Machine Learning Model to Predict Mortality due to Cardiovascular Diseases

Abstract. A popular phrase for conditions affecting the heart or blood arteries is "Cardiovascular Disease." They are the cause for the highest number of deaths worldwide annually. Most of these diseases can be prevented by altering to healthier lifestyles and addressing behavioural risk factors. Machine learning plays a significant role in the prediction of cardiovascular diseases, and thus be integral in preventing the resulting deaths. The heart disease dataset was considered from Kaggle. The machine learning models used in this work are, Gaussian Naive Bayes, Support Vector Classifier, Logistic Regression, Decision Tree and Random Forest Classifier, K-Nearest Neighbours Classifier, with the highest accuracy achieved being 96% by Stacking Classifier model.

Keywords: Cardiovascular Disease, Machine Learning, Stacking Classifier, Gaussian Naïve Bayes, Logistic Regression, Support Vector Classifier.

1 Introduction

One of the most vital parts of the human body is the heart. Malfunctioning of heart can eventually affect other body parts, often results in fatalities. Greater risk of atherosclerosis and blood clots in veins are associated with cardiovascular disorders. Heart pumps the blood, which supplies oxygen and nutrients to various parts of the human body. Deficiency of blood in human body leads to malfunctioning of the heart, often resulting in immediate death. Angina (more commonly known as chest pain) happens when the heart's blood supply is momentarily cut off. In some cases, arteries in different organs including kidneys, heart, eyes, brain, and so forth might get damaged due to Cardiovascular Diseases (CVD).

Heart diseases remain one of the biggest causes of death across the globe. In India alone, an estimated four people dies every minute. Most of these deaths are observable in the age bracket of 30-50 years, as per the Indian Heart Association [1]. With the advent of modern medicine, great improvements have been achieved, however, more can be done in all areas concerning healthcare [2-4] including diagnostics. Therefore, an efficient and timely diagnosis would be an immense help in preventing loss of lives. Advancement towards Machine Learning (ML) in recent years has benefitted data analysis and prediction [5-7].

In this paper, a Kaggle dataset “Heart Disease Clinical Dataset” containing 299 samples, and 13 attributes is used to predict the mortality due to Heart Failure (HF). It predicts HF using different ML algorithms such as Logistic Regression (LR), K-Nearest Neighbour (KNN), Gaussian Naïve Bayes (GNB), Support Vector Classifier (SVC), Random Forest (RF), Decision Tree (DT) and Stacking Classifier (SC) model. The highest accuracy achieved on the test set was 94% with LR and SVC, and 96%

with SC. The goal of this study is to attain better prediction of life threats through CVD, which can prevent life loss. Following sections make up the structure of this study: Section 2 discusses various ML models implemented in this work, Section 3 presents the existing works related to HF predictions, Sections 4 and 5 address the methodology, and discussion of results, respectively. At last, conclusion and future works are delineated in Section 6.

2 Machine Learning Models

Various ML models used in this work are discussed as follow [5, 7]: SVC is a model which creates best decision boundaries or best lines to divide plane into n-dimension classes to map new data points in an appropriate category. LR is a supervised learning technique which gives categorical or discrete outcomes and provides probabilistic value in the range of 0 to 1. GNB is a Bayes classification model which predicts probability of the target outcomes. It works with Gaussian distribution and in case of continuous values, it assumes that values are sampled from normal distribution. DT is a tree structure classifier that works well on classification and regression problems. RF is an ensemble learning classification algorithm which divides dataset into the number of subsets, applying DT on each of them. The final outcome is the average of all DT sub models. KNN is a classification technique which uses the concepts of Euclidean distance, also this model is non-parametric i.e., it does not make any assumptions on underlying data. SC uses the concept of ensemble learning, in which two or more high performing base models are fed onto a meta-model. This model learns from the predictions of the individual base models. The results predicted by this ensembled model outperforms the individual base models.

3 Related Works

Several works have presented the use of ML models to detect heart diseases as shown in Table 1. For each existing work, the year of publication, technique used, summary along with the results are shown in tabular form.

Table 1. Description of existing works

S. No.	Ref.	Year	Techniques	Description	Results (Accuracy in %)
1	[8]	2019	KNN, SVM, DT, RF	As per agile methodology, various ML algorithms were trained using heart dataset for predictions. Later, deployed to the web using flask, the model, as per DTC Algorithm, shows an accuracy of about 98.83% and KNN having 97.4% approximately.	DTC-98.83
2	[10]	2019	DT, KNN, K-means clustering, Adaboost	Prediction and classification of heart diseases using DT and data mining with assistance of divergent data mining tools like, DT, KNN, NB, bagging algorithm, kernel density, sequential minimal optimization and neural networks, straight Kernel self-organizing map and SVM. The accurateness of the structure can be upgraded by creating various combinations of data mining techniques and by parameter tuning.	NN-100
3	[11]	2019	NB, GLM, LR1, DP, RF, DT, GBT, SVM, VOTE, HRFLM	Heart disease was predicted with increasing accuracy despite the presence of other conflicting health ailments. Performance was enhanced with an accuracy level of 88.7% through the prediction model for heart disease with HRFLM. The Cleveland heart dataset was used to improve the performance of heart disease.	HRFLM-88.7
4	[9]	2020	SVM, CNN, RF	The ML Algorithm goes through databases related to HF and determines successful implementation in clinical practice. SVM and boosting algorithms get a special mention.	AUC-(BA-91, CBA-93, SVM-92, CNN-90)
5	[12]	2020	DT, NB, LR1, RF	UCI ML repository dataset was used to detect heart diseases by using most efficient ML algorithm. Comparing the accuracy score of DT, LR1, RF and NB algorithms, RF algorithm provided highest accuracy of 90.16%.	RF-90.16
6	[14]	2020	SVM, DT, LR, KNN	Using algorithms such as KNN, DT, LR1 and SVM, on a dataset from UCI repository, heart disease was predicted with an increased accuracy where KNN found to be the most suitable model.	KNN-87
7	[15]	2020	GBC, RF Classifier, SVC, ETC, LR1, MLP Classifier	A model predicted heart disease in Cleveland dataset, where 14 attributes were analyzed, classification algorithms such as GBC, RFC, SVM, Extremely Randomized Trees Classifier (ETC), LR1 and MLP Classifier were used for the classification of heart disease. SVM and MLP provided highest accuracy of 91.7%.	SVM and MLP - 91.7
8	[16]	2021	Smote Technique (DT, SVM (linear), LR, KNN, NB, SVM(RBF), MLP, RC, RF, QDA, Adaboost, GB, LDA, ETC, EGB, LGB, CC, DF)	Analyzed and compared the performance of 18 different ML models for heart failure prediction based on 12 clinical features. Techniques such as Z-score or min-max normalization and SMOTE were used for evaluation.	Z-score Normalization-(No SMOTE- SVM (linear), RF, LR-86.67, SMOTE-QDA, LGB, ETC-86.67)
9	[17]	2021	LR1, SVM, KNN, DT, RF	Segregated heart disease patients (around 300 records) based on their risk factors. Five different classifiers i.e., LR1, SVM, KNN, DT and RF were used, and the accuracies were measured and compared. RF outperformed others in terms of accuracy as it increases DT accuracy by reducing overfitting. However, KNN does not perform well due to the small dataset.	SVM- 87.91
10	[19]	2021	-	Average lifetime risk for HF differs significantly across and within sex-race groups. Adults between the ages of 20 to 59 years and free of CVD from 5 sections of population were included in the study	-
11	[20]	2021	RF	RF algorithm was implemented for the prediction of heart disease on a dataset taken from Kaggle. It was the most efficient algorithm. This system can also be used for the prediction of various diseases by applying other ML algorithms such as NB, DT, KNN, LR1, fuzzy logic for better accuracy.	RF-93.3
12	[13]	2022	DT, NB	To predict CVDs, HF dataset containing 13 attributes was used. It contains an in-depth analysis of two classifiers, GNB, DT and records the best accuracies, which are 86.0% and 82.0% respectively.	GNB-86
13	[18]	2022	LR, GNB, DT, RF, SVM, Stack Model	Mortality from HF was predicted using ensemble learning with 90% accuracy on heart disease dataset from Kaggle. Various performance parameters (accuracy, recall, precision) were used to provide a comparative study of considered ML models.	Stack model-90

AUC: Area Under Curve, BA: Boosting Algorithms, CBA: Custom-Built Algorithms, CNN: Convolutional Neural Network, DP: Deep Learning, DT: Decision Tree, DTC: Decision Tree Classifier, EGB: Extreme Gradient Boosting, ETC: Extra Trees Classifier, GBC: Gradient Boosting Classifier,GBT: Gradient-Boosted Trees, GLM: Generalized Linear Model, GNB: Gaussian Naïve Bayes, HRFLM: Hybrid Random Forest with Linear Model, KNN: K-Nearest Neighbour, LDA: Linear Discriminant Analysis, LGB: Light Gradient Boosting, LR: Linear Regression, LR1: Logistic Regression, NB: Naive Bayes, NN: Neural Network, QDA: Quadratic Discriminant Analysis, SMOTE: Synthetic Minority Oversampling Technique, SVM: Support Vector Machine, RBF: Radial Basis Function, RC: Rating Curve, DF: Decision Forest, RFC: Random Forest Classifier, MLP: Muti-Layer Perception, HF: Heart Failure

4 Methodology

The step-by-step process used in this work (see Fig. 1) is discussed in this section. It is described as follows:

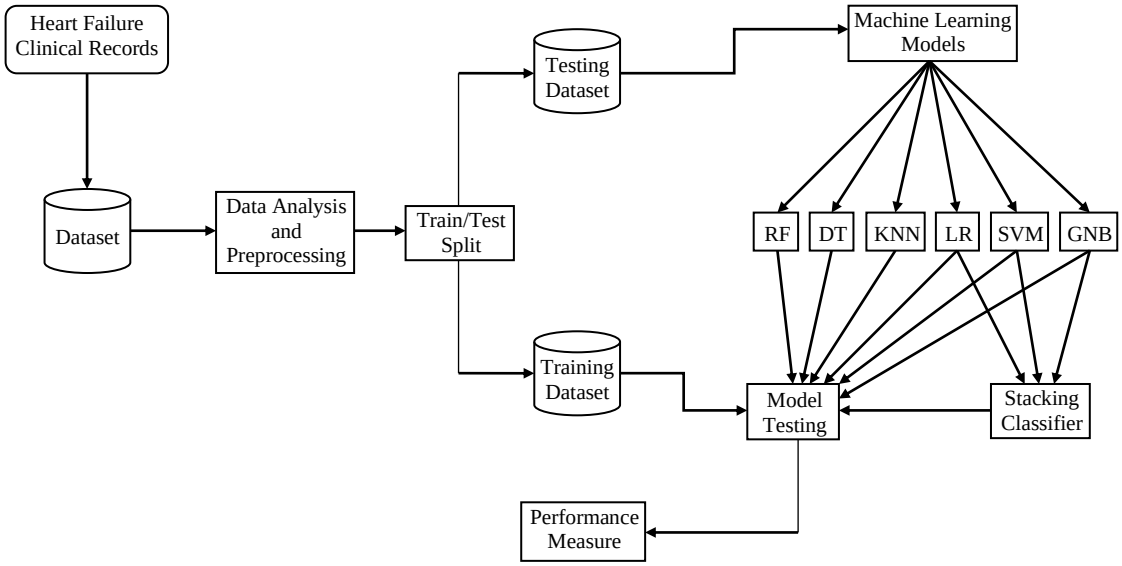

Fig. 1. Flowchart outlining the proposed work

4.1 Data acquisition

The Kaggle website provided the access to 13-attribute dataset (https://bit.ly/3Sl15B8) that was used in this study. An overview of the dataset is presented in Table 2.

Table 2. Dataset analysis

Dataset Statistics	Number of variables: 13
	Number of observations: 299
	Missing cells: 0
	Duplicate rows: 0
Variable types	Numeric: 7
	Categorical: 6

4.2 Pre-processing data and analysis

The quality of data [21-24] is critical for obtaining better results using ML models. The considered dataset is pre-processed to verify the same. Out of 13 features, 12 most significant were selected for training the models (see Fig. 2). Table 2 provides the analysis of dataset where there were no duplicate rows and missing values. Fig. 2 signifies the correlation between the features. It also demonstrates that there is a very low correlation among the features.

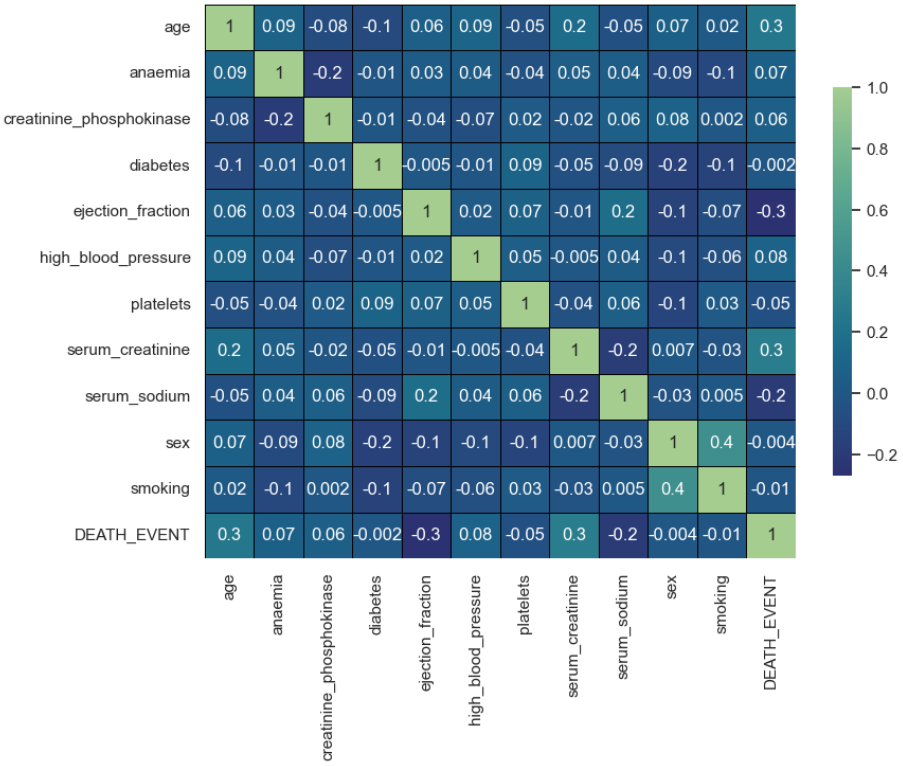

Fig. 2. Correlation matrix

Fig 3. demonstrates the distribution of the data with respect to individual pairs of the features in the dataset. It presents the most separated clusters mapping the relationship between pairs of features.

4.3 Train-test data split

Feature scaling was performed on the dataset using StandardScaler. Thereafter, the overall dataset was split into training and testing subsets with a ratio of 7:3. Random state was selected to be an optimum in a range of all possible split subsets.

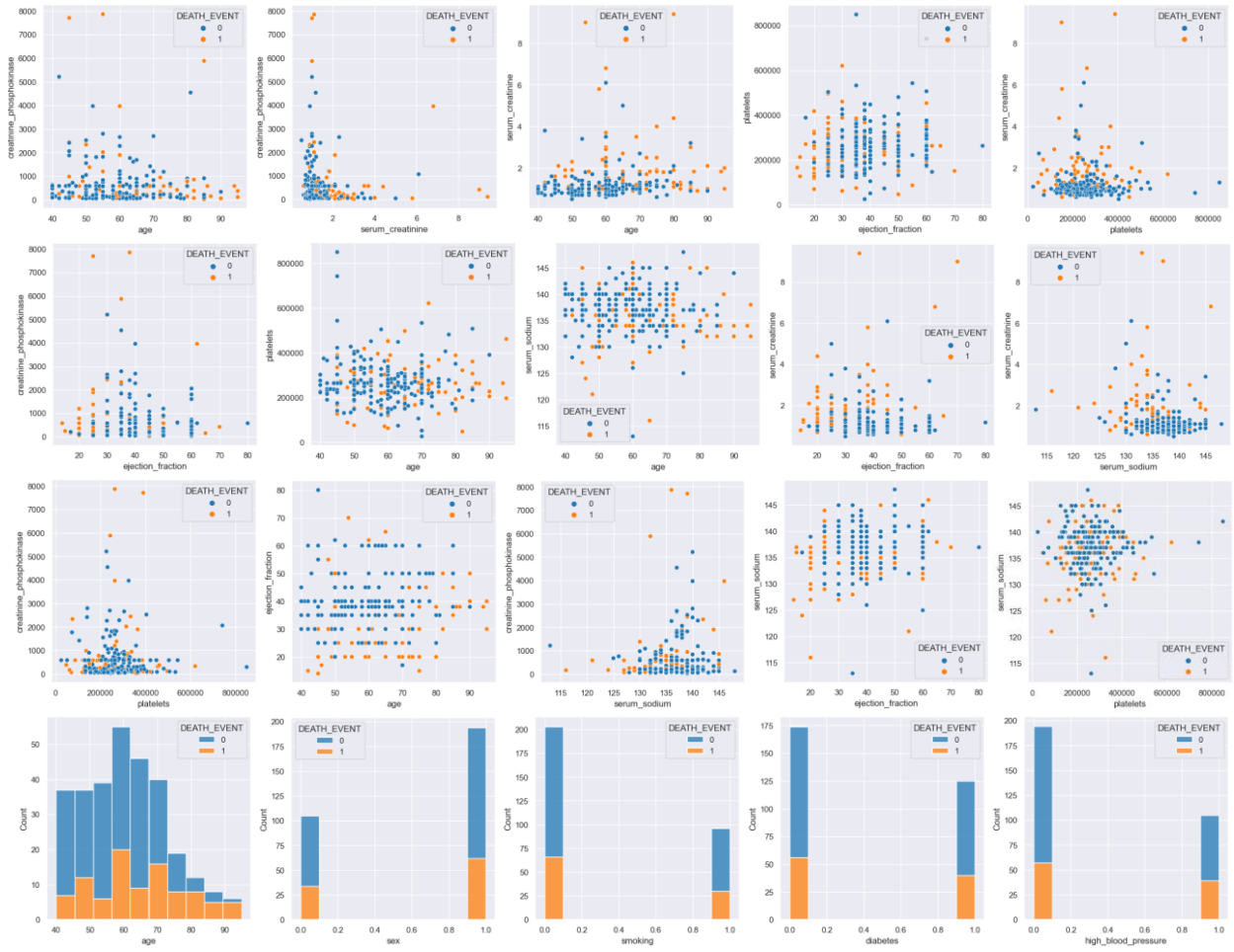

Fig. 3. Data distribution

4.4 Models Implementation

The training data was then fed to several ML models, including SVM, GNB, KNN, DT, RF, LR and SC (see Table 3).

4.5 Results

The evaluation metrics used for this work include confusion matrix, accuracy, precision, recall and F1 score. The performance measures were obtained and compared with the existing works as shown in Table 4.

5 Results and Discussion

This section describes the experimental environment, the results, and comparative analysis with existing works. The model was developed to provide a solution in detection of CVD, based on different medical test results tabulated in the selected dataset. The model has shown remarkable performance against the existing problem statement as shown in Table 3.

5.1 Hardware and Software Specifications

The overall experiment was conducted on a machine with Windows 10 operating system with Ryzen 7-5800H octa-core processor, 16 GB physical memory and 4GB Nvidia RTX 3050Ti graphics. The training and testing of the models were conducted on a local Jupyter Server with Python 3.10.64 environment. Any operating system, including Windows, MacOS and the Linux kernel distributions, with Python libraries (pandas, NumPy, sklearn, matplotlib) installed in it, can be used for the similar experiments.

5.2 Experimental Observations

This sub-section describes the results attained after performing experiments. ML models mentioned in sub-section 4.3 were used for the experimental work. The results of the models are shown in Table 3. The presented model used the concepts of ensemble learning, with the implementation of SC. In this model, 3 level-0 models (base models) i.e., LR (94%), GNB (79%) and SVC (94%), were fed onto the level-1 model (meta model) i.e., SVC. The final SC model produced the highest performance of 96% accuracy (see Table 3). The comparison of Recall and Precision in test prediction showed acceptable outcomes, which justifies the model.

Table 3. Results of performance parameters

	LR	**GNB**	**DT**	**RF**	**SVC**	**KNN**	**SC**
Accuracy	94	79	84	90	94	77	96
Precision	91	72	79	85	91	68	92
Recall	96	74	80	90	96	68	97
F1 Score	93	73	80	87	93	68	94

A key feature of this experiment was the selection of optimum training data using the most suitable random state parameter for the target dataset. This parameter was obtained after performing a large number of iterations to train the model and by comparing the results in each step. The confusion matrix in Fig 4. depicts the number of binary outcomes on the target which is either correct or incorrect predictions on positive or negative facts. These results are used to formulate the performance parameters of the applied ML models as discussed in Table 3.

LR		Predicted	
		Negative	Positive
Actual	Negative	63	5
	Positive	0	22

GNB		Predicted	
		Negative	Positive
Actual	Negative	57	11
	Positive	8	14

SVC		Predicted	
		Negative	Positive
Actual	Negative	63	5
	Positive	0	22

DT		Predicted	
		Negative	Positive
Actual	Negative	60	8
	Positive	6	16

RF		Predicted	
		Negative	Positive
Actual	Negative	61	7
	Positive	2	20

KNN		Predicted	
		Negative	Positive
Actual	Negative	58	10
	Positive	11	11

Stacking CLF		Predicted	
		Negative	Positive
Actual	Negative	64	4
	Positive	0	22

Fig 4. Confusion matrix of all models

5.3 Comparison with existing works

The presented work has been compared with prior studies [13,16,18] as shown in Table 4. Notably, in [16] the data split was 90:10 for the training and testing while [13] and [18] had used the standard 70:30 ratio. The presented work also used the general 7:3 ratio which is widely used by researchers. The maximum accuracies achieved by prior works were 86% using GNB and DT models [13], 86.7% using LR, Linear SVM and RF models [16], and 90% using SC model of GNB, DT and RF [18]. The proposed model (SC) outperformed all the previous works [13][16][18], in terms of accuracy, precision, F1-score and recall as observed in column 8 of Table 4.

Table 4. Results of existing vs current models

	Accuracy				**Precision**				**Recall**				**F1 Score**			
Models	[16]	[13]	[18]	Presented Models	[16]	[13]	[18]	Presented Models	[16]	[13]	[18]	Presented Models	[16]	[13]	[18]	Presented Models
LR	0.76	-	0.86	0.94	-	-	0.82	0.91	-	-	0.8	0.96	0.6	-	0.81	0.93
SVC	0.87	-	0.88	0.94	-	-	0.85	0.91	-	-	0.8	0.96	0.8	-	0.84	0.93
KNN	0.8	-	-	0.77	-	-	-	0.68	-	-	-	0.68	0.7	-	-	0.68
RF	0.83	-	0.89	0.9	-	-	0.86	0.85	-	-	0.9	0.9	0.7	-	0.86	0.87
GNB	-	0.86	0.86	0.79	-	0.73	0.83	0.72	-	0.73	0.8	0.74	-	-	0.8	0.73
DT	0.83	0.82	0.8	0.84	-	0.64	0.75	0.79	-	0.69	0.8	0.8	0.7	-	0.76	0.8
SC	-	-	0.9	**0.96**	-	-	0.88	**0.92**	-	-	0.9	**0.97**	-	-	0.87	**0.94**

6 Conclusion

In this work, six ML algorithms (LR, GNB, DT, RF, SVC, KNN) were implemented on a heart failure clinical dataset from Kaggle. Further, three of them, which include SVC, LR, GNB, were ensembled into an SC model. After analyzing the 12 selected features of the used dataset, the greatest accuracy on the test set was 94% for SVC and LR separately, and 96% for the SC model for the prediction of heart disease. The Recall and Precision results were comparable and justifiable, with False Negative errors being less than False Positive Errors. This study is currently restricted to a dataset of 299 observations; however, large datasets can be used in the future works. In this context, it is remarkable, that this model and the algorithm developed can be deployed and applied against any larger medical datasets for detection of different health diseases. This model has produced optimum performance with only 2 level of Ensemble Learning technique against the heart failure clinical dataset. Further in context of future research or application deployment of this model, the number of meta-learning levels can be increased for better performance upon different healthcare datasets.

References

1. Kannan, R., Vasanthi, V.: Machine learning algorithms with roc curve for predicting and diagnosing the heart disease. Soft Computing and Medical Bioinformatics, pp. 63-72. Springer (2019).
2. Rana, M., Bhushan, M.: Advancements in healthcare services using deep learning techniques. In: 2022 International Mobile and Embedded Technology Conference (MECON), pp. 157-161, IEEE (2022). doi: 10.1109/MECON53876.2022.9752020.
3. Singh, V. J., Bhushan, M., Kumar, V., Bansal, K. L.: Optimization of Segment Size Assuring Application Perceived QoS in Healthcare. In: Proceedings of the World Congress on Engineering, vol. 1 (2015).
4. Pathan, S., Bhushan M., Bai, A.: A study on health care using data mining techniques. Journal of Critical Reviews, 7 (19), 7877-7890 (2020). doi:10.31838/jcr.07.19.896
5. Singh, S. N., Bhushan, M.: Smart ECG monitoring and analysis system using machine learning. In: 2022 IEEE VLSI Device Circuit and System (VLSI DCS), pp. 304-309, IEEE (2022). doi: 10.1109/VLSIDCS53788.2022.9811433.
6. Pal, S., Mishra, N., Bhushan, M., Kholiya, P. S., Rana, M., Negi, A.: Deep Learning Techniques for Prediction and Diagnosis of Diabetes Mellitus, 2022 International Mobile and Embedded Technology Conference (MECON), pp. 588-593 (2022). doi: 10.1109/MECON53876.2022.9752176.
7. Verma, U., Garg, C., Bhushan, M., Samant, P., Kumar, A., Negi, A.: Prediction of students' academic performance using Machine Learning Techniques. In: 2022 International Mobile and Embedded Technology Conference (MECON), pp. 151-156, IEEE (2022). doi: 10.1109/MECON53876.2022.9751956.
8. Taylor, O., Ezekiel, P., Okuchaba, F.: A Model to Detect Heart Disease using Machine Learning Algorithm. In: International Journal of Computer Sciences and Engineering, vol. 7, no. 11, pp. 1-5 (2019). doi: 10.26438/ijcse/v7i11.15.

9. Krittanawong, C., Virk, H. U. H., Bangalore, S., Wang, Z., Johnson, K. W., Pinotti, R. et al.: Machine learning prediction in cardiovascular diseases: a meta-analysis. Scientific Reports, vol. 10, no. 1, pp. 1-11 (2020).
10. Golande, A., Kumar T., P.: Heart Disease Prediction Using Effective Machine Learning Techniques. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8, Issue-1S4 (2019).
11. Mohan, S., Thirumalai, C., Srivastava, G.: Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques. IEEE Access, vol. 7, pp. 81542-81554 (2019). doi: 10.1109/ACCESS.2019.2923707.
12. Rajdhan, A., Sai, M., Agarwal, A., Rav, D., Ghul, P.: Heart disease prediction using machine learning, International Journal of Engineering Research & Technology (IJERT), 9(04), (2020).
13. Reddy, V. S. K., Meghana, P., Reddy, N. S., Rao, B. A.: Prediction on Cardiovascular disease using Decision tree and Naïve Bayes classifiers. In: Journal of Physics: Conference Series, vol. 2161, no. 1, p. 012015 (2022). doi: 10.1088/1742-6596/2161/1/012015.
14. Singh, A., Kumar, R.: Heart Disease Prediction Using Machine Learning Algorithms. In: 2020 International Conference on Electrical and Electronics Engineering (ICE3), pp. 452-457 (2020). doi: 10.1109/ICE348803.2020.9122958.
15. Arunachalam, S.: Cardiovascular Disease Prediction Model using Machine Learning Algorithms. International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; Volume 8 Issue VI (2020). doi:http://doi.org/10.22214/ijraset.2020.6164.
16. Wang, J.: Heart Failure Prediction with Machine Learning: A Comparative Study. In: Journal of Physics: Conference Series, vol. 2031, no. 1, p. 012068 (2021). doi: 10.1088/1742- 6596/2031/1/012068.
17. Anusha, M., Suresh, K., Chandana, M.: In: Earlier Prediction on the heart disease based on supervised machine learning techniques. 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS) (2021). doi:10.1109/ICICCS51141.2021.9432212
18. Kedia, S., Bhushan, M.: Prediction of Mortality from Heart Failure using Machine Learning. In: 2nd International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET), pp. 1-6 (2022). doi: 10.1109/ICEFEET51821.2022.9848348.
19. Khan S. et al.: Development and Validation of a Long-Term Incident Heart Failure Risk Model. Circulation Research, vol. 130, no. 2, pp. 200-209 (2022). doi: 10.1161/circresaha.121.319595.
20. Pal, M., Parija, S.: Prediction of Heart Diseases using Random Forest. In: Journal of Physics: Conference Series, vol. 1817, no. 1, pp. 012009 (2021). https://doi.org/10.1088/17426596/1817/1/012009.
21. Bhushan, M., Negi, A., Kaur, K.: Method to resolve software product line errors. In: International Conference on Information Communication and Computing Technology, pp. 258-268 (2017). doi: https://doi.org/10.1007/978-981-10-6544-6_24.
22. Bhushan, M., Goel, S., Kumar, A., Negi, A.: Managing software product line using an ontological rule-based framework. In: 2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS), pp. 376-382 (2017). doi: 10.1109/ICTUS.2017.8286036.
23. Bhushan, M., and Goel, S.: Improving software product line using an ontological approach. In: Sadhana, vol. 41, no. 12, pp. 1381-1391 (2016). doi: 10.1007/s12046- 016-0571-y.
24. Bhushan, M., Goel, S., and Kaur, K.: Analyzing inconsistencies in software product lines using an ontological rule-based approach. In: Journal of Systems and Software, vol. 137, pp. 605-617 (2018). doi: http://dx.doi.org/10.1016/j.jss.2017.06.002.