International Conference on Intelligent Technologies & Science - 2022
(ICITS-2022)

Machine Learning Prediction of Cardiac Arrest and Recommendations for Lifestyle Changes to Prevent It

Neeraj Kumar[1], Sushil Kumar Maurya[2]
Computer science and Engineering Department,
Bansal Institute of Engineering and Technology, Lucknow
nk364987@gmail.com, maurya.susheel@gmail.com

Abstract: The main problems with earlier study were a skewed distribution of the data and greater numbers caused by incorrect pre-processing of the data. As a result, the model would be biassed because of the skewness of the data, processing the data would take longer since it was not scaled, and it would be more difficult to train the model to understand the linear and non-linear relationships between the features. Without even hyper-tuning the models, the data were used to train them directly. This makes it more challenging for the models to respond to training data because they would be built using default parameters. The most crucial step in any learning process is feature extraction. The heart disease dataset was subjected to machine learning algorithms in this study, and we suggested a recommendation method based on demographic data like age, sex, cholesterol level, etc. To help the patient recover, the system will also suggest some dietary and exercise changes.

Keywords-Heart Attack, Machine Learning, KNN, Random Forest, Artificial Neural Networks.

1. Introduction:

In the past 40 years, India has seen a remarkable decrease in communicable diseases while experiencing a fourfold increase in mortality and morbidity from non-communicable diseases. Mortality and morbidity from non-communicable diseases have increased four times in relation to heart disease. The detailed estimates of cardiovascular diseases published in The Lancet and its affiliated journals show an increase in their frequency in every Indian state between 1990 and 2016, but there is significant regional heterogeneity. In India, the prevalence of heart disease grew across the board from 1990 to 2016, by more than 50%. Compared to 15.2% in 1990, heart disease and stroke together caused 28.1% of all deaths in India in 2016. Heart disease caused 17.8% of all fatalities and stroke caused 7.1% of all deaths. Men were much more disabled by heart disease than women were, although stroke-related impairment was similar for both sexes.

Cardiovascular disease-related deaths increased from 13 lakh in 1990 to 28 lakh in 2016. Cardiovascular disease prevalence has increased from 2.57 crore cases in 1990 to 5.45 crore cases in 2016. Kerala, Punjab, and Tamil Nadu had the greatest prevalence rates, followed by Andhra Pradesh, Himachal Pradesh, Maharashtra, Goa, and West Bengal. People under the age of 70 made up more than half of all cardiovascular disease fatalities in India in 2016. This percentage was highest in less developed states, which raises serious concerns about the difficulties facing the health systems.

All Indian states must take immediate measures to reduce the number of early cardiovascular disease fatalities in the economically productive age groups, according to the experts.

The study demonstrates that each state's content must be appropriately reflected in the response. This study will assist in allocating health system resources to optimise impact through early prevention and efficient treatment by placing the spotlight on specific disease loads that each state must target. Professor K Srinath Reddy, President of the Public Health Foundation of India and a joint senior author of the series, clarified. Dr. Sourya Awaminathan and Dr. Lalit Dandora are additional joint senior authors.

2. Related Work:

For the prediction of coronary sickness, Sharma Purushottam et al. [2] presented the c45 rule (choice tree) and midway tree technique. The Cleveland Clinic dataset, which is available on the UCI, was used for this study. In order to construct a decision tree with the aid of a slope climbing calculation and thorough calculation, numerous principles were discussed in this work. The limits used for slope climbing calculations were limit, minItemsets (least number of leaves under a hub), certainty (insignificant certainty by a hub to be considered as a hub in the tree, esteem used was 0.25), and minItemsets (it was utilised for discovering best subset under a hub assuming the worth is not as much as edge, a thorough calculation is utilised or, more than likely slope climb calculation is utilized, esteem is 10). The KEEL and WEKA tools were used to complete the paper. In contrast to the decision tree's accuracy, which was 73.5, the proposed system's results revealed precision of 86.7%. The information used to create the model was not checked for skewness or missing properties, which is a shortcoming of the analysis. The skewness of the data would cause the model to be either over- or under-fitted. K-Fold cross approval was not used for the results and examination. By using the mean of the qualities, the missing attributes were replaced.

A Particle Swarm Optimization (PSO) combined with Ant Colony Optimization (ACO) and Fast Correlation-Based

Feature Selection (FCBF) based upgraded AI classifier was proposed by Youness Khourdifi et al. in [3]. The UCI coronary sickness data was the informational collection that was used. The execution used the WEKA gadget. K-Nearest Neighbors, Support Vector Machine, Naive Bayes, Random Forest, and a Multilayer Perception | Artificial Neural Network developed by PCO and ACO were the characterisation models used. After simplifying, K-Nearest Neighbors and Random Forest were the most effective models. The dataset was preprocessed using Quick Correlation-Based Feature Selection, which involved reducing measurement and removing redundant data. Following the simplification, K-Nearest Neighbors achieved an accuracy of 99.6%, SVM 83%, Random Woods 99.7%, NB 85%, and MLP 90%. The examination was limited by the fact that it made use of the biassed and incomplete uci Cleveland dataset. The component determination measure using subterranean insect settlement streamlining uses the skewed dataset; nevertheless, no such effort is done with regard to the AI computation. For the purpose of determining the provisions, the insect state streamlining divides the data into various more compact pieces, following which relationship bunches are created. Due to class inequality, there is a greater risk of overfitting for the AI models used, such as SVM, KNN, Logistic Relapse, and ML.

A Decision Emotionally Supportive Network Using Fake Neural Organization was proposed by Costa W.L et al (ANN). The ANN being used only has one hidden layer. The improvement was completed by using Nesterov's energy streamlining agent to find the global least. The secret layer was composed of 6 neurons and served as the initiation work with a learning rate of 0.28. The Cleveland Clinic dataset, which is available in the UCI Machine Learning repository, was used in this study. The model's accuracy, review, and f1score are all 0.91, and its exactness is 90.76%. The model's hyper boundary adjustment was done using an Amazon EC2 cloud worker. The examination was hampered by the fact that the skewness of the data was left uncorrected and that the middle of the attributes were used to fill in the gaps. The bounds were hyper-tuned to work the model. The sigmoid capacity was used at a 0.28 learning rate. The sigmoid capacity would be hit in a nearby least and hence be unable to determine an optimal value. Additionally, because to hyper boundary adjustment, the model's performance for fresh data as it is tailored for the specific example couldn't be better.

3. Methodology:

Ada Boosting

AdaBoost is an abbreviation for Adaptive Boosting. It was proposed in 1996 by Yoav Freund and Robert Schapire [16]. It was the first successful algorithm for ensemble boosting classification. It is also called Discrete AdaBoost because it is applied usually to classification problems. It is best for converting weak learners to strong learners. AdaBoost classifier combines a number of poor performing classifiers in order to build a strong classifier builds a strong classifier which has high degree of accuracy. The concept behind this algorithm is to set the weight of each classifier and training data in each iteration such that it yields correct predictions for even unusual observations. We can use any machine learning algorithm as a base classifier if and only if it assigns weights to the data in the training set. AdaBoost should satisfy the following two conditions:

- Training of base classifiers must be done on a number of weighted training samples.
- In each iteration, algorithm tries to minimize the training error and hence provide the best fit for these samples.

Working of AdaBoost

Let there be a labeled train dataset DS consisting of n tuples (X1, y1), (X2, y2)…….(Xn,yn). A tuple Xi is assigned a class label yi. Initially the algorithm assigns equal weight to each tuple which is equal to the probability of occurrence of each tuple and is calculated as 1/d. If we require k classifiers for an ensemble, the algorithm must run through k iterations. In the ith iteration, we take tuples from the training set D to form a sample Di of size d. Similar to bagging, we use sampling with replacement. The weight of the tuple determines its chances of being selected in the sample. The training tuples of DS_i constitute the classifier model M_i. Then we calculate the error by using DS_i as the test set. The predictions of the test set are used to modify the weights of the tuples for the next iteration. The weight of the correctly classified tuples is reduced and that of misclassified tuples is increased. This is how sequential classifiers are built with an improvement over the previous one. In order to determine the error rate of the model Mi, we find the sum of weights of all tuples in Di that were misclassified i.e.,

$$\mathbf{Error(Mi)} = \sum_{\mathbf{j}}^{\mathbf{d}} \mathbf{wj\ X\ err\ (Xj)}$$

where err (Xj) is the misclassification error of tuple Xj. The value of err (Xj) is equal to 1 if the tuple Xj was misclassified else it is equal to 0. We ignore the classifier Mi if its error is so poor that it exceeds 0.5. In that case we try to generate a new classifier Mi from the dataset Di by generating a new sample. According to the error the weights of model Mi are updated. If the tuple is correctly classified in i^{th} iteration the current weight of that tuple is multiplied by (1- error(Mi)). Thereafter normalization of the weights is done so that the sum of the weights of all the tuples remains the same as it was initially.

ADASYN Algorithm

Calculate the ratio of minority to majority examples using:
d=ms/ml

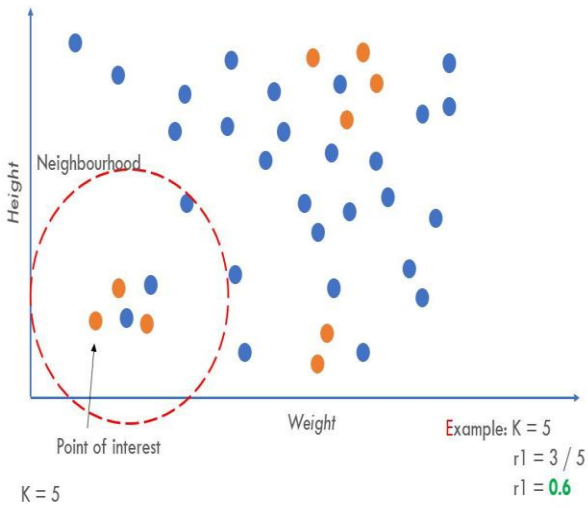

Fig. 1. k-Nearest Neighbour

Normalize the r_i values so that the sum of all r_i values equals to 1.

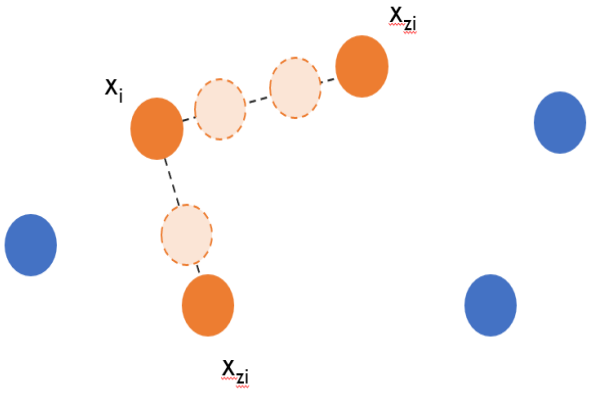

Fig. 2. Linear Combination of Xi and Xz

There are two ways by which we have analyzed the accuracy/error of Classification Models. Accuracy refers to the number of tuples in test data that have been correctly predicted by the learning model. One method that I have used to measure accuracy is by means of Confusion Matrix and the second one is by means of Classification Report. I will discuss both the methods here and then depict the results for both the classification problems for Decision Tree Classifiers, Random Forest classifiers and AdaBoost Classifiers.

4. Result and Discussion:

For implementation of the proposed model python programming language is used. Python is a popular technology due its versatility of been used in the field of web site development, app development, IoT (internet of things) and in the field of research and analytics. For implementation of our model we have used the various tools of python like scikit-learn, pandas, imblearn, matplotlib, seaborn, jupyter notebook. For the frontend is developed by using the tkinter tool available in python. The dataset is obtained from the www.kaggle.com website. Kaggle is a website which provides a platform for researchers and data scientist, it offers various dataset for research purpose.

The result shows that the optimized random forest classifier has outperformed all the other models. The accuracy of the optimized machine learning model is 90% where the accuracy of SVC is 69% that is it is a under fitted model, the decision tree classifier resulted an accuracy of 76 % which is better in comparison of the SVC but under fitted model when compared to the optimized random forest classifier, and the accuracy of Adaboost classifier is 78% which is better than bot SVC and Decision tree but not better than the proposed model. The result is shown in the figure 3.

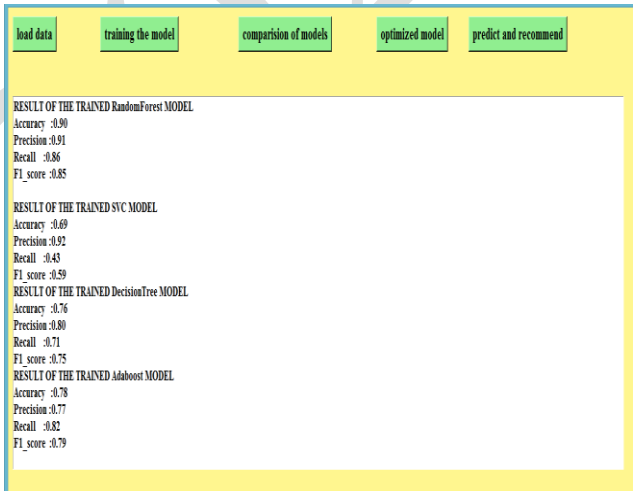

Fig 3: result analysis of the models

The result of every model is visualized by using the bar graph plotted with the help of matplotlib library. The graph is shown in the figure 4.

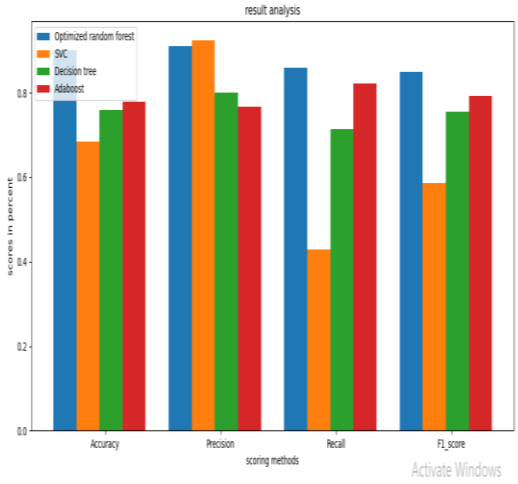

Fig 4. Graphical representation of result of the models.

The figure 5 shows the user interface for the prediction system. The user needs to fill the details required in the columns available in the left side of the GUI. After filling the details in the GUI one needs to click the predict button firstly which make sure that all the details are filled then the result is shown in the left side of the GUI. The result includes both the presence and absence result along with the recommendations based on the irregularities. It will recommend food and lifestyle changes to the patient in order to cure the patient.

Fig 5: User interface for the prediction and recommendation

5. Conclusion:

In the proposed work, we have improved the prediction model for heart disease in patients. The model was evaluated against SVC and decision tree classifier, two other machine learning techniques. All other machine learning models were outperformed by the suggested model. The prediction accuracy of this model is 90%. And we've put in place the first-ever recommendation system. This would instruct the user on the actions to take to combat reversible heart illnesses. The Cleveland dataset is utilised to create the model. By combining the naive Bayes technique, ADASYN over sampler, and Pearson correlation, an appropriate feature extraction mechanism was created. The suggested approach advises patients to alter their lifestyles in order to become better.

References:

[1]. Coronary heart disease and risk factors in India – On the brink of an epidemic?, M.N. Krishnan Indian Heart J. 2012 Jul; 64(4): 364–367. Doi: 10.1016/j.ihj.2012.07.001

[2]. Sharma Purushottam, Dr. Kanak Saxena, Richa Sharma, "Heart Disease Prediction System Evaluation using C4.5 Rules and Partial Tree", Springer, Computational Intelligence in Data Mining,vol.2, 2015, pp.285-294

[3]. Khourdifi, Youness and M. Bahaj. "Heart Disease Prediction and Classification Using Machine Learning Algorithms Optimized by Particle Swarm Optimization and Ant Colony Optimization." International Journal of Intelligent Engineering and Systems 12 (2019): 242-252.

[4]. Costa W.L., Figueiredo L.S., Alves E.T.A. (2019) "Application of an Artificial Neural Network for Heart Disease Diagnosis". In: Costa-Felix R., Machado J., Alvarenga A. (eds) XXVI Brazilian Congress on Biomedical Engineering.[online] IFMBE Proceedings, vol 70/2. Springer, Singapore. Availabe: https://doi.org/10.1007/978-981-13-2517-5_115

[5]. Amarbayasgalan T., Lee J.Y., Kim K.R., Ryu K.H. (2019) "Deep Autoencoder Based Neural Networks for Coronary Heart Disease Risk Prediction". In: Gadepally V. et al. (eds) Heterogeneous Data Management, Polystores, and Analytics for Healthcare. DMAH 2019, Poly 2019. Lecture Notes in Computer Science, vol 11721. Springer, Cham[online].Available: https://doi.org/10.1007/978-3-030-33752-0_17

[6]. K. G. Dinesh, K. Arumugaraj, K. D. Santhosh and V. Mareeswari, "Prediction of Cardiovascular Disease Using Machine Learning Algorithms," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), Coimbatore, 2018, pp. 1-7, doi: 10.1109/ICCTCT.2018.8550857.

[7]. S. Mohan, C. Thirumalai and G. Srivastava, "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques," in IEEE Access, vol. 7, pp. 81542-81554, 2019, doi: 10.1109/ACCESS.2019.2923707.

[8]. N. L. Fitriyani, M. Syafrudin, G. Alfian and J. Rhee, "HDPM: An Effective Heart Disease Prediction Model for a Clinical Decision Support System," in IEEE Access, vol. 8, pp. 133034-133050, 2020, doi: 10.1109/ACCESS.2020.3010511.

[9]. J. P. Li, A. U. Haq, S. U. Din, J. Khan, A. Khan and A. Saboor, "Heart Disease Identification Method Using Machine Learning Classification in E-Healthcare," in IEEE Access, vol. 8, pp. 107562-107582, 2020, doi: 10.1109/ACCESS.2020.3001149.

[10]. Apurb Rajdhan , Avi Agarwal , Milan Sai , Dundigalla Ravi, Dr. Poonam Ghuli, 2020, Heart Disease Prediction using Machine Learning, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 04 (April 2020)

[11]. Shah, Devansh & Patel, Samir & Bharti, Drsantosh. (2020). Heart Disease Prediction using Machine Learning Techniques. SN Computer Science. 1. 10.1007/s42979-020-00365-y.

[12]. RAJESH, N et al. Prediction of Heart Disease Using Machine Learning Algorithms. International Journal of Engineering & Technology, [S.l.], v. 7, n. 2.32, p. 363-366, may 2018. ISSN 2227-524X.

[13]. Wikipedia, Bayes' theorem [online]. Available: https://en.wikipedia.org/wiki/Bayes%27_theorem

[14]. Kim, Y.J., Saqlian, M. & Lee, J.Y. "Deep learning–based prediction model of occurrences of major adverse cardiac events during 1-year follow-up after hospital discharge in patients with AMI using knowledge mining". Pers Ubiquit Comput (2019)[online].Available: https://doi.org/10.1007/s00779-019-01248-7

[15]. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.