Prediction of Heart Disease

*Raphael Joseph*

*George Mason University*

*Hap-780-002*

*rjoseph5@masonlive.gmu.edu*

**Abstract**

*Identifying the risk of cardiovascular disease is highly necessary for health risk prevention. Data mining applications in cardiovascular disease prevention can greatly benefit all parties involved and perhaps most importantly, save lives. The initial objective of the analysis was to discover data mining models to predict cardiovascular disease by using key health attributes that influence its development. Another objective was to extract useful information (pattern) from arrhythmia raw data by using regression learning techniques. To develop this project, it will be used the Waikato Environment for Knowledge Analysis (WEKA) machine learning Workbench since this one allows users to quickly try out and compare different machine learning methods on new data sets.*

KEYWORDS • Naive Bayes; data mining; classification algorithms;

**1.1 Introduction**

Cardiovascular disease (CVD) is an illness that involves narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke [5]. Cardiovascular disease is a commercially targeted group since it is preventable around the globe [5]. CVD continues to increase in population size, due to work/life burdens, impoverishment, and limited resources. Heart disease kills 17 million people a year, which is accounting for one-third of all deaths worldwide [5]. By 2020, heart disease will become the leading cause of both death and disability globally, a number that is expected to grow to more than 23.6 million by 2030, according to the American Heart Association [5].

Therefore, there is a need for new potential data mining strategies to predict the risk factors of cardiovascular disease. This research study is regarding cardiovascular risk factors, such as resting heart rate, serum cholesterol, heart reactivity to stress, and also on the varying degrees of chest pain, to predict an individual’s probability of developing heart disease.

**1.2 Analysis of the data**

The first step of any classification problem is to quickly observe the data that we are using to train and test a classification model. Here, we have 270 patients who possess or do not possess heart disease. In this case, the classes in which the model will classify these patients as are 0 (no heart disease) or 1 (with heart disease). The data is then sorted into binary values via smoothing. Smoothing is a form of generalization, normalization, and data cleaning also serve as forms of data reduction. Normalization is defined as the process of organizing data in a database. The attribute is normalized by scaling its values so that they fall within a small specified range, such as 0 to 1.0. Thus the normalization creates a binary table and relationships are established between each attribute. The binary table is created to transform the database into a more flexible unit leading towards the elimination of redundancy, inconsistent dependency and minimizing memory costs. We have a roughly even sample of patients, therefore no sampling is to be done.

Furthermore, we have 23 features that indicate whether a patient possesses heart disease; some of these include age, sex, resting blood pressure, serum cholesterol, fasting blood sugar level, maximum heart rate achieved, etc.

**2.1 Choosing the best classification algorithm**

The no free lunch theorem suggests that there is no universally best learning algorithm. In other words, the choice of an appropriate algorithm should be based on its performance for that particular problem and the properties of data that characterize the problem.

Evaluating the performance of classifier concerns measuring its effectiveness, rather than its efficiency. As a result, the classifier's ability to correctly predict the category of an unseen instance is evaluated, and not its computational complexity.

Given a test dataset, classification is evaluated relative to the training dataset, producing four outputs: instances that are predicted as being positive, when they are indeed positive (TP), instances that are predicted as being negative, when they are indeed negative (TN), instances that are predicted as positive, when in fact, they are negative (FP), and instances that are predicted as negative, when in fact, they are positive (FN).

There are several evaluation techniques that are used to evaluate how well a classifier performs. The most common measures used are precision, recall, F-measure, and accuracy. Often, the goal is to maximize all measures, which range from 0 to 1. Therefore, higher values correspond to better classification performance. Other measures of evaluation include the ROC curve and the confusion matrix.

**2.2 Precision, Recall, F-measure, and Accuracy**

Precision and recall are two metrics that are often simultaneously used to verify classification performance. ‘Classic’ precision and recall are derived from the ratios of relevant instances and non-relevant instances. They also consider the relevant instances that are not retrieved. More specifically, precision measures the number of retrieved instances that are relevant, whilst recall measures the number of all the relevant instances that are successfully retrieved. Both metrics can be calculated using these equations:

Precision = TP / TP + FP

Recall = TP / TP + FN

Precision and recall are not often taken into account alone. The two measures are often used together in the F-measure, which provides a single weighted metric to evaluate the overall performance. F-measure can be measured by calculating the harmonic mean of precision and recall:

F-measure = 2 \* ( (Precision \* Recall) / (Precision + Recall))

Others use accuracy as a metric to measure the performance of a classifier. Accuracy measures the number of instances that were correctly classified. However, the problem of using accuracy to measure the effectiveness of a classifier is that if the classifier always predicts one class, a strategy that defeats the purpose of building a classifier, it will achieve high accuracy. This is known as the accuracy paradox.

Accuracy = (TP + TN) / (TP + FN + FP + TN)

We used Weka, a popular suite of machine learning software, to perform 10-fold cross-validation experiments in order to determine which classifier is best for our problem. 7 classifiers distributed as part of Weka were selected. Classifiers included generative models that consider conditional dependencies in the dataset or assume conditional independence (e.g. Bayesian Network, Naive Bayes), and discriminative models that aim to maximise information gain or directly maps data to their respective classes without modelling any underlying probability or structure of the data (e.g. J48 Decision Tree, Support Vector Machine). The following table demonstrates the results form cross-validation. The overall performance represents weighted-average results.

|  |  |  |  |
| --- | --- | --- | --- |
| Classifier | Precision | Recall | F-measure |
| BayesNet | 82.6 | 82.6 | 82.6 |
| Naïve Bayes | 86.0 | 85.9 | 85.9 |
| Support Vector Machine | 84.1 | 84.1 | 84.0 |
| Random Tree | 71.9 | 71.5 | 71.6 |
| Random Forest | 81.8 | 81.9 | 81.8 |
| J48 | 77.7 | 77.8 | 77.7 |
| Neural Network | 81.4 | 81.1 | 81.2 |

*figure 1.1*

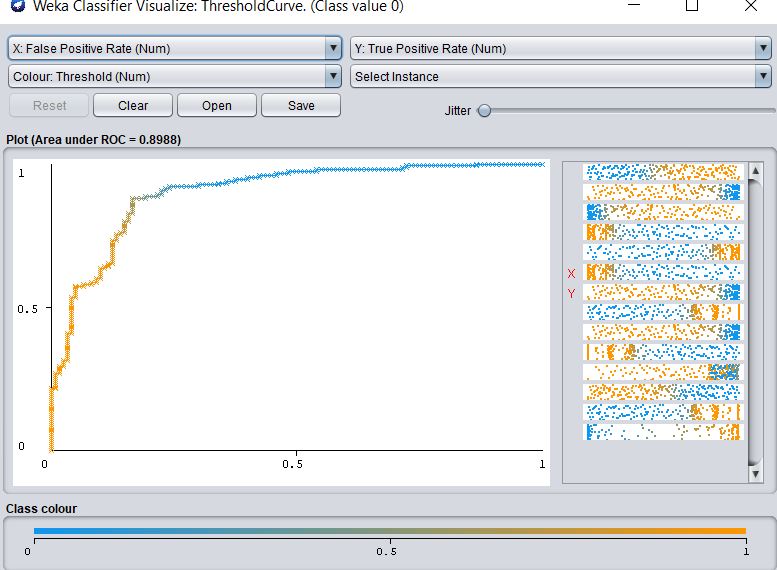
The results in figure 1.1 show that the best classifier for this problem is Naïve Bayes (precision of 86.0%, recall of 85.9% and F-measure of 85.9). The model has demonstrated to make the least incorrect predictions, misclassifying 39 out of 270 patients (figure 1.2).

The area under the ROC curve ( AUC ) is a measure of how well a parameter can distinguish between two diagnostic groups (with heart disease and without). The X-axis shows the false positive rate (100-Specificity) of the dataset while the Y-axis shows the true positive rate (Sensitivity) of the data both are used to compare predictions. Figure 1.3 shows the performance of the models across all possible states. The model ideal line would be at a 45-degree angle, given 50% data tested and 50% would be correctly predicted. If the entire​

the population is processed, the Naïve Bayes model appears to perform better than the other two as it gives the highest number of correct predictions (89.88%) followed by Neural Network (86.23%) and Random Trees (75.18%).​

|  |  |  |  |
| --- | --- | --- | --- |
| Predicted | |  |  |
| 0 | 1 |  |  |
| 132 | 18 | 0 | Actual |
| 21 | 99 | 1 |

*figure 1.2*



*figure 1.3*

**3.1 Applying model on unseen data**

We can further evaluate the performance of Naïve Bayes on our data set by splitting the data set to training and testing sets. The norm is to have more training instances than testing, as this is the data the classifiers learn from to make predictions on the unseen data (the testing data).

We use Weka’s percentage split function which splits our dataset into 60% training (162 patients) and 40% testing (108 patients). We apply the Naïve Bayes algorithm, which achieves a precision of 86.1%, recall of 85.2% and F-measure of 85.2%.

|  |  |  |  |
| --- | --- | --- | --- |
| Predicted | |  |  |
| 0 | 1 |  |  |
| 47 | 4 | 0 | Actual |
| 12 | 45 | 1 |

*figure 1.4*

**4. Conclusion**

In conclusion, the models are trained and validated against a test dataset. The most effective classifier again shows Naive Bayes promising results in terms of classification performance followed by Neural Network and Decision Trees. The Naïve Bayes algorithm misclassifies 16 patients out of 106 with heart disease when they do not possess any problems. In this context, neither classifiers are the best performing for this problem. As this is a problem which deals with human life, a classifier needs to achieve a significantly higher classification performance to be considered as being suitable for classifying whether patients have heart disease or not. The reason for this is that, if a doctor is to rely on the results, they could be dispensing drugs to patients who have been classified as having heart disease when they indeed do not have it. This could have more of an impact on their health.

In order to achieve a higher classification performance, we would focus on reducing the false positives (misclassifications). To do this, we could add more data or include more specific features that are more of an indication of heart disease. Other approaches to incorporate additional streams of data instead of just categorical data is the use of Text Mining. Since the majority of valuable health knowledge is summarized in medical notes, incorporating unstructured data can be invaluable.

**5. References**

[1] C. Raju, E. Philipsy, S. Chacko, L. Padma Suresh, and S. Deepa Rajan, "A Survey on Predicting Heart Disease using Data Mining Techniques," 2018 Conference on Emerging Devices and Smart Systems (ICEDSS), Tiruchengode, 2018, pp. 253-255.

[2] Abdar, M. (2015). Using decision trees in data mining for predicting factors influencing heart disease. Carpathian Journal of Electronic and Computer Engineering

[3] Zriqat, I. A., Altamimi, A. M., & Azzeh, M. (2016). A comparative study for predicting heart diseases using data mining classification methods. International Journal of Computer Science and Information Security, 14(12), 868-879.

[4] S. Palaniappan and R. Awang, "Intelligent heart disease prediction system using data mining techniques," 2008 IEEE/ACS International Conference on Computer Systems and Applications, Doha, 2008, pp. 108-115.

[5] Journal of the American Heart Association. Oxford: Published on behalf of the American Heart Association Inc. by Wiley-Blackwell, 2012. Print.