PREDICTION OF HEART DISEASE USING ARTIFICIAL NEURAL NETWORK

SHAHID MEHMOOD AWAN*, MUHAMMAD USAMA RIAZ2, ABDUL GHAFFAR KHAN3

1Department of Software Engineering, University of Management and Technology, Lahore, Pakistan
Email: shahid.awan@umt.edu.pk

ABSTRACT. Heart disease is increasing rapidly due to number of reasons. If we predict cardiac arrest (dangerous conditions of heart) in the early stages, it will be very helpful to cured this disease. Although doctors and health centres collect data daily, but mostly are not using machine learning and pattern matching techniques to extract the knowledge that can be very useful in prediction. Bioinformatics is the real world application of machine learning to extract patterns from the datasets using several data mining techniques. In this research paper, data and attributes are taken from the UCI repository. Attribute extraction is very effective in mining information for the prediction. By utilizing this, various patterns can be derived to predict the heart disease earlier. In this paper, we enlighten the number of techniques in Artificial Neural Network (ANN). The accuracy is calculated and visualized such as ANN gives 94.7% but with Principle Component Analysis (PCA) accuracy rate improve to 97.7%.

Keywords: heart disease; artificial neural network; principal component analysis

1. Introduction. The heart is the main organ of the human body. If it stops working correctly our whole body can get infected and in most cases, this disease leads to death. From 1990 to 2013 41% cardiovascular disease increased globally. Life is not possible without a functioning heart and so human beings cannot survive without it. It controls our blood pressure, body temperature and many other vital health aspects including oxygen regulation in the blood. Some of the most common factors that contribute to the heart disease include family genetic problems which come from generation to generation, high blood pressure, Cholesterol, Sex, Age mostly above 30, Poor diet, Calcium rate, a blood vessel of heart got over stretched, lack of exercise etc.

Heart disease is a persistent apprehension of the whole world as it is frequently increasing in ratio per person. The heart is the main regulating organ of the human body.

According to the Institute for Health Metrics and Evaluation (IHME) death rate due to this disease has shown an alarming 41% increase from 1990 to 2013 and a 9.83% further increase in 2014. Survey analysis of (WHO) shows 17 million deaths due to heart disease [5].

If any disease gets predicted at earlier stages, then it becomes easier to find or apply a cure before it gets dangerous. Just like that, if a heart disease is predicted earlier, it becomes easier to find a cure. This type of prediction problem, related to medical diagnosis comes under the branch of computer science i.e., bioinformatics. In bioinformatics, for prediction, we need previous historical data of the patient and some pattern matching algorithms which can be trained on the respective data to generate results. In clinics and health centers, a lot of patients are being diagnosed daily so there is a lot of data collection, for example, in the form of Medical Reports. Collection of data from these reports can be used for extracting knowledge from data. This can be done by a well-known model of bioinformatics i.e., Knowledge Discovery from Data (KDD).
A. Neural Networks. A single neural is also known as a perceptron. The human brain consists of more than ten billion neurons. In ANN, there are two parts of a single perceptron, one is the sum function and the other is a transfer function. Sum function is like almost every model of the ANN but there are different transfer functions with different natures according to their theories.

\[ Z = \sum_{i=1}^{n} X_i W_i \]  

(1)

After calculating the sum of the product of all inputs and weights using equation (1). The other step is transfer function. The transfer function is of different types such as sigmoid function, sign function, tan hyperbolic function, ReLU function and a step function.

\[ y = \frac{1}{1+e^{-z}} \]  

(2)

Equation (2) shows the sigmoid function. Its output value remains between 0 and 1. While Sign function is just replacing any positive number to +1 and any negative number to -1 as shown in equation (3)

\[ f(x) = \begin{cases} -1, & x < 0 \\ +1, & x \geq 0 \end{cases} \]  

(3)

Tan Hyperbolic function (4) gives the output in the form of a value from -1 and 1. It's also used for the continues value output as a signed function, but the difference is that it also gives the negative value.

\[ \tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}} \]  

(4)

ReLU function is used in the convolutional neural network a type of deep learning neural network for images. It just replaces negative values with the zero as shown in (5).

\[ \text{ReLU}(x) = \max(0, x) \]  

(5)

The basic simplest transfer function is the step function. It is used for the linear problems. As shown in equation (6) controls the y-intercepts.

\[ f(x) = \begin{cases} 0, & x < \theta \\ 1, & x \geq \theta \end{cases} \]  

(6)

Perceptron is the mathematical model of the neuron. It is a single neuron connected to inputs and it has only one output. It is used to classify linear problems. It has two parts sum function and transfer function.

\[ \begin{array}{c} x \\
1 \\
x2 \\
x3 \\
\sum \\
\end{array} \xrightarrow{\begin{array}{c} w1 \\
w2 \\
w \end{array}} \begin{array}{c} \text{Output} \\
\end{array} \]

Figure 1 Perceptron Model showing x1, x2, x3 inputs with corresponding weights w1, w2, w3. There are two parts of the perceptron one is sum function and another one is step function as the activation function. The sum function is as shown in (1) and step function is shown in (6).

B. Multi-Layer Perceptron. An artificial neural network (ANN), also known as a short “neural network” (NN) and Multi-Level Perceptron, is a mathematical model or computational model based on the neural network found in human anatomy. It is built on the analogy of a single neuron of the human brain. It has multiple perceptrons at multiple levels. Every perception has its weight which affects the value and output. A hidden layer where value is predictable and last output layer where result is generated.
C. Feed Forward Neural Network. Feed Forward Neural Network (FFNN) is the simple neural network with multiple layers as shown in Figure 1. It has input layer which contains the neurons equal to the number of features, hidden layers, and output layers. Number of the neuron of output layer usually equal to the number of classes. Back Propagation is used to optimize the weights to reduce error.

\[
E = \frac{1}{2} \sum_{i=1}^{n} (\text{expected} - \text{actual})^2
\]  

(7)

Mean Square Error (7) is calculated and then weights are updated according to the gradient decent of each weight.

D. Recurrent Neural Network. Recurrent Neural network (RNN) is similar to the FFNN with the difference that it contains one loopback cycle. There are directed connections i.e. a connection goes from hidden layer to input layer, hidden to hidden and vice versa (has many variations).
E. Nonlinear Autoregressive Exogenous. Nonlinear autoregressive exogenous model (NARX) model is also like the FFNN and it’s a type of RNN. The difference is that the input to hidden layer is from input layer and the output layer with some delays.

It gives much more accuracy when the data is in time series like Hidden Markov’s Model (HMM) does. Figure 4 shown NARX model.

F. Cascade-Forward Neural Networks. Cascade forward neural networks (CFNN) is like the FFNN with the only difference being that all input neurons have significantly more connections to every other neuron of every layer of the model.

This research is related to heart disease prediction using ensemble technique. We use KDD and Neural Network techniques for the prediction of heart disease, so we can compare the results before PCA and after PCA. We try to find the results, so we can predict a heart disease for a patient. Also, want to prove that algorithm accuracy increases after applying PCA and the ensemble technique. The paper is drieved in following sections.

2. Related Work. In Paper [13], uses the data [19] from UCI data repository. They use a fuzzy genetic algorithm for pre-processing and then they test their data on naïve Bayes, decision List Tree, and nearest neighborhood. Accuracy rate for naïve Bayes is 96.5%, Decision Tree 99.2% and for classification via clustering 88.3%. An Intelligent Heart Disease Prediction System (IHDPDS) is developed by using data mining techniques. Naïve Bayes, Neural Network, and Decision Trees as a solution were proposed by Sellappan Palaniappan et al. [20]. Every technique has its own particular quality to get proper outcomes. To construct this framework, shrouded examples and relationships between them are utilized. It is online, easy to understand and expandable. To develop the multi-parametric segment with straight and nonlinear characteristics of HRV (Heart Rate Variability) a novel strategy was proposed by Heon Gyu Lee et al. [21]. To finish this, they have used a couple of classifiers e.g. Bayesian Classifiers, CMAR (Classification in perspective of Multiple Association Rules), C4.5 (Decision Tree) and SVM (Support Vector Machine).In [24], it analyzed about the desire of the coronary sickness using data mining frameworks like decision trees, Naïve Bayes, Neural Networks, gathering and Genetic Algorithm. His gave the examination of various systems used as a part of the desire of the coronary disease. As for other heart diagnosis problems, classification systems have been used for heart disease diagnosis problem, too. When the studies in the literature related with this classification application are examined, it can be seen that a great variety of methods were used which reached high classification accuracies using the dataset taken from the UCI machine learning repository. Among these, [8] ToolDiag, RA obtained 50.00% classification accuracy by using IB1- 4 algorithms. [4] WEKA, RA obtained a classification accuracy of 58.50% using InductH algorithm while ToolDiag, RA reached to 60.00% with RBF algorithm. [8] Again, WEKA, RA applied FOIL algorithm to the problem and obtained a classification accuracy of 64.00%. [8] MLP+BP algorithm that was used by ToolDiag, RA reached to 65.60%. [8] The classification accuracies obtained with T2, 1R, IB1c and K* which were applied by WEKA, RA are 68.10%, 71.40%, 74.00% and 76.70%, respectively. [8] Robert Detrano used logistic regression algorithm and obtained 77.0% classification accuracy. The result of this fuzzy expert system in 79% as a well as the expert did. Moreover, Cheung utilized C4.5, Naive Bayes, BAND and BNNF algorithms and reached the classification accuracies 81.11%, 81.48%, 81.11% and 80.96%, respectively [8].

3. Methodology. As we are following the model of the KDD which involves the collection of data sets, its preprocessing, building patterning matching classifier model with ANN. After training them we will ensemble them to get the better decision we use Weka tool for the implementation of the methodology as shown in the figure of “implementation design”. Figure 5 In first step data is preprocessed for the training using PCA algorithm then the processed data is input in the ANN.

4. Datasets. As we are following Heart disease diagnosis medical reports data used “cleaver land heart data” is taken from UCI data repository [16]. As shown in table 1 only 13 attributes are select for the research such as age (years), sex (1 = male; 0 = female), cp, tretbps (resting blood pressure in mm Hg), chol (cholesterol in mg/dl), fbs(fasting blood sugar), resectg (resting electrocardiographic results), oldpeak (ST depression induced by exercise relative to rest), slope(the slope of the peak exercise ST segment ), ca(number of major vessels), thal(reversible defect) and Num(Class attribute 1=heart disease; 0 for no heart disease)
5. Data Pre-Processing. Data pre-processing evolves the step of the feature extraction and feature selection. Feature selection means selection of the feature from the big set of features to the small and relevant set of features, but feature extraction transforms the big set of features to the new generated small set of features. Feature selection gives original features, but feature extraction gives transformed features. We have used feature extraction by one of the well-known methods, Principle component analysis (PCA). Figures and Tables

A. Principle Component Analysis. Principle component analysis (PCA) is used for transforming the big dimension to lower one. For this Correlation Matrix Eigenvalues and Eigen Vectors is calculated based on which new feature is extracted with more relevancy.

6. Perceptron with LEARNP. A perceptron model is made to elaborate working. It contains only a single neuron with 13 inputs inside and a step function is used as a transfer function (6). There is a bias also which controls the y-intercepts and to each input, there is a separate weight
For learning, LEARNP algorithm is used. LEARNP algorithm updates weights by taking the product of input and error. If error is positive then the change in weight is positive.

\[
W_{\text{new}} = W_{\text{old}} + \beta \times x \times \epsilon
\]  

(8)

In equation (8) \(\beta = -1\) when error (\(\epsilon\)) is negative, \(\beta = 1\) when error is positive, and \(\beta = 0\) when \(\epsilon\) is 0. This weight update happens in every iteration of every epoch.

4. **Confusion Matrix.** Confusion Matrix is used to calculate and show the true positive, true negative, false positive and false negative. i.e.,

| True Positive | True Negative |
|---------------|---------------|
| **Predicted Positive** | 135 | 94 |
| **Predicted Negative** | 4 | 70 |

only 23.1% of the times 0 is required and 0 is achieved and 1.3% of records related to when required is 0 and 1 is achieved. Also, there are 31.0% records when 1 is desired and 0 comes and 43.6% when 1 is desire and 1 is achieved. So, the accuracy becomes 67.7%.

7. **Feed Forward Neural Network.** A Feed Forward Neural Network (FFNN) consists of the multiple layers of the neurons the first layer has the input layer, the second is hidden and last one is output layer. Each neuron of every layer is connected to every neuron of the next layer. 13 input neurons in input layer because of 13 attributes, 10 neurons in the hidden layer and 1 in the output layer [32, 33, 34]

![Feedforward Neural Network](image)

**Figure-8:** Feedforward Neural Network diagram with 13 input neurons, 10 in hidden layers and 1 in output layer.

A. **Confusion Matrix.** By the analysis, 48.5% of the times 0 is required and the result 0 is achieved. 7.9% of records are achieved when required is 0 and the result comes 1. Also, there are 5.6% records when 1 is desired and 0 comes and 38.0% when 1 is desired and 1 is actual. So, the accuracy becomes 86.5%.

| True Positive | True Negative |
|---------------|---------------|
| **Predicted Positive** | 115 | 17 |
| **Predicted Negative** | 24 | 147 |

8. **FFNN with Resilient Propagation.** Feedforward neural network with resilient propagation (FFNN - RP) is analyzed. For optimizing FFNN, Resilient Propagation is the weighs optimizing technique [24]. Model has
1 hidden layers and 13 input layer there is one output neuron. For the Optimization and learning of the network, Resilient Propagation "rprop" is used

**A. Confusion Matrix.** This analysis results only 47.9% of the times 0 is required and the result 0 is achieved. 10.9 % of records are achieved when required is 0 and the result comes 1.

|                | True Positive | True Negative |
|----------------|---------------|---------------|
| **Predicted Positive** | 106           | 19            |
| **Predicted Negative**  | 33            | 145           |

Also, there are 6.3% records when 1 is desired and 0 comes and 35.0 % when 1 is desired and 1 is the result. So, the accuracy becomes 82.8%.

**9. Recurrent Neural Network**

Recurrent Neural Network (RNN) is simpler than the FFNN with the difference being a loopback from the hidden layer to the input layer i.e. the output of the hidden layer again goes as an input with some delays.

![Network Diagram](image)

**Figure- 9: Network Diagram**

**A. Confusion Matrix.** 49.5% of the times 0 is required and the result 0 is achieved. 17.9% of records are achieved when required is 0 and the result comes 1.

|                | True Positive | True Negative |
|----------------|---------------|---------------|
| **Predicted Positive** | 115           | 14            |
| **Predicted Negative**  | 24            | 150           |

Also, there are 4.6% records when 1 is desired and 0 comes and 38.0 % when 1 is desired and 1 is the result. So, the accuracy becomes 87.5%.

**10. Cascade Forward Neural Networks.** Cascade forward neural networks (CFNN) is like the FFNN the only difference is that the all input neurons has some more connections to every other neuron of every layer of the model

**A. Confusion Matrix.** 51.2% of the times 0 is required and the result 0 is achieved. 2.3 % of records are achieved when required is 0 and the result comes 1
|               | True Positive | True Negative |
|---------------|---------------|---------------|
| Predicted Positive | 112           | 14            |
| Predicted Negative   | 27            | 150           |

Also, there are 3.0% records when 1 is desired and 0 comes and 43.6% when 1 is desired and 1 is the result. So, the accuracy becomes 94.7%.

11. NARX Model. NARX model is also like the FFNN and it’s a type of the RNN the difference is that the input to hidden layer is from input layer and the output layer with some delays

![Network Diagram](image)

**Figure-10**: Network Diagram

A. Confusion Matrix. There are 4.6% records when 1 is desired and 0 comes and 37.0% when 1 is desired and 1 is the result. So, the accuracy becomes 86.5%.

|               | True Positive | True Negative |
|---------------|---------------|---------------|
| Predicted Positive | 112           | 14            |
| Predicted Negative   | 27            | 150           |

Only 49.5% of the times 0 is required and the result 0 is achieved. 8.9% of records are achieved when required is 0 and the result comes 1.

12. Cascade Forward Neural Networks With PCA. After getting results we train to find if this model is giving more accuracy. We achieve higher accuracy levels by applying PCA on data and giving the model, which really increases the accuracy.

A. Confusion Matrix. The following Confusion Matrix observation only 52.8% of the times 0 is required and the result 0 is achieved. 1.0% of records are achieved when required is 0 and the result comes 1.

|               | True Positive | True Negative |
|---------------|---------------|---------------|
| Predicted Positive | 136           | 4             |
| Predicted Negative   | 3             | 160           |

Also, there are 1.3% records when 1 is desired and 0 comes and 44.9% when 1 is desired and 1 is the result. So, the accuracy becomes 97.7%.
13. Results

| Algorithm     | Sensitivity | Precision | F1 Score | Accuracy |
|---------------|-------------|-----------|----------|----------|
| Perceptron    | 97.12       | 58.95     | 73.37    | 67.7     |
| LEARNP        |             |           |          |          |
| FFNN          | 82.73       | 87.12     | 84.87    | 86.5     |
| FFNN RP       | 76.26       | 84.80     | 80.03    | 82.8     |
| RNN           | 82.73       | 89.15     | 85.82    | 87.5     |
| NARX          | 80.58       | 88.89     | 84.53    | 86.5     |
| CFNN          | 80.58       | 88.89     | 84.53    | 94.7     |
| CFNN PCA      | 97.84       | 97.14     | 97.49    | 97.7     |

Fig. 1. Example of a figure caption. (figure caption)

Conclusion. For clear understanding, results/prediction rate for each of the algorithm are summarized in a tabular form as well as graph representations and the best prediction rate obtained in each of the techniques/methodologies is summarized by studying, analyzing and performing an ensemble base technique. Different variations of ANN give different accuracy rates. This shows the accuracy of 'before Applying PCA' and 'after applying' is different i.e., before 94.7% and after applying PCA is up to 97.7%. A huge different accuracy is observed. So, we can do heart disease prediction. It can be further increased by changing the setting and making them more optimized according to each algorithm and nature of data.

REFERENCES

[1]. Moftah, R. A., Maatuk, A. M., & White, R. (2016, September). Methods to access structured and semi-structured data in bioinformatics databases: A perspective. In Engineering & MIS (ICEMIS), International Conference on (pp. 1-5). IEEE.
[2]. Rahm, E., & Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. the VLDB Journal, 10(4), 334-350.
[3]. Wadler, P. (1987, October). Views: A way for pattern matching to cohabit with data abstraction. In Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages (pp. 307-313). ACM.
[4]. Gandapur, A. S. K., Yar, S., & Majid, T. (2012). Study of risk factors in coronary heart disease. Pakistan Heart Journal, 21(4).
[5]. Mitchell, T. M. (1997). Machine learning (mcgraw-hill international editions computer science series).Hotelling, Harold. "Analysis of a complex of statistical variables into principal components." Journal of educational psychology 24.6 (1933): 417.
Polat, K., Güneş, S., & Tosun, S. (2006). Diagnosis of heart disease using artificial immune recognition system and fuzzy weighted pre-processing. Pattern Recognition, 39(11), 2186-2193.

Novakovic, J. (2009, November). Using information gain attribute evaluation to classify sonar targets. In 17th Telecommunications forum TELFOR (pp. 1351-1354).

Asim, M., & Khan, Z. Mobile Price Class prediction using Machine Learning Techniques.

Pfähringer, B., Holmes, G., & Kirkby, R. (2007, December). New options for hoeffding trees. In Australasian Joint Conference on Artificial Intelligence (pp. 90-99). Springer, Berlin, Heidelberg.

Pyle, D., 1999. Data Preparation for Data Mining. Morgan Kaufmann Publishers, Los Altos, California

Masethe, H. D., & Masethe, M. A. (2014, October). Prediction of heart disease using classification algorithms. In Proceedings of the world Congress on Engineering and computer Science (Vol. 2, pp. 22-24).

Xing, Y., Wang, J., & Zhao, Z. (2007, November). Combination data mining methods with new medical data to predicting outcome of coronary heart disease. In Convergence Information Technology, 2007. International Conference on (pp. 868-872). IEEE.

Engel, J. (1988). "Polytomous logistic regression". Statistica Neerlandica. 42 (4): 233. doi:10.1111/j.1467-9574.1988.tb01238.x

Bhatla, N., & Jyoti, K. (2012). An analysis of heart disease prediction using different data mining techniques. International Journal of Engineering, 1(8), 1-4.

Masethe, H. D., & Masethe, M. A. (2014, October). Prediction of heart disease using classification algorithms. In Proceedings of the world Congress on Engineering and computer Science (Vol. 2, pp. 22-24).

Wu, Z. H. Z. J. X., & Chen, Y. J. S. F. (2001, August). Genetic algorithm based selective neural network ensemble. In IJCAI-01: proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, Seattle, Washington.

Palaniappan, S., & Awang, R. (2008, March). Intelligent heart disease prediction system using data mining techniques. In Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on (pp. 108-115). IEEE.

Lee, H. G., Noh, K. Y., & Ryu, K. H. (2007, May). Mining biosignal data: coronary artery disease diagnosis using linear and nonlinear features of HRV. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 218-228). Springer, Berlin, Heidelberg.

Sonawale, S. A., & Ade, R. (2015). Dimensionality reduction: an effective technique for feature selection. International Journal of Computer Applications, 117(3).

Gupta, T. K., Kumar, C., Prakash, S., & Prasad, M. (2015). Dimensionality reduction techniques and its applications. Journal of Computer Science & Systems Biology, 8(3), 170.

Kavitha, R., & Kannan, E. (2016, February). An efficient framework for heart disease classification using feature extraction and feature selection technique in data mining. In Emerging Trends in Engineering, Technology and Science (ICETETS), International Conference on (pp. 1-5). IEEE.

Mohamad, M. A., Nasien, D., Hassan, H., & Haron, H. (2015). A review on feature extraction and feature selection for handwritten character recognition. International Journal of Advanced Computer Science and Applications, 6(2), 204-212.

Chitra, R., & Seenivasagam, V. (2013). Review of heart disease prediction system using data mining and hybrid intelligent techniques. ICTACT journal on soft computing, 3(04), 605-09.

[25] Mosca, Alan, and George D. Magoulas. "Adapting resilient propagation for deep learning." arXiv preprint arXiv:1509.04

Butt, A. H., Rasool, N., & Khan, Y. D. (2017). A treatise to computational approaches towards prediction of membrane protein and its subtypes. The Journal of membrane biology, 250(1), 55-76.

Butt, A. H., Rasool, N., & Khan, Y. D. (2017). A treatise to computational approaches towards prediction of membrane protein and its subtypes. The Journal of membrane biology, 250(1), 55-76.

Akmal, M. A., Rasool, N., & Khan, Y. D. (2017). Prediction of N-linked glycosylation sites using position relative features and statistical moments. PloS one, 12(8), e0181966.

Ehsan, A., Mahmood, K., Khan, Y. D., Khan, S. A., & Chou, K. C. (2018). A novel modeling in mathematical biology for classification of signal peptides. Scientific reports, 8(1), 1039.

Abid, Adnan; Hussain, Naveed; Abid, Kamran; Ahmad, Farooq; Farooq, Muhammad Shoaiab; Farooq, Uzma; Khan, Sher Afzal; Khan, Yaser Daanial; Naeeem, Muhammad Azhar; Sabir, Nabeel; "A survey
on search results diversification techniques, Neural Computing and Applications, 27(5), 1207-1229, 2016, Springer London

[30]. Khan, Y. D., Ahmad, F., & Anwar, M. W. (2012). A neuro-cognitive approach for iris recognition using back propagation. World Applied Sciences Journal, 16(5), 678-685.

[31]. Khan, Y. D., Ahmed, F., & Khan, S. A. (2014). Situation recognition using image moments and recurrent neural networks. Neural Computing and Applications, 24(7-8), 1519-1529.