Prediction of Heart Diseases Using Machine Learning Techniques

[1] Randhi Hemasree, [2] M.Sampath Kumar

[1]M.Tech, Department of Computer Science and Systems Engineering, Andhra University, Visakhapatnam, AP, India

[2]Proffessor, Department of Computer Science and Systems Engineering, Andhra University, Visakhapatnam, AP, India.

ABSTRACT: The heart is the next major organ comparing to the brain, which has more priority in the Human body. Thus, preventing Heart diseases has become more than necessary. Good data-driven systems for predicting heart diseases can improve the entire research and prevention process, making sure that more people can live healthy lives. This is where Machine Learning comes into play. Machine Learning helps in predicting the Heart diseases, and the predictions made are quite accurate. The project involved analysis of the heart disease patient dataset with proper data processing. Then, different models were trained, and predictions are made with different algorithms KNN, Decision Tree, Random Forest, SVM, Logistic Regression etc. this is the Jupiter notebook code and I've used dataset from kaggle.com and UCI repository for various diseases-based datasets. I've used a variety of Machine Learning algorithms, implemented in Python, to predict the presence of heart disease in a patient. This is a classification problem, with input features as a variety of parameters, and the target variable as a binary variable, predicting whether heart disease is present or not.

KEYWORDS: Heart disease, Data Mining Techniques, Decision Tree, Naïve Bayesian, Support Vector Machine (SVM), K-NN classifier.

I. INTRODUCTION

Data Mining is the way toward extracting interesting patterns and knowledge from huge amount of information. The Data Mining process is a combination of choosing, analyzing, planning, interpreting and evaluating the outcomes [1]. Numerous clinical finding achievement in the data mining techniques for prediction and clustering. Data mining comprises of the different specialized approaches including machine learning, database system and statistic

The healthcare industry assembles immense measure of healthcare data which are not abundant to discover hidden information for effective decision making. Utilizing distinctive medical profiles such as sex, blood pressure, age, hypertension, lack of physical activity, blood sugar it can find the probability of patients getting a coronary illness [3]. Diagnosing machines or frameworks are quite useful in this procedure because not every doctor must have the learning of each and every kind of problem of disease. In this manner automated diagnosing machine is used by them to diagnose the problem accurately.

The WHO consortium has shared this information that ten a great many passing happen in this world is a consequence of coronary illness. so, it was an extremely dangers problem in world. These systems typically create huge amounts of information which appear as numbers, charts and images. There are numerous sort of heart disease such as coronary heart disease, cardiomyopathy disease and cardiovascular sickness. Cardiovascular disease is an illness which specifically impacts the blood circulation in the body and blood vessels which are associated to the heart. Increment in heart disease is because of many facts like great BP, smoking, Family history etc. some unique factors that likewise causes heart illnesses are elevated cholesterol level, hyper solidness, improper diet etc.[4].

MACHINE LEARNING:

Machine Learning, as the name suggests, is the science of programming a computer by which they are able to learn from different kinds of data. A more general definition given by Arthur Samuel is – "Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed." They are typically used to solve various types of life problems.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven't surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, "to make decisions, based on data, with efficiency and scale". Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programing logic, in the problems that cannot be programmed inherently. The fact is that we can't do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Applications of machine learning:

Machine learning is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that which makes it more similar to humans: The ability to learn. Machine learning is actively being used today, perhaps in many more places than one would expect. We probably use a learning algorithm dozen of time without even knowing it. Applications of Machine Learning include:

- → Web Search Engine: One of the reasons why search engines like google, Bing etc work so well is because the system has learnt how to rank pages through a complex learning algorithm.
- → Photo tagging Applications: Be it Facebook or any other photo tagging application, the ability to tag friends makes it even more happening. It is all possible because of a face recognition algorithm that runs behind the application.
- → Spam Detector: Our mail agent like Gmail or Hotmail does a lot of hard work for us in classifying the mails and moving the spam mails to spam folder. This is again achieved by a spam classifier running in the back end of mail application.

Today, companies are using Machine Learning to improve business decisions, increase productivity, detect disease, forecast weather, and do many more things. With the exponential growth of technology, we not only need better tools to understand the data we currently have, but we also need to prepare ourselves for the data we will have. To achieve this goal we need to build intelligent machines. We can write a program to do simple things. But for most of times Hardwiring Intelligence in it is difficult. Best way to do it is to have some way for machines to learn things themselves. A mechanism for learning – if a machine can learn from input then it does the hard work for us. This is where Machine Learning comes in action.

Challenges in Machines Learning:

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are

- → Quality of data: having good quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data pre-processing and feature extraction.
- → Time-Consuming task: another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.
- → Lack of specialist persons: As ML technology is still in its infancy stage, availability of expert resources is a tough job.
- → No clear objective for formulating business problems: Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.
- → Issue of overfitting & under fitting: If the model is overfitting or under fitting, it cannot be represented well for the problem.
- → Curse of dimensionality: another challenge ML model faces is too many features of data points. This can be a real hindrance.
- → Difficulty in deployment: Complexity of the ML model makes it quite difficult to be deployed in real life.

II. LITRATURE SURVEY

Bala Sundar V et.al examined in this paper real and artificial dataset that have been used to predict diagnosis of heart diseases with the help of a K-mean clustering technique results to check its accuracy [1]. Sayali D. Jadhav et.al proposed that the issue goes for taking in the relationship between an arrangement of feature variables and a target variable of interest [2].

Mr. P Sai Chandrasekhar Reddy, he proposed work separates proposed system in two sections such as execution model and prediction model. Performance model is intended to assess the general performance of the application [4].Dursun Delen, et.al [5] presented prescient models so as to investigate the immense databases of patients across the nation. . The performance results of mechanisms that include decision trees and neural networks are less exact that the ones with SVM calculations. David L. Olson, et.al [6] amongst the normal accuracy and decision tree size a tradeoff is given among this paper. Minimum parameters can be used keeping in mind the end goal to control the decision tree's size.

Akhilesh Kumar Yadav, et.al proposed algorithm has been tested by performing distinctive experiments on it that gives excellent outcome on essential data sets. In real world problem enhanced outcomes are accomplished utilizing foggy k-mean clustering algorithm as compared to existing simple k-means clustering algorithm. Daljit Kaur et.al [8]explained in this paper that data contained similar objects has been divided using clustering. The proposed algorithm has been tested and results shows that it is able to reduce efforts of numerical calculation, complexity along with maintaining an easiness of its implementation and also able to solve dead unit problem. Sanjay Chakrabotryet.al [9] stated that powerful tool clustering is used as different forecasting tools. The weather determining has been performed utilizing proposed incremental K-mean clustering generic technique.

K. Raja lakshmi et.al[10]stated that K-means algorithm has been utilized to study distinctive extant illness. The cost Amandeep Kaur et al, International Journal of Advanced Research in Computer Science, 9 (2), March-April 2018,569-572 effectiveness and human effects has been reduced using proposed prediction system-based data mining. Mustafa A. Al-Fayoumi[11]proposed an Associative Classification based on Incremental Mining (ACIM) algorithm in order to maintain the huge amount of information.

Sajida Perveen, et.al[12] presented that J48 decision tree was utilized in order to apply ad boost and bagging ensemble methods in order to differentiate patients that are suffering from diabetes mellitus based on different elements that can cause diabetes. Basma Boukenze, studied the prediction of kidney disorder by using numerous machine learning methods is the prior aim of this research. The algorithms that are included within this study are SVM, Decision Tree , and Bayesian Network (BN).

Nancy. P, et.al[14]investigated that performance of around fifteen data mining classification algorithms utilized within the data mining systems. Johan Holmgren, et.al [15] presented that utilization of support vector regression such that the numbers of bicycles that are being registered are predicted. Min Chen, et.al[16]proposed a novel Convolutional Neural Network based Multimodal Disease Risk Prediction (CNN-MDRP) algorithm and 94.8% of prediction accuracy was achieved here along with the higher convergence speed in comparison to other similar enhanced algorithms.

III. ANALYSIS OF THE DATA MINING TECHNIQUES

Various data mining techniques are usable and still lot of research is going on to find new techniques that can produce exact outcomes.

1. Decision Tree

The Decision tree is a classification strategy in which classification is done by the dividing criteria. The decision tree is a schema like a tree structure that gatherings instances by sorting them in perspective of the feature values. Each and every node in a decision tree depicted the features in a case to be classified. Decision tree makes the rule for the classification of the data set. The three fundamental algorithms are extensively utilized that are ID3, CART and C4.5.

2. Naive Bayesian:

The Bayesian Classification express as supervised learning strategy and statistical technique for classification. Assumes a hidden probabilistic model and it empower us to capture vulnerability about the model in a right way by deciding probabilities of the results. It can take care of diagnostic and predictive issues.
Naive Bayes algorithm depends on Bayesian Theorem.
Bayesian Theorem:
Given training data X, back likelihood of a theory, H, P(H|X), takes after the Bayes hypothesis P(H|X)=P(X|H)P(H)/P(X)

3. SVM:

Support Vector Machine algorithm make good judgement for data points that are outside the preparing set. There are two classes of information in SVM. The data points are separated such that they could draw a horizontal line on the figure. The line is made in a way that it isolates every one of the focuses on one side of one class and every one of the focuses on the reverse side of alternate class. When such circumstance happens, then the data are linearly separable. The line used to isolate the dataset is known as a separating hyperplane. The points nearest to the isolating hyperplane are called as support vectors. Kernels are utilized to extend SVMs to a bigger number of datasets. Mapping of one feature space to another is finished by kernel. Kernel method maps the information (in some cases likewise called as nonlinear information) from a little dimensional space to an extensive dimensional space. In a bigger measurement, it decides straight issue that is nonlinear in smaller-dimensional space. The Radial Bias Function (RBF) is a prominent kernel that measures the separation among two vectors.

4. K-Nearest Neighbor classifier (KNN):

KNN is a basic, lazy and nonparametric classifier. KNN is preferred when every one of the features are persistent. KNN is likewise called as case-based reasoning and has been utilized in numerous applications like statistical estimation, pattern recognition. Classification is distinguishing the closest neighbor to decide the class of an unknown sample. KNN is favored over other classification algorithms due to its high merging velocity and straightforwardness. KNN characterization has two phases:
Find the k number of examples in the dataset that is nearest to instance S
These k number of examples at that point vote to decide the class of instance S
The Accuracy of KNN relies upon separate metric and K value. Different methods for estimating the separation between two instances are cosine, Euclidian distance. To evaluate the new unidentified sample, KNN figure out its KNN and assign a class by dominant part voting.

5. Multi-layer perceptron (MLP) ANN:

Neural networks are of particular interest because they offer a means of efficiently modeling large and complex problems in which there may be hundreds of predictor variables that have many interactions. (Actual biological neural networks are incomparably more complex.) Neural nets may use in classification problems (where the output is a categorical variable) or for regressions (where the output variable is continuous).The architecture of the neural network shown in figure.4 consists of three layers such as input layer, hidden layer and output layer. The nodes in the input layer linked with a number of nodes in the hidden layer. Each input node joined to each node in the hidden layer. The nodes in the hidden layer may connect to nodes in another hidden layer, or to an output layer. The output layer consists of one or more response variables. A main concern of the training phase is to focus on the interior weights of the neural network, which adjusted according to the transactions used in the learning process. For each training transaction, the neural network receives in addition the expected output. This concept drives us to modify the interior weights while trained neural network used to classify new images.

6. Random forest classifier:

Random forest is a supervised learning algorithm. It can be used both for classification and regression. It is also the most flexible and easy to use algorithm. A forest is comprised of trees. It is said that the more trees it has, the more robust a forest is. Random forests create decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance. Random forests have a variety of applications, such as recommendation engines, image classification and feature selection. It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the base of the Boruta algorithm, which selects important features in a dataset.

7. Logistic regression:

Logistic regression is named for the function used at the core of the method, the logistic function.
The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

1 / (1 + e^-value)

Where e is the base of the natural logarithms (Euler's number or the EXP() function in your spreadsheet) and value is the actual numerical value that you want to transform. Below is a plot of the numbers between -5 and 5 transformed into the range 0 and 1 using the logistic function.

IV. ANALYSIS RESULTS

A. MEASURES FOR PERFORMANCE EVALUATION

In this study, the accuracy of two data mining techniques is compared. Although such metrics are used more often in the field of information retrieval, it is considered as they are related to other existing metrics such as specificity and sensitivity. These metrics can be derived from the confusion matrix and can be easily converted to true-positive (TP) and false-positive (FP) metrics.

i. Accuracy Measures

Accuracy measure represents how far the set of tuples are being classified correctly.TP refers to positive tuples and TN refers to negative tuples classified by the basic classifiers. Similarly, FP refers to positive tuples and FN refers to negative tuples which is being incorrectly classified by the classifiers. The accuracy measures used here are sensitivity and specificity

ii. Confusion matrix

The confusion matrix contains four classification performance indices: true positive, false positive, false negative, and true negative as shown in Table1. These four indices are also usually used to evaluate the performance the two-class classification problem. The four classification performance indices included in the confusion matrix is shown in Table I.

Table I. Confusion Matrix

Actual Class	Predicted Class	
	Positive	Negative
Positive	True Positive(TP)	False Negative(FN)
Negative	False Positive(FP)	True Negative(TN)

iii. Cross Validation

Cross-validation is a standard tool in analytics and is an important feature for helping you develop and fine-tune data mining models. You use cross-validation after you have created a mining structure and related mining models to ascertain the validity of the model. Cross-validation has the following applications:

Validating the robustness of a particular mining model. Evaluating multiple models from a single statement.

Building multiple models and then identifying the best model based on statistics.

iv. Sensitivity Analysis

A sensitivity analysis is a technique used to determine how different values of an independent variable impact a particular dependent variable under a given set of assumptions. Sensitivity (also called the true positive rate, the recall, or probability of detection [1] in some fields) measures the proportion of positives that are correctly identified as such (e.g., the percentage of sick people who are correctly identified as having the condition).

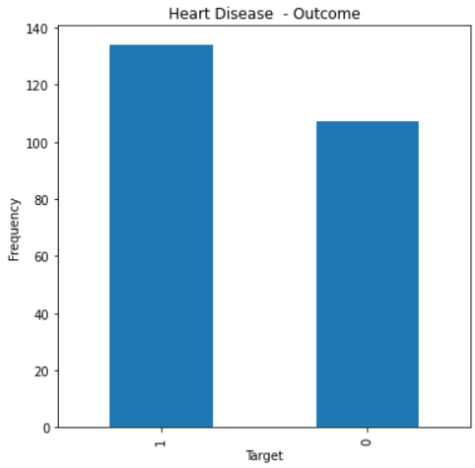
v. Specificity Analysis

Specificity (also called the true negative rate) measures the proportion of negatives that are correctly identified as such (e.g., the percentage of healthy people who are correctly identified as not having the condition).

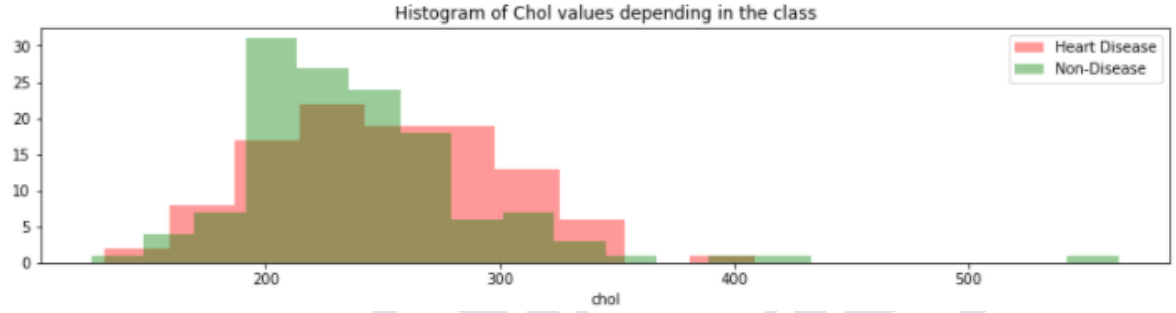
5. RESULTS

Heart Disease – outcome: The following output shows how many people have heart disease and how many people does not have disease [1 – have heart disease ; 0 – does not have heart disease]

```
df['target'].value_counts().plot(kind='bar', figsize=(6,6))
plt.title('Heart Disease  - Outcome')
plt.xlabel('Target')
plt.ylabel('Frequency')
plt.show()
```

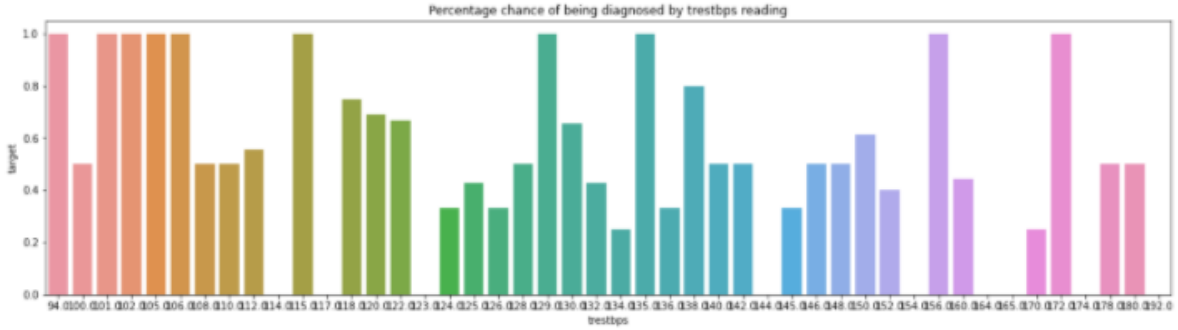



```
plt.figure(figsize=(14,3))
Insulin_plt = df.groupby(df['chol']).target.count().reset_index()
sns.distplot(df[df.target == 0]['chol'], color='red', kde=False, label='He
sns.distplot(df[df.target == 1]['chol'], color='green', kde=False, label='
plt.legend()
plt.title('Histogram of Chol values depending in the class')
plt.show()
```



Histogram of chol values depending in the class

```
plt.figure(figsize=(20,5))
glucose_plt = df.groupby('trestbps').target.mean().reset_index()
sns.barplot(glucose_plt.trestbps, glucose_plt.target)
plt.title('Percentage chance of being diagnosed by trestbps reading')
plt.show()
```



Percentage chance of being diagnosed by trestbps reading

DecisionTreeClassifier

```
from sklearn.tree import DecisionTreeClassifier
DT = DecisionTreeClassifier()
DT.fit(scaled_x_train, y_train)
predictions = DT.predict(scaled_x_test)
val3 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for DT: ", val3, "\n")
print("*Confusion Matrix for DT: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for DT: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for DT:  75.34246575342466

*Confusion Matrix for DT:
[[24  9]
 [ 9 31]]
*Classification Report for DT:
              precision    recall  f1-score   support

           0       0.73      0.73      0.73        33
           1       0.78      0.78      0.78        40

    accuracy                           0.75        73
   macro avg       0.75      0.75      0.75        73
weighted avg       0.75      0.75      0.75        73
```

KNeighborsClassifier

```
from sklearn.neighbors import KNeighborsClassifier
KNN = KNeighborsClassifier()
KNN.fit(scaled_x_train, y_train)
predictions = KNN.predict(scaled_x_test)
val4 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for KNN: ", val4, "\n")
print("*Confusion Matrix for KNN: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for KNN: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for KNN:  79.45205479452055

*Confusion Matrix for KNN:
[[28  5]
 [10 30]]
*Classification Report for KNN:
              precision    recall  f1-score   support

           0       0.74      0.85      0.79        33
           1       0.86      0.75      0.80        40

    accuracy                           0.79        73
   macro avg       0.80      0.80      0.79        73
weighted avg       0.80      0.79      0.79        73
```

LogisticRegression

```
from sklearn.linear_model import LogisticRegression
LR = LogisticRegression()
LR.fit(scaled_x_train, y_train)
predictions = LR.predict(scaled_x_test)
val5 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for LR: ", val5, "\n")
print("*Confusion Matrix for LR: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for LR: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for LR:  80.82191780821918

*Confusion Matrix for LR:
[[25  8]
 [ 6 34]]
*Classification Report for LR:
              precision    recall  f1-score   support

           0       0.81      0.76      0.78        33
           1       0.81      0.85      0.83        40

    accuracy                           0.81        73
   macro avg       0.81      0.80      0.81        73
weighted avg       0.81      0.81      0.81        73
```

MLP -ANN

```
from sklearn.neural_network import MLPClassifier
MLP = MLPClassifier()
MLP.fit(scaled_x_train, y_train)
predictions = MLP.predict(scaled_x_test)
val6 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for MLP: ", val6, "\n")
print("*Confusion Matrix for MLP: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for MLP: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for MLP:  75.34246575342466

*Confusion Matrix for MLP:
[[24  9]
 [ 9 31]]
*Classification Report for MLP:
              precision    recall  f1-score   support

           0       0.73      0.73      0.73        33
           1       0.78      0.78      0.78        40

    accuracy                           0.75        73
   macro avg       0.75      0.75      0.75        73
weighted avg       0.75      0.75      0.75        73
```

GaussianNB

```
from sklearn.naive_bayes import GaussianNB
GNB = GaussianNB()
GNB.fit(scaled_x_train, y_train)
predictions = GNB.predict(scaled_x_test)
val7 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for GNB: ", val7, "\n")
print("*Confusion Matrix for GNB: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for GNB: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for GNB:  82.1917808219178

*Confusion Matrix for GNB:
[[27  6]
 [ 7 33]]
*Classification Report for GNB:
              precision    recall  f1-score   support

           0       0.79      0.82      0.81        33
           1       0.85      0.82      0.84        40

    accuracy                           0.82        73
   macro avg       0.82      0.82      0.82        73
weighted avg       0.82      0.82      0.82        73
```

Random Forest Classifier

```
from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier()
RF.fit(scaled_x_train, y_train)
predictions = RF.predict(scaled_x_test)
val2 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for RF: ", val2, "\n")
print("*Confusion Matrix for RF: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for RF: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for RF:  82.1917808219178

*Confusion Matrix for RF:
[[27  6]
 [ 7 33]]
*Classification Report for RF:
              precision    recall  f1-score   support

           0       0.79      0.82      0.81        33
           1       0.85      0.82      0.84        40

    accuracy                           0.82        73
   macro avg       0.82      0.82      0.82        73
weighted avg       0.82      0.82      0.82        73
```

Support Vector Machine

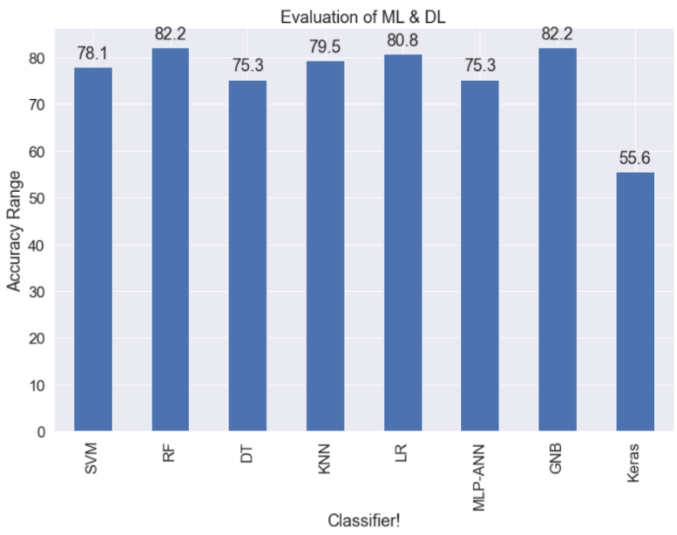
```
from sklearn.svm import SVC
SVM = SVC()
SVM.fit(scaled_x_train, y_train)
predictions = SVM.predict(scaled_x_test)
val1 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for SVM: ", val1, "\n")
print("*Confusion Matrix for SVM: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for SVM: ")
print(classification_report(y_test, predictions))
```

```
*Accuracy score for SVM:  78.08219178082192

*Confusion Matrix for SVM:
[[28  5]
 [11 29]]
*Classification Report for SVM:
              precision    recall  f1-score   support

           0       0.72      0.85      0.78        33
           1       0.85      0.72      0.78        40

    accuracy                           0.78        73
   macro avg       0.79      0.79      0.78        73
weighted avg       0.79      0.78      0.78        73
```



Accuracy comparison of machine learning algorithms

6. CONCLUSION

This project summarizes about the research functionality in establishing huge frameworks in healthcare, using the applications of different machine learning algorithms for prognosis of the disease. The contribution made by many researchers for application of different machine algorithms along with accuracy is comparatively shown. This project discusses the various machine learning algorithms such as support vector machine, Naive Bayes, decision tree, k- nearest neighbor, neural networks, random forest, logistic regression which were applied to the data set. It utilizes the data such as blood pressure, cholesterol, diabetes and then tries to predict the possible coronary heart disease patient. Family history of heart disease can also be a reason for developing a heart disease. So, this data of the patient can also be included for further increasing the accuracy of the model. By applying different machine learning algorithms Random Forest and naive Bayes algorithm yields the highest accuracy of 82%. This work will be useful in identifying the possible patients who may suffer from heart disease.

7. FUTURE WORK

In future, this may help in taking preventive measures and hence try to avoid the possibility of heart disease for the patient. So when a patient is predicted as positive for heart disease, then the medical data for the patient can be closely analyzed by the doctors. An example would be - suppose the patient has diabetes which may be the cause for heart disease in future and then the patient can be given treatment to have diabetes in control which in turn may prevent the heart disease. The heart disease prediction can be done using other machine learning algorithms. Logistic regression can also perform well in case of binary classification problems such as heart disease prediction. Random forests can perform well than decision trees. The results can be compared and improvised.

REFERENCES

[1] Sonam Nikhar, A.M. Karandikar “Prediction of Heart Disease Using Machine Learning Algorithms” in International Journal of Advanced Engineering, Management and Science (IJAEMS) June-2016 vol-2
[2] Deeanna Kelley “Heart Disease: Causes, Prevention, and Current Research” in JCCC Honors Journal
[3] Nabil Alshurafa, Costas Sideris, Mohammad Pourhomayoun, Haik Kalantarian, Majid Sarrafzadeh "Remote Health Monitoring Out-come Success Prediction using Baseline and First Month Intervention Data" in IEEE Journal of Biomedical and Health Informatics
[4] Ponrathi Athilingam, Bradlee Jenkins, Marcia Johansson, Miguel Labrador "A Mobile Health Intervention to Improve Self-Care in Patients with Heart Failure: Pilot Randomized Control Trial" in JMIR Cardio 2017, vol. 1, issue 2, pg no:1
[5] DhafarHamed, Jwan K. Alwan, Mohamed Ibrahim, Mohammad B. Naeem "The Utilisation of Machine Learning Approaches for Med-ical Data Classification " in Annual Conference on New Trends in Information & Communications Technology Applications - march-2017
[6] Applying k-Nearest Neighbour in Diagnosing Heart Disease Patients Mai Shouman, Tim Turner, and Rob Stocker International Journal of Information and Education Technology, Vol. 2, No. 3, June 2012
[7] Amudhavel, J., Padmapriya, S., Nandhini, R., Kavipriya, G., Dhavachelvan, P., Venkatachalapathy, V.S.K., "Recursive ant colony optimization routing in wireless mesh network", (2016) Advances in Intelligent Systems and Computing, 381, pp. 341-351.
[8] Alapatt, B.P., Kavitha, A., Amudhavel, J., "A novel encryption al-gorithm for end to end secured fiber optic communication", (2017) International Journal of Pure and Applied Mathematics, 117 (19 Special Issue), pp. 269-275.
[9] Amudhavel, J., Inbavalli, P., Bhuvaneswari, B., Anandaraj, B., Vengattaraman, T., Premkumar, K., "An effective analysis on har-mony search optimization approaches", (2015) International Journal of Applied Engineering Research, 10 (3), pp. 2035-2038.
[10] Amudhavel, J., Kathavate, P., Reddy, L.S.S., Bhuvaneswari Aadharshini, A., "Assessment on authentication mechanisms in distributed system: A case study", (2017) Journal of Advanced Re-search in Dynamical and Control Systems, 9 (Special Issue 12), pp. 1437-1448.
[11] Amudhavel, J., Kodeeshwari, C., Premkumar, K., Jaiganesh, S., Rajaguru, D., Vengattatraman, T., Haripriya, R., "Comprehensive analysis on information dissemination protocols in vehicular ad hoc networks", (2015) International Journal of Applied Engineering Re-search, 10 (3), pp. 2058-2061.
[12] Amudhavel, J., Kathavate, P., Reddy, L.S.S., Satyanarayana, K.V.V., "Effects, challenges, opportunities and analysis on security based cloud resource virtualization", (2017) Journal of Advanced Research in Dynamical and Control Systems, 9 (Special Issue 12), pp. 1458-1463.