Heart Disease Prediction

Thushani Kalamullage- 001178245

Abstract

Heart and circulatory diseases are causing 160,000 deaths each year. 7.6 million people have cardiovascular disease in the UK. There were 900,000 people having heart failure, which results in 100,000 hospital admissions each year.(*Facts and figures* n.d.) This project scope is based on predicting heart attacks which might be able to save a considerable amount of lives in the future. The proposed solution is to experiment with the machine learning classification methods to predict the result. In this project, 8 machine learning techniques were used to predict heart patients using the Cleveland dataset. K nearest neighbour classifier performs better when tuning the hyperparameter with an accuracy of 90.16%. Finally, the support vector machine is the best classifier for predicting heart disease.

1. Introduction

Heart disease is a major cause of death in the world's population. One person dies every three minutes in the UK because of heart disease. In Clinical data analysis, cardiovascular disease prediction is considered one of the most important matters. 85% of deaths are caused by heart attacks and strokes in cardiovascular disease.(n.d.) This project is for predicting heart diseases using the machine learning algorithm. Because of the heavy health care data and factors, getting a decision about patients is difficult. Data mining helps to convert raw medical data into information, and it makes decisions easy. Considering the size of the data that I have used in this project is not something that converts into useful information efficiently

Identifying heart disease is a little bit difficult and risky because a lot of factors have to be considered. Using Machine Learning techniques helps to identify the risk factors of heart diseases and make predictions accurately and fast manner. As a modern approach, machine learning has the capability to handle a large amount of data very accurately and it is the best way to predict heart disease. In this project, I have used a machine learning approach to classify a heart patient. I have used the Cleveland Heart Disease dataset taken from UCI Repository. In their 76 attributes but this project, I only used 14 attributes. According to this data set, classification is used to identify a heart patient. I have used Support vector machine(SVM), Logistic regression, Naïve bays, Stochastic Gradient Descent(SGD), K- nearest neighbour(KNN), Decision tree and Random forest as machine learning techniques for evaluation.

Generally, These are the set of most used classification techniques for solving classification problems. The above techniques are proposed based on the Prediction of Cardiac Disease using Supervised Machine Learning Algorithms paper.(Princy et al. 2020) The specific reasons for that I have chosen the above techniques will be covered in the method section.

2. Methods

As classification models, I have used Kernel SVM, Linear SVM, Logistic regression, Naïve Bays, Stochastic Gradient Descent, K -Nearest Neighbours, Decision tree and Random Forest to classify the heart patients.

The hypothesis function for classification can show as,

$$y = x^{\top}\theta \tag{1}$$

$$h_\theta(x) = \begin{cases} ClassN & if g_\theta(x) < 0 \\ ClassP & if g_\theta(x) > 0 \end{cases} \tag{2}$$

The $g_\theta(x)$ is the discriminant function. The cost function $J(\theta)$ is given by,

$$J(\theta) = L(\theta) + R(\theta) \tag{3}$$

$$R(\theta) = \lambda||\theta||_2^2. \tag{4}$$

Loss function $L(\theta)$ differs from the model and it shows how much data will fit the model. $R(\theta)$ is a Regularisation term and it shows how much effect it has on the training sample. It can reduce overfitting.

2.1. Logistic regression

Logistic regression is used for classifying binary output. To do so, there is a logistic function also known as the sigmoid function(ς). The discriminant function of logistic regression can express as follow,

$$g_\theta(x) = x_i^{\top}\theta \tag{5}$$

This discriminant function shows a log odd ratio. The log odd ratio can be expressed as follows,

$$g_\theta(x) = \frac{P[y^i = 1]}{1 - P[y^i = 1]} = x_i\theta \tag{6}$$

After deriving this equation probability of the y equals to one can express as follows,

$$P[y^i = 1] = \frac{1}{1 + exp(-x_i\theta)} = \varsigma(x_i\theta)$$

$$P[y_i|x_i\theta] = \begin{cases} 1 - \varsigma(x_i\theta)) & if y_i = -1 \\ \varsigma(x_i\theta)) & if y_i = +1 \end{cases} \tag{7}$$

To get the score of this logistic regression, have to maximise the $P[y|X\theta]$with respect to θ. After optimisation, I have used this for prediction.

$$P[y_(i)|x_i\theta] < 0.5 \rightarrow x_i^\top\theta < 0$$
$$P[y_(i)|x_i)\theta] > 0.5 \rightarrow x_i^\top\theta > 0$$

Then the prediction rule will be,

$$y_i = \begin{cases} -1 & if x_i^\top\theta < 0 \\ +1 & if x_i^\top\theta > 0 \end{cases} \tag{8}$$

The logistic regression is matched for the data set that I used in this project because I have to classify the binary output. In Python, I used the "sklearn.linear_model" library to create a logistic regression object and used an inbuilt function to train the model.

2.2. Support Vector Machine(SVM)

SVM is another model I used in this project to predict heart patients. This model has the capability to handle an infinite number of features. The SVM model helps to find complex relationships over the factors that are provided. Then it is advantageous for this dataset for prediction. In this project, I have used the radial basis function(RBF) kernel and linear kernel. Kernels are the group of mathematical functions used in SVM algorithms. SVM chooses the best hyperplane to separate the group using the maximum distance of each point.

Hypothesis function can be denoted by,

$$h_\theta(x) = q[\phi(x)]^\top\theta \tag{9}$$

And prediction rule will be,

$$y_i = \begin{cases} -1 & if [\phi(x)]^\top\theta < 0 \\ +1 & if [\phi(x)]^\top\theta > 0 \end{cases} \tag{10}$$

In this project, I have used the LinerSVC SVM module and SVC with kernel RBF SVM module to predict heart disease.

2.3. Naïve Bays

Naive Bayes is based on the simple rule that the existence of a feature in a class is not related to any other features in the class. All the properties separately support the probability of the last result. Bayes theorem represents the calculation of posterior probability as below,

$$P(c|X) = \frac{P(x|c)P(c)}{P(x)} \tag{11}$$

Naive Bayes is easy to implement and consumes a very less amount of time to predict, compared to most other classifications. Naive Bayes has a better performance compared to logistic regression when we have less amount of test data. I have proposed this technique to be used here, because of the better performance. To create the naive Bayes model, I have used the GaussianNB module in sklearn.naive_bayes library. (*Machine Learning Archives* n.d.)

2.4. Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent means descending a slope to archive the lowest value in the set. It starts with a random point on that particular function and descends down in the slop. This will iteratively reach the lowest point until it gets the gradient as almost 0. This model is suitable for our data set because of its efficiency.

Equation of the hinge log function can derive as below,

$$L(y_i, f(x_i)) = max(0, 1 - y_i f(x_i)) \tag{12}$$

I have used SGDClassifier in sklearn.linear_model library to create the SGD object. As a loss parameter, I have passed hinge, modified_huber and log. When passing the hinge value as a parameter is gives the highest accuracy. This model with hinge loss is equivalent to the SVM model. And SGD with log loss function act as logistic regression. (*User guide: Contents* n.d.)

2.5. K -Nearest Neighbours

This model is a supervised learning algorithm, and it shows how to classify neighbour points. This method classifies the data by measuring the distance. In commonly used Euclidean distance to meassure distance of nearest neighbours.

The parameter k depends on the data set that we used. Running the code for different times to identify the best k for the data set according to the accuracy. If k=n, find n nearest neighbours to measure distance. KNN models classify the data into classes considering the lowest distance between

the class and data points. As per the distance measurement Euclidean Distance, Hamming Distance, Minkowski Distance and Manhattan Distance methods can be used. But euclidean distance is commonly used with this model.

Euclidean distance can show as below,

$$d(x, y) = \sum_{i=1}^{m} (x_i - y_i)^2 \quad (13)$$

This model is used to classify the dataset by the distance of the attributes. In this project, I have used KNeighborsClassifier from sklearn.neighbors library to create the model. k is the only hyperparameter in this model. (*User guide: Contents* n.d.)

2.6. Decision Tree

The decision tree is able to be used to solve the regression as well as the classification problems.

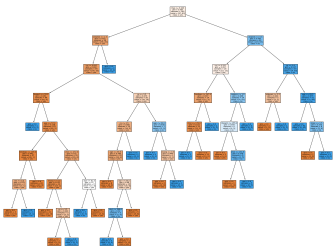

Figure 1. Decision Tree

It will Calculate the entropy of each attribute to divide the data with features that have maximum information gain or minimum entropy. Entropy is basically the uncertainty of the data set which measures the disorder of the data.

This will generate the flowchart like a tree structure result of feature-based classification as shown in the Figure 1. (*Machine Learning Archives* n.d.)

$$Entropy(S) = \sum_{i=1}^{c} -P_i log_2 Pi \quad (14)$$

$$Gain(S, A) = S - \sum_{i=\nu \in Values(A)} \frac{|S_v|}{|S|} \dot{E}ntropy(S_v) \quad (15)$$

In the implementation, sklearn.tree library was used to create the DecisionTreeClassifier object.Because easiness of the capturing the Non-linear pattern in the dataset, this model will suit our dataset. As a criterion, I have used the gini criterion because entropy criterion reduces the accuracy and F1 score. (*User guide: Contents* n.d.)

2.7. Random Forest

Random Forest is a supervised learning technique that is used very often for classification problems. This is capable of handling data sets with continuous variables and categorical variables as well. Mostly it has an accurate result on the classifications. One of the concerns over the Random Forest is the slowness. It doesn't follow any formulas and it selects the observations of a decision tree for getting average output. This is the improved version of the decision tree. Controlling the overfitting is one of the advantages of this technique and this advantage led me to choose this technique as a model for this project. As described in the decision tree, the same parameter is used in the random forest. The RandomForestClassifier is created by sklearn.ensemble library with gini impurity.

3. Experiments

I have used the Cleveland processed dataset as a .dat file in the UCL repository. It included 14 attributes such as age, sex, pain type, Resting blood pressure, cholesterol , blood sugar, resting ECG, heart rate, exercise induces angina, depression after exercise, Slope of the peak, number of major vessels coloured by fluoroscopy, the status of the heart and target.

3.1. Experimental settings

In the pre-processing section, I removed the missing value in the ca and thal attribute. And I change the target attribute value to 0 and 1 because 1 – 4 represents the risk of heart diseases and 0 represent no risk. Then split the data into 3 sets for training, validating and testing as shown in Figure 2. After that, I have normalised the dataset to improve accuracy. To do this I have used sklearn.preprocessing library to create StandardScaler object.

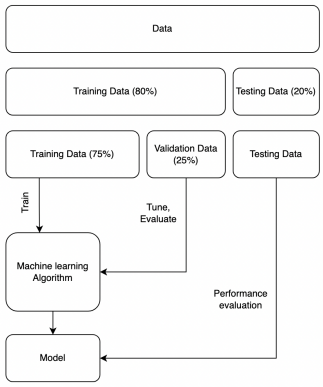

Figure 2. Dataset splitting

Then I have applied machine learning models such as SVM(Support Vector Machine), Logistic Regression Naïve bays, SGD, KNN, Decision Tree and random forest classifiers for prediction. Hyperparameters of the models are shown in the Table 1.

Machine Learning models	Parameter Name	Value
Kernel SVM, Linear SVM and Logistic regression	Regularization strength C	$1.0 \times 10^{\pm 0}$
Kernel SVM	Kernel	RBF Kernel
KNN	n_neighbors	4
Random Forest	n_estimators	10
SGD	loss	hinge

Table 1. Hyperparameter of diffrent models

3.2. Evaluation criteria

As evaluation criteria, I have mainly used the F1 score because it is good for evaluating a model with imbalanced data. Moreover, I used accuracy, precision and recall as well. Evaluating the performance for classification models confusion matrix can be used.

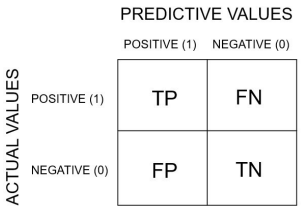

Figure 3. Confusion Matrix

According to the confusion matrix precision, accuracy, recall, and F1 score can be measured as follows,

$$\boldsymbol{Accuracy} = \frac{(\boldsymbol{TP} + \boldsymbol{TN})}{(\boldsymbol{P} + \boldsymbol{N})}$$

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{(\boldsymbol{TP} + \boldsymbol{FP})}$$

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{(\boldsymbol{TP} + \boldsymbol{FN})}$$

$$\boldsymbol{F1score} = \frac{\mathbf{1}}{\frac{\mathbf{1}}{\mathbf{2}}(\frac{\mathbf{1}}{\boldsymbol{Precision}} + \frac{\mathbf{1}}{\boldsymbol{Recall}})}$$

3.3. Results

According to the tabulated results in Table 2, Linear SVM, Logistic regression, KNN has the highest accuracy. But comparing the F1 score, Linear SVM has the highest value and it is 0.896552. Logistic regression and KNN have similar values in accuracy and F1 score. Considering the accuracy and F1 score, Linear SVM is the best model for predicting heart disease.

Machine Learning models	Accuracy	Precision	Recall	F1 score
Kernel SVM	0.868852	0.888889	0.827586	0.857143
Linear SVM	0.901639	0.896552	0.896552	0.896552
Logistic regression	0.901639	0.925926	0.862069	0.892857
Naïve Bays	0.868852	0.888889	0.827586	0.857143
SGD	0.885246	0.866667	0.896552	0.881356
KNN	0.901639	0.925926	0.862069	0.892857
Decision Tree	0.836066	0.787879	0.896552	0.83871
Random forest	0.868852	0.888889	0.827586	0.857143

Table 2. Performance of machine Learning techniques

3.4. Discussion

As per the observation in the Table 2 that I had, when it comes to the larger data set, We can expect faster convergence in the LinearSVC. Logistic regression and KNN are the second-highest accurate model and still, it has a 0.003695 F1 score difference compared to the Linear SVM. It's not a considerable gap with other results. Kernel SVM and the Naive Bayes tend to be getting similar accuracy and the F1 score.

4. Conclusion

Heart diseases are one of the most common health risks in the present world due to lack of exercise and bad food habits. According to the statistics in the UK, we are looking at 160,000 deaths each year.*Facts and figures* (n.d.) If we can work on this scope, it's a large number of people we can save per year. All so a this is a widely researched area in medical history. In this project, I'm considering the 8 Machine learning classification algorithms. With the observation that I had around the mentioned techniques, the Linear SVM algorithm is the one we achieve with the highest accuracy and the highest F1 score. We achieved 90.16% accuracy in the Linear SVM compared to other models. We can consider Logistic regression and KNN as second-highest accuracy and the F1 score archiving 90.16% and 89.29% accordingly. Tuning the hyperparameter of the KNN and random forest gives the better performance to the dataset.

References

(N.d.). URL: https://archive.ics.uci.edu/ml/datasets/Heart+Disease.

Facts and figures (n.d.). URL: https://www.bhf.org.uk/what-we-do/news-from-the-bhf/contact-the-press-office/facts-and-figures.

Machine Learning Archives (n.d.). URL: https://www.analyticsvidhya.com/blog/category/machine-learning/?utm_source=blog_navbar&utm_medium=machine_learning_button.

Princy, R. Jane Preetha et al. (2020). "Prediction of Cardiac Disease using Supervised Machine Learning Algorithms". In: *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 570–575. DOI: 10.1109/ICICCS48265.2020.9121169.

User guide: Contents (n.d.). URL: https://scikit-learn.org/stable/user_guide.html.