning algorithms is not possible without a complete dataset. In addition to the problem of incomplete dataset, another problem that arises is that of insufficient dataset. Machine Learning algorithms need a considerable amount of observation to train the algorithm on. An insufficient data prevents this and in turn affects the accuracy of the algorithm as well. The pre-processing of data becomes a difficult task and if not done correctly, leads to inaccurate and unreliable results.

* Margin of Error in algorithms

The accuracy of a machine learning algorithm is rarely 100%. This fact leads to the conclusion that even the most efficient algorithm can be wrong about a prediction by a considerable margin. This is one of the greatest difficulties as it is very difficult to differentiate between a correct and a wrong prediction.

* Complex GUI

Graphical user interface of the system might be difficult to understand for the user and thus the user may not be able to use the system to their advantage. Thus, it is very important to design a user interface that can be easily comprehended by the user.

* Dataset with unmanageable number of attributes

Dataset is the most integral part of a machine learning algorithm as the algorithm is trained on the dataset. In addition to having a dataset, it is also very important to select the right number of attributes. The selected attributes should be such that they contribute to the algorithm. Enormous number of attributes in your dataset leads to difficulty in selecting the right factors upon which an accurate prediction has to be made.

* Smaller dataset

Another requirement to obtain an accurate and efficient machine learning algorithm is that the algorithm should be trained on a dataset with considerable number of observations. Training small datasets may not give as much accuracy as one could obtain by training on larger dataset.

* Unavailability of Information

There are some cases in which the user might not have accurate information required for a correct prediction. In such cases the accuracy of the prediction decreases as the algorithm is trained on wrong information and thus would create a model which would predict on the basis of the inaccurate information, thus giving inaccurate results.

**1.2 PURPOSE OF PLAN**

The main purpose of this project is to create a system that can accurately predict coronary heart diseases in an individual on giving input about various characteristics related to the health and habits of the individual. This can help them increase their awareness about the chances they have to incur a coronary heart disease and thus help them take preventive measures for the same. The project intends to minimize the effort and cost required for a prediction while increasing the availability.

People who are probable to be diagnosed with a heart disease might not be getting regular health checkups due to cost, effort or psychological factors. This technology can promote a more accessible and easier method that can be used to get a diagnosis at minimal effort and cost.

A majority of the population suffers from various factors giving way to heart disease such as high cholesterol, obesity etc. yet they fail to take any preventive measures to counter this condition due to the lack of knowledge. This system plans to tackle that problem and provide a solution in the form of a prediction.

**1.3 PROJECT GOALS AND OBJECTIVE**

* The project aims to increase the accuracy i.e. the correctness of the prediction algorithm so that the predicted results can be utilized as much as possible without the need to identify whether the prediction is accurate or not.
* The project also aims to build models based on machine learning algorithms which perform in the best possible manner without costing too much time and effort.
* Once the model has been made based on the various machine learning algorithms, the patient only needs to input various attributes to get the prediction. This requires minimal effort from the patient.
* Researching and studying about various machine learning algorithms that can be suitable to build models for the project.
* Gathering relevant information from various research papers which relate to the topic of this project.
* Building an appropriate workflow for the project which is followed to build the models, improve them and give a graphical user interface so that the system can be used by patients.
* Collecting various datasets and choosing the most appropriate dataset for prediction. Further choosing the right number of attributes which can contribute towards the prediction.
* Make efficient, accurate and reliable models based on various machine learning algorithms which can predict the likeliness of coronary heart disease to occur in a patient.
* To help people get an easy and economical method for prediction.
* Input attributes from individuals and give a prediction regarding the possibility of a coronary heart disease
* Make a graphical user interface which can be easily used by the patient to predict the possibility of a coronary heart disease.
* Accomplish the project goal within defined time period.

**CHAPTER 2. LITERATURE REVIEW**

In recent years, there has been a significant increase in the number of people having complex diseases, heart diseases and ailments being one of the commonly occurring diseases. With this sudden increase, the medical data related to the patients has also exponentially increased. This data can be utilized to develop systems which can accurately predict the probability of whether a person will have a disease in the future based on his medical history. This can be made possible with the use of various machine learning algorithms on the recorded data present with various medical houses where the algorithm can first be trained on the present data and then be used to give results. A lot of research has also been done on the following topic which can form the basis of what position or accuracy level the current algorithms possess to predict the occurrence of a disease, specifically a chronic heart disease in this case.

The research conducted by G. E. Sakr, I. H. Elhajj and H. A. Huijer[4] aims to focus on detection of agitation as well as the transition towards agitation in patients suffering from dementia. The research focusses on the use of Support Vector Machines where 3 primary factors are taken into account- the skin temperature, the heart rate and the galvanic skin response. Two SVM architectures that are proposed and are trained on these 3 factors, show a high accuracy of 91.4 % which is more accurate in comparison to the conventional SVM.

The research performed by Seyedamin Pouriyeh, Sara Vahid, Giovanna Sannino, Giuseppe De Pietro, Hamid Arabnia, Juan Gutierrez[5] focusses on comparison of different classification techniques as well as the combinations of Ensemble Machine Learning techniques. The various machine learning algorithms that have been compared are Naïve Bayes, Decision Tree, Multilayer Perceptron, K- Nearest Neighbors, Support Vector Machine, Radial Basis Function, Multilayer Perceptron, Single Conjunctive Rule Learner and Ensemble Learning Methods. The paper then shows the result of applying the algorithms mentioned above individually as well as in combination using 10-Fold Cross Validation. The mentioned algorithms have been applied on the Cleveland Heart Disease dataset. The paper focusses on depicting the following performance measurements- Precision, F-Measure, ROC Area, Accuracy and Recall of every algorithm. It was observed that the performance increased when Ensemble Machine Learning Methods were used in comparison to using the Machine Learning Algorithms individually. It was also observed that among all the algorithms SVM improved the most on applying boosting method on it.

The authors, C.-L. Chang and C.-H. Chen[6], focused on creating a predictive model for dermatology where they used, they conducted a set of five experiments which focused on six major skin diseases. They used a combination of decision tree with neural networks to construct their predictive model. The results depicted that an accuracy of 92.62% when the neural network model was used. But on combining with the decision tree model, the accuracy reduced to 80.33%.

In the paper written by Saangyong Uhmn, Dong -Hoi Kim, Jin Kim, Sung Won Cho, Jae Youn Cheong[7] Support Vector Machines, Decision Rules and Decision Trees have been used to predict how likely chronic hepatitis and cirrhosis, both of which are liver diseases, are likely to occur. The experiment has been performed on Single Nucleotide Polymorphism data which shows Decision Rule has the highest accuracy of 73.20% to distinguish between chronic and normal hepatitis, followed by an accuracy of 72.60% by Decision Tree. SVM was observed to have the least accuracy rate of 67.53%, thus proving Decision Rule and Decision Tree to be the most suitable tools of prediction.

The authors, S. Ghumbre, C. Patil, and A. Ghatolm[8], have made a system which is based on the combination of Support Vector Machine with a Radial Basis Function to be used for heart disease diagnosis. The system will then decide what type of heart disease is likely to occur. First, the SVM algorithm with SMO (Sequential Minimal Optimization) is applied to the dataset. This is then followed by the application of RBF network structure using OLS (Orthogonal Least Square) algorithm to the same dataset that the SVM was applied on. The experimental results then show the average performance of SVM and RBF to be 86.42 % and 80.81%. With 5-Fold Cross Validation the accuracy of SVM decreased to 85.04% but the accuracy of RBF increased to 82.71%. Further on applying 10-Fold Cross Validation the accuracy of SVM remained the same but the accuracy of RBF decreased to 82.24%.

The authors, S. Palaniappan and R. Awang[9], have used 3 data mining techniques- Naïve Bayes, Weighted Associative classifier and Apriori. The system mainly uses the CRISP-DM methodology to build the models mentioned above. Further the performance has been evaluated with methods such as Classification matrix, Bar charts, Lift charts and Pie charts. The classification matrix is used to display the number of correct and incorrect predictions. Further it is observed that WAC gives 84% accurate prediction followed by Naïve Bayes with an accuracy of 78%.

The research conducted by Edward Choi, Andy Schuetz, Walter F Stewart,and Jimeng Sun[10] on the data taken from the Electronic Health Records of a health system to observe whether deep learning can improve the performance of a model using conventional methods to predict heart failure. Recurrent Neural Networks is implemented using Gated Recurrent Units within a time frame of 12 to 18 months and then compared to regular Neural Network, Support Vector Machine, Logistic Regression and K-Nearest Neighbour classifiers. The area under curve for the RNN model implemented within 12, timeframe is 0.777 in comparison to area under curve for K-Nearest Neighbour which is 0.730, Support Vector Machine which is 0.743, Logistic Regression which is 0.747 and Multilayer Perceptron with a single hidden layer which is 0.765. It is then observed that on increasing the timeframe to 18 months, the area under curve for the RNN model increases to 0.883.

The research conducted by Chen, Min & Yixue, Hao & Hwang, Kai & Wang, Lu & Wang, Lin[11] is based on modifying prediction models and experiments on predicting chronic diseases of cerebral infarction. Convolutional Neural Networks based diseases prediction algorithm is suggested and is trained on structured and unstructured data. In comparison to the normal prediction algorithms, the accuracy for the following suggested algorithm that is the CNN based multimodal reaches 94.8% which is considerable higher than that of CNN based unimodal algorithm. Machine learning algorithms like Naïve Bayes, Decision Tree and K-Nearest Neighbour has been used for structured dataset. For Naïve Bayes conditional probability and Gaussian distribution is used. Classification and Regression Tree algorithm is chosen for the Decision Tree implementation. K-Nearest Neighbour uses Euclidean distance for measuring the distance and takes 10 as the k value. Among the three traditional models, Decision Tree has the highest accuracy of 63% which is followed by an almost equal accuracy shown by the Naïve Bayes as well as the Support Vector Machine algorithms. For Structured and Text Data, accuracy of Convolutional Neural Networks-UDRP (unimodal predictor) for Text data is 94.20% and recall is 0.9808. Also, the accuracy of Convolutional Neural Networks-MDRP (multimodal predictor) for Structured and Text data is 94.80% and recall is 0.99923. It is then observed that the difference between the accuracy of the two methods mentioned above is very minimal but the recall of Convolutional Neural Network-MDRP for Structured and Text data is better than the recall of Convolutional Neural Network-UDRP for Text data which means that the former model is better in performance.

The research conducted by Isra’a Ahmed Zriqat, Ahmad Mousa Altamimi, Mohammad Azzeh[12] highlights the development of an intelligent medical system using 5 classification algorithms namely, Naïve Bayes, Decision Tree, Support Vector Machine, Random Forest and Discriminant. The following have been implemented on two datasets which are the Cleveland and the Statlog datasets using MATLAB tools. It is observed that the most accurate algorithm is the decision tree algorithm according to the results obtained from calculating accuracy, F-measure and precision. One of the other important observations is that Support Vector Machine does not give as good or accurate results in comparison to other algorithms as it should. It is also observed that the accuracy of algorithms across two datasets is the most stable for Decision Tree algorithms which is followed by Random Forest classifier.

The author, Zhang Et Al[13], argues that Support Vector Machine is applied on the dataset consisting of attributes- blood pressure Glucose, Plasma Lipid and UA which is then used to identify coronary heart disease in individuals. The Support Vector Machine classifier is built with radial basis kernel. Then Particle Swarm Optimization (PSO) is used to optimize the kernel parameter as well as the factor C. It is then observed that among the Logistic Regression model, non-optimized Support Vector Machine, Artificial Neural Network (with back propagation model) and the model tried in the following paper, The RBF-SVM model is higher in accuracy which is 94.51 %, specificity which is 96.67% and sensitivity which is 92.31%.

In all the discussed research papers, the algorithms taken up vary and the dataset on which the algorithms are trained also vary. The most commonly occurring algorithms that have been taken up in the research papers are namely Logistic Regression, Support Vector Machine, Naïve Bayes, K-Nearest Neighbour, Decision Tree, Random Forest and different implementations of Artificial Neural Networks. The accuracy of the algorithms as well as the dataset on which they have been trained vary. Thus, the aim of this project is to train the above different algorithms on a dataset in the most optimal manner and try to make them more accurate so that they can be compared in various factors such as their accuracy, precision, recall, etc. to analyze their individual performance.

**CHAPTER 3. METHODOLOGY**

1. **SYSTEM DESIGN**

In this project, the proposed system takes into account various machine learning algorithms and makes models based upon these algorithms. The accuracies of the algorithms are then compared on a dataset. The dataset is based on the likeliness of occurrence of coronary heart diseases within the next 10 years and takes into account various factors concerning the patient such as their gender, cholesterol, systolic blood pressure, diastolic blood pressure, body mass index, if the patient is a current smoker or not etc. The algorithms are given a graphical user interface in the form of a desktop application where the user can input these attributes and the system will predict whether the patient is likely to have a coronary heart disease within the next 10 years. In addition to that the patient can even print the prediction in pdf format and can see the accuracies of different algorithms on the application in form of a graph.

1. **SOFTWARE AND LIBRARIES USED**
2. Python

The project uses Python as its programming language. Python is a popular language which is used for various purposes such as

* Development of web-based applications
* Connection to database systems
* Performing complex mathematical operations for big data and machine learning
* Creation of workflows
* Creation of rapid prototypes

Python is an object oriented, interactive and interpreted language which can work on variety of platforms such as Windows, Linux, Mac, etc. It has a very easy to read and learn syntax which is very similar to English.

The software used are PyCharm and Spyder.

PyCharm- An integrated development environment took kit. It is used to build or develop software and programs which is used for programming in Python.

PyCharm provides the following:

* An intelligent code editor which not only supports python but also JavaScript, CSS, etc. It also includes features such as error detection, language aware code fixes, etc.
* Smart code navigation is used to jump to the required class or file using the smart search tool.
* PyCharm also contains a wide range of built in developer tools such as a debugger with a graphical user interface, various version control systems, ability for debugging on remote machines, etc.
* It also offers a lot of web development frameworks such as Django, Flask, Pyramid, etc. Support for JavaScript, HTML and CSS is also provided.
* Presence of an interactive console.
* Python console and a built-in support for various scientific libraries such as NumPy, Pandas, Matplotlib, etc.

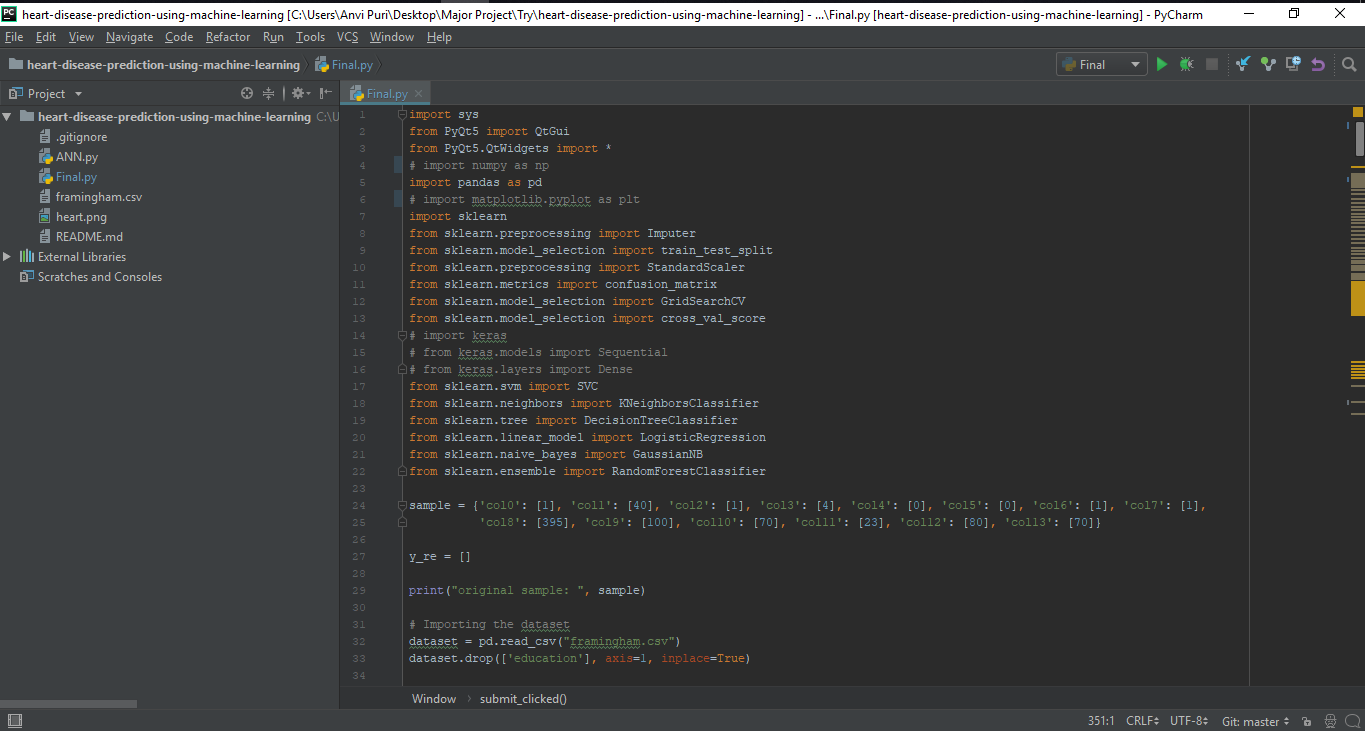


Figure 1. PyCharm IDE

Spyder

Spyder is a Python Development Environment used for scientific purposes. It is written in Python and provides a multi-language editor for efficient editing and the ability to analyze and debug. It provides visualization and data exploration capabilities. It can be used as an extension library for PyQt. Availability of an interactive I python console, presence of a GUI interface, a variable explorer for interacting ad modifying the variables and a help tool to instantly view documentations.

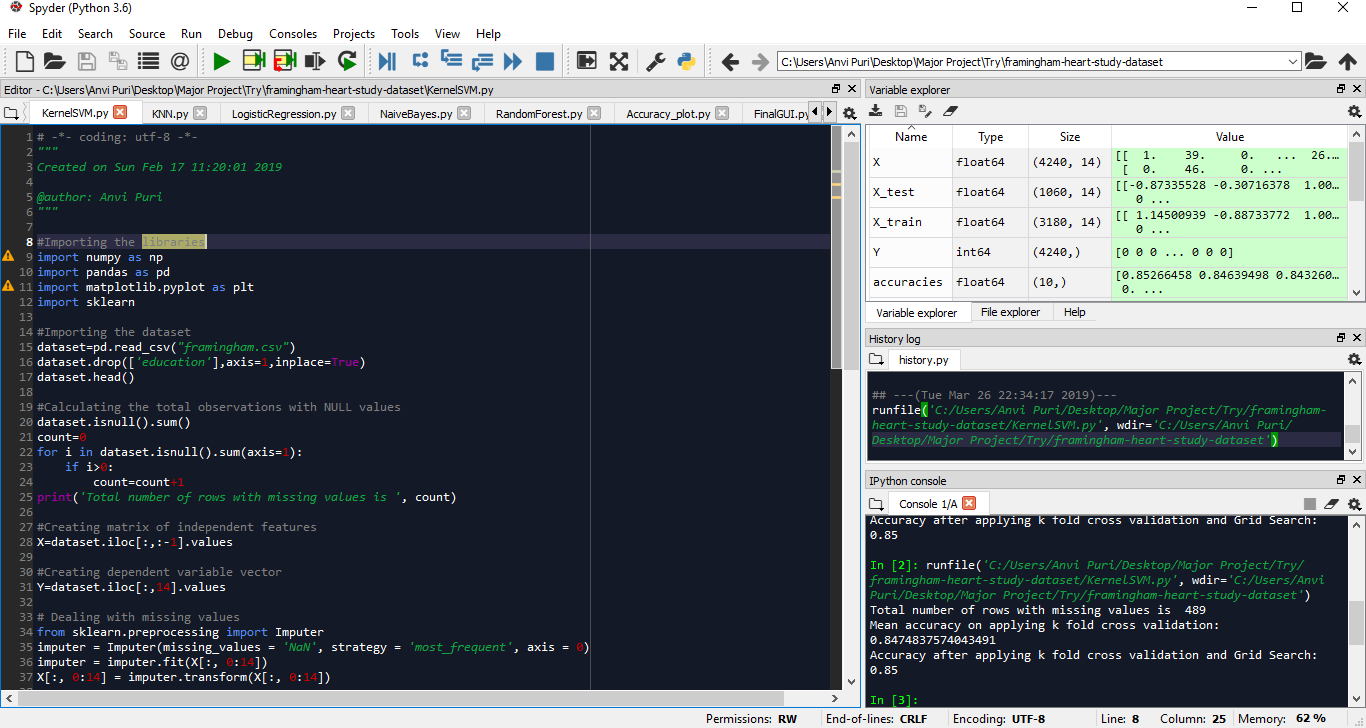


Figure 2. Spyder IDE

1. Pandas

An open source library for Python which is used to provide tools for ease in analysis of data. In the context of this project, it is mainly used to convert data in a CSV file to a data frame. A data frame is a Python object consisting of rows and columns and makes it easier to analyze data.

1. Scikit-learn

An open source, simple and efficient library which is built on SciPy, NumPy and Matplotlib libraries. Scikit-learn is used for data mining and analysis of data. In the context of this project, the library has been used to make models based on classification algorithms- Support Vector Machine, Decision Tree, Random Forest, K-Nearest Neighbour. Logistic Regression and Naïve Bayes algorithms.

1. Keras

An open source library for neural networks which is written in Python. Keras is an easy to use and fast library which runs on top of TensorFlow or Theano. It is a high-level API which cannot perform low level computations and thus uses the Backend library. In the context of this project, Keras has been used to make an Artificial Neural Networks model.

1. Matplotlib

It is a 2D plotting library which has been written in Python and is used to plot histograms, bar charts, scatterplots, error charts, etc. Matplotlib can be used for plotting in web application servers, Python scripts, Python shells, etc.

1. PyQt

PyQt is a Graphical User Interface toolkit and a powerful cross platform library. A combination of Python and Qt library, it is a Python interface for Qt. In the context of this project, this toolkit has been used to develop a GUI which can be used by patients to predict the likeliness of them having a coronary heart disease in the next 10 years.

1. NumPy

NumPy is fundamental general-purpose library within Python for scientific computing. It contains N-dimensional array object, provides a wide range of tools to work with these arrays, wide range of functions and is useful for linear algebra, random number capabilities and Fourier transform. It also provides tools to integrate C, C++, etc. NumPy among various other uses can be used as a multi-dimensional container of generic data.

1. **DATA SOURCE**

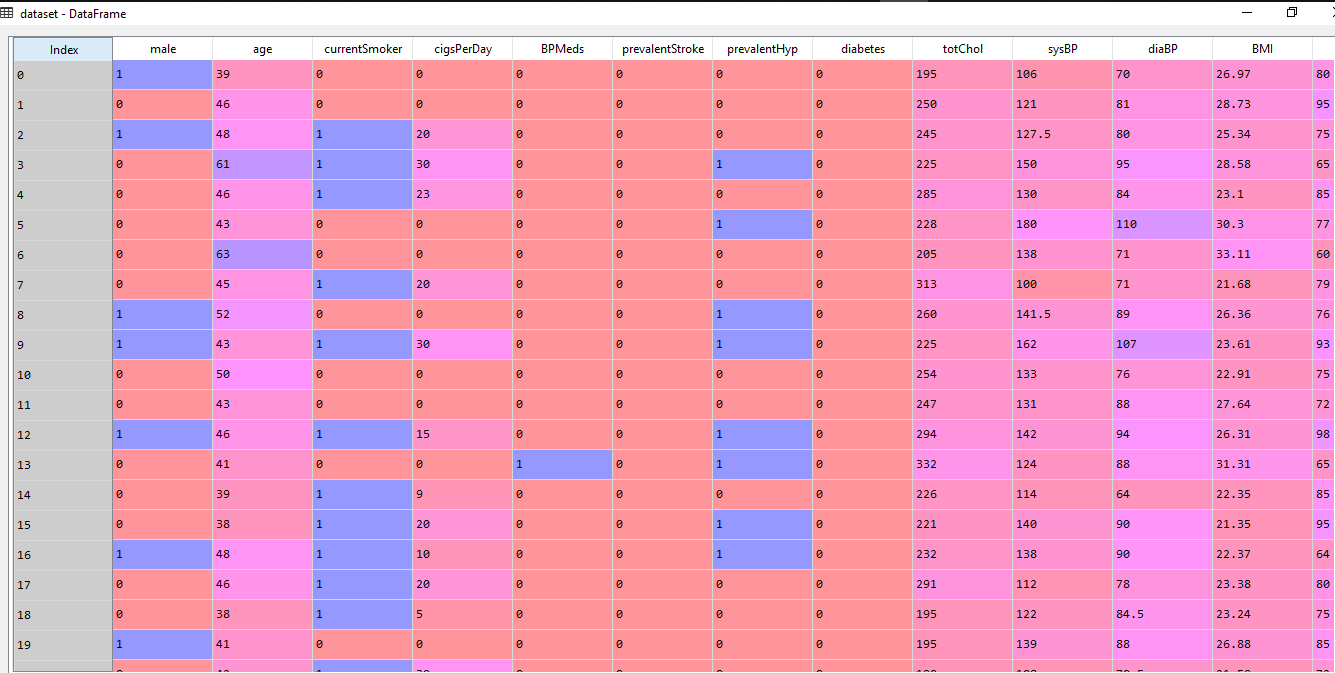
Clinical databases have gathered and thus contain a large amount of information about various kinds of patients along with their respective medical conditions. A lot of this information contains medical data related to heart diseases which mainly comprise of diseases like the Coronary heart disease, Cardiovascular diseases, etc. which can be used to determine the probability of these diseases to occur.

The dataset for this project has been taken from the Framingham Heart Study Dataset[14] and is available in CSV file format. Coronary heart diseases can be predicted with the help of the following database by finding significant patterns in the data available through machine learning algorithms. The dataset has been split into a training dataset and a testing dataset where the training dataset is fed to the algorithm to learn the logic and the testing dataset is used to judge how accurate the predictions are.

The attributes in the dataset can be divided into Input Attributes and Predictable Attributes. The Input Attributes are those attributes which are taken as an input for the algorithm to predict the likeliness of a certain disease to occur. The Predictable Attributes on the other hand are the attributes which are predicted on the basis of the result of the algorithm and are compared with the already existing values in the data to see the accuracy of the algorithm.

|  |  |
| --- | --- |
| **INPUT ATTRIBUTES** | |
| Attribute Name | Detail |
| sex | Males are denoted by 1  Females are denoted by 0 |
| education | Denotes the level of education the patient has had. The following attribute is dropped as it is of no relevance to predicting Coronary Heart Diseases. |
| currentSmoker | 0=patient is a non-smoker  1=patient is a smoker |
| cigsPerDay | The number of cigarettes the patient smokes per day on average. |
| BPMeds | 0= Not taking Blood Pressure medication  1=Taking Blood Pressure medication |
| prevalentStroke | 0=patient has never suffered from a stroke  1=patient has suffered from a stroke |
| prevalentHyp | 0=patient has/had hypertension  1=patient does not have hypertension |
| diabetes | In mg/Dl  0=patient does not have diabetes  1=patient has diabetes |
| totChol | In mg/dl  Total cholesterol level |
| sysBP | In mmHg  The systolic blood pressure of the patient. |
| diaBP | In mmHg  The diastolic blood pressure of the patient. |
| BMI | Body Mass Index of the patient. |
| heartRate | In Beats/min  The heart rate of the patient. |
| glucose | In mg/dL  The Glucose Level |
| **PREDICTABLE ATTRIBUTES** | |
| Attribute Name | Detail |
| TenYearCHD | The risk of having Coronary Heart Disease in 10 years.  0=No  1=Yes |

Table 1. Dataset Attributes



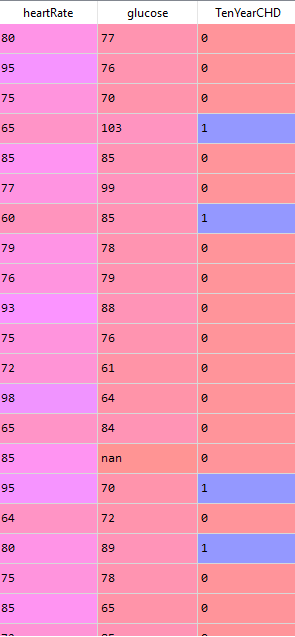


Figure 3. Framingham Dataset

1. **GRAPHICAL USER INTERFACE**

The Graphical User Interface (GUI) has been built to provide ease of access to the user in regards to entering parameters, getting a result based on their entered parameters, and some other functions. The user has to enter their name, age, number of cigarettes consumed per day, cholesterol, systolic blood pressure, diastolic blood pressure, body mass index, heart rate & glucose manually in the given text box made using the ‘Line Edit’ widget. The gender can be selected from a drop-down menu using the ‘Combo Box’ widget. Checkboxes are provided for if the person is a current smoker, whether they are taking blood pressure medication, if there are prevalent conditions regarding stroke and hypertension and if the patient is diabetic using the ‘Check Box’ widget. In the second section ‘Radio Button’ widgets provide a selection between seven algorithms. Once you select an algorithm you can check the accuracy of your selected algorithm or you can submit your entries and get a prediction against them.

|  |
| --- |
| Figure 4. The Graphical User Interface |

We as humans have a tendency to make mistakes and it is possible that the fields provided might not be filled properly or may have missing data. Factors like body mass index, age, cholesterol can’t be more than a certain value. In case the user tries to enter a value that is not within the scope of the field value to be entered (for example trying to enter age as 189 years), the field doesn’t allow for it. It will only take a certain number of characters as input and thus prevent the user from mistakenly entering incorrect or wrong information. Also, in the ‘cigarettes per day’ field, if the user has not checked the ‘current smoker’ checkbox, the value for it is automatically taken to be zero, but if they do check mark the ‘current smoker’ checkbox and submit the form without entering any data in the ‘cigarettes per day’ field they get an error message in the form of a message box. Similarly, if you don’t enter all the parameters provided you will get an error message in the form message box which will notify you about which field has incomplete data and allow the user to correct it.

|  |  |
| --- | --- |
|  |  |
| Figure 5. Examples of error messages | |

When you select the algorithm and click on ‘Check Accuracy’ button, it will show a message box displaying the accuracy of the algorithm allowing user to have better information about their algorithms.

|  |
| --- |
| Figure 6. ‘Check Accuracy’ message |

Further, user can see a graph using the ‘Show Graph’ button that will give you a graphical representation in the form of a chart detailing both the old and the current improved accuracy of each algorithm. This graph has been formed using the ‘Matplotlib’ library. This library enables us to represent the accuracies of the algorithms in a visual format which entails both their original accuracy and their accuracy after application of K-Fold Cross Validation and Grid Search. They are properly indicated using legend and labels. If an algorithm has been selected, then the accuracy of that algorithm is highlighted with a colour that differs from the usual and given a label in the legend to provide easy visualization of information about the selected algorithm for the users.

|  |
| --- |
|  |
|  |
|  |
| Figure 7. Examples of ‘Show Graph’ when an algorithm is selected |

Among the seven algorithms, Artificial Neural Networks takes the longest time and can give the appearance of the application being non-responsive. So, if you select the Artificial Neural networks and try to check the accuracy or submit the information for a prediction you get a message box informing you the highly complex nature of this algorithm and how it’s going to take longer than usual to provide a result just so the user doesn’t become restive due to the slow responsive time.

|  |
| --- |
| Figure 8. ‘Patience’ message |

When the user submits the information using the ‘Submit’ button, the user gets a notification about the result. Using the algorithm selected by the user, the entered information is processed and based on it gives the user a result in the form of a message box with the information about the prediction. The result will tell you either you are at ‘high risk’ or at ‘low risk’ of getting a Coronary Heart Disease within the next ten years. This message box further asks you if you would like to get your results in the form of a pdf file or not.

|  |  |
| --- | --- |
| Figure 9. ‘Low risk’ message | Figure 10. ‘High risk’ message |

If the user decides to print the pdf, then they get a pdf file in a pre-set location. This is done using the FPDF library of python. It is a famous qt library that is popularly used to get output from a python file in the form of a .pdf file. We can set the location or the destination at which we want the PDF file to be saved on the machine. The generated result can then be accessed freely without the use of any software other than the one used to read pdf files. We utilize it to provide the user with something that is more tangible than just some numbers and words on a screen.

|  |
| --- |
|  |

Figure 11. An example of generated pdf

1. **WORKFLOW**

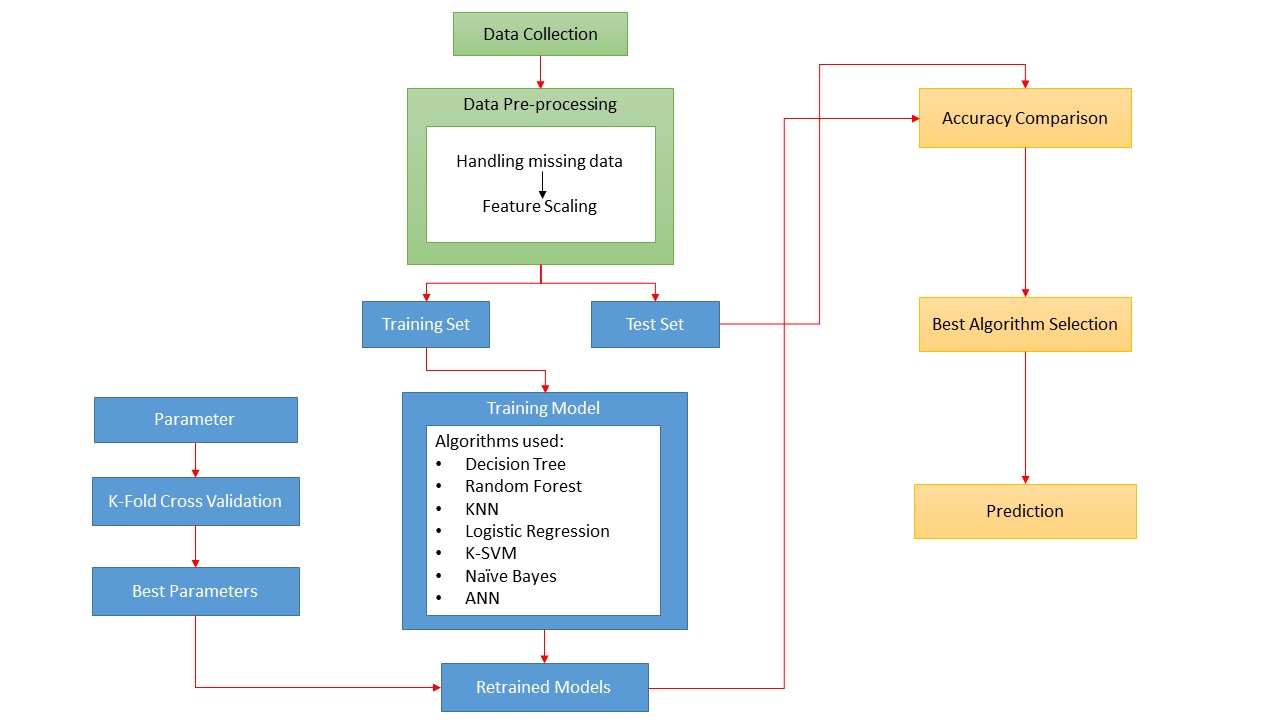


Figure 12. Workflow Diagram

1. Data Collection

Framingham Heart Study Dataset[14] has been used for the following project. Another dataset that was looked into was the Heart Disease Dataset from UCI[15]. However, it was rejected due to the lack of enough observations to run the algorithms on.

1. Data Pre-Processing

This is divided mainly into 5 steps i.e.

* Importing libraries

Libraries are a collection of functions which help in performing specific tasks without the need to write the code. There are various libraries in Python used to make the job of machine learning easier. However, to make use of these libraries, the first step would be to import the required libraries.

* Importing the dataset

The next step after importing libraries is to load the dataset that we will be using for the project. The format that is used for the dataset is CSV files. The dataset can be loaded in a number of ways such as loading the CSV file using the Python Standard Library, loading the CSV file using the NumPy Library or loading the CSV file using the Pandas Library. For the following project the Pandas library has been used to load the dataset. The CSV data can also be loaded directly from a URL. However, before loading the csv file it is important to set the working directory containing the data.csv file and saving it. After importing the dataset, it is important to distinguish between the independent variables i.e. the matrix of features and dependent variables.

* Handling the missing values using the Imputer class

Situations might occur when some fragments of data may be absent due to various reasons such as corrupt information, data failure, etc. In such case s it is very important to handle the missing data correctly to generate efficient data models. The various ways in which the missing data can be handled are deletion of rows with null values or a column which has 70-80% of missing values, Replacement with mean, median or mode is suitable for features having numeric data.

* Feature scaling using normalization or standardization.

There are times when variables don’t have their values in the same range or vary in range and units. To avoid the domination of one variable from the other variable, the variables are transformed through Standardization or Normalization. If the variables are not feature scaled in the cases when one variable dominates the other, on taking the Euclidean distance, it will be dominated by one factor. Since most of the machine learning algorithms use Euclidean distance in computing two data points, the following can be a problem. Thus, it is very important to have values within the same range so that the larger value does not dominate the smaller values.

1. Division of Dataset into Training and Testing sets-

The dataset is further divided into training and testing sets where the training set comprises of 75 percent of the dataset.

1. Training Model

Classification algorithms and Deep Learning algorithms are then implemented on the training dataset to create a training model.

Classification algorithms used are:

* Kernel-SVM

Support Vector algorithm aims to find a hyperplane in an N-dimensional space that separates the data points into two classes. Hyperplane is a decision boundary which helps in classifying the data points on the basis of the side of hyperplane on which they fall. A hyperplane can be a line in case of 2 input features or a plane in case of 3 or more than 3 input features.

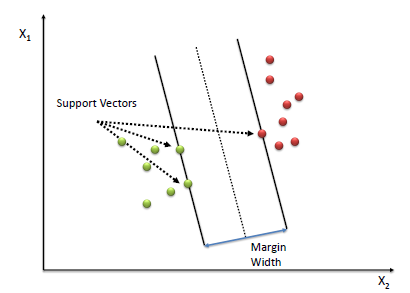


Figure 13. Support Vector Machine[16]

The main objective of the algorithm is to find a line/plane that has the maximum margin. Maximum margin refers to the line/ plane that will divide the dataset into two classes in a way that the data points of two classes have maximum distance between them. Maximizing the margin distance increases the probability of getting a right prediction.

In addition to this data points that are used to decide the orientation as well as the position of the hyperplane are known as support vectors. These are used to maximize the margin of the classifier. If the positions of the support vectors are changed, the position of the hyperplane will change as well.

Kernel is defined as a set of mathematical functions in SVM algorithms. In situations where it is difficult to find a boundary and the data is not linearly separable the kernel takes data as input and maps it into a higher dimension which becomes linearly separable. Kernel functions can be of different types such as linear, polynomial, radial basis function, sigmoid and nonlinear.

In this project, Radial Basis Function (RBF) has been used. This kernel is used when no prior information about the data is available. The kernel is based on the following formula:

K ()=

Formula 1. RBF kernel formula

In the above formula, x is the point in dataset is the landmark and σ is the fixed parameter. Based on the fixed parameter, the circumference of points to be included are chosen from the dataset. Greater the fixed parameter, more the points to be chosen. The points which are not chosen are considered as 0. Thus, the fixed parameter classifies the points that are wanted in the classification from the points that are not wanted, thus creating a decision boundary. In addition to this, other parameters taken for the K-SVM implementation are the penalty parameter C which is set to 1 and gamma. Gamma parameter is a kernel coefficient for RBF, poly and sigmoid kernels. Gamma parameter has its value set to 0.3. Larger the value of gamma, smaller is the variance making the support vectors not having a widespread influence.

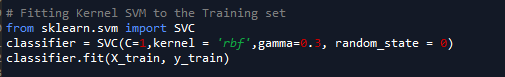


Figure 14. Implementing Kernel SVM

* Logistic Regression

The logistic regression algorithm is based on the sigmoid function which has the following formula:

P=

Formula 2. Sigmoid function formula for Logistic Regression

Logistic Regression algorithm is used to predict results with binary outcomes with the help of a given set of independent variables. Probabilities are predicted using the L8ogistic Regression algorithm. To calculate the outcome a particular point is chosen as the threshold above which all probabilities are considered to have 1 as its outcome and all probabilities below the point are considered to have 0 as its outcome.



Figure 15. Logistic Regression Vs Linear Regression[17]

In the following project, the parameters used are penalty, solver and maximum iterations. The penalty parameter is used to specify the norm that is used in penalization and is set to l2. The solver parameter refers to the algorithm that has been used in optimization and is set to newton-cg. The maximum iterations parameter refers to the maximum number of iterations taken for the solvers to converge and is set to 50.

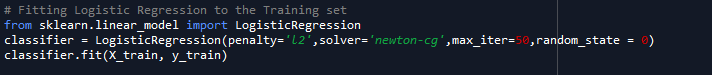


Figure 16. Implementing Logistic Regression

* Naïve Bayes

The following algorithm is a classification technique (Naïve Bayesian classifier) which is derived from the Bayes theorem. The theorem is based on the following formula:

P

Formula 3. Bayes Theorem formula

The main assumption of the following algorithm is that there should be an independence among the features / attributes i.e. they should not be interrelated or dependent on each other. In case of a dependency, these features still contribute independently to the probability of the result. The following classifier is very useful for a large dataset.

where,

Pis the Posterior probability of some class based on the value of predictor.

P() is the Likelihood i.e. the probability of some predictor provided the class is given.

P (C) is the Class Prior Probability which in this case is the probability of whether a person has a coronary heart disease or not.

P (X) is the Predictor Prior Probability

P P

Formula 4. Posterior Probability formula

Posterior Probability can be calculated using this theorem using P ( ), P (C), P (X) where in the above case X denotes the value of some predictor on some class C.

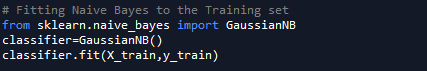


Figure 17. Implementing Naïve Bayes

* K-Nearest Neighbour

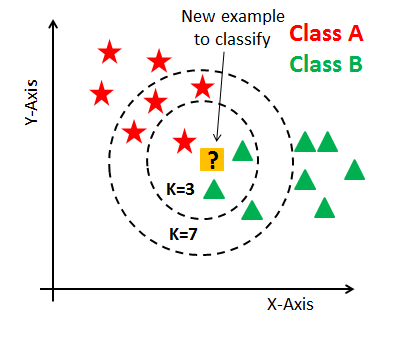


Figure 18. Different cases of K- Nearest Neighbour[18]

This algorithm is based on classification on the bases of the neighbours of a data point. The number of neighbours to be considered around the data point is chosen. The data point is then used to choose its k nearest neighbours based on a distance criterion like Euclidean distance, etc. Among these k data points, the number of data points in each category is counted and the category with maximum number of data points is chosen as the outcome.

In the following project the chosen parameters are n\_neighbours which denotes the number of neighbours and is set to 10, weights which is set to uniform in this case denoting that all points in the neighbourhood are weighed equally, ball\_tree algorithm is chosen, metric is used to denote the distance metric that is to be used for the tree and p is the power parameter for minkowski metric and is set to 2. When the default metric is chosen to be minkowski with the power parameter p being set to 0, this combination is equivalent to Euclidean metric.

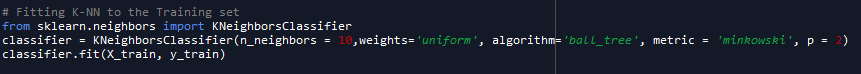


Figure 19. Implementing K-Nearest Neighbour

* Decision Tree

A decision tree algorithm which is a supervised algorithm, classifies data on the basis of attributes. The data can be classified into edges and nodes where edges are used to represent the output of a test on some attribute and the nodes are used to represent the test that will be performed on some attribute. If- Then rule is used to build the decision tree which can handle numerical as well as categorical data. In the following algorithm the data points are sliced in iterations so that the data points with a similar output remain together. The following algorithm performs better on large amounts of data.

The following steps are included to build the decision tree:

Step 1: Attribute present in the dataset are placed at the root of the tree.

Step 2: The dataset is split into subsets in such a way that each subset contains data with the same value for some attribute.

Step 3: Steps 1 and 2 are repeated on every subset till leaf nodes in all branches of the leaf nodes are found

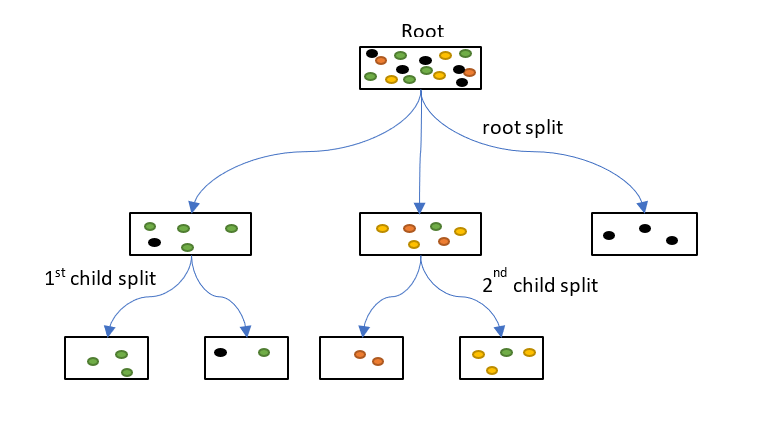


Figure 20. Decision Tree- Dividing a dataset[19]

The algorithm first calculates the entropy of every attribute. The dataset is then split on the basis of the variables (or predictors) with minimum entropy or maximum information gain.

The decision tree algorithm does not take Euclidean distance into account. Thus, it is not necessary to perform Feature Scaling on Dataset in this case. The parameters used are criterion, splitter and maximum depth. The criterion parameter is used to measure the quality of the split. In this project, the criterion has been set to entropy for information gain (higher the information gain, more the homogeneity). Splitter parameter is used to choose the best possible strategy at each node. In this project, random has been se to choose the best random split. Maximum depth is another parameter which decides the maximum depth of the tree. In this project maximum depth is taken as 10.

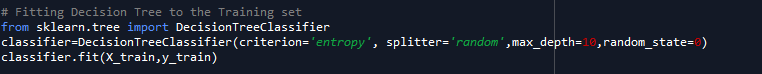


Figure 21. Implementing Decision Tree

* Random Forest

Random Forest classification technique is based on Ensemble learning where multiple machine learning algorithms are considered to make one machine learning algorithm. In random forest classification n data points are chosen from dataset at random and a decision tree is built around these n data points. The number of trees to be made is chosen and trees are built by choosing n number of data points and decision trees around them. Each decision tree is used to predict the outcome corresponding to a particular data point and the outcome with majority votes i.e. the outcome that was predicted on the maximum number of trees is chosen as the final outcome.

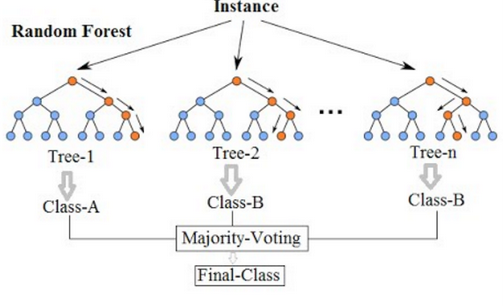


Figure 22. Random Forest [20]

In this project the parameters chosen for the Random Forest classifier are n\_estimators which sets the number of sets in the forest, criterion to measure the quality of the split and max\_features refer to the number of features to consider when looking for the best split. The number of trees is set to 14, criterion is set to entropy and the maximum number of features of be considers is set to 14.

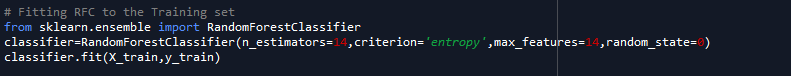


Figure 23. Implementing Random Forest

Artificial Neural Networks are implemented in Deep Learning.

Artificial Neural Networks are comprised of three layers namely Input layer which takes the input, Hidden Layer which performs the computations and the Output Layer which gives the output. Inputs are independent variables and need to be feature scaled necessarily. The output signal/value can be a value, binary outcome or categorical. The inputs get assigned with weights which are very important for the neural network training. These weights also get adjusted.

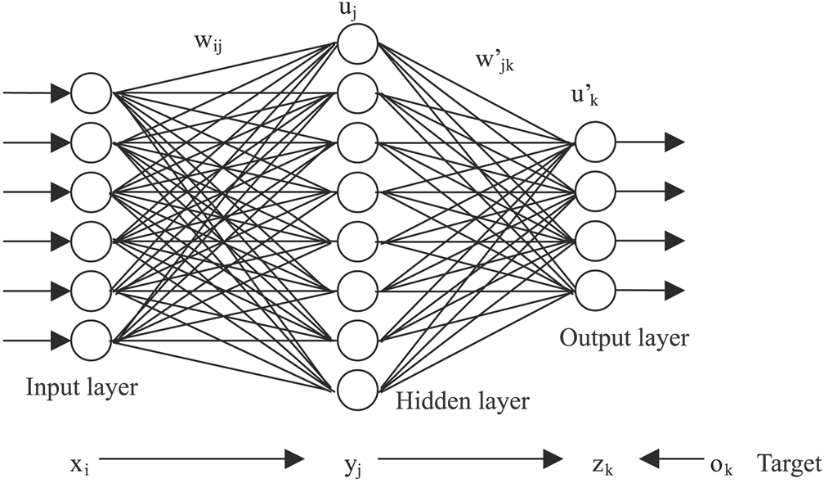


Figure 24. Artificial Neural Networks[21]

Further an activation function is applied to the weighted sum of inputs and depending on the function, the signal may or may not be passed further to another hidden layer or output layer. There are various activation functions like threshold function, sigmoid function, rectifier function, hyperbolic tangent function, etc.

In the following project, the artificial neural network has 2 hidden layers where rectifier function has been applied on hidden layers and sigmoid function has been applied on the output layer. As the dependent variable is binary, sigmoid function has been applied on the output layer. Rectifier function has been applied on the hidden layers as the neuron will only fire once the property will cross a certain mark.

The sigmoid function has the following formula:

Φ(x)=

Formula 5. Sigmoid function formula for ANN

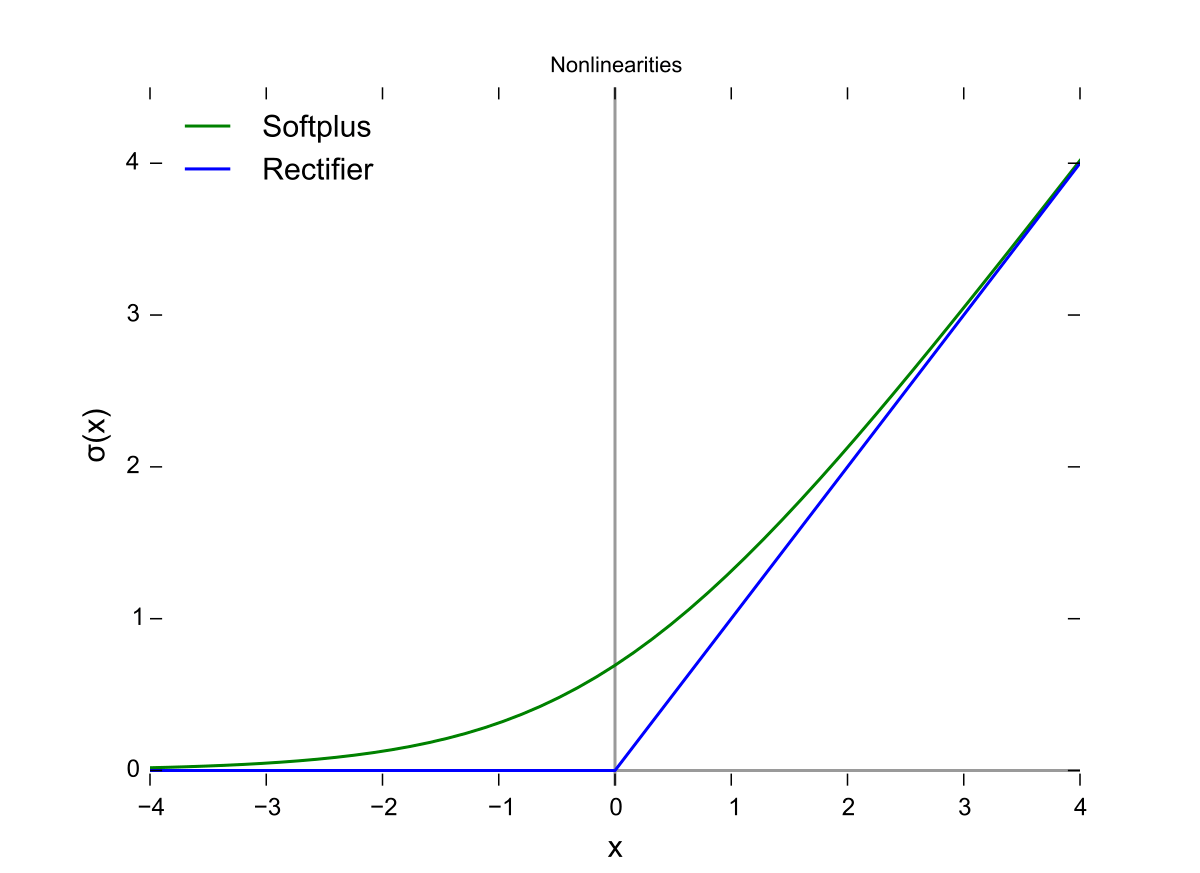


Figure 25. Rectifier Function[22]

The rectifier function has the following formula:

Φ(x)= max(x,0)

Formula 6. Rectifier function formula

When an output is predicted the predicted output is compared with the expected output on the basis of a cost function which tells the amount of error in prediction. The cost function is then fed back to the neural network and the weights are adjusted accordingly. The aim is to minimize the cost function.

The cost function is minimized using Gradient Descent where all rows are plugged into the neural network together and the cost function is calculated and the weights are adjusted. However, it requires the cost function to be convex. However, if the cost function is not convex local minimum may be formed instead of maximum one. In this case, Stochastic gradient descent is applied where the rows are fed one by one to the neural network, the cost function is calculated individually and the weights are adjusted individually. This helps to avoid the problem of local minimum. The classifier has taken the batch size of 10 i.e. 10 rows are fed into the artificial neural network at once and 200 epochs. An epoch is an iteration over the whole data. In addition to this the optimizer is set to adam, metrics is evaluated on the basis of accuracy and loss is set to binary\_crossentropy due to only 2 possible outputs i.e. 1 or 0.

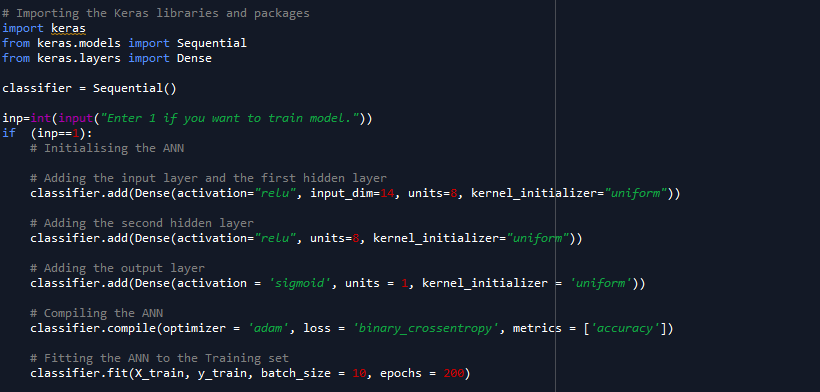


Figure 26. Implementing Artificial Neural Networks

1. K-Fold Cross Validation and Grid Search

A model trained on one training set may give a different accuracy as compared to a model trained on some other training set. On making a cross validation model, accuracy of every combination is returned. Various combinations of parameters are chosen and K-Fold Cross Validation and Grid search are applied on them to find the best combination of parameters. Grid search can only be applied on trained models. The previously trained models are trained again on the basis of the new chosen set of parameters to optimize the results.

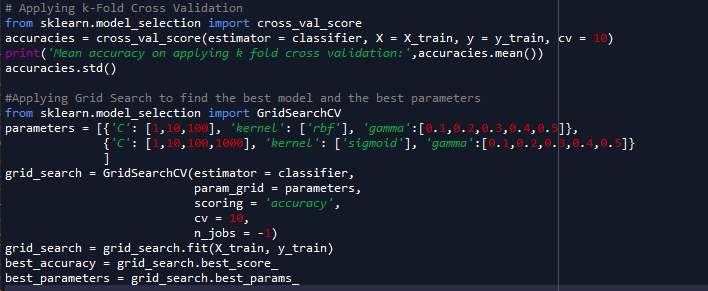


Figure 27. Applying K-Fold Cross Validation and Grid Search for Kernel SVM

1. Accuracy Comparison and Prediction

Accuracies of all the implemented models are compared to obtain the most accurate model. Further, data entered by the user in the desktop application is used to predict how likely is it for the patient to have a coronary heart disease in the next ten years.

**CHAPTER 4: RESULT**

A set of classification algorithms are trained on the training dataset to create a trained model. The model is then tested on the testing dataset and the predicted result is then compared with the expected result of the testing set. K-Fold Cross Validation is also applied on the training set by dividing the dataset into 10 sets and running the algorithm individually on each set. The mean accuracy on applying K-Fold Cross Validation is also recorded.

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Initial Accuracy** | **Mean Accuracy**  **(K-Cross Validation)** | **Final Accuracy** |
| **Classification Algorithms** | | | |
| Logistic Regression | 85.47% | 85.31% | 85.47% |
| Kernel SVM | 84.71% | 85.03% | 85.00% |
| K-Nearest Neighbour | 84.00% | 83.49% | 84.53% |
| Random Forest | 84.81% | 84.75% | 84.40% |
| Naïve Bayes | 81.89% | 82.51% | 81.89% |
| Decision Tree | 77.26% | 83.08% | 81.51% |
| **Deep Learning** | | | |
| Artificial Neural Network | - | 84.86% | 85.09% |

Table 2. Comparison of Accuracy

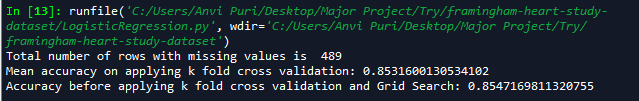


Figure 28a. Accuracy of Logistic Regression model



Figure 28b. Accuracy of Logistic Regression model after k fold cross validation and Grid Search

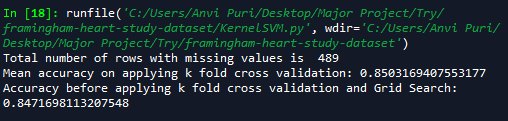


Figure 29a. Accuracy of Kernel SVM model



Figure 29b. Accuracy of Kernel SVM model after k fold cross validation and Grid Search



Figure 30a. Accuracy of Random Forest model



Figure 30b. Accuracy of Random Forest model after k fold cross validation and Grid Search

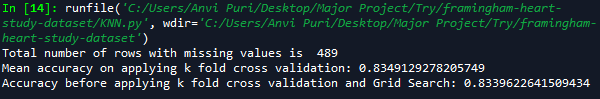


Figure 31a. Accuracy of K-Nearest Neighbour model



Figure 31b. Accuracy of K-Nearest Neighbour model after k fold cross validation and Grid Search

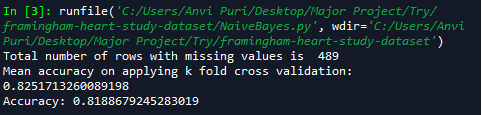


Figure 32. Accuracy of Naïve Bayes model

https://cdn.discordapp.com/attachments/529627191110139907/577187368738488326/unknown.png

Figure 33a. Accuracy of Decision Tree model



Figure 33b. Accuracy of Decision Tree model after k fold cross validation and Grid Search



Figure 34. Accuracy of Artificial Neural Network model

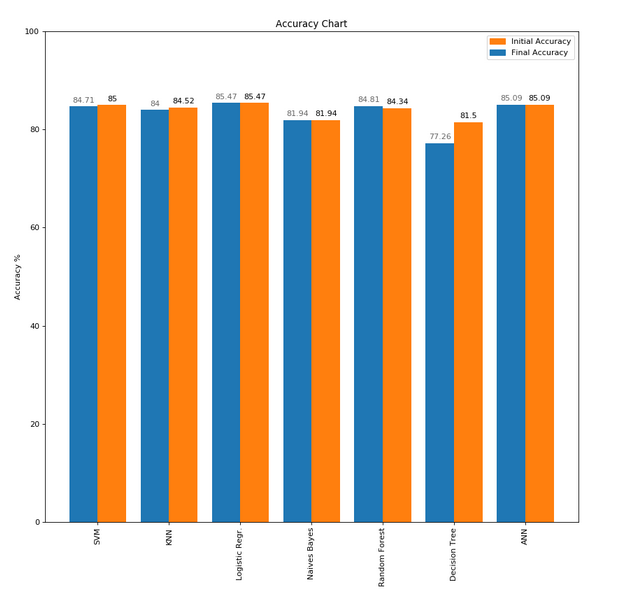


Figure 35. Accuracy Comparison Chart

**CHAPTER 5: ANALYSIS AND DISCUSSION OF RESULTS**

Table 2 shows a comparison of the final accuracies of the various implemented machine learning algorithms. The comparison shows the accuracy of each algorithm which can help decide which accuracies are the best to implement in the system. The accuracy of the algorithm is calculated on the basis of comparison of the testing dataset outputs and the predicted outputs. The dataset is divided into the training and testing datasets. The machine learning model is built upon training the algorithm on the training dataset. The model is then used to predict result of input attributes of the testing dataset as the predicted outputs which is then compared with outputs of the testing dataset. The results that have been obtained in the project have been analyzed and explained as follows:

* The model based on Logistic Regression shows the highest accuracy of 85.47% as shown in Figure 28a. On applying K Fold Cross Validation, the mean accuracy of the model decreases to 85.31%.
* The model based on Kernel SVM shows an accuracy of 84.71% as shown in Figure 29a. On applying K Fold Cross Validation, the mean accuracy of the model increases to 85.03%.
* The model based on Random Forest shows an accuracy of 84.81% as shown in Figure 30a. On applying K Fold Cross Validation, the mean accuracy of the model decreases to 84.75%.
* The model based on K-Nearest Neighbour shows an accuracy of 84.34% as shown in Figure 31a. On applying K Fold Cross Validation, the mean accuracy of the model increases to 83.50%.
* The model based on Naïve Bayes shows an accuracy of 81.89% as shown in Figure 32. On applying K Fold Cross Validation, the mean accuracy of the model increases to 82.52%.
* The model based on Decision Tree shows an accuracy of 77.26% as shown in Figure 33a. On applying K Fold Cross Validation, the mean accuracy of the model increases to 83.08%.
* A feed forward Artificial Neural Network algorithm is trained on the training set. When tested on the testing set, the model shows an accuracy of 85.09% as shown in Figure 34. On applying K Fold Cross Validation, the mean accuracy of the model increases to 84.86 (+/-0.33) %.
* Grid Search is applied on every model by selecting various combinations of values of parameters as well as various combinations of parameters that can occur in the algorithm. Grid Search in turn selects the best set of values of parameters as well as the best combination of parameters i.e. the combinations which will give the highest accuracy and performance for every model. The parameters are then used to retrain the model on the training set to make a more optimal model. However, when we applied Grid Search on the algorithms, it was observed that in some cases the accuracy increased but in others, the accuracy decreased.
  + The accuracy of the Logistic Regression model remained the same (as shown in Figure 28b) even after applying Grid Search on the parameters of the model.
  + The accuracy of the Kernel SVM model decreased to 85% (as shown in Figure 29b) after applying Grid Search on the parameters of the model.
  + The accuracy of the Random Forest model decreased to 84.40% (as shown in Figure 30b) after applying Grid Search on the parameters of the model.
  + The accuracy of the K-Nearest Neighbour model increased to 84.53% (as shown in Figure 31b) after applying Grid Search on the parameters of the model.
  + The accuracy of the Decision Tree model increased to 81.51% (as shown in Figure 33b) after applying Grid Search on the parameters of the model.
  + Grid search is not applied for the Naïve Bayes algorithm as there is not hyper parameter that can be changed or tuned in the classifier. Thus, the accuracy of the algorithm stays the same.

The analysis of the results led to the conclusion that Logistic Regression algorithm had the highest accuracy of 85.47% which was followed by Artificial Neural Networks with an accuracy of 85.09%. The third and fourth highest accuracy were that of Kernel SVM and K-Nearest Neighbour standing at 85% and 84.53% respectively. This is followed by Random Forest algorithm with an accuracy of 84.40% and Naïve Bayes algorithm with an accuracy of 81.89%. The lowest accuracy is that of Decision Tree algorithm i.e. 81.51%.

Once the results are analyzed and improved a Graphical User Interface is made for the patients in the form of a Desktop application to predict whether they will have a coronary heart disease within the next 10 years. They can also print the results in the form of a pdf file and check the accuracy of each algorithm.

**CHAPTER 6: CONCLUSION**

Heart Diseases are one of the few main causes of deaths around the world. People realize that they may be suffering from a heart disease only when it is too late to prevent it and the only solution that remains is to treat the heart disease. However, with advancing technologies, machine learning makes it possible to predict a heart disease based on various factors related to the health and habits of the patient. This gives the patient the advantage of changing their habits and taking precautions in advance so that they can try to prevent the heart disease from occurring. This project thus aimed to calculate accuracies of various algorithms for prediction of coronary heart diseases so that a suitable system could be built which could be used by the patients for heart disease prediction.

All the machine learning algorithms which consist of classification and deep learning algorithms were implemented for a dataset which consisted of various matrix of features like gender, systolic blood pressure, diastolic blood pressure, cholesterol, heart rate, body mass index etc. and a dependent variable which told whether the patient suffered from coronary heart disease within 10 years of collecting the inputs. Logistic Regression, Support Vector Machine, K-Nearest Neighbour, Random Forest, Decision Tree, Naïve Bayes and Artificial Neural Network algorithms were implemented.

The respective accuracies of the implemented machine learning algorithms were compared. It was found that in this stage Logistic Regression had the highest accuracy with 85.47%. Then K-Fold Cross Validation and Grid Search were applied for all the algorithms, which had varying effects on each algorithm. Decision Tree saw the maximum increase. The accuracies of Kernel-SVM & Random Forest actually decreased. Again, Logistic regression had the highest accuracy but Artificial Neural Networks surpassed it in some repeated simulations but due to its varying accuracy was not considered to be the best. The application successfully utilizes the information provided the dataset to give out a prediction.

The algorithms give a prediction based on the dataset’s fourteen attributes. In the dataset the education attribute was excluded as it is not a factor that effects Coronary Heart Diseases. A PyQt based Graphical User Interface was developed, which works like a tool that provides an easy access to an end user to get a prediction easily. The user can enter their data get a prediction as a message and if they want get a report in the form of a PDF based upon the data they entered. They can also check the accuracies of all the algorithms from the tool and are even provided with charts made using Matplotlib which provides further information for an easier decision. Further the patient can also print the prediction as a report in form of a pdf file. This application reduces the need for a profession to provide a diagnosis.

**CHAPTER 7: FUTURE WORK**

**7.1 Future Scope:**

* The current project focusses on coronary heart disease prediction specifically. However, the algorithms can be further improved upon to work on a more generalized dataset so as to further increase applicability of the system. It can be used to predict not only coronary heart diseases but also various other disease like cancer detection, etc.
* This project provides a Graphical User Interface in form of a desktop application where patients can input their required data and get a prediction for coronary heart disease accordingly. This system can be incorporated in the workings of medical labs to provide faster results.
* The scope of desktop application might be limited as web applications and mobile applications are used more in comparison. Thus, the project can be incorporated in a mobile application and a web application for on the go prediction where the patient does not want to go through the effort of downloading a desktop application.
* There can be an application developed which incorporates data from health devices to automate the process of entering information and then provide an on-the-go prediction accordingly. Devices such as Fitbit and other health watches already communicate with smartphones to provide relevant data and that data can be incorporated with this process further increasing ease of usage.
* A machine learning algorithm performs better if the dataset that it has trained on is large. Since the dataset of this project contains limited observations, it can be made dynamic where new entries are added to the dataset upon which the algorithms can be further trained. This will make the dataset larger and thus would enable the algorithms to have more observations to train on. This might result in the algorithms giving higher accuracy.

**7.2 Limitations:**

* Available datasets are exceedingly varied in their attributes so it is difficult to develop an application for already existing dataset.
* The end user might not have all the required information for the application to work.
* Entry of incorrect data could lead to a wrong prediction and if information were to be dynamically added to dataset it would ambiguously affect the accuracies of the algorithms.

**REFERENCES**

1. T.Nagamani, S.Logeswari, B.Gomathy. (2019). Disease Prediction using Data Mining with Mapreduce Algorithm. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)ISSN: 2278-3075, Volume-8 Issue-3*
2. https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/
3. https://www.dezyre.com/article/top-10-machine-learning-algorithms/202
4. G. E. Sakr, I. H. Elhajj and H. A. Huijer. (2010). Support vector machines to define and detect agitation transition. *IEEE Transactions on Affective Computing, vol. 1, pp. 98-108*
5. Seyedamin Pouriyeh, Sara Vahid, Giovanna Sannino, Giuseppe De Pietro, Hamid Arabnia, Juan Gutierrez. (2017)*.* A Comprehensive Investigation and Comparison of Machine Learning Techniques in the Domain of Heart Disease. *22nd IEEE Symposium on Computers and Communication (ISCC 2017)*
6. C.-L. Chang and C.-H. Chen. (2009). Applying decision tree and neural network to increase quality of dermatologic diagnosis. *Expert Syst. Appl., vol. 36, no. 2, Part 2, pp. 4035–4041*
7. Saangyong Uhmn, Dong -Hoi Kim, Jin Kim, Sung Won Cho, Jae Youn Cheong. (2007). Chronic Hepatitis Classification Using SNP Data and Data Mining Techniques. *Frontiers in the Convergence of Bioscience and Information Technologies, FBIT.81-86*
8. S. Ghumbre, C. Patil, and A. Ghatol. (2011). Heart disease diagnosis using support vector machine. *International Conference on Computer Science and Information Technology (ICCSIT), pp. 84-88*
9. S. Palaniappan and R. Awang. (2008). Intelligent heart disease prediction system using data mining techniques. *International Journal of Computer Science and Network Security, pp. 343-350*
10. Edward Choi, Andy Schuetz, Walter F Stewart,and Jimeng Sun.(2017). Using recurrent neural network models for early detection of heart failure onset*. Journal of the American Medical Informatics Association*, *24(2), 361–370*
11. Chen, Min & Yixue, Hao & Hwang, Kai & Wang, Lu & Wang, Lin. (2017). Disease Prediction by Machine Learning Over Big Data From Healthcare Communities. *IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2694446*
12. Isra’a Ahmed Zriqat, Ahmad Mousa Altamimi, Mohammad Azzeh. (2016). A Comparative Study for Predicting Heart Diseases Using Data Mining Classification Methods. *International Journal of Computer Science and Information Security (IJCSIS), Vol. 14, No. 12*
13. Zhang Et Al. (2012). Studies on Application of Support Vector Machine in Diagnose of Coronary Heart Disease. *Electromagnetic Field Problems and Applications Sixth International Conference (ICEF), Dalian,IEEE*
14. https://www.kaggle.com/amanajmera1/framingham-heart-study-dataset
15. https://archive.ics.uci.edu/ml/datasets/heart+disease
16. https://www.saedsayad.com/support\_vector\_machine.htm
17. https://www.saedsayad.com/logistic\_regression.htm
18. https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn
19. https://www.displayr.com/how-is-splitting-decided-for-decision-trees/
20. https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d
21. https://www.extremetech.com/extreme/215170-artificial-neural-networks-are-changing-the-world-what-are-they
22. https://en.wikipedia.org/wiki/Rectifier\_(neural\_networks)