International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 8 Aug 2021, pp: 1198-1212 www.ijaem.net ISSN: 2395-5252

Prediction of Heart Stroke Disease Stages using ML Techniques

V.M.Sivagami,N.Devi

[1], *Associate professor,Sri Venkateswara College of Engineering, Sriperumbudur,Tamilnadu.*
[2],*Assistant Professor,Sri Venkateswara College of Engineering, Sriperumbudur,Tamilnadu.*

Submitted: 05-08-2021 Revised: 18-08-2021 Accepted: 20-08-2021

ABSTRACT: This research work is about to predict the occurrence of a heart stroke disease among patients .Predictive analytical techniques for heart stroke using machine learning model is applied on the given hospital dataset. The main objective is to design predictive analytics model which diagnoses heart stroke stages of patients. In addition, the performance of the model was applied on the given hospital dataset with evaluation of classification report and identify the confusion matrix using supervised machine learning algorithms. Also, the behaviour of the proposed model was evaluated against performance of various machine learning algorithms from the given healthcare department dataset and with evaluation classification reports were made. Identification of confusion matrix and the categorization of data from priority and the result shows that the effectiveness of GUI based proposed machine learning algorithm technique can be compared with best accuracy metrics precision, Recall and F1 Score.

KEYWORDS: Fast Co-relation Filtering Algorithm(FCBF), Fast Co-relation Filtering Algorithm(SVM), K-Nearest Neighbour,Confusion Matrix,Heart Stroke.

I. INTRODUCTION

Heart attacks diseases are now happening among many people. Medical field is conducting different surveys on heart diseases and gather information to analyze the reason for heart attacks among patients . Data Mining and machine learning algorithms are applied to predict Stroke based on patient treatment history and health data Many works have been applied data mining techniques to pathological data or medical profiles for prediction of stroke. Some approaches try to do prediction on control and progression of disease. Machine learning is to predict the future from past data. Machine learning (ML) is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of Computer Programs that can change when exposed to new data and the basics of Machine Learning, implementation of a simple machine learning algorithm using python. Machine learning can be roughly separated in to three categories. There are supervised learning, unsupervised learning and reinforcement learning. Supervised learning. This work aims to observe which features are most helpful in predicting the patient diseases based on the attributes and finding the symptoms of heart diseases to see the general trends that may help us in model selection and hyper parameter selection. To achieve this, we used machine learning classification methods to fit a function that can predict the discrete class of new input. By applying the fundamental concepts of machine learning from an available dataset and the evaluation is carried out and the interpreted results justify the decision made for diagnosing the heart stroke among patients.

In [1], Radar technology is used as a burden-free method for continuous heart sound monitoring, which can be used to detect cardiovascular diseases. This paper gives a comprehensive study on recorded sound signals from radar, a bistatic radar system built and installed at the university hospital. Under medical supervision, heart sound data were recorded from 30 healthy test subjects.Different state-of-the-art pattern classification algorithms were evaluated for the task of automated signal quality determination and the most promising one was optimized and evaluated using leave-one-subject-out cross validation. This paper introduced an ensemble classifier that is able to perform automated signal quality determination of radar-recorded heart sound signals with a high accuracy and the method enables contactless and continuous heart sound monitoring for the detection of cardiovascular diseases

In [2] , developed a cloud-computing platform monitored by physicians, which can receive 12-lead ECG records and send back diagnostic reports to users The objective was to lessen the physicians' workload, they implemented an analysis algorithm that can identify abnormal heart rate, irregular heartbeat, abnormal amplitude, atrial fibrillation and abnormal ECG in it. A large number of testing samples were used to evaluate performance and it received and analyzed ECG records in real time.

In paper [3] they proposed energy management system for human-electric hybrid vehicles that used an optimal control approach to regulate the heart rate of the cyclist. The system consists of a control stage and a planning stage. In the control stage, a model predictive controller regulates the heart rate by changing the motor power and gear ratio to maintain a user-defined exertion while considering constraints. The planning stage processes a priori information about the user and the route to estimate the power demand during different sections of the trip and to calculate the optimal motor power for each section. This system helped people with limited physical capabilities to safely engage in physical activity.

In [4] the authors analyzed the influence of social anxiety on the autonomic nerve control of the heart in two social exposure events: public speaking and thesis defending. In an experiment of public speaking, 59 human subjects were tested, and 11 conventional heartbeat measures and a heartbeat measure named the range of local Hurst exponents (RLHE) were evaluated for their capabilities to reveal the onset of social anxiety.With the combination of three conventional features and the RLHE feature, a support vector machine classifier obtained true positive rate and true negative rate of 84.88 and 97.29 percent in the five-fold cross validation process of binary classification between high anxiety status and low anxiety status; the classifier also realized a generalization accuracy of 81.82 percent in detecting the high anxiety status in the thesis defense method.

Heart rate (HR) estimation and monitoring is of great importance to determine a person's physiological and mental status. Recently, it has been demonstrated that HR can be remotely retrieved from facial video-based photoplethysmographic signals captured using professional or consumer-level cameras. Many efforts have been made to improve the detection accuracy of this noncontact technique. In [5], researchers presented a timely, systematic survey on such video-based remote HR measurement approaches, with a focus on recent advancements that overcome dominating technical challenges arising from illumination variations and motion artifacts.

In[6] machine learning algorithms are applied to compute predictive computational techniques for heart stroke on a given hospital dataset. Atrial fibrillation is a significant risk factor for cardiac attack in patients, and it shares many of the same factors that predict stroke. When a dataset is analysed using a controlled machine learning algorithm, variables such as variable recognition, univariate analysis, bivariate and multivariate analysis, missed value therapies, mathematical methods, and so on are all recorded. The aim of the predictive analytics model is to recognise the various stages of heart stroke in patients. Discuss the output of the provided hospital dataset, as well as the evaluation of the classification study and the uncertainty matrix.

The goal of this work [7] was to use machine learning algorithms to classify patient-reported outcomes (PROs) using activity tracker data in a cohort of patients with stable ischemic heart disease (SIHD). A population of 182 patients with SIHD were monitored over a period of 12 weeks. Each subject received a Fitbit Charge 2 device to record daily activity data, and each subject completed eight Patient-Reported Outcomes Measurement Information Systems short form at the end of each week as a self-assessment of their health status

II PROPOSED SYSTEM

This research work focuses on the development of a graphical user interface (GUI) model for the prediction of stroke using different machine learning algorithms.The data used for this paper was taken from a Kaggle repository and from International Stroke trial Database. Database includes patient information, patient history, hospital details, Country, risk factors and symptoms. The data has various labelled features such as gender, age, if a person has hypertension or not, if a person has heart disease or not, if a person is married or not, work type, residential type, average glucose level, BMI, smoking status, if a person has a stroke or not. There are a total of fourteen features in which four are numeric, four alphanumeric and the rest categorical. .

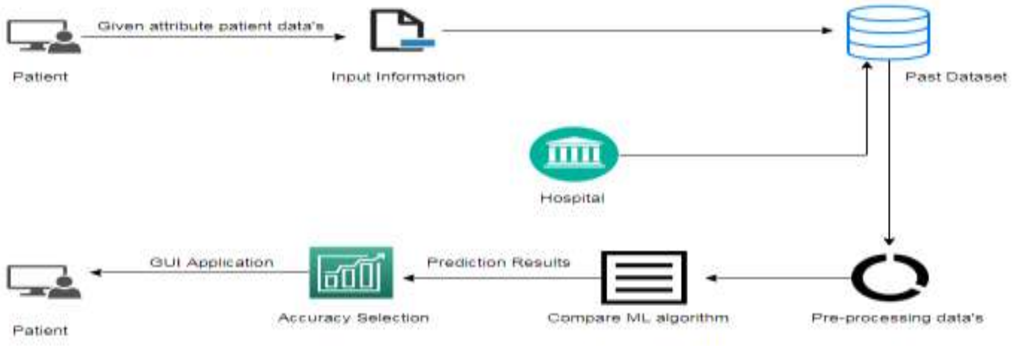


Figure.1 Heart Stroke Prediction Architecture Diagram

2.1 Classification of tasks

From the perspective of automatic learning, heart disease detection can be seen as a classification or clustering problem. On the other hand, we formed a model on the vast set of presence and absence of file data; we can reduce this problem to classification. For known families, this problem can be reduced to one classification only - having a limited set of classes, including the heart disease sample, it is easier to identify the right class, and the result would be more accurate than with clustering algorithms. In this section, the theoretical context is given on all the methods used in this research. For the purpose of comparative analysis, five Machine Learning algorithms are discussed. The different Machine Learning (ML) algorithms are K-Nearest Neighbour (KNN), Random Forest (RF), Support Vector Machine (SVM), Naïve Bayes and Artificial Neural Network (ANN). The reason to choose these algorithms is based on their popularity .

In the heart disease datasets, the number of features can reach up to tens of thousands; the heart disease dataset has 14 attributes. Since a large

In this paper, we applied machine learning algorithms on heart stroke dataset to predict heart diseases. Our goal was to compare different classification models and define the most efficient one. From all the tables above, different algorithms performed better depending upon the situation whether cross-validation, grid search, calibration and feature selection is used or not. Every algorithm has its intrinsic capacity to outperform other algorithm depending upon the situation. For example, Random Forest performs much better with a large number of datasets than when data is small while Support Vector Machine performs better with a smaller number of data sets. Performance of algorithms decreased after boosting in the data, which did not feature, selected while algorithms were performing better without boosting in feature selected data. This shows the necessity that the data should be feature selected before applying to boost.

number of irrelevant and redundant attributes are involved in these expression data, the heart disease classification task is made more complex. If complete data are used to perform heart disease classification, accuracy will not be as accurate, and calculation time and costs will be high. Therefore, the feature selection, as a pre-treatment step to machine learning, reduces sizing, eliminates unresolved data, increases learning accuracy, and improves understanding of results. The recent increase in the dimensionality of the data poses a serious problem to the methods of selecting characteristics with regard to efficiency and effectiveness. The FCBF's reliable method is adopted to select a subset of discriminatory features prior to classification, by eliminating attributes with little or no effect, FCBF provides good performance with full consideration of feature correlation and redundancy. In this document, we first standardized the data and then selected the features by FCBF in WEKA. The number of heart disease attributes increased from 14 to 7.

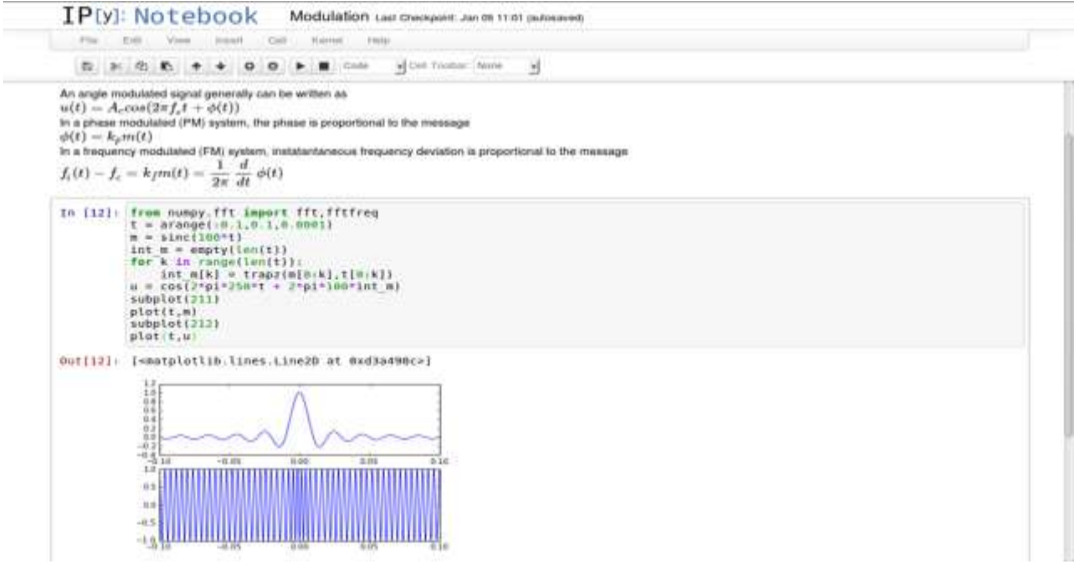
For the comparison of the dataset, performance metrics after feature selection, parameter tuning and calibration are used because this is a standard process of evaluating algorithms. The precision average value of the best performance without optimization it’s for SVM and NB with 83.6% than RF with 81.4%. These shows SVM and NB are performing on average, after optimized by we find the best performance of precision it’s for MLP with 84.2% than NB with 84%. In the last stage, we compared the different algorithms with the proposed optimized model. we find the best one is K-NN we find the best one is K-NN with 99.7 % than RF with 99.6%.

2.2Pre-Processing

Data pre-processing is a data mining technique which is used to transform the raw data in a useful and efficient format. The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc. The tuples are ignored. The missing values are filled either manually, by attribute mean or the most probable value.

2.3DataReduction

Since data mining is a technique that is used to handle huge amount of data. While working with huge volume of data, analysis became harder in such cases. In order to get rid of this, we uses data reduction technique. It aims to increase the storage efficiency and reduce data storage and analysis costs.



An angle modulated signal generally can be written as

$u(t) = A_c \cos(2\pi f_c t + \phi(t))$

In a phase modulated (PM) system, the phase is proportional to the message

$\phi(t) = k_p m(t)$

In a frequency modulated (FM) system, instatantaneous frequency deviation is proportional to the message

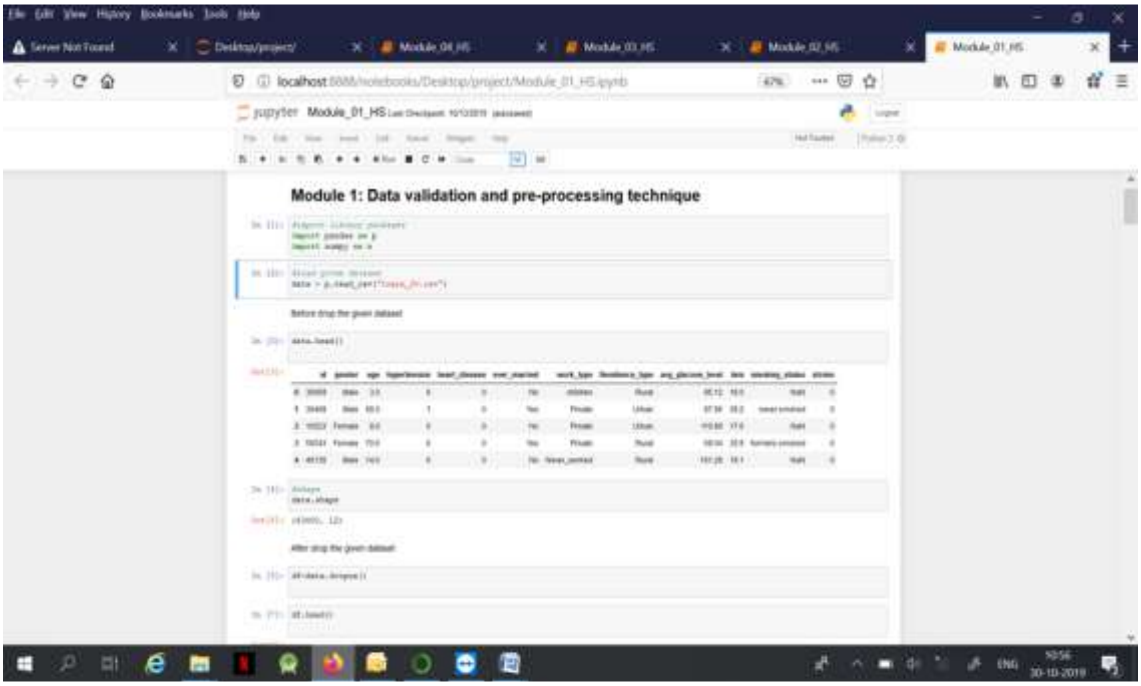
$f_i(t) - f_c = k_f m(t) = \frac{1}{2\pi}\frac{d}{dt}\phi(t)$

```
In [12]: from numpy.fft import fft,fftfreq
         t = arange(-0.1,0.1,0.0001)
         m = sinc(100*t)
         int_m = empty(len(t))
         for k in range(len(t)):
             int_m[k] = trapz(m[0:k],t[0:k])
         u = cos(2*pi*250*t + 2*pi*100*int_m)
         subplot(211)
         plot(t,m)
         subplot(212)
         plot(t,u)

Out[12]: [<matplotlib.lines.Line2D at 0xd3a498c>]
```

Figure 2 :IPython Notebook interface

23 Variable Identification Process / data validation process

Validation techniques in machine learning are used to get the error rate of the Machine Learning (ML) model, which can be considered as close to the true error rate of the dataset. If the data volume is large enough to be representative of the population, you may not need the validation techniques. However, in real-world scenarios, to work with samples of data that may not be a true representative of the population of given dataset. To finding the missing value, duplicate value and description of data type whether it is float variable or integer. The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper parameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. It as machine learning engineers uses this data to fine-tune the model hyper parameters. Data collection, data analysis, and the process of addressing data content, quality, and structure can add up to a time-consuming to-do list. During the process of data identification, it helps to understand your data and its properties; this knowledge will help you choose which algorithm to use to build your model. For example, time series data can be analyzed by regression algorithms; classification algorithms can be used to analyze discrete data.



Module 1: Data validation and pre-processing technique

```
#import library packages
import pandas as p
import numpy as n
```

```
#load given dataset
data = p.read_csv("train_2v.csv")
```

Before drop the given dataset

```
data.head()
```

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	30669	Male	3.0	0	0	No	children	Rural	95.12	18.0	NaN	0
1	30468	Male	58.0	1	0	Yes	Private	Urban	87.96	39.2	never smoked	0
2	16523	Female	8.0	0	0	No	Private	Urban	110.89	17.6	NaN	0
3	56543	Female	70.0	0	0	Yes	Private	Rural	69.04	35.9	formerly smoked	0
4	46136	Male	14.0	0	0	No	Never_worked	Rural	161.28	19.1	NaN	0

```
#shape
data.shape
```

```
(43400, 12)
```

After drop the given dataset

```
df=data.dropna()
```

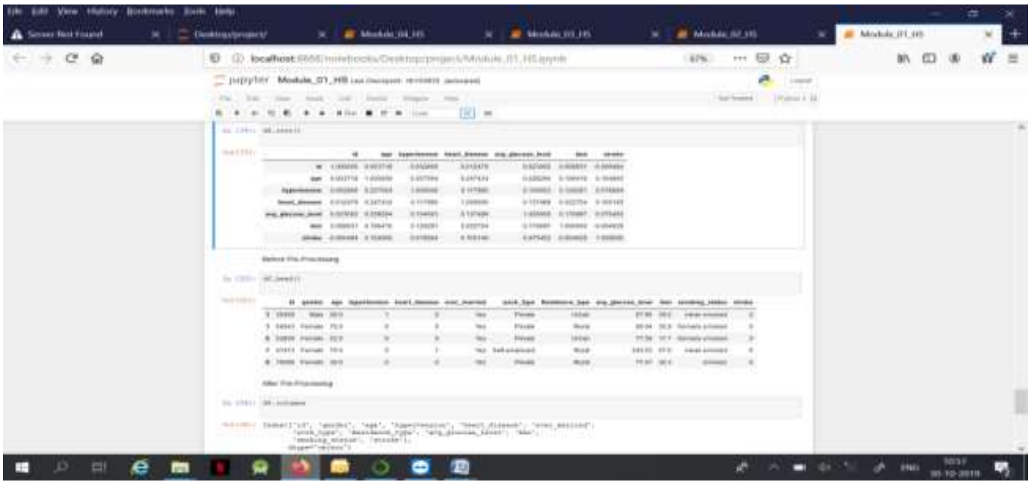
```
df.head()
```

Figure 3. DataValidation/Cleaning/Preparing Process

Importing the library packages with loading given dataset. To analysing the variable identification by data shape, data type and evaluating the missing values, duplicate values. A validation dataset is a sample of data held back from training your model that is used to give an estimate of model skill while tuning model's and procedures that you can use to make the best use of validation and test datasets when evaluating your models. Data cleaning / preparing by rename the given dataset and drop the column etc. to analyze the uni-variate, bi-variate and multi-variate process. The steps and techniques for data cleaning will vary from dataset to dataset. The primary goal of data cleaning is to detect and remove errors and anomalies to increase the value of data in analytics and decision making.

2.4 Data Pre-processing

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Pre-processing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis. To achieving better results from the applied model in Machine Learning method of the data has to be in a proper manner. Some specified Machine Learning model needs information in a specified format; for example, Random Forest algorithm does not support null values. Therefore, to execute random forest algorithm null values have to be managed from the original raw data set. And another aspect is that data set should be formatted in such a way that more than one Machine Learning and Deep Learning algorithms are executed in given dataset.



Before Pre-Processing

After Pre-Processing

Figure 4: Data before pre-processing and data after pre-procssing

2.5 Exploration data analysis of visualization

Data visualization is an important skill in applied statistics and machine learning. Statistics does indeed focus on quantitative descriptions and estimations of data. Data visualization provides an important suite of tools for gaining a qualitative understanding. This can be helpful when exploring and getting to know a dataset and can help with identifying patterns, corrupt data, outliers, and much more. With a little domain knowledge, data visualizations can be used to express and demonstrate key relationships in plots and charts that are more visceral and stakeholders than measures of association or significance. Data visualization and exploratory data analysis are whole fields themselves and it will recommend a deeper dive into some the books mentioned at the end.

Sometimes data does not make sense until it can look at in a visual form, such as with charts and plots. Being able to quickly visualize of data samples and others is an important skill both in applied statistics and in applied machine learning. It will discover the many types of plots that you will need to know when visualizing data in Python and how to use them to better understand your own data. Many machine learning algorithms are sensitive to the range and distribution of attribute values in the input data. Outliers in input data can skew and mislead the training process of machine learning algorithms resulting in longer training times, less accurate models and ultimately poorer results.

Even before predictive models are prepared on training data, outliers can result in misleading representations and in turn misleading interpretations of collected data. Outliers can skew the summary distribution of attribute values in descriptive statistics like mean and standard deviation and in plots such as histograms and scatterplots, compressing the body of the data. Finally, outliers can represent examples of data instances that are relevant to the problem such as anomalies in the case of fraud detection and computer security.It could not fit the model on the training data and can't say that the model will work accurately for the real data. For this, we must assure that our model got the correct patterns from the data, and it is not getting up too much noise. Cross-validation is a technique in which we train our model using the subset of the data-set and then evaluate using the complementary subset of the data-set.

The three steps involved in cross-validation are as follows:

1. Reserve some portion of sample data-set.
2. Using the rest data-set train the model.
3. Test the model using the reserve portion of the data-set.

Advantages of train/test split

1. This runs K times faster than Leave One Out cross-validation because K-fold cross-validation repeats the train/test split K-times.
2. Simpler to examine the detailed results of the testing process.

Advantages of cross-validation

1. More accurate estimate of out-of-sample accuracy.
2. More "efficient" use of data as every observation is used for both training and testing.

2.7 Training the Dataset

The first line imports iris data set which is already predefined in sklearn module and raw data set is basically a table which contains information about various varieties.For example, to import any algorithm and train_test_split class from sklearn and numpy module for use in this program.To encapsulate load_data() method in data_dataset variable. Further divide the dataset into training data and test data using train_test_split method. The X prefix in variable denotes the feature values and y prefix denotes target values.This method divides dataset into training and test data randomly in ratio of 67:33 / 70:30. Then we encapsulate any algorithm.In the next line, we fit our training data into this algorithm so that computer can get trained using this data. Now the training part is complete.

2.6 Testing the Dataset

Now, the dimensions of new features in a numpy array called ‘n’ and it want to predict the species of this features and to do using the predict method which takes this array as input and spits out predicted target value as output.So, the predicted target value comes out to be 0. Finally to find the test score which is the ratio of no. of predictions found correct and total predictions made and finding accuracy score method which basically compares the actual values of the test set with the predicted values.

Figure 5.Pie Chart for the DataSet

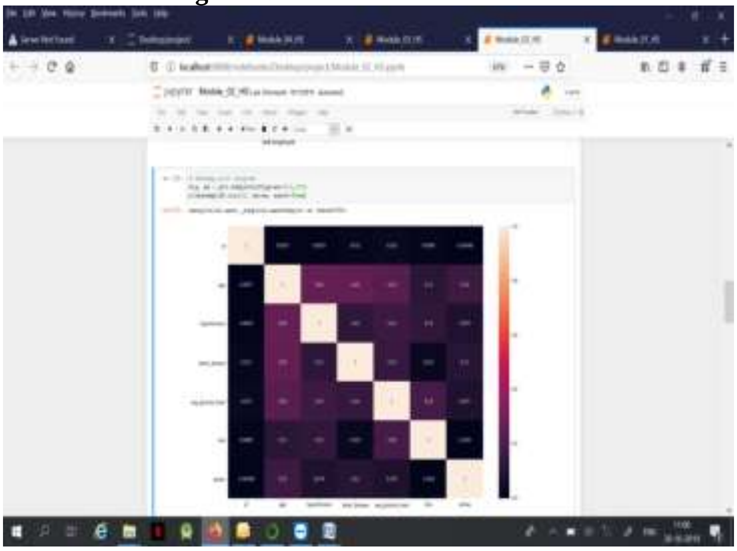


Figure 6: Correlation Matrix for the data set

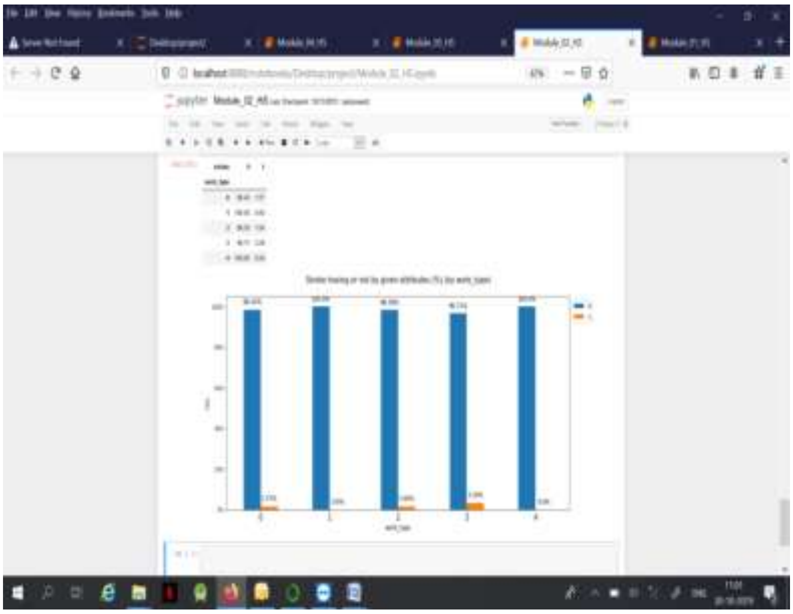


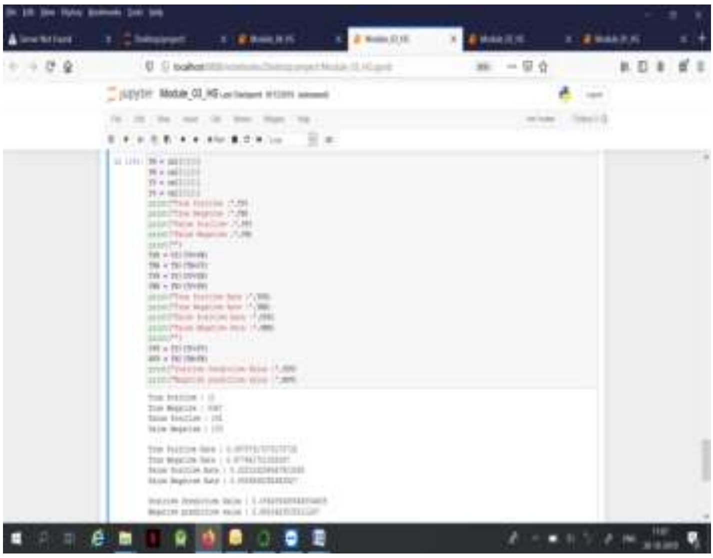
Figure 7: A bar chart for the data set

2.7 Logistic Regression

It is a statistical method for analysing a data set in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable . The goal of logistic regression is to find the best fitting model to describe the relationship between the dichotomous characteristic of interest and a set of independent variables. Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 or 0 no, failure, etc.

2.8 Decision Tree

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. A decision node has two or more branches and a leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data. Decision tree builds classification or regression models in the form of a tree structure. It utilizes an if-then rule set which is mutually exclusive and exhaustive for classification. The rules are learned sequentially using the training data one at a time. Each time a rule is learned, the tuples covered by the rules are removed. This process is continued on the training set until meeting a termination condition. It is constructed in a top-down recursive divide-and-conquer manner. All the attributes should be categorical. Otherwise, they should be discretized in advance. Attributes in the top of the tree have more impact towards in the classification and they are identified using the information gain concept. A decision tree can be easily over-fitted generating too many branches and may reflect anomalies due to noise or outliers.



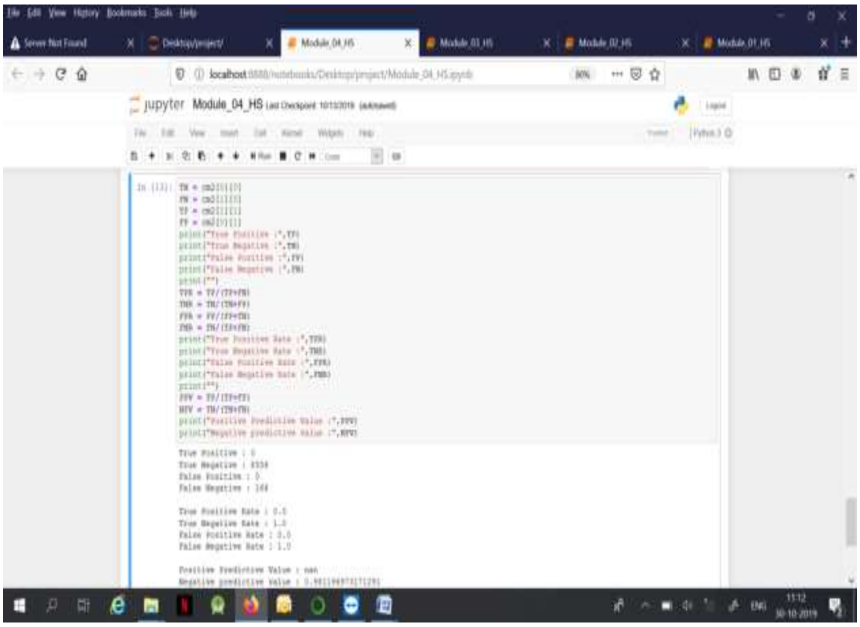
```
In [13]: [illegible]
```

.

Figure 8: Output for Decision Tree Classifier

2.9 Support Vector Machines (SVM)

A classifier that categorizes the data set by setting an optimal hyper plane between data. I chose this classifier as it is incredibly versatile in the number of different kernelling functions that can be applied and this model can yield a high predictability rate. Support Vector Machines are perhaps one of the most popular and talked about machine learning algorithms.



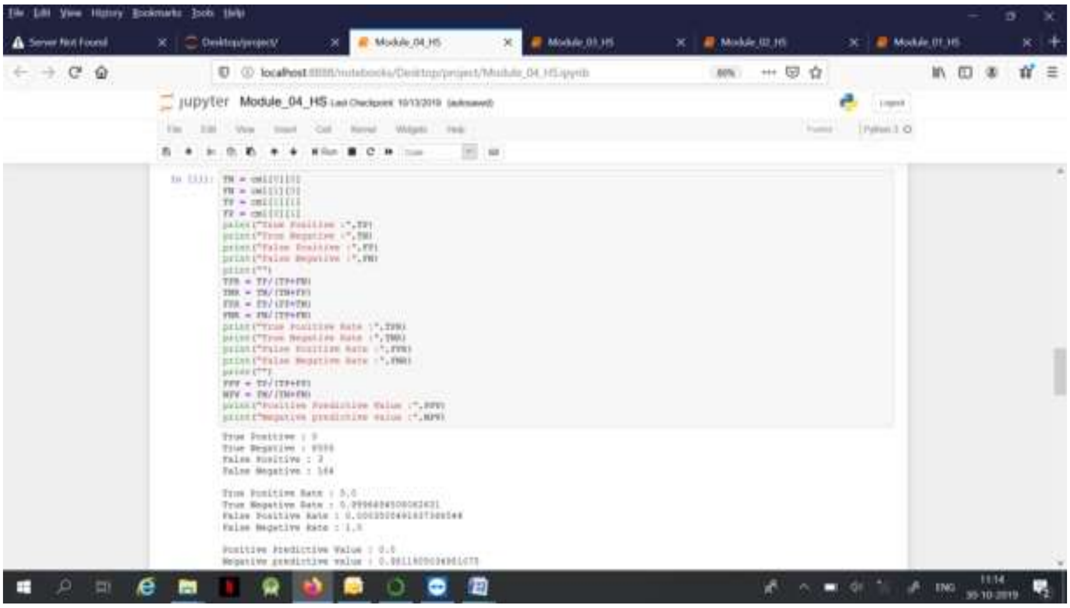
```
In [13]: TN = cm2[0][0]
         FN = cm2[1][0]
         TP = cm2[1][1]
         FP = cm2[0][1]
         print("True Positive :",TP)
         print("True Negative :",TN)
         print("False Positive :",FP)
         print("False Negative :",FN)
         print("")
         TPR = TP/(TP+FN)
         TNR = TN/(TN+FP)
         FPR = FP/(FP+TN)
         FNR = FN/(TP+FN)
         print("True Positive Rate :",TPR)
         print("True Negative Rate :",TNR)
         print("False Positive Rate :",FPR)
         print("False Negative Rate :",FNR)
         print("")
         PPV = TP/(TP+FP)
         NPV = TN/(TN+FN)
         print("Positive Predictive Value :",PPV)
         print("Negative predictive Value :",NPV)

True Positive : 0
True Negative : 8338
False Positive : 0
False Negative : 168

True Positive Rate : 0.0
True Negative Rate : 1.0
False Positive Rate : 0.0
False Negative Rate : 1.0

Positive Predictive Value : nan
Negative predictive Value : 0.98[illegible]
```

Figure 9: Output for SVM algorithm

2.10 Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of over fitting to their training set. Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in a forest of trees, hence the name "Random Forest". The random forest algorithm can be used for both regression and classification tasks.

The following are the basic steps involved in performing the random forest algorithm. In case of a regression problem, for a new record, each tree in the forest predicts a value for Y (output). The final value can be calculated by taking the average of all the values predicted by all the trees in forest. Or, in case of a classification problem, each tree in the forest predicts the category to which the new record belongs. Finally, the new record is assigned to the category that wins the majority vote.



```
In [11]: TN = cm[illegible]
         FN = cm[illegible]
         TP = cm[illegible]
         FP = cm[illegible]
         print("True Positive :",TP)
         print("True Negative :",TN)
         print("False Positive :",FP)
         print("False Negative :",FN)
         print("")
         TPR = TP/(TP+FN)
         TNR = TN/(TN+FP)
         FPR = FP/(FP+TN)
         FNR = FN/(TP+FN)
         print("True Positive Rate :",TPR)
         print("True Negative Rate :",TNR)
         print("False Positive Rate :",FPR)
         print("False Negative Rate :",FNR)
         print("")
         PPV = TP/(TP+FP)
         NPV = TN/(TN+FN)
         print("Positive Predictive Value :",PPV)
         print("Negative predictive value :",NPV)

         True Positive : 0
         True Negative : 9596
         False Positive : 2
         False Negative : 164

         True Positive Rate : 0.0
         True Negative Rate : 0.[illegible]
         False Positive Rate : 0.[illegible]
         False Negative Rate : 1.0

         Positive Predictive Value : 0.0
         Negative predictive value : 0.[illegible]
```

Figure 10: Output for Random Forest

2.11 K-Nearest Neighbor (KNN)

K-Nearest Neighbor is a supervised machine learning algorithm which stores all instances correspond to training data points in n-dimensional space. When an unknown discrete data is received, it analyzes the closest k number of instances saved (nearest neighbors) and returns the most common class as the prediction and for real-valued data it returns the mean of k nearest neighbors. In the distance-weighted nearest neighbor algorithm, it weights the contribution of each of the k neighbors according to their distance using the following query giving greater weight to the closest neighbors.

Usually KNN is robust to noisy data since it is averaging the k-nearest neighbors. The k-nearest-neighbors algorithm is a classification algorithm, and it is supervised: it takes a bunch of labelled points and uses them to learn how to label other points. To label a new point, it looks at the labelled points closest to that new point (those are its nearest neighbors), and has those neighbors vote, so whichever label the most of the neighbors have is the label for the new point (the “k” is the number of neighbors it checks). Makes predictions about the validation set using the entire training set. KNN makes a prediction about a new instance by searching through the entire set to find the k “closest” instances. “Closeness” is determined using a proximity measurement (Euclidean) across all features.

2.12 Accuracy calculation

False Positives (FP): A person who will pay predicted as defaulter. When actual class is no and predicted class is yes. E.g. if actual class says this passenger did not survive but predicted class tells you that this passenger will survive.

False Negatives (FN): A person who default predicted as payer. When actual class is yes but predicted class in no. E.g. if actual class value indicates that this passenger survived and predicted class tells you that passenger will die.

True Positives (TP): A person who will not pay predicted as defaulter. These are the correctly predicted positive values which means that the value of actual class is yes and the value of predicted class is also yes. E.g. if actual class value indicates that this passenger survived and predicted class tells you the same thing.

True Negatives (TN): A person who default predicted as payer. These are the correctly predicted negative value which means that the value of actual class is no and value of predicted class is also no. E.g. if actual class says this passenger did not survive and predicted class tells you the same thing.

III EXPERIMENTAL STUDY

3.1 LOGISTIC REGRESSION

It is a statistical method for analysing a data set in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). The goal of logistic regression is to find the best fitting model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.).

3.1.1 IMPLEMENTATION

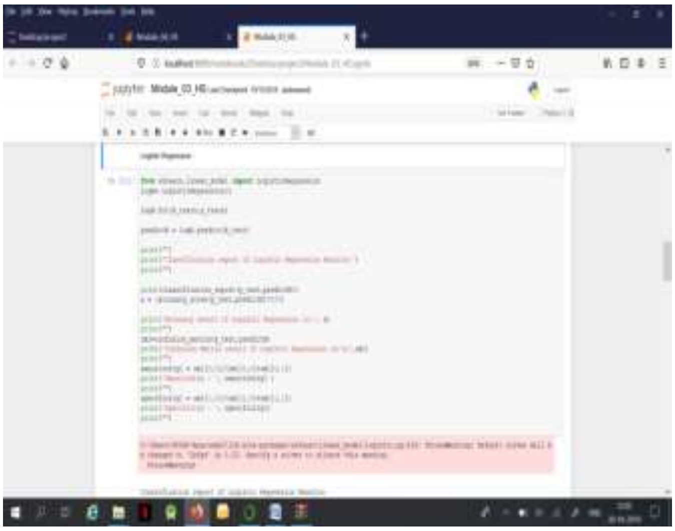


Figure 11 : Logistic Regression

3.1.2 OUTPUT

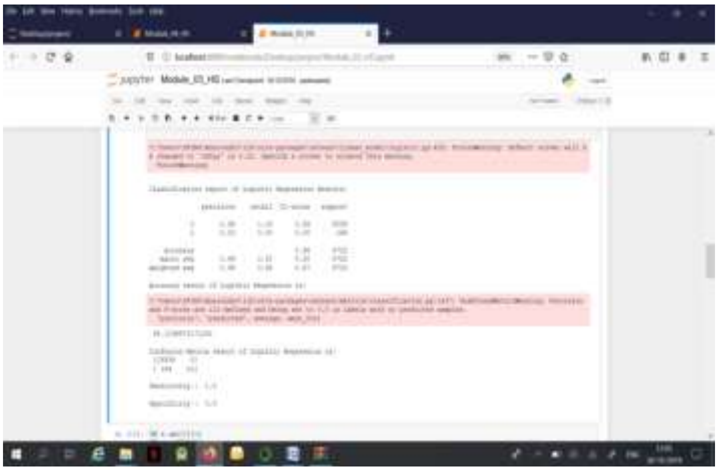


Figure 12 Classification report of logistic Regression Results

3.2 DECISION TREE ALGORITHM

It is one of the most powerful and popular algorithm. Decision-tree algorithm falls under the category of supervised learning algorithms.

3.2.1 IMPLEMENTATION

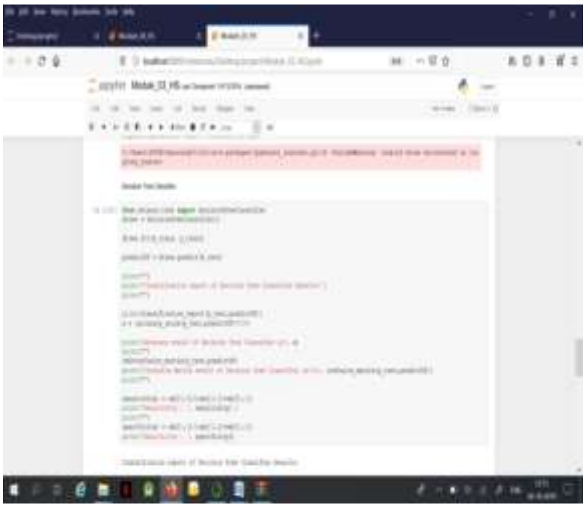


Figure 13 Decision Tree Classifier

3.2.2 OUTPUT

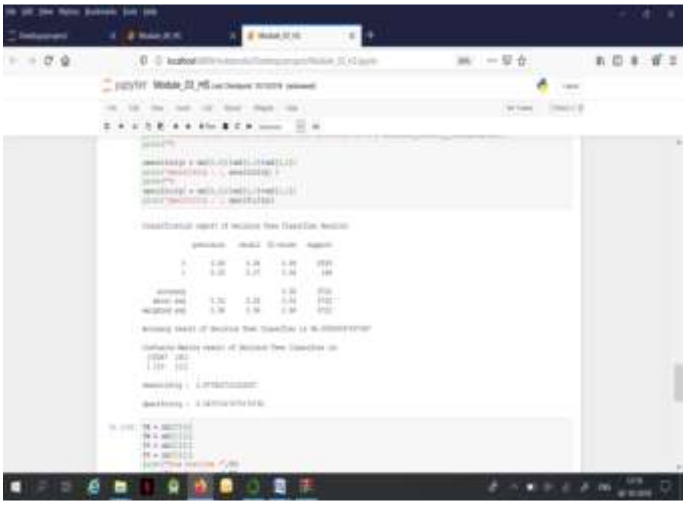
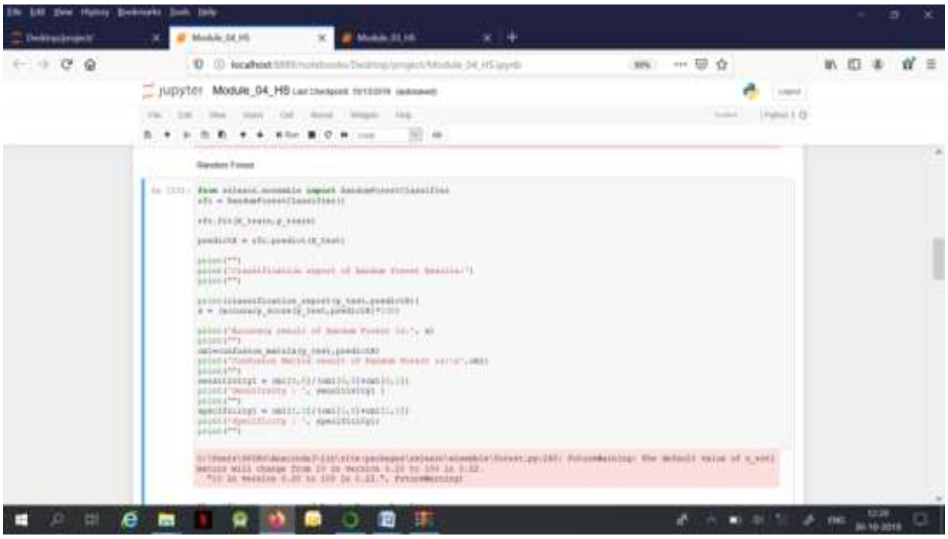


Figure 14 Classification report of Decision Tree Classifier results

3.3 RANDOM FOREST

Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in a forest of trees, hence the name "Random Forest". The random forest algorithm can be used for both regression and classification tasks.

3.3.1 IMPLEMENTATION



Random Forest

```
In [131]: from sklearn.ensemble import RandomForestClassifier
          clf = RandomForestClassifier()

          clf.fit(X_train,y_train)

          predictR = clf.predict(X_test)

          print("")
          print('Classification report of Random Forest Results:')
          print("")

          print(classification_report(y_test,predictR))
          x = (accuracy_score(y_test,predictR)*100)

          print('Accuracy result of Random Forest is:', x)
          print("")
          cm1=confusion_matrix(y_test,predictR)
          print('Confusion Matrix result of Random Forest is:\n',cm1)
          print("")
          sensitivity1 = cm1[0,0]/(cm1[0,0]+cm1[0,1])
          print('Sensitivity : ', sensitivity1 )
          print("")
          specificity1 = cm1[1,1]/(cm1[1,0]+cm1[1,1])
          print('Specificity : ', specificity1)
          print("")

C:\Users\[illegible]\Anaconda3\lib\site-packages\sklearn\ensemble\forest.py:245: FutureWarning: The default value of n_estimators will change from 10 in version 0.20 to 100 in 0.22.
  "10 in version 0.20 to 100 in 0.22.", FutureWarning)
```

Figure 15 Random Forest

3.3.2 OUTPUT

Figure 16: Output for SVM algorithm

3.4 SUPPORT VECTOR MACHINE

3.4.1 IMPLEMENTATION

Support Vector Machines are perhaps one of the most popular and talked about machine learning algorithms.

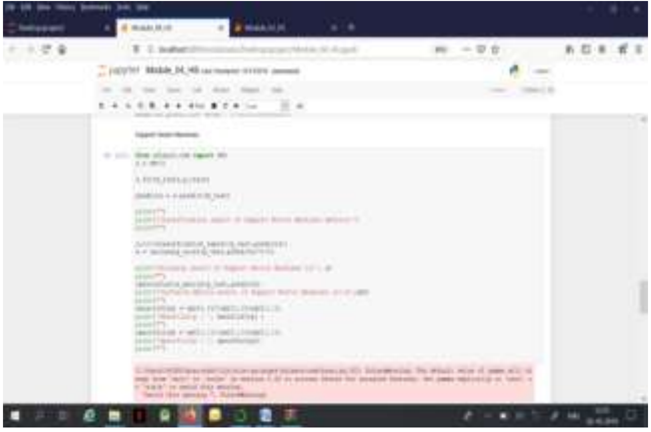


Figure 17: Code for SVM architecture

3.4.2 OUTPUT

Figure 18: Classification report of Support Vector Machines Results

IV CONCLUSION AND FUTURE WORK

This work aims to observe which features are most helpful in predicting the patient diseases by given attribute symptoms of heart disease or not and to see the general trends that may help us in model selection and hyper parameter selection. To achieve this we used machine learning classification methods to fit a function that can predict the discrete class of new input. The fundamental concepts of machine learning algorithms are applied from an available dataset and evaluated and interpreted the results and justified the interpretation based on observed dataset. Notebooks in Python was created that serve as computational records and document revealed and investigated the patient details whether patient affected by disease or not from the analysis of data set. statistical and visualized results were presented which find the standard patterns for all regiments.

In future, this work may be extended to apply for very large dataset and give preventive measures to patients and Doctors in the early detection stage of the disease.

REFERENCES

[1]. Shi K, Schellenberger S, Michler F, Steigleder T, Malessa A, Lurz F, Ostgathe C, Weigel R, Koelpin A. Automatic signal quality index determination of radar-recorded heart sound signals using ensemble classification. IEEE transactions on biomedical engineering. 2019 Jun 5;67(3):773-85.

[2]. Jin LP, Dong J. Intelligent health vessel ABC-DE: An electrocardiogram cloud computing service. IEEE Transactions on Cloud Computing. 2018 Apr 11;8(3):861-74

[3]. Meyer D, Körber M, Senner V, Tomizuka M. Regulating the Heart Rate of Human–Electric Hybrid Vehicle Riders Under Energy Consumption Constraints Using an Optimal Control Approach. IEEE Transactions on Control Systems Technology. 2018 Jul 24;27(5):2125-38.

[4]. Wen W, Liu G, Mao ZH, Huang W, Zhang X, Hu H, Yang J, Jia W. Toward constructing a real-time social anxiety evaluation system: Exploring effective heart rate features. IEEE Transactions on Affective Computing. 2018 Jan 11;11(1):100-10.

[5]. Chen X, Cheng J, Song R, Liu Y, Ward R, Wang ZJ. Video-based heart rate measurement: Recent advances and future prospects. IEEE Transactions on Instrumentation and Measurement. 2018 Nov 29;68(10):3600-15.

[6]. Kadtan YP, Chauhan AP, Brindha R. GUI based Prediction of Heart Stroke Stages by finding the accuracy using Machine Learning algorithm. Annals of the Romanian Society for Cell Biology. 2021 May 17:4571-7

[7]. Meng Y, Speier W, Shufelt C, Joung S, Van Eyk JE, Merz CN, Lopez M, Spiegel B, Arnold CW. A machine learning approach to classifying self-reported health status in a cohort of patients with heart disease using activity tracker data. IEEE journal of biomedical and health informatics. 2019 Jun 11;24(3):878-84

[8]. Ashish Chhabbi,Lakhan Ahuja,Sahil Ahir, and Y. K. Sharma,19 March 2016,“Heart Disease Prediction Using Data Mining Techniques”, International Journal of Research in Advent Technology,E-ISSN:2321-9637,Special Issue National

Conference "NCPC-2016", pp. 104-106.

[9]. Shadab Adam Pattekari,and Asma Parveen, 2012,"Prediction System for Heart Disease using Naive Bayes", International Journal of Advanced Computer and Mathematical Sciences, ISSN: 2230-9624,Vol. 3, Issue 3, pp. 290-294.

[10]. Boshra Bahrami, and Mirsaeid Hosseini Shirvani,February 2015,"Prediction and Diagnosis of Heart Disease by Data Mining Techniques",Journal of Multidisciplinary Engineering Science and Technology(JMEST), ISSN:3159- 0040, Vol. 2, Issue 2, pp. 164-168.