



# A Novel Approach for Specification Testing on Heart Disease Detection Using Feed-Forward Neural Network

E. Naresh<sup>1</sup> · S. L. Shiva Darshan<sup>1</sup> · N. N. Srinidhi<sup>2</sup> · M. Niranjanmurthy<sup>3</sup> · P. Dayananda<sup>1</sup>

Received: 12 February 2023 / Accepted: 19 July 2023  
© The Author(s) 2023

## Abstract

The proposed work highlights the importance of testing in machine-learning applications and the ensuing need to increase model quality to decrease the likelihood of errors. The proposed work considers patient health information that may be utilized for decision-making or prediction utilizing various computations, and in this instance, it emphasizes the development of artificial neural networks with the multilayer perceptron method to forecast cardiac abnormalities. The dataset employed in this work includes information about the patient's demographics, clinical measures, and medical background. Utilizing this labelled dataset, one may estimate if each patient has heart disease or not. We developed a GUI to collect data about a new patient to accomplish the task specified. After the model is generated, effective functionality testing was performed on the model. As a result of the testing report's assistance in identifying faults and defects, the artificial neural network's accuracy ultimately improved as a result of its improved accuracy following correction. After undergoing feature testing, the application's accuracy ascended to 96.69%, which is a higher percentage than was attained using any alternative method.

**Keywords** ANN · Functional testing · Machine learning · Heart disease · Multilayer perceptron technique

---

This article is part of the topical collection “Machine Intelligence and Smart Systems” guest edited by Manish Gupta and Shikha Agrawal.

---

✉ S. L. Shiva Darshan  
shivadarshan.sl@manipal.edu

E. Naresh  
naresh.e@manipal.edu

N. N. Srinidhi  
srinidhinagesh@gmail.com

M. Niranjanmurthy  
niruhsd@bmsit.in

<sup>1</sup> Department of Information Technology, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India

<sup>2</sup> Department of Computer Science and Engineering, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India

<sup>3</sup> Department of AI&ML, BMSITM, Avalahalli, Bengaluru, India

## Introduction

Testing is essential for every software project, whether simple or complex. Every software must go through the software lifecycle to achieve a successful outcome and simple maintenance. Most of the time is spent identifying bugs and errors that must be fixed. There are numerous types of testing, including functionality testing, system testing, unit testing, and integration testing, among others. This testing contributes to the enhancement of the proposed model, so that no crashes or problems are encountered during its execution.

Ad-hoc testing with test-driven development can also be very beneficial in reducing the number of defects [1]. One of the most common tasks in machine learning is data classification. Machine learning is one of the most important features for mining information from massive databases of enterprise operational records. Machine Learning in Medical Health Care is a rapidly growing field with enormous potential for providing prediction and a deeper understanding of medical data. Most machine-learning methods rely on a set of features that define the learning algorithm's behaviour and, in turn, influence the performance and complexity of the resulting models. In the past decade, heart disease has been the leading cause of death across the globe.

Several machine-learning techniques have been applied to the diagnosis of heart disease in the past. Neural networks and logistic regression are two of the few machine-learning techniques that have shown some success in the diagnosis of heart disease [2].

For the application, we accessed the UCI repository of datasets for machine-learning models. We are considering patient data from a Cleveland hospital containing approximately 400 patients with their names removed from a column to protect their privacy. The application under consideration falls under the umbrella of machine learning. Numerous researchers, businesses, and organizations use machine learning to improve without being explicitly programmed [3, 4]. This method relies on patterns and inference rather than explicit instructions to complete a task. There are three primary types of learning: supervised, unsupervised, and reinforcement.

In preprocessing, supervised learning considers labelled data, whereas unsupervised learning uses unlabeled data. For application modelling, we are considering artificial neural networking with multilayer perceptron. A multilayer perceptron is a very useful approach for making the model learn; for the given data set, which contains around 400 patient records, we used ReLu for forward propagation [5]. By comparing the above algorithms to various other approaches, we have overserved an improvement in the accuracy of the result.

We discovered that accuracy improved for the model once the developer was able to rectify it and backpropagation of the multilayer perceptron improved the weights which used the Adams optimizer once developer was able to rectify and backpropagation of the multilayer perceptron improved the weights which used the Adams optimizer.

We tested different epoch values until the difference in accuracy was stable. The results show increased accuracy, precision, recall, and sensitivity. Only by testing with various test scenarios can the problems that will be encountered be analyzed and understood, which can be a challenge in and of itself [6].

The remaining sections have the following distribution: the literature review talks about various papers referred to that have been proposed in this research field, followed by the methodology which consists of the proposed model and procedure approached, next is the results obtained and comparison and then finally conclusion with future enhancement.

## Literature Review

Models for the heart disease dataset have undergone extensive research and development over time. The majority of individuals research several algorithms for the diagnosis of heart disease, including neural network, nearest neighbors,

Naive Bayes, and logistic regression as well as hybrid strategies using the aforementioned algorithms [7]. The work by author [8] gives a learning-driven approach that analyzes the dataset and learns the pattern to identify the results. At first logistic regression was utilized by the author to acquire 77% exactness. Whereas another Author [9, 10] used Naive Bayes calculations on the dataset and came to the characterization correctness of 81.48% which was a good improvement. One of the authors examined looking at changed information and mining procedures in the conclusion of coronary illness patients, these systems included naive Bayes, and choice tree, and furthermore, the neural system which was used for the first time on the dataset and the outcomes demonstrated that the naive Bayes could accomplish the best precision in the determination of coronary illness patients [11].

Kmeans bunching [12] is a standout among the most well-known grouping systems; anyway beginning centroid determination is a basic issue that emphatically influences its outcomes. This paper researches applying various techniques for introductory centroid determination, for example, go, inlier, anomaly, arbitrary quality qualities, and arbitrary column strategies for the kmeans grouping method in the conclusion of coronary illness patients. Indira [13] utilizes probabilistic fake neural systems BF arrange. This which is a class of spiral premise work RB strategy is valuable for programmed design acknowledgement, non-linear mapping what's more, estimation of probabilities of class enrollment, and probability proportions. The information utilized in the tests was taken from Cleveland coronary illness database with all out 576 records and 13 restorative properties. The best exactness execution achieved 94.60%. KNearestNeighbour is a standout among the most generally utilized information mining strategies in arrangement issues [14, 15]. Its straightforwardness and moderately high assembly speed settle on it a famous decision. Anyway, a principle burden of KNN classifiers is the enormous memory prerequisite expected to store the entire example. At the point when the example is enormous, reaction time on a consecutive PC is likewise huge [16].

The fundamental activity of software engineering is software testing. The software is executed by an operation to detect mistakes or bugs. The testing methods and the software testing strategies are discussed in detail here. The techniques of developing the test instances are important phases in the testing process. Numerous test methods have to be developed to evaluate as all the bugs in the software cannot be found.

Himanshi Babbar [17] discussed the significance of software testing in the software development life cycle. Roughly 60% of resources are used for software testing. Manual or automated testing may be possible. Software testing is an activity which focuses on evaluating the program's capacity and requires that it genuinely achieves outcomes in quality.

Testing is widely divided into three levels: testing unit, testing integration, and testing system. Testing is used at this moment to create bug-free software. The most common test cases and testing technologies for error detection are defined here. Many test instances assist detection of bugs.

Software product line (SPL) engineering was created by Isis Cabral et al. [18] and has been used to improve the effectiveness and quality of software generated. With SPL engineering, engineers may methodically create families of products with clearly defined and managed reusable asset sets. The results of the two software product lines' case studies are documented. With both SPLs, the same error detection outcomes are achieved as all products can be tested. More analyses show that the grouped version of the basic path algorithm can also be used for testing subfamilies of the SPL identified by alternatives. In this technique, the feature model was used to control the selection of test cases and the test effort required can be decreased using a graphical selection algorithm to retain the deletion capability. In testing every product, errors are identified using the FIG Basis Path Method, testing only 6% or 24% of our SPL goods and only 10% of the test instances as are all products in the best situation. The Covering Array method was the most efficient non-graphical technique, which needed us we tested between 13 and 54% of goods in the same schemes. This method of failure identification can be economically adaptable in the case of the topic with only optional characteristics.

Naresh et al. [19] projected that in addition to the elevated level of process maturity, the implementation of the defect preference approach is also a most precious investment. The detection of mistakes in the development life cycle enables avoidance of the passing of mistakes from the design to the coding. Analyses conducted across three businesses demonstrate the significance of using defect avoidance methods to deliver a high-quality product. Investment in the correct DP operations rather than investing in rework that is seen as a result of uncaught failures is the main focus of quality costs' investment. There are several defect avoidance methods, techniques, and procedures. The most effective defect detection and avoidance method have proven to be software inspection. The objective of continuously achieving 99% defect-free software relies heavily on efficient methods for defect avoidance.

Antonia Bertolino [20] summarized that one observation concerns the many successful connections between software testing and other fields of study. By addressing particular issues of software testing, numerous interesting possibilities are ignored at the boundary between testing and other disciplines. Some have just been discussed here, for example, model control methods or the use of search-based methods to generate test inputs or the use of test methods to evaluate efficiency characteristics. A more holistic strategy to study tests by software readers can discover and enjoy numerous

fresh and exciting synergies across the software engineering study fields.

Rasneet Kaur Chauhan et al. [21] emphasized the software testing strategies, methodologies, principles, and tools. A software testing approach integrates different software case design techniques in a well-scheduled sequence of steps leading to effective software testing. Therefore, software test strategies for testing are essential. The strategy for software testing is usually created by experts, project managers, and software engineers. It involves four methodologies and some rules to be followed accurately. Several software testing instruments are accessible on the market. Some were used for a very long period and fresh instruments with many fresh features were also created [22]. Every software project focuses primarily on quality. Software testing techniques are methods of quality measurement. The driving force of growth and application is software testing research. It is also essential to constantly synthesize new accomplishments and new features, and suggest various ideas to support the research on system engineering software testing and facilitate fast developments in the field of software testing [23].

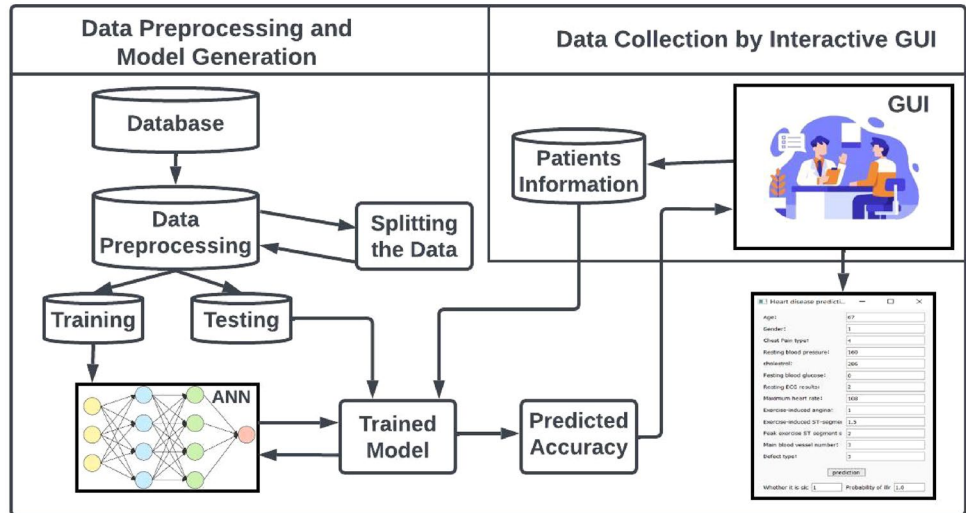
Shah et al. [24] gathered various surveys on testing that can be useful before starting the testing process and it is important to overcome the issues faced. Zain Amin and Ali [25] have applied the multilayer perceptron on a dataset from large-scale healthcare from daily-based information which can help save a life; his study shows a caesarian section for an expecting mother to do surgery that utilizes the MLP with backpropagation to determine the medical operation. Some of the main emergencies of the caesarian are immediate threat to the child or mother or no maternal but early delivery required the author has considered the dataset on 80 patients who are pregnant. The case study resulted in 95% accuracy. We can improve this by enlarging the dataset and including some more grave attributes.

## Proposed Methodology

In the proposed model, we are using the approach of neural networking that comes under artificial intelligence which is widely being used to make the model behave more intelligent, pattern recognition, optimization, associative memory, and predictions can be done using the above algorithms, and the dataset needs to be a supervised labelled set have some features that help in analysis and learning in the hidden layers of the network. The overall flow diagram of the proposed model is shown in Fig. 1.

“Shallow learning architecture” describes classifiers based on machine learning that typically have just one layer of non-linear feature transformation. Support vector machines, logistic regression, decision trees, XGB, and other systems are examples of shallow structured

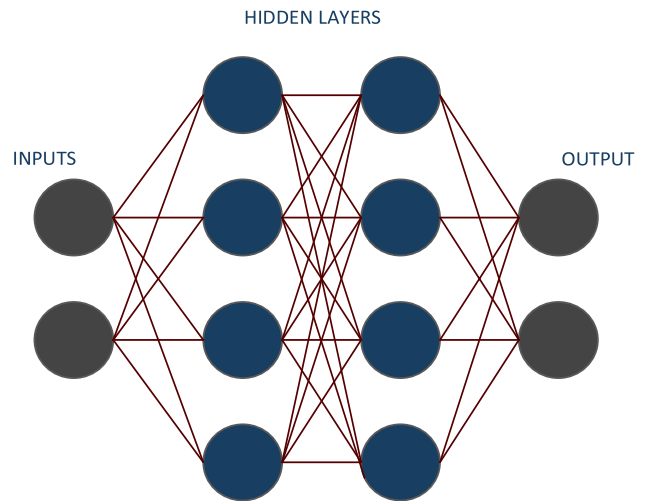
**Fig. 1** Overview of the proposed model



architectures. Even though they have demonstrated efficient performance in straightforward or well-constrained scenarios, their limited modelling and representation capabilities struggle to handle the vast training data sets. Due to their restricted capacity for learning, machine-learning models with shallow structures perform poorly as the training data grow increasingly large. An alternate solution to address this is the Artificial Neural Network (ANN). As a result of their effective learning capacity, ANN-based prediction models are proposed in this work.

The artificial neural network is a computational model that works like a human brain; this produces artificially structured neurons that gives a gateway for data to transfer. A computational system can produce a linear and non-linear response with the help of multiple, simple, highly connected elements. These elements process the information to produce a dynamical state response based on external inputs.

From Fig. 2, we can see the different hidden layers along with the input and output layer, and this is a general representation of the artificial neural network. The type of neural network that is used for our model is a multilayer perceptron. In the case of multilayer perceptron, it can learn non-linear functions and has more than one hidden layer, so that arbitrary accuracy of the activation function increases, depending on the number of neurons each layer has used. At any hidden layer neuron node, a computation takes place where it receives the input from the previous node along with an associated weight, which gets valued and assigned randomly depending on the relative importance to other input layer values. At the node, the function  $f$  is applied, and we have used the ReLu for the activation function, ReLu (rectified linear unit) generally takes a real-valued input and gives a threshold to zero, it replaces the negative values to zero



**Fig. 2** Neural network general structure

$$\text{Neuron} = (\text{weight1} * x1 + \text{weight2} * x2 + \dots + \text{bias}). \tag{1}$$

The above function is called an activation function, where the numerical inputs such as  $x1 \times x2$  and so on along with the weight associated with each node are calculated which can be used as the input in the next hidden layer if present or the output layer. Bias gives a trainable constant value addition to input values received by the neuron

$$f(x) = \max(0, x). \tag{2}$$

The above ReLu equation is used for the activation function in our model, and the general representation of the graph is shown in Fig. 3. ReLu is generally lesser

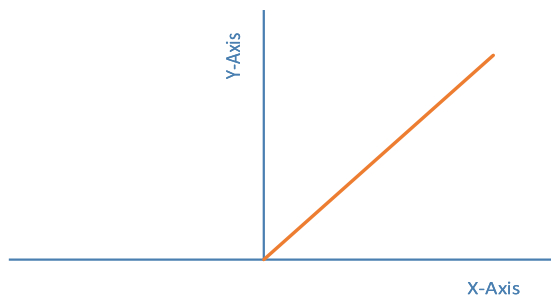


Fig. 3 ReLu graphical representation

expensive for computation compared to tanh and sigmoid as it consists of the less complex.

The dataset is divided into training and testing, the training data result is compared with the predicted result, and the error rate between the predicted and actual result is compared, so that the error is reduced and the model learns. We are using Adam’s optimizer for the backward propagation. As ANN involves correlation-based feature selection approach, it measures the correlation between each feature and the target variable. Features that have a high correlation with the target variable are considered more important and are selected for training the ANN. This approach helps to identify the most relevant features in the proposed work.

The proposed framework has calculated the output error and propagated these errors back through the model using the backpropagation to calculate the gradient value to update the parameter for the model. To achieve, we are using Adam’s optimizer which is similar to the classical stochastic gradient descent, but it uses the iterative based on the dataset that gives faster and better results, generally in gradient descent maintains a learning rate in a single phase for all weights and the learning rate does not change during the training of the train data of the dataset. In the case of Adam’s optimizer, learning rate is maintained for each

model weight which is used for the neuron and individually adapted as learning unfolds.

Figure 4 shows the complete model implementation concerning the artificial neural network having forward propagation and backward propagation.

This helped in the analysis of the model, so the learning of the model used various numbers of epochs, so that we could get better accuracy. Learning started with 500 epochs and with an increment of the epoch by 100 could observe the accuracy increased.

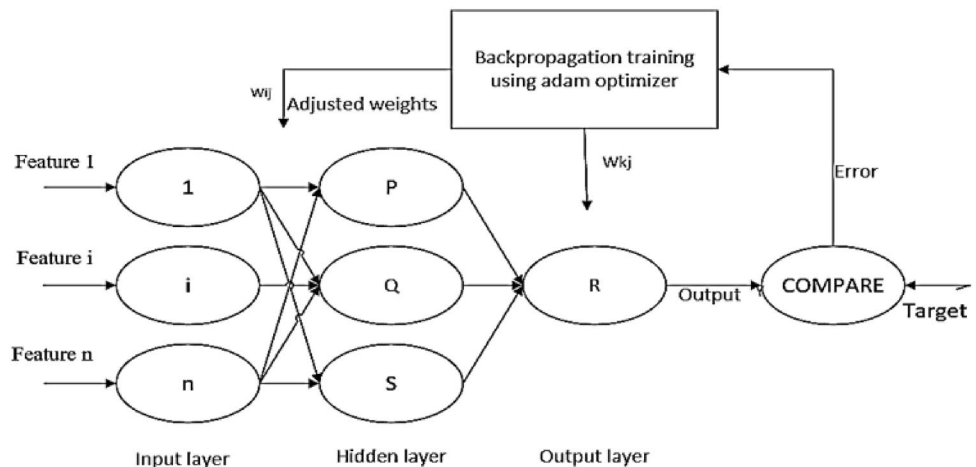
For the testing part, here, the functionality testing of the application is used and the accuracy of the model improved as the testing phase showed a few errors and bugs, once the error was rectified the results improved and helped in better prediction. For the prediction of a new patient, a GUI interface is used that helps in predicting if the patient can have heart disease or not.

### Results

The experimental findings for the ANN algorithm employed for this work are reported in this section. The ANN algorithm’s effectiveness was evaluated using classification datasets and benchmark functions. The benchmark functions and classification datasets are the two sections that make up this section. Experiments were executed having the specifications of the system as follows: Intel i5-3470 3.20 GHz, Intel Windows 10, and 6 GB of memory.

From the proposed model the accuracy increased, compared to other supervised learning algorithms, the ANN learns the model and predicts the results more accurately. The learning takes some time and gives different measurements such as sensitivity, recall, precision, and accuracy. These measurements help in understanding the success of the model.

Fig. 4 Multilayer perceptron model





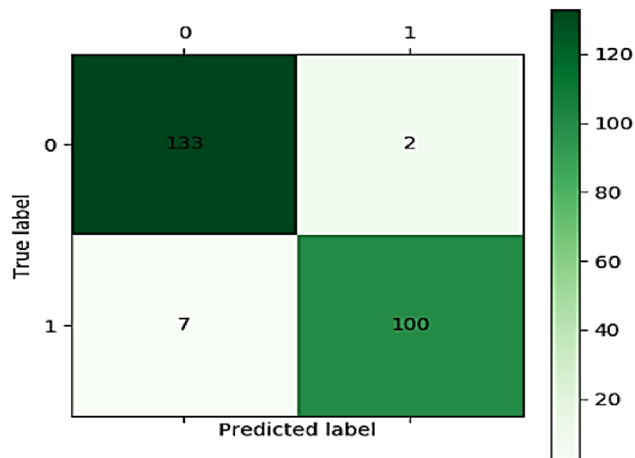


Fig. 5 Confusion matrix

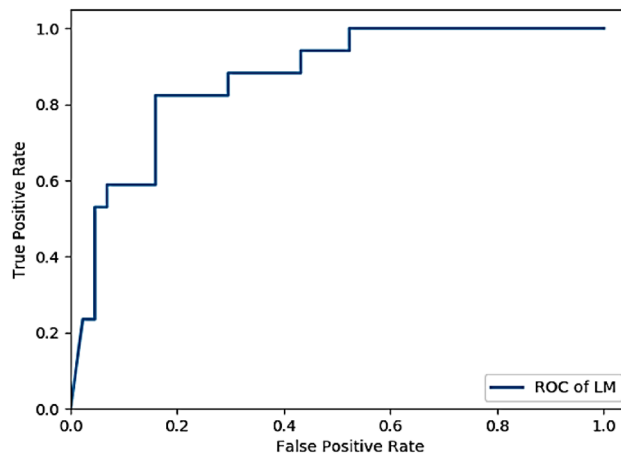


Fig. 6 The area under the curve

Table 1 Results obtained from various measurements

	Precision	Recall	F1-score	Support
Prob=0	0.99	0.95	0.97	133
Prob=1	0.93	0.98	0.96	100
Micro-average	0.96	0.96	0.96	233
Macro-average	0.96	0.97	0.95	233
Weighted average	0.95	0.96	0.95	233

Figure 5 shows the confusion matrix of the dataset. It shows the true positive and false negative

$$\text{Sensitivity} = \left[ \frac{\text{no. of classes correctly predicted as true + ve}}{\text{total no. of actual changes for true + ve case}} \right] * 100.$$

$$\text{Sensitivity} = \left[ \frac{\text{True + ve}}{\text{True + ve + False - ve}} \right]$$

$$\text{Precision} = \left[ \frac{\text{True - ve}}{\text{true + ve + False + ve}} \right]$$

$$\text{Recall} = \left[ \frac{\text{True + ve}}{\text{True + ve + False - ve}} \right]$$

$$\text{F1-score} = 2 * \left[ \frac{\text{precision * recall}}{\text{precision + recall}} \right].$$

Table 1 shows the results obtained by the various measurements. The total average of the model after the functional testing and rectifying the error is equal to 96.69%. Figure 6 shows the area under the curve which helps in measuring pure prediction discrimination that

helps to obtain the accuracy of the test depending on how well the test separates the group being tested into those with and without heart diseases.

Finally, Fig. 7 shows an example of the GUI used to predict for new patients if they are having heart disease or not.

### Conclusion and Future Work

This work contributes to a better understanding of the fundamentals of learning with artificial neural networks by applying them to labelled datasets. After carrying out functionality testing on the application, the accuracy of the application scales to 96.69%, which is a higher percentage than can be obtained by any other means. From a future

perspective, researchers can improve the results using more machine-learning algorithms or undertaking further end-to-end testing, such as system testing, which can uncover any integration issues or flaws. In addition to creating a steadier outcome, increasing the amount of the dataset including information on heart problems collected from hospitals can also help deliver a more accurate prediction.

Heart disease predicti...

Age: 67

Gender: 1

Chest Pain type: 4

Resting blood pressure: 160

cholestrol: 286

Fasting blood glucose: 0

Resting ECG results: 2

Maximum heart rate: 108

Exercise-induced angina: 1

Exercise-induced ST-segmei: 1.5

Peak exercise ST segment s: 2

Main blood vessel number: 3

Defect type: 3

prediction

Whether it is sic 1 Probability of illr 1.0

**Fig. 7** Application showing the result

**Funding** Open access funding provided by Manipal Academy of Higher Education, Manipal.

## Declarations

**Conflict of Interest** All the authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Mohankrishna Rayudu D, Naresh E, Vijaya Kumar BP. The impact of test-driven development on software defects and cost: a comparative case study. *Int J Comput Eng Technol*. 2014;5(2):98–107.
- Sharma A, Pal T, Jaiswal V. Heart disease prediction using convolutional neural network. In: *Cardiovascular and coronary artery imaging*. Cambridge: Academic Press; 2022. p. 245–72.
- Deepika D, Balaji N. Effective heart disease prediction using novel MLP-EBMDA approach. *Biomed Signal Process Control*. 2022;72: 103318.
- Song G, Zhang J, Mao D, Chen G, Pang C. A multimodel fusion method for cardiovascular disease detection using ECG. *Emerg Med Int*. 2022. <https://doi.org/10.1155/2022/3561147>.
- Bokhare A, Bhagat A, Bhalodia R. Multi-layer perceptron for heart failure detection using SMOTE technique. *SN Comput Sci*. 2023;4(2):182.
- Schlingloff BH, Roggenbach M. Specification-based testing. In: *Formal methods for software engineering: texts in theoretical computer science*. Cham: Springer; 2022. [https://doi.org/10.1007/978-3-030-38800-3\\_5](https://doi.org/10.1007/978-3-030-38800-3_5).
- Hassan CAU, Iqbal J, Irfan R, Hussain S, Algarni AD, Bukhari SSH, Ullah SS. Effectively predicting the presence of coronary heart disease using machine learning classifiers. *Sensors*. 2022;22(19):7227.
- Nahar J. Computational intelligence for heart disease diagnosis: a medical knowledge driven approach. *Expert Syst Appl*. 2013;40:96–104.
- Detrano R, Janosi A, Steinbrunn W, Pfisterer M, Schmid J, Sandhu S, et al. International application of a new probability algorithm for the diagnosis of coronary artery disease. *Am J Cardiol*. 1989;6:304–10.
- Cheung N. Machine learning techniques for medical analysis. B.Sc. Thesis, School of Information Technology and Electrical Engineering, University of Queensland, 2001.
- Palaniappan S, Awang R. Web based heart disease decision support system data mining classification modeling techniques. *Proceedings of IIWAS*, 2007.
- Bramer M. Principles of data mining. Cham: Springer; 2007.
- Dessai ISF. Intelligent heart disease prediction system using probabilistic neural network. *Int J Adv Comput Theory Eng (IJACTE)*. 2013;2(3):2319–526.
- Moreno Seco F, Mico L, Oncina JA. Modification of the LAESA algorithm for approximated k-NN classification. *Pattern Recognit Lett*. 2003;24:47–53.
- Shorewala V. Early detection of coronary heart disease using ensemble techniques. *Inf Med Unlocked*. 2021;26: 100655.
- Alpaydin E. Voting over multiple condensed nearest neighbors. *Artif Intell Rev*. 1997;11:115–32. <https://doi.org/10.1023/A:1006563312922>.
- Babbar H. Software testing: techniques and test cases. *Int J Res Comput Appl Robot*. 2017;5(3):44–53.
- Niranjnamurthy M, Navale S, Jagannatha S, Chakraborty S. Functional software testing for web applications in the context of industry. *J Comput Theor Nanosci*. 2018;15(11):3398–404.
- Shankar V, Kumar V, Devagade U, et al. Heart disease prediction using CNN algorithm. *SN Comput Sci*. 2020;1:170. <https://doi.org/10.1007/s42979-020-0097-6>.
- Bertolino A. Software testing research: achievements, challenges, dreams. In: *Future of software engineering (FOSE'07)*. IEEE; 2007. p. 85–103.
- Chauhan RK, Singh I. Latest research and development on software testing techniques and tools. *Int J Curr Eng Technol*. 2014;4(4):2368–72.
- Bareja K, Singhal A. A review of estimation techniques to reduce testing efforts in software development. In: *15th International Conference on Advanced Computing & Communication Technologies*, 2015.
- Kanewala U, Bieman JM. Using machine learning techniques to detect metamorphic relations for programs without test oracles. In: *2013 IEEE 24th international symposium on software reliability engineering (ISSRE)*. IEEE; 2013. p. 1–10.

24. Shah D, Patel S, Bharti SK. Heart disease prediction using machine learning techniques. *SN Comput Sci.* 2020;1:345. <https://doi.org/10.1007/s42979-020-00365-y>.
25. Amin MZ, Ali A. Application of multilayer perceptron (MLP) for data mining in healthcare operations. In: 2017 3rd International conference on biotechnology. Lahore, Pakistan; 2017. p. 2e11.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.