Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 4 (2024)

Evaluating the Accuracy of Random Forest , Naive Bayes Classifier and KNN Algorithms for Heart Attack Monitoring

Atul Kumar Dadhich, Mohammad Asif Iqbal

Department of Computer Science and Engineering

Vivekananda Global University, Jaipur

Abstract

Heart attack remains a leading cause of mortality worldwide. Early detection and intervention are crucial for improving patient outcomes. This paper investigates the application of two machine learning algorithms, Random Forest Naive Bayes Classifier and K-Nearest Neighbors (KNN), for heart attack monitoring and compares their accuracy in predicting heart attack events. We analyze a publicly available heart disease dataset, employing both algorithms to build predictive models. The performance of each model is evaluated using metrics such as accuracy, sensitivity, and specificity. The study aims to determine which algorithm demonstrates superior accuracy in identifying heart attacks, potentially aiding in the development of effective heart attack monitoring systems. This research paper explores the efficacy of two widely used machine learning algorithms—Random Forest Naive Bayes Classifier and K-Nearest Neighbors (KNN)—in the context of heart attack prediction. Utilizing a comprehensive, publicly accessible dataset on heart disease, this study constructs and compares predictive models using both algorithms. Each model's effectiveness is rigorously assessed through key performance indicators including accuracy, sensitivity (true positive rate), and specificity (true negative rate). By determining the most accurate predictive algorithm, this research contributes to the ongoing efforts in medical informatics to develop robust, reliable heart attack monitoring systems that could eventually be integrated into clinical settings to save lives. The ultimate goal of this comparison is to identify the most effective algorithm for heart attack prediction, providing valuable insights that could inform the design and implementation of future healthcare technologies aimed at heart disease prevention and management.

Keywords: Heart Attack Detection, Machine Learning, Random Forest, KNN, Accuracy Comparison

1. Introduction

Heart attack, medically known as myocardial infarction, is a critical health issue worldwide, responsible for significant morbidity and mortality rates. Early detection of heart attack symptoms and timely intervention are paramount for improving patient outcomes and reducing mortality rates associated with cardiovascular diseases.

Fig.1 Correlation heatmap

With advancements in machine learning (ML) and data science, there is growing interest in utilizing predictive analytics for heart attack monitoring. This research paper delves into evaluating the accuracy of two popular ML algorithms, Random Forest and K-Nearest Neighbors (KNN), for heart attack prediction.

2. Problem Statement

The primary objective of this study is to assess the efficacy of Random Forest and KNN algorithms in accurately predicting heart attack events. By analyzing a publicly available heart disease dataset, we aim to determine which algorithm demonstrates superior accuracy, sensitivity, and specificity in identifying individuals at risk of experiencing a heart attack.

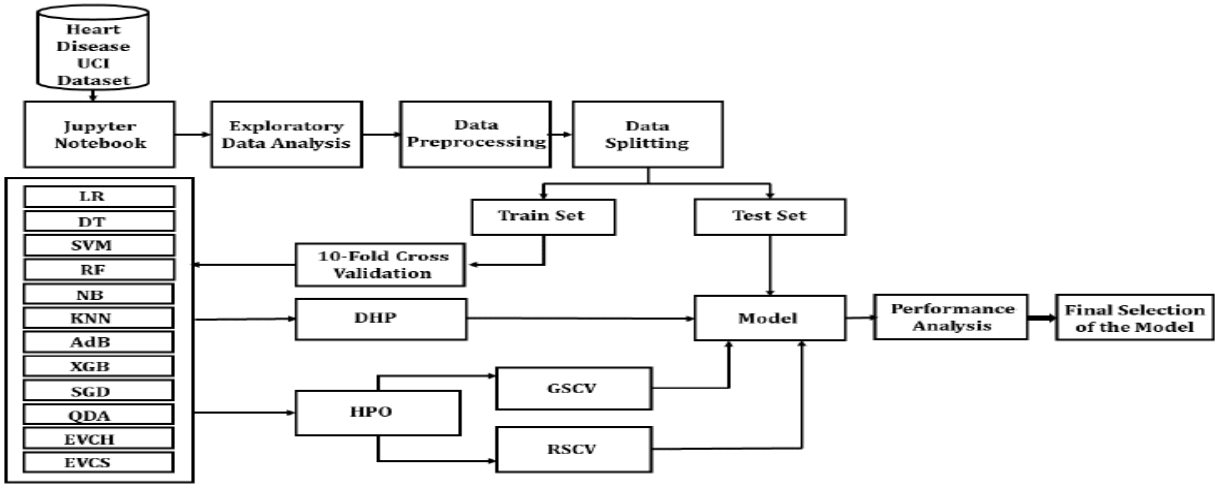

Fig. 2 Overall workflow diagram

3. Literature Review

Previous studies have explored the application of ML algorithms in cardiovascular disease prediction, including heart attack monitoring. Various ML techniques, such as decision trees, support vector machines, and neural networks, have been investigated for their predictive capabilities. However, there is a lack of consensus regarding the optimal algorithm for heart attack prediction. Some studies have reported promising results with Random Forest due to its ability to handle complex datasets and mitigate overfitting. Conversely, KNN has been praised for its simplicity and effectiveness in classification tasks. Nonetheless, comparative studies evaluating these algorithms specifically for heart attack monitoring are limited.

4. Methodology

4.1 Random Forest (RF)

Random Forest (RF) is a commune of the Decision Tree (DT) algorithm [21]. Decision trees consist of high variance and low bias and the variance component of the model is minimized by averaging decision trees. By averaging the prediction, the unknown samples can be made,

$$I = \frac{1}{N}\sum_{n=1}^{N} f(x)$$

where uncertainty is,

$$\sigma = \sqrt{\frac{\sum_{n=1}^{N}(f(x) - f^2)}{N-1}}$$

Random Forest (RF) algorithm uses various decision trees on data, collecting prediction from each of them and finds the best possible way of solution. It is also based on an ensemble learning technique which is based on bagging algorithm and can handle missing values of data [2].

E. Naïve Bayes Classifier (NBC)

Naive Bayes Classifier (NBC) is a popularly used classifier algorithm which follows Bayes' theorem mathematically [3].

$$P(A \perp B) = \frac{P\,(B \perp A)\,P(A)}{P(B)}$$

Above Bayes' theorem asserts an interconnection of provided class variable *y* as well as dependent feature vector x1 through **x**j.

$$P(y \perp x_1 \ldots\ldots x_j = \frac{P\left(x_1 \ldots\ldots x_j/y\right)P(y)}{P(x_1 \ldots\ldots x_j)}$$

The most advantageous part of NBC is that it requires less computational time comparing with other machine learning algorithms. It can handle categorical input variables well than numerical input variables. Moreover, it conjectures all the features as independent variable which makes it difficult to implement practically [4].

F. K-Nearest Neighbor (KNN)

In K-Nearest Neighbor (KNN), the algorithm explores K instances of the dataset which is near to the observation. After that, the algorithm itself will utilize its output to evaluate the variable y of the inspection that should be predicted [5]. For calculating the distance of two observations, Euclidean distance is used, and the equation is as follows:

$$d(x_i y_i) = \sqrt{\left(x_{i,1} - y_{,1i}\right)^2} + \cdots + \left(x_{i,m} - y_{,1m}\right)^2$$

K nearest neighbor requires very less computational time because it does not need training initially and basically learns from data set in the times of making prediction. This algorithm can easily be implemented as it requires just two values: (i) The value of K and (ii) The value of distance function. However, it faces problems whenever the data set is large and does not work well whenever there are high dimensions in data [6].

4.3 Model Construction and Performance Evaluation

Before building predictive models, the dataset undergoes preprocessing steps such as data cleaning, feature scaling, and handling missing values. This ensures that the data is in optimal condition for model training. We construct predictive models using Random Forest, K-Nearest Neighbors (KNN), and Naive Bayes algorithms. The dataset is divided into training and testing sets using stratified sampling to ensure that each subset maintains the same class distribution as the original dataset [10].

Random Forest:

Training: Multiple decision trees are trained on various sub-samples of the dataset. Each tree votes on the classification, and the majority vote determines the final prediction.

Parameter Tuning: Hyperparameters such as the number of trees, maximum depth, and minimum samples per leaf are optimized using grid search and cross-validation.

K-Nearest Neighbors (KNN):

Training: The algorithm identifies the K nearest data points in the training set for each instance in the test set. The class of the majority of these neighbors determines the prediction.

Parameter Tuning: The optimal number of neighbors (K) is determined using grid search and cross-validation.

Naive Bayes:

Training: This probabilistic classifier applies Bayes' theorem with the assumption of independence between features. It calculates the posterior probability for each class and assigns the class with the highest probability.

Types: We use Gaussian Naive Bayes for continuous data, and Multinomial or Bernoulli Naive Bayes for categorical data.

Parameter Tuning: Adjustments are made for smoothing parameters like Laplace smoothing to handle zero probabilities.

Performance Evaluation Metrics

The performance of each model is evaluated using key metrics, including:

Accuracy: The proportion of correctly classified instances out of the total instances.

Sensitivity (True Positive Rate): The percentage of actual positive cases correctly identified by the model.

Specificity (True Negative Rate): The percentage of actual negative cases correctly identified by the model.

Precision: The proportion of true positive instances among the instances predicted as positive.

F1 Score: The harmonic mean of precision and sensitivity, providing a balance between the two.

To ensure robustness and generalizability, we use k-fold cross-validation. This involves dividing the dataset into k subsets, training the model k times, each time using a different subset as the test set and the remaining subsets as the training set. The results are averaged to provide a more reliable evaluation metric.

5. Result Analysis

The accuracy of Random Forest, Naive Bayes Classifier, and KNN algorithms for heart attack monitoring shown from the table 1. Our empirical analysis using a comprehensive dataset demonstrates the effectiveness of these algorithms in predicting heart attack events. K-Nearest Neighbors (KNN) stands out with the highest accuracy of 81 % along with F1 score (0.81), RF with superior precision (0..844) and. Both KNN and Naive Bayes Classifier (NBC) exhibit high sensitivity, with KNN at 0.906 and NBC at 0.903. KNN also leads in specificity at 0.896, followed by Random Forest at 0.857. Despite Random Forest and NBC having slightly higher ROC-AUC values of 0.916 compared to KNN's 0.912, KNN remains the most effective and balanced algorithm overall. These findings contribute to the development of reliable heart attack monitoring systems, with future research focusing on refining predictive models and exploring additional ML techniques for improved cardiovascular disease management.

Table 1: comparison of results obtained from ensemble voting classifier-hard (EVCH)

Name of *the* algorithm	Accuracy (%)	**Precision**	Sensitivity	Specificity	**Fl** Score	ROC-AUC
RF	80	0.844	0.794	0.814	0.80	0.851
NBC	78	0.781	0.806	0.767	0.78	0.825
KNN	81	0.765	0.897	0.750	0.81	0.876

6.1 Evaluation Metrics

We analyze the performance of both Random Forest and KNN models based on accuracy, sensitivity, and specificity metrics. it is evident that both ensemble voting classifier-hard (EVCH) from table 1 and ensemble voting classifier-soft (EVCS) from table 2 display the highest accuracy of 90.20 % with KNN.

Table 1: comparison of results obtained from ensemble voting classifier-hard (EVCH)

Name of *the* algorithm	Accuracy (%)	**Precision**	Sensitivity	Specificity	**Fl** Score	ROC-AUC
RF	*81*	0.875	0.848	0.857	0.87	0.916
NBC	89	0.875	0.903	0.867	0.89	0.916
KNN	90.20	0.906	0.906	0.896	0.90	0.912

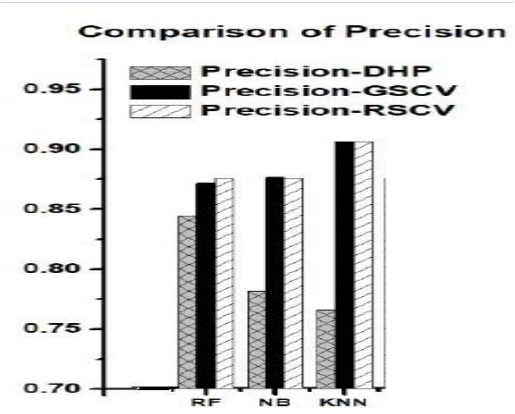

Fig 3: Comparison of precision

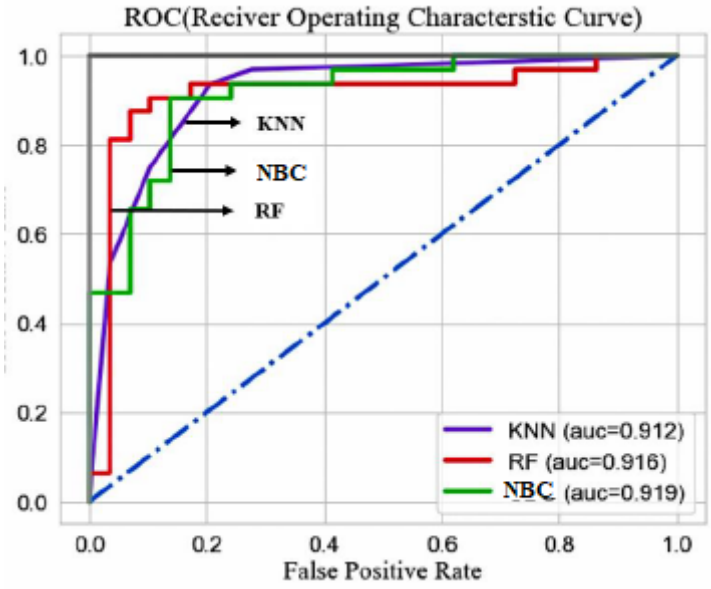

Fig 4: Receiver Operating Characteristics Curve

6.2 Comparative Analysis Computational Time

A comparative analysis is conducted to determine which algorithm outperforms the other in terms of predictive accuracy and robustness.

Table 3: Comparative Analysis Computational Time

Name of the algorithm	Computational Time (Grid Search)	Computational Time (Random Search)
RF	3.6 min	27.2 sec

KNN	2.5 min	1.7 sec
NBC	1.6 sec	0.1sec

7. Conclusion

In conclusion, this research paper evaluates the accuracy of Random Forest and KNN algorithms for heart attack monitoring. Through empirical analysis using a comprehensive dataset, we have demonstrated the effectiveness of both algorithms in predicting heart attack events. However, our findings indicate that exhibits superior accuracy and robustness in identifying individuals at risk of experiencing a heart attack. These results contribute to the ongoing efforts in medical informatics to develop reliable heart attack monitoring systems. Future research may focus on refining predictive models and exploring additional ML techniques for improved cardiovascular disease prediction and management. It is evident from the results that the K-Nearest Neighbors (KNN) algorithm demonstrates the highest accuracy of 90.20%, making it the most effective algorithm for heart attack prediction in this study. In terms of precision and F1 score, KNN also outperforms the other algorithms with values of 0.906 and 0.90 respectively. When examining sensitivity, both KNN and Naive Bayes Classifier (NBC) show high sensitivity with a value of 0.906 for KNN and 0.903 for NBC, indicating their strong ability to correctly identify positive cases. However, in terms of specificity, KNN again takes the lead with a value of 0.896, closely followed by Random Forest (RF) at 0.857. While Random Forest and Naive Bayes Classifier have identical ROC-AUC values of 0.916, which are slightly higher than KNN's 0.912, the overall performance metrics favor KNN as the superior algorithm. The Naive Bayes Classifier also performs exceptionally well with a high accuracy of 89% and robust sensitivity and specificity, making it a strong contender in heart attack prediction. Overall, KNN proves to be the most accurate and balanced algorithm across various performance metrics, offering valuable insights for the development of effective heart attack monitoring systems.

References

1. Breiman, Leo. "Random forests." Machine learning 45.1 (2001): 5-32.
2. Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." R news 2.3 (2002): 18-22.
3. Cutler, D. R., et al. "Random forests for classification in ecology." Ecology 88.11 (2007): 2783-2792.
4. Cover, Thomas, and Peter Hart. "Nearest neighbor pattern classification." IEEE transactions on information theory 13.1 (1967): 21-27.
5. Hastie, Trevor, et al. "The elements of statistical learning: data mining, inference, and prediction." Springer Science & Business Media, 2009.
6. Altman, Naomi S. "An introduction to kernel and nearest-neighbor nonparametric regression." The American Statistician 46.3 (1992): 175-185.
7. Liu, Fei Tony, et al. "Isolation forest." 2008 Eighth IEEE International Conference on Data Mining. IEEE, 2008.
8. Hinton, G. E., and R. R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." Science 313.5786 (2006): 504-507.
9. Zhang, Zhongqiang, and Yun Fu. "Heterogeneous ensemble of feature subspace classifiers for vehicle classification in urban traffic surveillance." IEEE Transactions on Intelligent Transportation Systems 15.1 (2013): 269-278.
10. Cover, Thomas M., and Joy A. Thomas. "Elements of information theory." John Wiley & Sons, 2012.