Performance Analysis of Machine Learning Methods for Heart Disease and Liver Cancer Detection

Xuhui Wang

February 2021

Introduction

Heart disease and cancers are the most prevalent diseases worldwide. In spite of the commonness, the diagnosis and detection for these diseases demand extensive study and experience in the domain of medicine. In this project, I apply different approaches with different optimization and tuning methods to detect these diseases, which involves the use of machine learning algorithms to identify diseases based on up to 33 attributes observed on analysing of the patients' clinical features. Upon training and testing for these two diseases, I achieved an accuracy of up to 95 percent.

Methods for Heart Disease Detection

Heart Disease

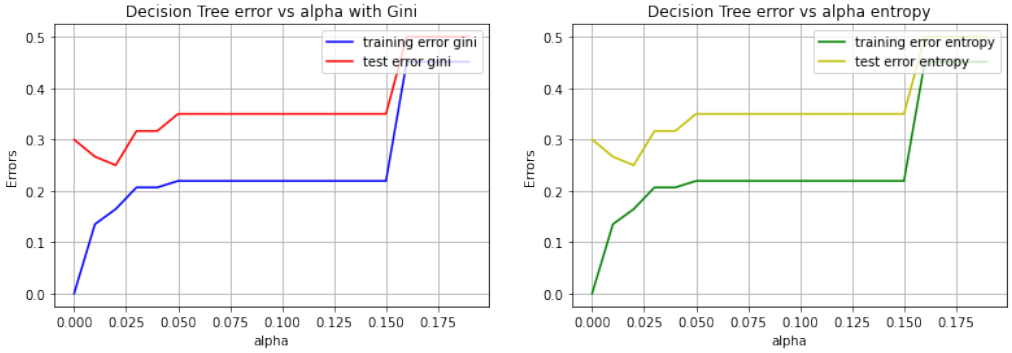
For this problem, the three following methods have been implemented: Decision tree, Random forest and Neural Network, each of which is applied on two binary classification problems. The experiment, performance, analysis and comparison between different methods on the same problem would be represented on the specific section of the problem.

Given is a heart disease dataset from UCI repository containing 297 individuals of 13 features. We would much focus on the changes of training error and test error based on controlled method, in order to figure out the best performance at different conditions and methods, as well as give the related analysis in terms of accuracy.

Decision Tree

Experiments

Firstly, post pruning is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 1 has two sub figures, where 1(a) shows the information of decision tree with Gini Index split criterion and 1(b) shows the information of decision tree with entropy criterion.



(a) Gini Index (b) Entropy

Figure 1: Decision Tree

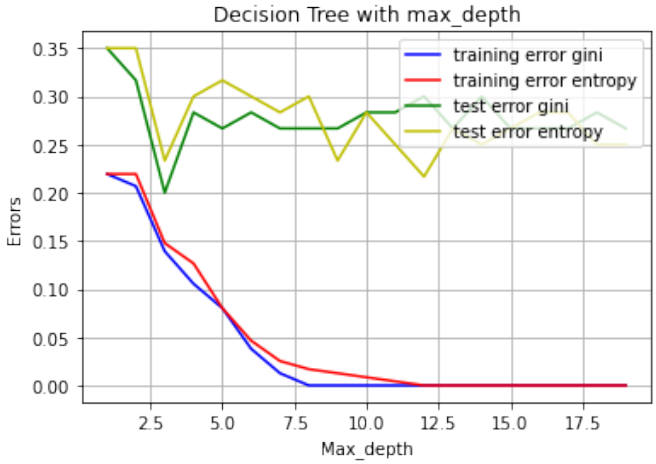
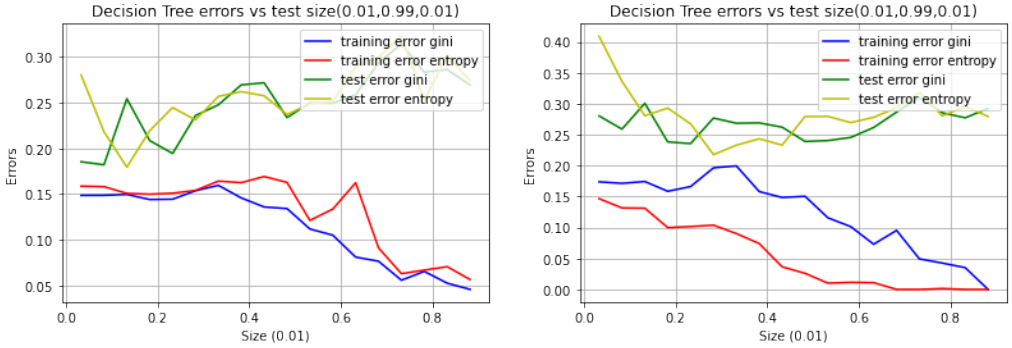


Figure 2: Max Depth

Secondly, pre pruning is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 2 shows how training error and test error with different criterion change respectively as the maximum depth changes within a specific range.

For camparison, pre pruning and post pruning are applied on the dataset. Figure 3 shows how training error and test error change respectively as the test size changes within a specific range, where 3(a) shows the information of pre pruning decision tree with *max depth* = 3 and 3(b) shows the information of post pruning decision tree with *complexity cost* = 0.02.



(a) Pre Pruning

(b) Post Pruning

Figure 3: Decision Tree

Analysis

For the post pruning decision tree method, both Gini Index and entropy criterion receive the best performance when α is around 0.02. The similar trend of lines in Figure 1(a) and 1(b) shows that there is no big gap of accuracy between the methods with different criterion. From the graph, it is clearly observed that training error and test error increase respectively as the cost complexity values increases until $\alpha = 0.05$, after which they both remain constant.

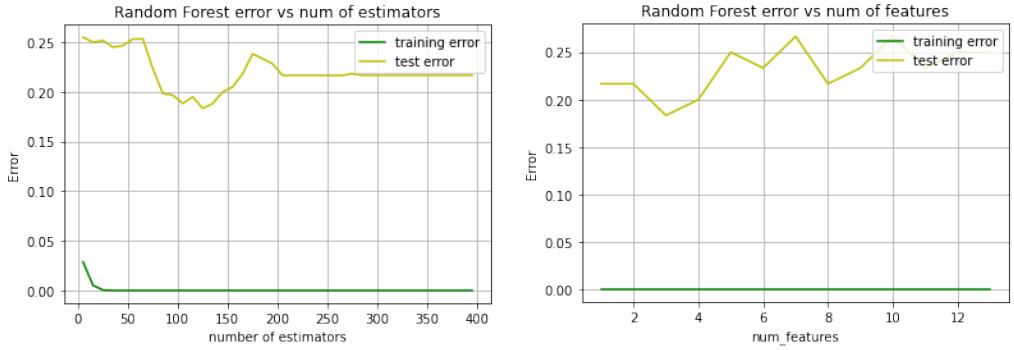
For the pre pruning decision tree method, the training error and test error with both Gini Index and entropy reach the best performance at $max\ depth = 3$. then followed by a small fluctuation.

Figure 3(a) and 3(b) show that there is an inversely proportional relationship between the training error and test error with the same testset capacity, no matter for pre pruning or post pruning decision tree. We could observe that for both figures, the test errors are very small with the test size in range $[0.1, 0.3]$.

Random Forest

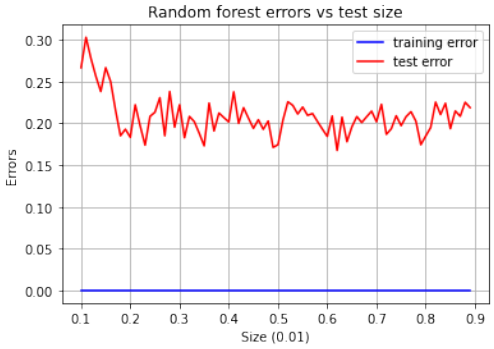
Experiments

Random forest is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 4 has three sub figures, where 4(a) shows how training error and test error change as the number of estimators changes, 4(b) shows how training error and test error change as the number of features changes with 120 estimators, and 4(c) shows how training error and test error change as the size of testset changes with 120 estimators and 3 features.



(a) estimator

(b) feature



(c) test size

Figure 4: Random Forest

Analysis

Figure 4(a) shows that the test error comes to the gap at the number of estimators = 125 and gradually becomes stable as the number of estimators increases. Training error reaches the best performance at the number of estimators = 25. To summarize, the number of estimators = 125 is best suited for both training error and test error for this problem.

Figure 4(b) and Figure 4(c) show that the training error always remains 0 no matter how the number of features and testset size changes. But, the test error receives the best performance at the number of features = 3 and keeps small fluctuation when the test set capacity is greater than 0.2.

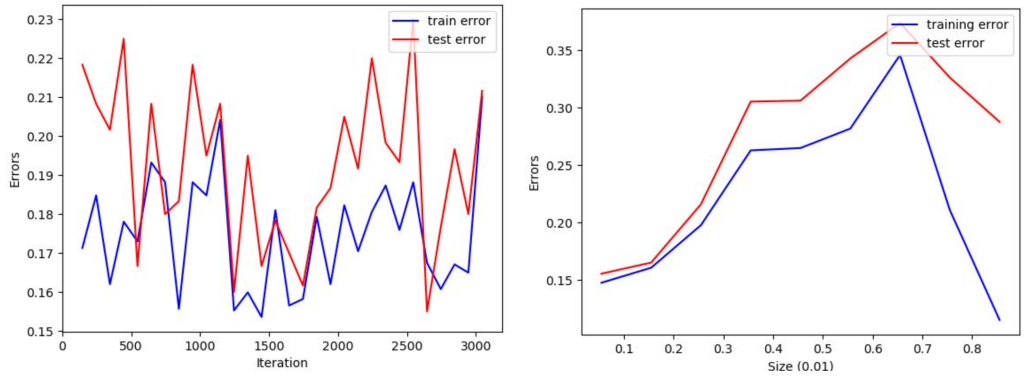
Neural Networks

Experiments

Data was normalized before being feed to the neural network to ensure the uniformity and consistency of the experiments.

Figure 5(a) shows that the training error and test error change as the number of iteration changes. There are 120 nodes in 1 hidden layer and 0.03 learning rate. Figure 5(b) shows that the training

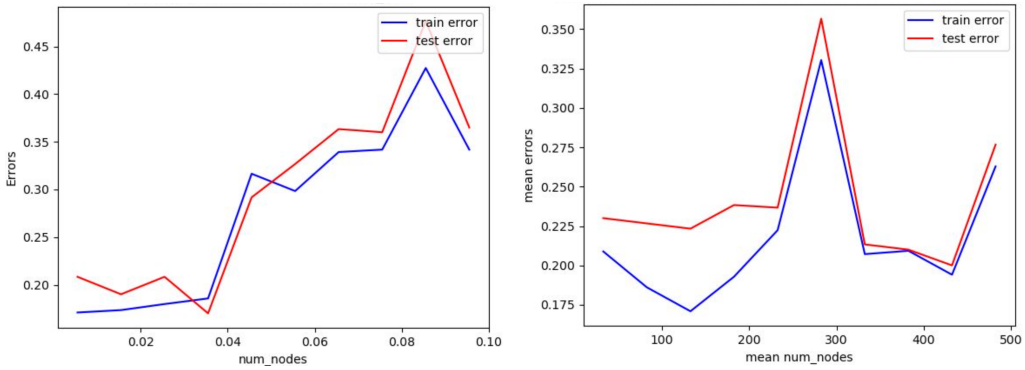
error and test error change as the test size changes. There are 120 nodes in 1 hidden layer, 0.03 learning rate and 1750 iterations.



(a) NN errors vs iteration (b) NN errors vs test size

Figure 5: Neural Network

Figure 6(a) shows that the training error and test error change with the different learning rate. There are 120 nodes in 1 hidden layer and 1000 iterations. Figure 6(b) shows that the training error and test error change with the different number of nodes in the hidden layer. There are 0.03 learning rate and 1000 iterations.



(a) NN errors vs learning rate (b) NN errors vs number of nodes

Figure 6: Neural Network

Analysis

From Figure 5(a), it could be observed that the training error and the test error change unstably with the change of number of iterations. Despite of the big fluctuation, it is safe to say that these two errors decrease to a gap at 1250-1750 times of iteration. Specifically, the test error reach the lowest point around 1250 times of iteration.

Figure 5(b) shows that the training error and test error both increase as the test size increase

until the test size = 0.65, and then they decrease. As shown in this figure, the convention of selecting test size within $[0.1, 0.2]$ is reasonable and feasible.

Figure 6(a) shows that the training error and the test error increase with the increasing learning rate. Despite of the considerable increase, they achieve good performances, lower than 0.2, at the learning rate within a range of $[0.02, 0.04]$.

From Figure 6(b), it could be observed that the training error and the test error have similar trend of change. With the increasing number of nodes in the hidden layer, they start to increase from number of nodes = 100, reach the peak around number of nodes = 300 and then start to decrease. However, the test error reaches its best performance, lower than 0.17, at the number of nodes = 130, which might be an idea choice for this problem.

Methods for Liver Cancer Detection

Liver Cancer

For this problem, the three following methods have been implemented: Decision tree, Random forest and Neural Network, each of which is applied on two binary classification problems. The experiment, performance, analysis and comparison between different methods on the same problem would be represented on the specific section of the problem.

Cancer is a disease in which cells in the body grow out of control. When cancer starts in the liver, it is called liver cancer. Each year in the United States, about 24,000 men and 10,000 women get liver cancer, and about 18,000 men and 9,000 women die from the disease. The percentage of Americans who get liver cancer has been rising for several decades, but may be beginning to level off. Liver cancer is more common in other parts of the world than in the United State. Curing liver cancer is one of the most challenging tasks in human society.

With an accurate machine learning prediction model, it is non-trivial to find by how much conditions and medical test results are related to the disease, and therefore find ways for people to prevent, treat, and cure cancer.

Given is a liver cancer dataset from the Internet containing 569 subjects of 33 features including diagnosis and so on. These features are related to the liver cancer based on the previous publications, and the data is collected from real patients. We would much focus on the changes of training error and test error based on controlled method, in order to figure out the best performance at different conditions and methods, as well as give the related analysis in terms of accuracy. The machine learning methods adopted in this study can guarantee their accuracy without feature selection, and therefore it enables good comparison of the methods.

Decision Tree

Experiments

Firstly, post pruning is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 7 has two sub figures, where 7(a) shows the information of decision tree with Gini Index split criterion and 7(b) shows the information of decision tree with entropy

criterion. From the graph, it is clearly observed that training error and test error increase respectively as the cost complexity value increases within a specific range, but the increases stop and remain constant at $\alpha = 0.05$.

(a) Gini Index (b) Entropy

Figure 7: Decision Tree

Secondly, pre pruning is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 8 shows how training error and test error with different criterion change respectively as the maximum depth changes within a specific range.

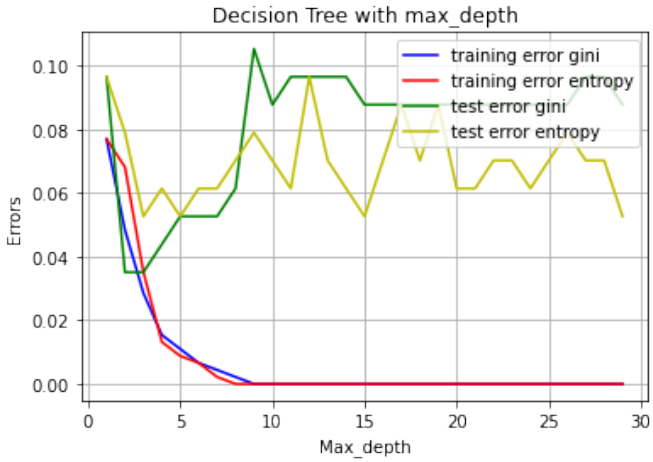
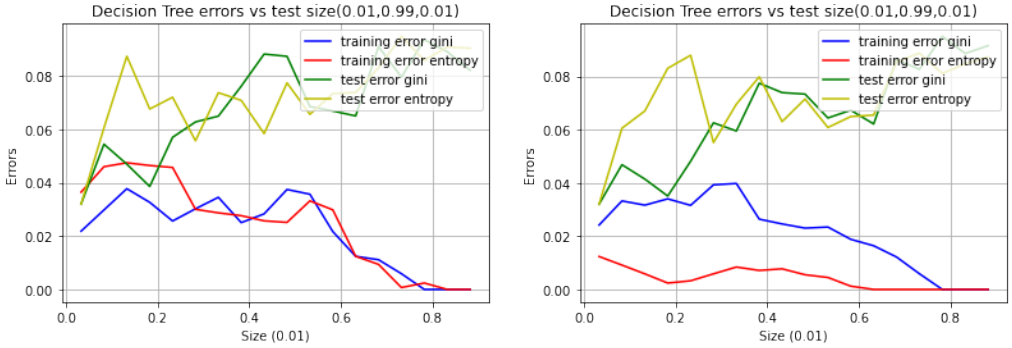


Figure 8: Max Depth



(a) Pre Pruning

(b) Post Pruning

Figure 9: Decision Tree

For camparison, pre pruning and post pruning are applied on the dataset. Figure 9 shows how training error and test error change respectively as the test size changes within a specific range, where 9(a) shows the information of pre pruning decision tree with *max depth* = 3 and 9(b) shows the information of post pruning decision tree with *complexity cost* = 0.01.

Analysis

Regarding to the cost complexity, for the post pruning decision tree method, both Gini Index and entropy criterion receive the best performance when α is around 0.01. The similar trend of lines in Figure 7(a) and 7(b) shows that there is no big gap of accuracy between the methods with different criterion.

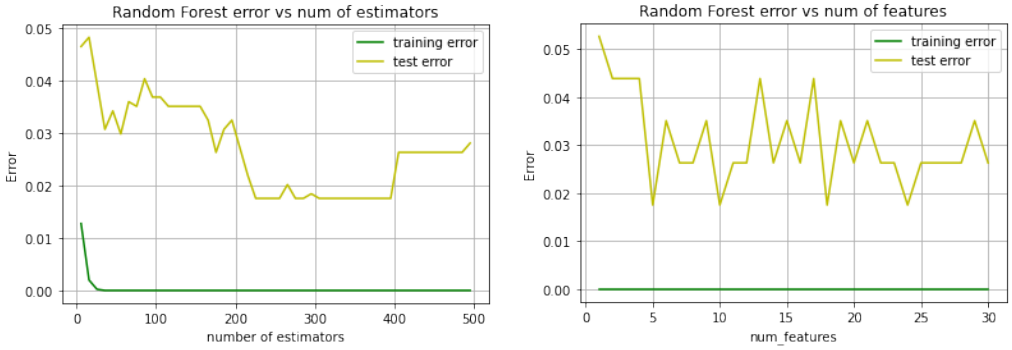
For the pre pruning decision tree method, the training error and test error with both Gini Index and entropy reach the best performance at *max depth* = 3, then followed by a big fluctuation of test error.

Figure 9(a) and 9(b) show that there is an inversely proportional relationship between the training error and test error with the same testset capacity, no matter for pre pruning or post pruning decision tree. However, we could reach a relatively ideal test error with the test size in range $[0.1, 0.2]$, which shows that this may be a feasible and reasonable test set/training set ratio.

Random Forest

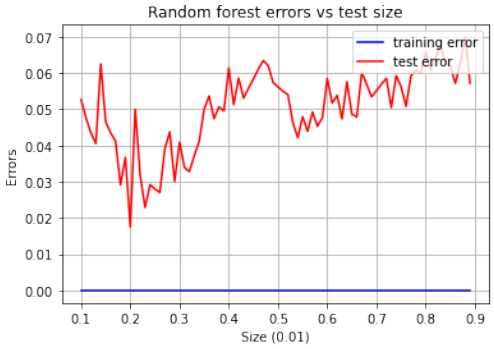
Experiments

Random forest is applied on the dataset separated into training set with 80% data and test set with 20% data. Figure 10 has three sub figures, where 10(a) shows how training error and test error change as the number of estimators changes, 10(b) shows how training error and test error change as the number of features changes with 230 estimators, and 10(c) shows how training error and test error change as the size of testset changes with 230 estimators and 5 features.



(a) estimator

(b) feature



(c) test size

Figure 10: Random Forest

Analysis

Figure 10(a) shows that the training error remains 0 since the number of estimators is 30. Meanwhile, test error keeps decreasing and remains the lowest value, lower than 0.02, since the number of estimators reach 225. To summarize, the number of estimators = 225 is best suited for both training error and test error for this problem.

Figure 10(b) shows that the test error experiences a fluctuation in range $[0.02, 0.05]$ as the number of features increases. It is observe that the test error hits 0.02 when the number of features = 24, so we know that, at least, the increasing number of features would not bring troubles to achieving good performances.

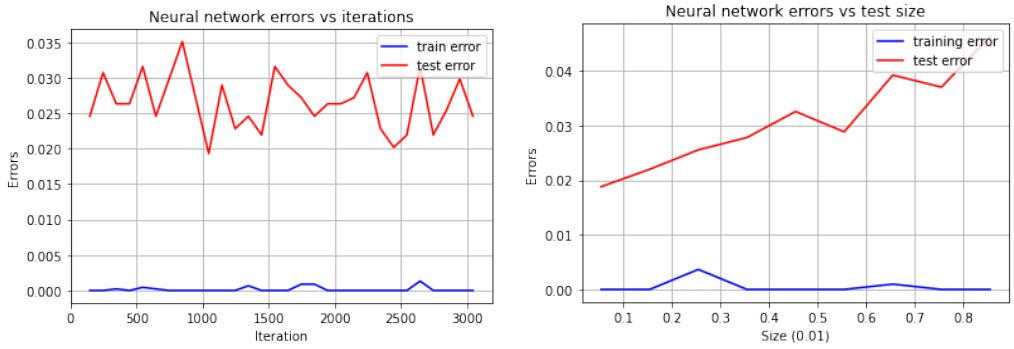
Figure 10(c) shows that despite of fluctuation, the test error keeps increasing as the test size increases. It reaches the best performance, 0.02, at the test size = 0.2. Furthermore, a good way to improve the accuracy of random forest is to find and delete the bad predictors. However, increasing the number of features does not always enhance the accuracy and deleting important related features would cause low accuracy.

Neural Networks

Experiments

Data was normalized before being feed to the neural network to ensure the uniformity and consistency of the experiments.

Figure 11(a) shows that the training error and test error change as the number of iteration changes. There are 130 nodes in 1 hidden layer and 0.02 learning rate. Figure 11(b) shows that the training error and test error change as the test size changes. There are 130 nodes in 1 hidden layer, 0.02 learning rate and 2000 iterations.

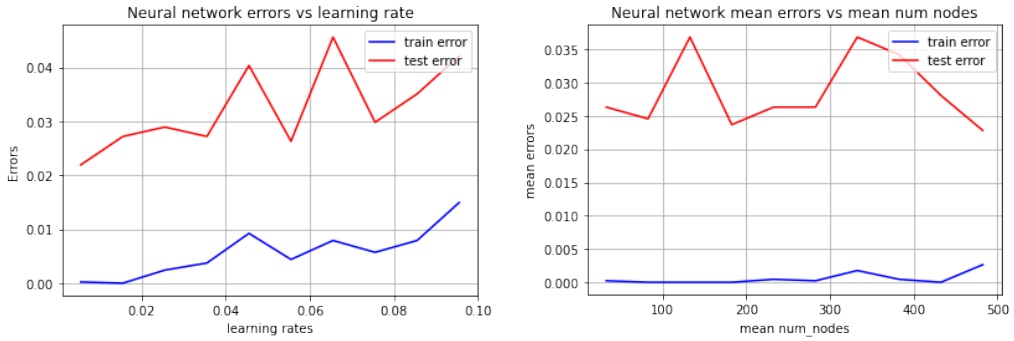


(a) NN errors vs iteration

(b) NN errors vs test size

Figure 11: Neural Network

Figure 12(a) shows that the training error and test error change with the different learning rate. There are 200 nodes in 1 hidden layer and 1000 iterations. Figure 12(b) shows that the training error and test error change with the different number of nodes in the hidden layer. There are 0.02 learning rate and 1000 iterations.



(a) NN errors vs learning rate

(b) NN errors vs number of nodes

Figure 12: Neural Network

Analysis

From Figure 11(a), it could be observed that the test error change unstably with the change of number of iterations, while the training error remains almost 0 all the time. Despite of the big fluctuation, it is safe to say that the error decreases to a gap at 1000 times of iteration.

Figure 11(b) shows that the test error stably increases as the test size increase. As shown in this figure, the convention of selecting test size ≤ 0.2 is reasonable and feasible.

Figure 12(a) shows that the training error and the test error increase with the increasing learning rate. Despite of the small fluctuation, they achieve good performances, lower than 0.03, at the learning rate within a range of $[0.01, 0.04]$.

From Figure 12(b), it could be observed that the test error experiences a big fluctuation. With the increasing number of nodes in the hidden layer, it starts to increase from number of nodes = 80, reaches the peak around number of nodes = 130, start to decrease and then achieves excellent performance at the number of nodes = 180, which is a good point to minimize the error.

Conclusion

As illustrated in the above information for heart disease detection, decision tree method achieves good test error with the range of $[0.2, 0.3]$; For random forest, the test error ranges in $[0.2, 0.3]$; However, neural network receives the best test error with the range of $[0.1, 0.2]$. Therefore, we state that though decision tree and random forest are considerably desirable, neural network is best suited for the prediction for this problem.

As illustrated in the above information for liver cancer detection, decision tree method achieves good test error with the range of $[0.05, 0.1]$; For random forest, the test error ranges in $[0.02, 0.05]$; However, neural network receives the best test error with the range of $[0.02, 0.04]$. Therefore, we state that though decision tree and random forest are considerably desirable, neural network is best suited for the prediction for this problem.

Upon applying three different methods on 2 different problems. Given the accuracy of up to 95%, I conclude that machine learning is an excellent method to perform this difficult task, disease detection. Except the results, analysis and summation above, some problems came up and blocked me from going further and acquire better performance. There are some ways to enhance accuracy of these three models: Apply more data, especially high quality data. It is also important to treat missing and outlier values well. Feature engineering helps to extract more information from existing data. Finally, use multiple algorithms and better parameter tuning.