|  |
| --- |
| Data Science with R II (JEM220) – Charles University Prague, FSV |
| Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients |
| by Anchana Khemphila & Veera Boonjing |

|  |
| --- |
| Analysis and improvements by Erik Nemcik & Paul Mainka  17.5.2020 |

Overview

In their study Anchana Khemphila and Veera Boonjing compare the performance of logistic regression, decision trees and artificial neural networks in classification problems. For this purpose, the authors employ the Cleveland Heart Disease dataset on the development of heart disease among individuals.

In order to evaluate the performance of the three different methods, the authors use measurements of area under the curve (AUC), sensitivity, specificity, accuracy and the error rate. Since artificial neural networks have the lowest error rate and the highest accuracy in their estimations, they conclude that artificial neural networks is the most suitable of the three classification techniques assessed for this data set.

# Data

The data set used by Boonjing and Khemphila includes 303 individuals, 13 explanatory variables and the dependent variable whether the patient has developed a heart disease or not. The data set is quite balanced with 164 patients having developed a heart disease, which corresponds to approximately 54% of all the patients. Of the 13 explanatory variables, 8 are factor variables and 5 are continuous variables.

Description of the eight factor variables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Name | Meaning | Different Levels | Frequency of occurrence |
| 1 | Sex | Patient gender | Female | 95 (32%) |
| Male | 206 (68%) |
| 2 | Chest Pain Type |  | Angina | 142 (47%) |
| Abnormal | 50 (17%) |
| Nonanginal pain | 86 (29%) |
| Asymptotic | 23 (8%) |
| 3 | Fasting Blood Sugar | Is fasting blood sugar less than 120? | True | 257 (85%) |
| False | 44 (15%) |
| 4 | Resting ECG | The electrocardiogram (ECG) measures the heart’s electrical activity, and a resting ECG is administered when the patient is at rest. | Normal | 146 (49%) |
| Abnormal | 155 (51%) |
| 5 | Induced Angina | Does the patient experience angina as a result of exercise? | True | 203 (67%) |
| False | 98 (33%) |
| 6 | Slope | Slope of the peak exercise ST segment. | Up | 21 (7%) |
| Flat | 139 (46%) |
| Down | 141 (47%) |
| 7 | Number Coloured Vessels | Number of major vessels coloured by fluoroscopy. | 0 | 173 (57%) |
| 1 | 65 (22%) |
| 2 | 38 (13%) |
| 3 | 25 (8%) |
| 8 | Thal | Thalassemias, blood disorders characterized by decreased hemoglobin production | Normal | 18 (6%) |
| Fixed defect | 166 (55%) |
| Reversable defect | 117 (39%) |

Description of the 5 continuous variables:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Name | Meaning | Min | Median | Mean | Max |
| 9 | Age | Age in years | 29 | 56 | 54.38 | 77 |
| 10 | Blood Pressure | Resting blood pressure upon hospital admission | 94 | 130 | 131.6 | 200 |
| 11 | Cholesterol | Serum cholesterol | 126 | 241 | 246.5 | 564 |
| 12 | Maximum Heart Rate | Maximum heart rate achieved. | 71 | 153 | 149.7 | 202 |
| 13 | Ole Peak | ST depression induced by exercise relative to rest. | 0 | 0.80 | 1.04 | 6.20 |

# Theory/Model

The authors take a data mining approach, implying that they do not make assumptions regarding the mode of influence of the various explanatory variables on the dependent variable. Instead, they let the data speak for itself. The aim of the paper is to determine which of the three presented techniques provides the best estimations when applied to new data. The three approaches used are Logistic regression, Artificial neural networks and Classification and Regression Trees. For this purpose, the authors of the study draw a random sample of 60% of the data set to train the models and then test their predictive power on the remaining 40%. By knowing the true outcomes of the test data set, they can then evaluate the accuracy of the prediction of the different models.

Improvements to the Study

We have a number of criticisms of the quality of the study and suggestions for improvement to increase the validity of the results.

First of all, we would like to criticize the form of the paper. The readability is partly limited, for instance due to missing spaces in the text and spelling mistakes.

More important, however, are technical comments, especially that Boonjing and Khemphila do not perform cross-validation. Our main contribution to improve the validity of the study is to utilize cross validation for the different methods (Logistic regression and neural networks) or to employ more sophisticated methods based on the same idea (bagged trees or random forest for decision trees). Without cross validation, the overall performance of the estimation methods rely heavily on the random training sample drawn. As we show in our study, the values of the quality measures area under the curve (AUC), sensitivity, specificity, accuracy and the error rate vary strongly with different splitting into training and test data set. A reliable statement about which data mining technique gives the best scores can therefore only be made by averaging the performance over many variations. Furthermore, we will use some more sophisticated methods, such as grid search to find the optimal parameters for our added models, to improve the robustness of the results.

More specifically, we will:

1. Replicate their estimations using Logistic Regression, Neural Networks & Decision Trees
2. Expand their methods by introducing Random Forests & Bagged Decision Trees
3. Check the validity of the results of 1. and 2. by Cross Validation using all techniques
4. Employ a Gradient Boosting for Decision Trees via xgboost & introducing a hyper parameter tuning algorithm for xgboost and Artificial Neural Network

Replication

To replicate the results of Boonjing and Khemphila we will pursue the same approach as them. We split the data set into a 60% training dataset and the remaining 40% we use as pseudo out of sample test data. Since it is a random split into test and training data, we cannot assume that we will obtain the same results as Boonjing and Khemphila. The decisive factor is whether we get a comparable ranking in the goodness of fit of the three different techniques. The authors of the study do not specify at which cutoff point they assign an estimated individual as result 1 (=has heart disease) or 0 (=has no heart disease). Since the aim of the study is to find the highest possible sensitivity, specificity and accuracy of the three classification techniques, we optimize the cutoff once for achieving maximum accuracy and once for finding the maximal point of a trade-off between sensitivity and specificity. We are aware of the fact that by doing so we are over-calibrating our models to some extent, but we consider it legitimate in this context, as we compare how well the models in the best case scenario perform. We calculate the error rate at the optimal cutoff for accuracy as at this point the error should be minimized. As an additional measure of performance, we will use the Area under the Curve value, which is threshold independent.

Comparison of performance in the training data

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Performance measure | Study results | Our results |
| Decision trees | Accuracy | 91.2 | 85.6 |
| Sensitivity | 90.1 | 87.4 |
| Specificity | 92.8 | 83.1 |
| Error rate | 0.082 | 0.183 |
| Area under the Curve |  | 0.895 |
| Neural networks | Accuracy | 92.3 | 90.6 |
| Sensitivity | 91.7 | 91.3 |
| Specificity | 92.9 | 89.6 |
| Error rate | 0.077 | 0.100 |
| Area under the Curve |  | 0.943 |
| Logistic regression | Accuracy | 91.7 | 88.9 |
| Sensitivity | 90.8 | 85.4 |
| Specificity | 92.8 | 93.5 |
| Error rate | 0.082 | 0.117 |
| Area under the Curve |  | 0.949 |

Comparison of performance in the test data

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Performance measure | Study results | Our results |
| Decision trees | Accuracy | 79.3 | 85.1 |
| Sensitivity | 81.7 | 86.9 |
| Specificity | 76 | 83.3 |
| Error rate | 0.21 | 0.174 |
| Area under the Curve |  | 0.8816 |
| Neural networks | Accuracy | 80.2 | 82.6 |
| Sensitivity | 81.1 | 78.7 |
| Specificity | 78.7 | 86.7 |
| Error rate | 0.198 | 0.182 |
| Area under the Curve |  | 0.851 |
| Logistic regression | Accuracy | 77.7 | 89.3 |
| Sensitivity | 81.2 | 90.2 |
| Specificity | 73.1 | 88.3 |
| Error rate | 0.22 | 0.116 |
| Area under the Curve |  | 0.927 |

Random Forests, Bagged Decision Trees for Decision Trees

We use three more sophisticated methods of decision trees - Bagged Decision Trees, Random Forests, Gradient boosting - to improve prediction performance.

Bagging is a version of bootstrap aggregation, whereby the random samples with replacement are drawn from the data. From each sample a decision tree is constructed, and the final model is then the average of all the individual decision trees. We use 500 bootstrap samples in our specification.

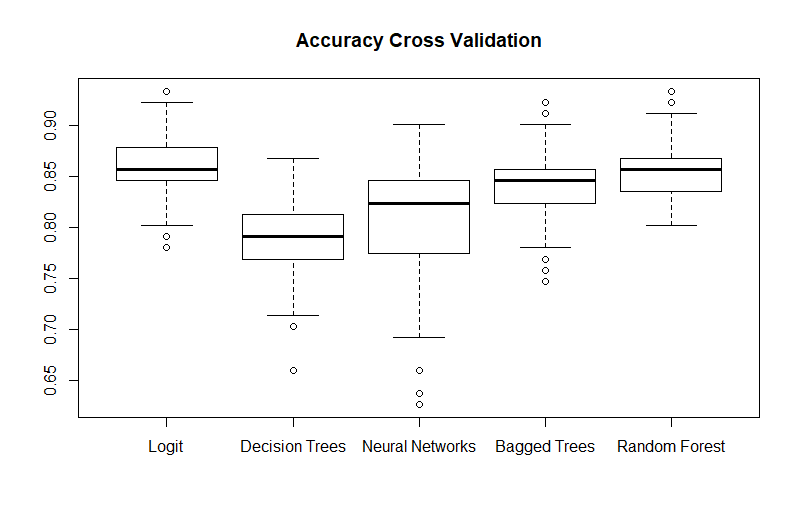
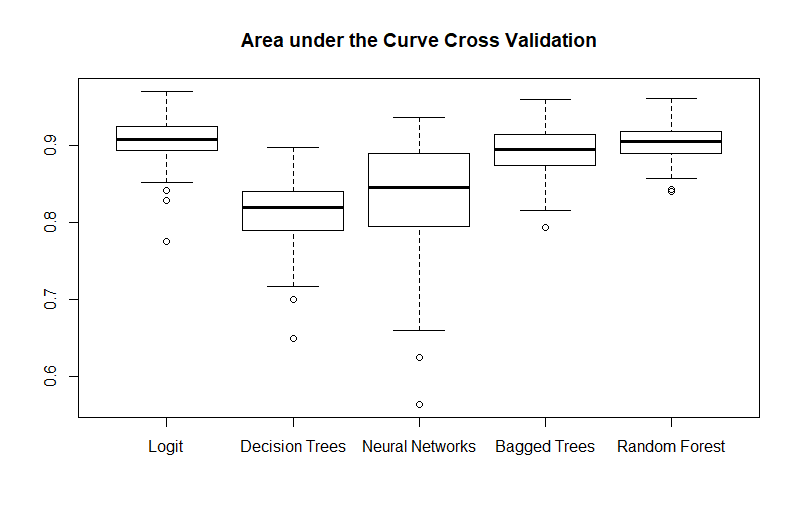
The Random Forest Approach operates similarly to Bagged Decision Tress but in addition to bootstrapping, it seeks to uncorrelate the individual trees of each random sample. This is achieved by randomizing the set of explanatory variables that each tree can use. For each node of a growing tree, a random subset of m variables is drawn from the total of variables p and the best variable and the best split from this subset m is selected. For our random forest we use the standard specification of the Random Forest package, i.e. the minimum node size is 1, no maximum node size is defined, the number of trees is 500, m is the square root of p and we have no importance measurement. The reason we use the standard specifications is that our data set does not contain any unusual values or many meaningless explanatory variables and in general, Random Forests perform well without further tuning of the parameters.

Comparison of performance on test data of new methods

|  |  |  |  |
| --- | --- | --- | --- |
| Performance measure | Decision Trees | Bagged Decision Trees | Random Forests |
| Accuracy | 85.1 | 87.6 | 88.4 |
| Sensitivity | 86.9 | 86.9 | 88.5 |
| Specificity | 83.3 | 88.3 | 88.3 |
| Error rate | 0.174 | 0.132 | 0.124 |
| Area under the Curve | 0.881 | 0.917 | 0.926 |

Cross Validation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | Performance measure |  | Our results | Study results |
| Decision trees | Accuracy | Min | 65.9 | 79.3 |
| Mean | 79.2 |
| Max | 86.8 |
| Sensitivity | Min | 64.3 | 81.7 |
| Mean | 81.1 |
| Max | 94 |
| Specificity | Min | 48.8 | 76 |
| Mean | 75.7 |
| Max | 89.2 |
| Error rate | Min | 0.143 | 0.21 |
| Mean | 0.257 |
| Max | 0.517 |
| Area under the Curve | Min | 0.649 |  |
| Mean | 0.815 |
| Max | 0.897 |
| Neural networks | Accuracy | Min | 62.6 | 80.2 |
| Mean | 80.5 |
| Max | 90.1 |
| Sensitivity | Min | 38 | 81.1 |
| Mean | 79.3 |
| Max | 93.2 |
| Specificity | Min | 35.7 | 78.7 |
| Mean | 80.7 |
| Max | 94.9 |
| Error rate | Min | 0.11 | 0.198 |
| Mean | 0.22 |
| Max | 0.56 |
| Area under the Curve | Min | 0.563 |  |
| Mean | 0.829 |
| Max | 0.937 |
| Logistic regression | Accuracy | Min | 78 | 77.7 |
| Mean | 86 |
| Max | 93.4 |
| Sensitivity | Min | 70.8 | 81.2 |
| Mean | 86.7 |
| Max | 98 |
| Specificity | Min | 74.4 | 73.1 |
| Mean | 84.1 |
| Max | 95.1 |
| Error rate | Min | 0.077 | 0.22 |
| Mean | 0.151 |
| Max | 0.231 |
| Area under the Curve | Min | 0.775 |  |
| Mean | 0.908 |
| Max | 0.971 |
| Bagged Decision Trees | Accuracy | Min | 74.7 |  |
| Mean | 84.2 |
| Max | 92.3 |
| Sensitivity | Min | 68.8 |  |
| Mean | 84.7 |
| Max | 98.1 |
| Specificity | Min | 69.6 |  |
| Mean | 82.2 |
| Max | 93.6 |
| Error rate | Min | 0.088 |  |
| Mean | 0.169 |
| Max | 0.264 |
| Area under the Curve | Min | 0.7941 |  |
| Mean | 0.895 |
| Max | 0.96 |
| Random Forest | Accuracy | Min | 80.2 |  |
| Mean | 85.4 |
| Max | 93.4 |
| Sensitivity | Min | 76.4 |  |
| Mean | 86.1 |
| Max | 96 |
| Specificity | Min | 71.1 |  |
| Mean | 83.4 |
| Max | 95.6 |
| Error rate | Min | 0.077 |  |
| Mean | 0.157 |
| Max | 0.209 |
| Area under the Curve | Min | 0.84 |  |
| Mean | 0.905 |
| Max | 0.962 |



Hyperparameter Tuning

Gradient boosting algorithm with application to decision trees produces an ensemble of weak prediction models and converts it into a strong learner. In practice, each newly produced tree is fit on a modified sample from the original data set. Each observation is then given a specific weight based on the model’s ability to classify it. Observations that are difficult to classify are given higher weights. On the adjusted data, a second tree is grown and prediction error is calculated on the ensemble of the two trees. A third tree is grown on the newly created residuals and the process repeats. The goal is to improve step by step the prediction ability of the final model. Therefore, the final model is a weighted average of the previously grown trees.

To further improve the predictive ability of the employed models, we have implemented a hyper parameter tuning algorithm for both xgboost and the artificial neural network. This way, we are able to optimize the parameters of the model while still maintaining the cross-validating function. For xgboost, we focus on the maximum depth of the grown trees, weights of the observations, subsampling and gamma (minimum loss reduction) and eta (adjusts learning rate to prevent overfitting) parameters. Because of the sheer number of parameters, we do not perform a grid search but rather perform 100 random combinations of the parameters to identify the best combination. For the neural network, we select the node size and decay (regularization parameter to avoid overfitting) parameters.

|  |  |  |
| --- | --- | --- |
| Method | Performance measure | Our results |
| Bagged Trees | Accuracy |  |
| Sensitivity |  |
| Specificity |  |
| Error rate |  |
| Area under the Curve |  |
| Random Forests | Accuracy |  |
| Sensitivity |  |
| Specificity |  |
| Error rate |  |
| Area under the Curve |  |
| xgboost (tuned) | Accuracy |  |
| Sensitivity |  |
| Specificity |  |
| Error rate |  |
| Area under the Curve |  |
| Neural Network (tuned) | Accuracy |  |
| Sensitivity |  |
| Specificity |  |
| Error rate |  |
| Area under the Curve |  |

Conclusion

In conclusion, the paper presented by Boonjing and Khemphila lacks basic scientific qualities. Our project therefore focuses on improvements to this paper to demonstrate proper techniques to be correctly employed. At first, we show that the results published by the paper are heavily dependent on the random sample generated from the original data. The resulting accuracy can vary anything between X and Y for logistic regression, X and Y for decision trees and X and Y for neural networks. We can clearly see that while the paper claims the superiority of the neural network approach, it seems that this might not be the case in reality.

To overcome this issue, we employ a cross-validating solution to randomly generate 100 samples from the dataset and train the models on these samples. In the end, we report the average accuracies (along other measures) to correctly show the actual predictive abilities. To further improve the results, we enhance the decision trees by utilizing bagged decision trees, random forests and gradient tree boosting. Furthermore, for our neural network and the gradient boosting algorithm (xgboost), we perform a parameter tuning. The resulting accuracies are X, Y, Z, W for bagged trees, random forests, xgboost (tuned) and neural network (tuned), respectively.

It can be clearly shown that the improved techniques dominate the original approach employed by the paper. The results are sample-robust and the run with close-to-optimal parameters. On average, we managed to improve the predictive ability (measured by accuracy) by X%. One might, therefore, believe that a large share of the information from the dataset is extracted by the models and further improvements would be only marginal.

# References

A. Khemphila and V. Boonjing, "Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients," 2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM), Krackow, 2010, pp. 193-198, doi: 10.1109/CISIM.2010.5643666