ISSN 2395-1621

Comparative Study of Different Machine Learning Techniques for Heart Disease Prediction

Aritra Ray
A. K. C. School of Information Technology,
University of Calcutta, Kolkata, India
aritra98.ray@gmail.com

Hena Ray
Centre for Development of Advanced Computing,
Kolkata, India
hena.roy@cdac.in

ABSTRACT

There is a large fraction of deaths attributed to heart diseases globally [1]. We propose for a preliminary preventive measure, barring the need of a trained medical practitioner, can be introduced just by performing some clinical laboratory tests, results of which when fed into a neural network can predict the possibility of an angiographic heart disease in the concerned individual with an accuracy of almost 90%. We have worked with K-Nearest Neighbours, Support Vector Machines and Neural Networks, using MATLAB as our tool for implementation, as the different models to identify which one among these would have the best accuracy and fine-tuned it to be used as an initial screening process to identify possible individuals having an angiographic heart disease as the ratio of number of trained medical practitioners to that of the number of individuals are alarmingly low in most of the developing nations particularly. Thus our focus is to ensure that we can diagnose a patient early to prevent mishaps.

Keywords— Heart Disease, KNN, SVM, Neural Networks

ARTICLE INFO

Article History

Received: 8th March 2020

Received in revised form :

8th March 2020

Accepted: 10th March 2020

Published online :

11th March 2020

I. INTRODUCTION

Cardiovascular disease (CVD) [1] is one of the major causes of deaths globally accounting for up to 30% deaths. The major of those include coronary heart diseases or ischemic ones, cerebrovascular disease or stroke, hypertension, heart failure, and rheumatic heart disease. The major affected among them are the developing countries where access to proper healthcare facilities are scarce. There were around 17.5 million deaths worldwide in 2015 related to CVD. Specifically, the Asian countries have a very high number of deaths related to heart diseases compared to the rest of the world owing to its share of population. Even in developed countries, the number of deaths related to such is also high. Taking the case of United States for instance, heart disease is the leading cause of death for men, women, and people of most racial and ethnic groups in the United States [2]. One person dies every 37 seconds in the United States from cardiovascular disease [2]. About 647,000 Americans die from heart disease each year—that's one in every four deaths. [3][4] Table 1 also represents some of the major communities where deaths related to heart diseases are very predominant from the United States [2].

Race of Ethnic Group	Percenatge of Deaths
American Indian or Alaska Native	18.3
Asian American or Pacific Islander	21.4
Black (Non-Hispanic)	23.5
White (Non-Hispanic)	23.7
Hispanic	20.3

Table 1: Represents some of the major communities in United States where there is a predominance of deaths owing to heart diseases.

Given the severity of the issue and the number of deaths worldwide, we devised to work on predicting the possibility of a heart disease in an individual based on a number of features, which are a result of clinical laboratory tests. Thereafter, machine learning techniques were deployed, experimentations were carried out on various parameters of the different machine learning techniques as presented in the paper. The dataset we used was from the UCI machine learning repository [5]. The results in our study are based on the Cleveland database which has simply attempted to distinguish presence of a heart disease, wherein the prediction parameter values are greater than one, from the absence of the heart disease, wherein the prediction parameter was noted to be zero. The dataset has thirteen parameters to be studied [5]. There were six patients in the concerned database where some records of some attributes were missing and thus have negated those from our study.

II. RELATED WORKS

There has been different models those were tried and varying levels of accuracy were noted. About 89% accuracy was found out using naïve bayes classifier [6] and ANN ensemble technique [7], ANN LQV showed an accuracy of about 80% [8], 84% when techniques like RBF, SVM, bagging, boosting and stacking were used [9] and about 86% in DT, NB and ANN techniques [10]. There were different datasets, apart from the one we considered here, were also studied for heart disease prediction and other techniques like data mining were also used to achieve the same goal.

However herein we have shown a detailed study of the how the accuracy varies with different distance metrics and number of nearest neighbours considered in K-Nearest Neighbours, and how different kernels affect the accuracy in Support Vector Machines and how changing the number of hidden layers in neural networks affect its accuracy. Alongside accuracy, other metrics like precision, recall and F-measure were also studied.

III. TECHNIQUES STUDIED

A. *K – Nearest Neighbours*

K – Nearest Neighbour is a simple algorithm that uses the concept of feature similarity to classify values of new data points, which means that the new data point will be assigned a value based on how closely it matches the points in the training set. KNN is used for classification, that is, an object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours. As it classifies new cases based on a similarity measure, that is the distance metric, we have studied how the accuracy different distance metrics vary with values of K, as K in KNN is the number of nearest neighbours used to classify the unknown data point. The underlying algorithm works like for every test data point, we calculate the distance between test data and each row of training data with any one of the distance metric, then based on the distance value, we sort them in ascending order, then choose the top K rows from the sorted array and finally assign a class to the test data point based on most frequent class of these rows.

1) **Euclidean Metric**

In Cartesian coordinates, if p = (p_1, p_2, … , p_n) and q = (q_1, q_2, … , q_n) then are two points in Euclidean *n*-space, then the distance (d) from p to q, or from q to p is given by $\sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$.

2) **Cosine Metric**

For two non-zero vectors of attributes A and B, using the Euclidean dot product, the cosine similarity cos(θ), is represented using $similarity = \cos(\theta) = \frac{A.B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$

3) **Minkowski Metric**

The Minkowski distance is a metric in a normed vector space which can be considered as a generalization of both the Euclidean distance and the Manhattan distance. The Minkowski distance D of order p between two points $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n) \in R^n$ is defined as $D(X,Y) = (\sum_{i=1}^{n}|x_i - y_i|^p)^{\frac{1}{p}}$.

4) **Mahalanobis Metric**

The Mahalanobis distance is a measure of the distance between a point P and a distribution D. The idea of measuring is, how many standard deviations away P is from the mean of D. The benefit of using mahalanobis distance is, it takes covariance in account which helps in measuring the strength/similarity between two different data objects. The distance between an observation and the mean can be calculated as $D_M(\vec{x}) = \sqrt{(\vec{x} - \vec{\mu})^T S^{-1} (\vec{x} - \vec{\mu})}$, where S is the covariance metric.

5) **Hamming Metric**

Hamming distance between two integers is the number of bits which are different at same position in both numbers and is calculated as the XOR of two numbers and the counting the number of set bits. Hamming distance D_H between two data points is given as

$D_H = \sum_{i=1}^{k}(|x_i - y_i|)$.

6) **Spearman Metric**

Spearman distance is the square of Euclidean distance between two rank vectors. For n rank vectors, it is given by $d_{ij} = \sum_{k=1}^{n}(x_{ij} - x_{jk})^2$.

7) **Jaccard Metric**

The Jaccard similarity index, or coefficient, compares members for two sets to see which members are shared and which are distinct. It is a measure of similarity for the two sets of data, with a range from 0% to 100%. The higher the percentage, the more similar the two populations. Jaccard index is calculated as the ratio of the number in both sets to the number in either set multiplied 100. The same formula in notation is given as $J(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}$.

The Jaccard distance is a measure of how dissimilar two sets are. It is the complement of the Jaccard index and can be found by subtracting the Jaccard Index from 100%. In set notation, we subtract from 1 for the Jaccard Distance as $D(X,Y) = 1 - J(X,Y)$.

B. *Support Vector Machines*

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for either classification or regression problems. Here we have utilized it for the former. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features in consideration) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well. A hyper-plane is a subspace of one dimension less than its ambient space. The dimension of a mathematical space (or object) is informally defined as the minimum number of coordinates (x, y and z axis) needed to specify any point within it while an ambient space is the space surrounding a mathematical object. A mathematical object is an abstract object arising in mathematics. An abstract object is an object which does not exist at any particular time or place, but rather exists as a type of thing, i.e., an idea, or abstraction. SVM algorithms use a set of mathematical functions that are defined as the kernel. The function of kernel is to take data as input and transform it into the required form.

Mathematically, $K(x, y) = < f(x), f(y) >$ where K is the kernel function, x, y are n dimensional inputs, f is a map from n-dimension to m-dimension space,$< x, y >$ denotes the dot product and usually m is much larger than n. There can be different types kernels like linear, nonlinear, polynomial, radial basis function (RBF), sigmoid and here we have tried out a couple of them.

1) Linear Kernel

The linear kernel is the simplest kernel function and thus training a support vector machine model with this is faster than any other kernel. It is given by the inner product $< x, y >$ plus an optional constant c. The kernel function can be given as $k(x, y) = x^T y + c$.

2) Polynomial Kernel

The polynomial kernel is a non-stationary one, given by$k\ (x, y) = (\propto x^T y + c\ )^d$. The adjustable parameters are the slope α, the constant term c and the polynomial degree d. The quadratic kernel is a specialised case where degree d is taken to be two, that is, $k\ (x, y) = (\propto x^T y + c\ )^2$ and the cubic kernel is where the degree d is taken to be three, that is, $k\ (x, y) = (\propto x^T y + c\ )^3$.

C. *Neural Networks*

A neuron in an artificial neural network is a set of input values and associated weights and a function that sums the weights and maps the results to an output. Neurons are organized into layers, namely, input, hidden and output. The input layer is composed not of full neurons, but rather consists simply of the record's values that are inputs to the next layer of neurons. The next is the hidden layer. Several hidden layers can exist in one neural network. We have experimented how various parameters change with changes in the number of hidden layers. The final layer is the output layer, where there is one node for each class. In the training phase, the correct class for each record is known (termed supervised training), and the output nodes can be assigned correct values.

IV. PARAMETERS MONITORED

A. *Accuracy*

Accuracy is an important parameter to be measured as it represents how close the predicted results comes to the actual test cases. We have monitored accuracy for the three techniques we have considered in our study. For neural networks, we obtained such from the confusion matrix which a table that is often used to describe the performance of our classifier on a set of test data for which the true values are known. It is measured as a ratio of sum of true positive values and true negative values, which reflects the total number of correct predictions, to the total number of predictions. [11]

B. *Recall*

Recall refers to the percentage of total relevant results correctly classified by our algorithm. It attempts to answer questions like what proportion of actual positives was identified correctly. It is calculated as a ratio of true positives to the sum of true positives and false negatives. [12]

a. *Precision*

It attempts to answer questions like what proportion of positive identifications was actually correct. It is calculated as a ratio of true positives to the sum of true positives and false positives. [12]

b. *F – Measure*

The F score, also called the F1 score or F measure, is a measure of a test's accuracy. The F score is defined as the weighted harmonic mean of the test's precision and recall. The F score is used to measure a test's accuracy, and it balances the use of precision and recall to do it. The F score can provide a more realistic measure of a test's performance by using both precision and recall. [13]

V. RESULTS

A. *K – Nearest Neighbours*

We computed the variation in accuracy of the discussed distance metrics by varying the values of K from one to ten with a five-fold cross validation. The results obtained are graphically represented in fig 1.

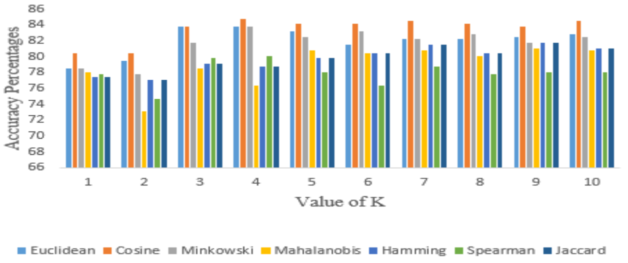

Figure 1. Accuracy Measure of KNN for different distance metrics by varying the value of K.

B. Support Vector Machines

Table 2 shows how the parameters of study varied when different kernels were implemented.

Parameters	**Kernel Function**		
	Linear	***Quadratic***	***Cubic***
Accuracy	0.83	0.79	0.79
Recall	0.87	0.83	0.81
Precision	0.83	0.80	0.81
F - Measure	0.85	0.81	0.81

Table 2: Representation of how each parameters of study varied with respect to their kernel functions.

C. Neural Networks

The number of hidden layers were altered from ten to twenty and observations were made in regards to accuracy, precision, recall and F – measure percentages. There were 207 training samples, 45 training and validation samples each. The recordings are shown as in fig 2, 3, 4, 5.

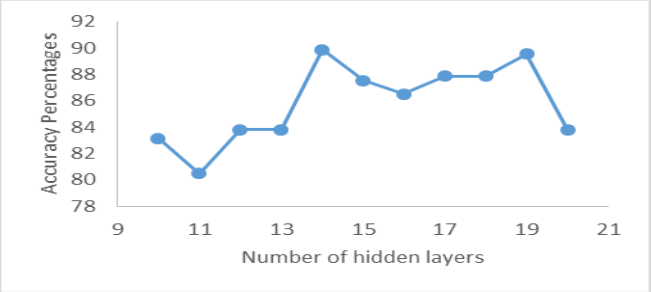

Figure 2. Accuracy measure as a percentage for different number of hidden layers in neural networks.

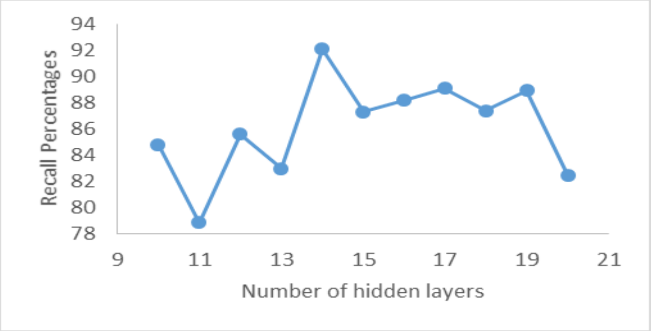

Figure 3. Recall measure as a percentage for different number of hidden layers in neural networks.

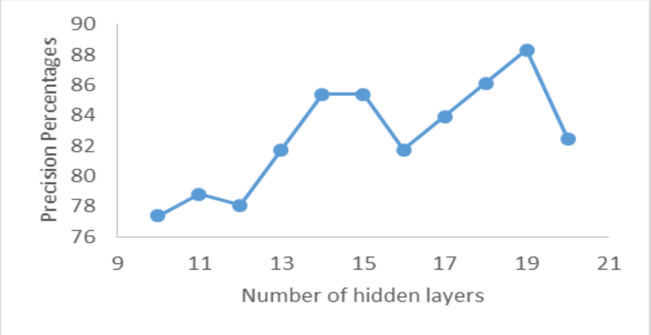

Figure 4. Precision measure as a percentage for different number of hidden layers in neural networks.

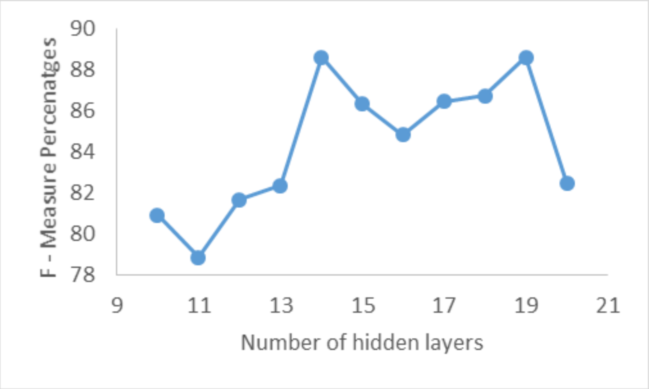

Figure 5. F- measure as a percentage for different number of hidden layers in neural networks.

VI. DISCUSSIONS

For the K – Nearest Neighbors, the cosine metric had the best accuracy regardless of the value of K we studied. The Euclidean and Minkowski metric had an increase in the accuracy initially and then almost remained a constant. Mahalanobis, Hamming and Jaccard metrics showed signs of improvement in the accuracy with increasing number of neighbours taken into consideration. However Spearman couldn't maintain a decent accuracy value throughout and was very unpredictable. The best accuracy of 84.8% was obtained by cosine metric when K was taken to be five. In regards of the Support Vector Machine, the best accuracy came from the linear kernel function at 83.0% and in other parameters of study, like precision, recall and F-measure percentage, linear kernel performed the best. For the Neural Networks, the best accuracy of 89.56% was achieved when the number of hidden layers were set to nineteen. The best recall percentage of 92.12% was attained for number of hidden layers to be fourteen, the best precision of 86.13% for eighteen number of hidden layers and best F-measure for nineteen hidden layers at 88.64%. Thus neural networks with nineteen hidden layers can be implemented in preliminary preventive measure unit, wherein if the desired clinical results values are fed in, the prediction of whether the person has a heart disease or not would have a accuracy of just under 90%.

REFERENCES

[1] Burden of Cardiovascular Disease in Asia: Big Challenges and Ample Opportunities for Action and Making a Difference, Yuling Hong, DOI: 10.1373/clinchem.2009.125369 Published July 2009

[2] Heron, M. Deaths: Leading causes for 2017. National Vital Statistics Reports;68(6). Accessed November 19, 2019.

[3] Benjamin EJ, Muntner P, Alonso A, Bittencourt MS, Callaway CW, Carson AP, et al. Heart disease and stroke statistics—2019 update: a report from the American Heart Association. Circulation. 2019;139(10):e56–528.

[4] Fryar CD, Chen T-C, Li X. Prevalence of uncontrolled risk factors for cardiovascular disease: United States, 1999–2010 pdf icon[PDF-494K]. NCHS data brief, no.

103. Hyattsville, MD: National Center for Health Statistics; 2012. Accessed May 9, 2019.

[5] https://archive.ics.uci.edu/ml/datasets/Heart+Disease

[6] D. Medhekar, M. Bote, and S. Deshmukh, “Heart disease prediction system using naive bayes,” International Journal of Enhanced Research In Science Technology & Engineering, vol. 2, no. 3, pp. 1–5, 2013.

[7] R. Das, I. Turkoglu, and A. Sengur, “Effective diagnosis of heart disease through neural networks ensembles,” Expert systems with applications, vol. 36, no. 4, pp. 7675–7680, 2009.

[8] A. Chen et al., “HDPS: Heart disease prediction system,” in Computting in Cardiology, Hangzhou, China: IEEE, 2011, pp. 557–560.

[9] S. Pouriyeh et al., “A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease,” in Proceedings of IEEE Symposium on Computers and Communications (ISCC). Heraklion, Greece: IEEE, July 2017, pp. 204–207.

[10] S. Palaniappan and R. Awang, “Intelligent heart disease prediction system using data mining techniques,” in IEEE/ACS International Conference on Computer Systems and Applications. Doha, Qatar, March 2008, pp. 108–115.

[11] https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

[12] https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall

[13] https://deepai.org/machine-learning-glossary-and-terms/f-score