|  |
| --- |
| 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  24.5.2020 |

Table of Contents

[Overview 2](#_Toc41079412)

[Data 2](#_Toc41079413)

[Theory/Model 3](#_Toc41079414)

[Improvements to the Study 3](#_Toc41079415)

[Replication 4](#_Toc41079416)

[Random Forests, Bagged Decision Trees for Decision Trees 5](#_Toc41079417)

[Cross Validation 6](#_Toc41079418)

[Hyperparameter Tuning 9](#_Toc41079419)

[Conclusion 10](#_Toc41079420)

[Table 1: Description of the eight factor variables 2](#_Toc41079421)

[Table 2: Description of the five continuous variables: 3](#_Toc41079422)

[Table 3: Comparison of performance in the training data 4](#_Toc41079423)

[Table 4: Comparison of performance in the test data 4](#_Toc41079424)

[Table 5: Comparison of performance on test data of new methods 6](#_Toc41079425)

[Table 6: Comparison of average performance on test data in cross validation 6](#_Toc41079426)

[Table 7: Parameter Tuning Test data performance 9](#_Toc41079427)

[Figure 2: Accuracy Cross Validation 8](file:///C:\Users\paulm\Desktop\Uni%20stuff\MA%20-%204.%20Sem\Datascience%20in%20R%20II\Projekt\R_Project\Paper.docx#_Toc41079428)

[Figure 1: Area under the curve Cross Validation 8](file:///C:\Users\paulm\Desktop\Uni%20stuff\MA%20-%204.%20Sem\Datascience%20in%20R%20II\Projekt\R_Project\Paper.docx#_Toc41079429)

# 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.

To improve the predictive ability of the model, we drop any observations with Thal = 0. This indicates missing/corrupted data. Furthermore, we perform merging of ca (colored vessels) variable equal to 4 with 3. This is since level 4 of said variable is extremely rare in the dataset. If the variable is kept in the dataset in its original form, a significant noise is introduced to the modelling prospects. Same argumentation is then used for merging of two levels of variable restecg (resting ECG), 2 and 1. In both cases, merging is accepted on the grounds of intuitiveness in the context of the variables.

Table 1: 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%) |

Table 2: Description of the five 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 split of the data into training and test sample. 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.

Table 3: 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 |  | 89.5 |
| **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 |  | 94.3 |
| **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 |  | 94.9 |

Table 4: 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 |  | 88.2 |
| **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 |  | 85.1 |
| **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 |  | 92.7 |

There are several noteworthy insights from our estimations. First, the performance of the three models - Logistic Regression, Decision Trees, and Neural Networks - applied to our randomly generated training data set is worse than for Boonjing and Khemphila. Conversely, our estimated models provide substantially superior scores for the test data. In fact, our estimated performance for Logistic Regression and Decision Trees is almost equally good for the training and test data, and in rare cases (like accuracy for Logistic Regression) even better for the test data. This is quite unusual and likely due to a split in training and test data that is very advantageous for training the models (another possible explanation is that our models are less prone to overfitting than Boonjing and Khemphila specifications).

Furthermore, we obtain a different ranking of the three methods in terms of quality compared to Boonjing and Khemphila. According to their results, Neural Networks was the method that provided the best estimates for the test data, followed by Decision Trees and finally Logistic Regression. For us, this order is reversed, Logistic Regression yields the best performance, followed by Decision Trees and then Neural Networks. Another noteworthy aspect is that our Neural Networks had the highest values of the performance measures in the training data set, but the lowest in the test data set. Thus, Neural Networks seems to be more prone to overfitting than the other techniques.

As discussed earlier, both Boonjing and Khemphila and our results are not robust as they are inferred from a single random split of the data into training and test data. However, our results indicate that the authors have worked in a careless manner, as we arrive at quite dissimilar results even though we only replicated their work steps.

# Random Forests, Bagged Decision Trees for Decision Trees

We introduce two more sophisticated methods of decision trees - Bagged Decision Trees and Random Forests - to obtain improved 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 calculated as the average of all the individual decision trees. We use quite a high number of 500 bootstrap samples in our specification. Including more bootstrap samples usually is connected to better prediction performance but comes with a cost of higher computational demand. We utilize 500 bootstrap samples as we make use of the same number for the Random Forests for which this is the default setting.

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.

Table 5: 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 | 88.2 | 91.7 | 92.6 |

The findings demonstrate that the introduction of Random Forest and Bagged Decision Trees was beneficial, at least if compared to Decision Trees. Random forests deliver in all of our performance measures better values than Bagged Decision Trees and Decision Trees, which was expected since Random Forests are an extension/improvement of (Bagged) Decision Trees. Bagged Decision Trees themselves are an improvement/extension of Decision Trees and accordingly perform here superior compared to the basic model.

# Cross Validation

In order to obtain robust estimates of the predictive power of the methods used by Boonjing and Khemphila and the additional techniques we have introduced, we perform cross-validation. More specifically, we sample 100 times different test and training data sets by randomly splitting our data. For each split, we then estimate the different methods based on the training data, and subsequently evaluate their predictive power on the respective test data. The average prediction performance over 100 iterations should be more indicative of the goodness of fit of the respective methods than a rather random value of a single split as performed by Boonjing and Khemphila. In addition, we examine the standard deviation of our performance measures to evaluate the reliability of the prediction performance of the models. Furthermore, we present two box plots that illustrate the in our opinion two most important performance measures - Accuracy and Area under the curve. For the sake of comprehensibility, we will refrain from listing all possible values of interest of the performance measures (such as minimum, maximum and quantiles) and the performance in the training data set here. Decisive in the end is the predictive power on the test data set and we believe that the values and graphs provided by us are sufficient for a meaningful interpretation.

Table 6: Comparison of average performance on test data in cross validation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | Performance measure | Study results (%) |  | Our results (%) |
| Decision trees | **Accuracy** | **79.3** | **Mean** | **78.7** |
| Std | 0.0381 |
| Sensitivity | 81.7 | Mean | 79.8 |
| Std | 0.0683 |
| Specificity | 76 | Mean | 75.9 |
| Std | 0.0696 |
| Error rate | 0.21 | Mean | 0.258 |
| Std | 0.0631 |
| Area under the Curve |  | Mean | 80.5 |
| Std | 0.0447 |
| Neural networks | **Accuracy** | **80.2** | **Mean** | **79.5** |
| Std | 0.0723 |
| Sensitivity | 81.1 | Mean | 78.6 |
| Std | 0.1125 |
| Specificity | 78.7 | Mean | 79.1 |
| Std | 0.1049 |
| Error rate | 0.198 | Mean | 0.222 |
| Std | 0.0861 |
| Area under the Curve |  | Mean | 82.1 |
| Std | 0.0882 |
| Logistic regression | **Accuracy** | **77.7** | **Mean** | **86.1** |
| Std | 0.0333 |
| Sensitivity | 81.2 | Mean | 87.2 |
| Std | 0.0476 |
| Specificity | 73.1 | Mean | 83.8 |
| Std | 0.0533 |
| Error rate | 0.22 | Mean | 0.15 |
| Std | 0.0333 |
| Area under the Curve |  | Mean | 90.7 |
| Std | 0.0316 |
| Bagged Decision Trees | **Accuracy** |  | **Mean** | **83.8** |
| Std | 0.0343 |
| Sensitivity |  | Mean | 84.8 |
| Std | 0.0538 |
| Specificity |  | Mean | 81.3 |
| Std | 0.0514 |
| Error rate |  | Mean | 0.173 |
| Std | 0.0343 |
| Area under the Curve |  | Mean | 89.2 |
| Std | 0.0302 |
| Random Forest | **Accuracy** |  | **Mean** | **85.1** |
| Std | 0.0309 |
| Sensitivity |  | Mean | 86.6 |
| Std | 0.0442 |
| Specificity |  | Mean | 82.1 |
| Std | 0.0529 |
| Error rate |  | Mean | 0.161 |
| Std | 0.0313 |
| Area under the Curve |  | Mean | 90.3 |
| Std | 0.0278 |

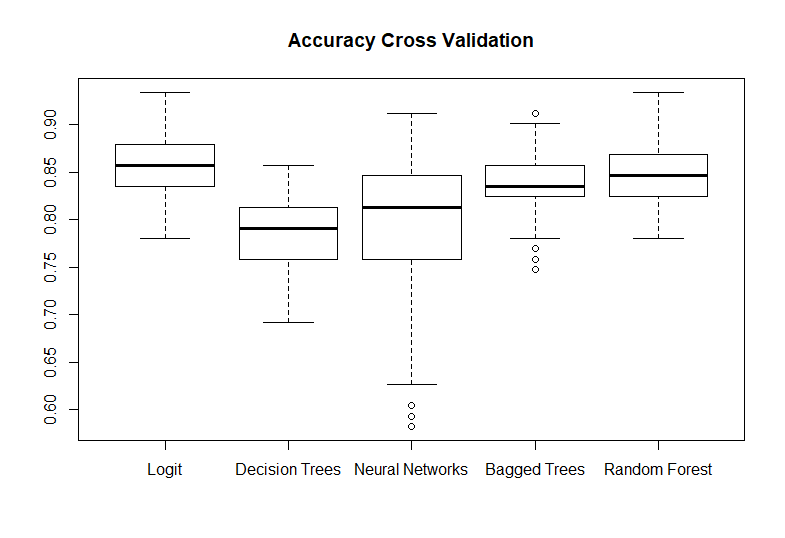
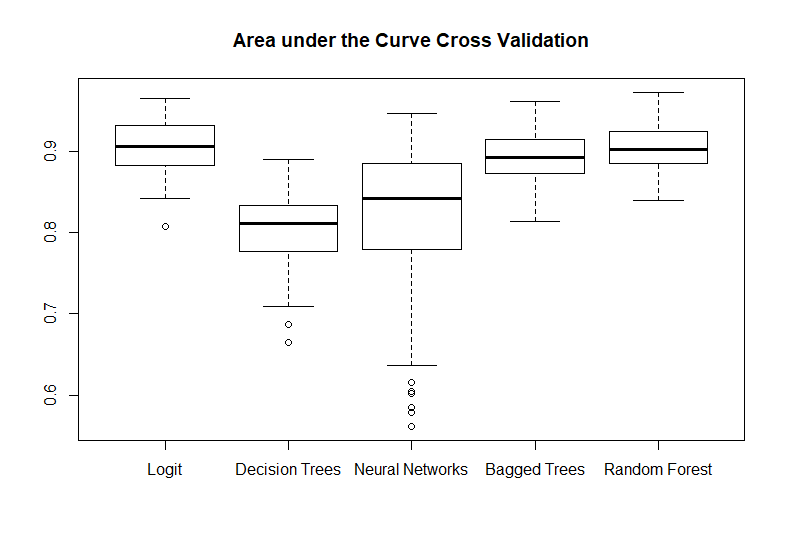
The mean values, standard deviation values and box plots provide a clear pattern: Random Forests and Logistic Regression provide the best estimates compared to the remaining methods, as they achieve the highest average values in all measurements investigated by us and at the same time have the lowest standard deviation across the measures. The discrepancies in quality between Random Forest and Logistic Regression itself are negligible. Bagged Decision Trees provide the next best estimates with slightly worse average performance values and little higher standard deviation. In the cross validation, it is proven what has already been observed in the previous chapter in the single run, that from the tree-based methods Random Forests provide the best results, followed by Bagged Decision Trees and then the basic model Decision Trees.

Figure : Accuracy Cross Validation

Figure : Area under the curve Cross Validation

Of the remaining two methods - Decision Trees and Neural Networks - Neural Networks have higher average performance values than Decision Trees, but also a significantly greater standard deviation. The large standard deviation of the performance measures of neural networks implies that the quality of the prediction depends heavily on the training data used. In our opinion, a method that delivers less variant but slightly worse average predictions is preferable to a method that is widely scattered in its quality, since the prediction quality can be better assessed for the first case. Accordingly, we consider neural networks to be the least suitable approach to perform estimations on this data set.

The cross-validation thus confirms once again our previous results that among the three methods used by Boonjing and Khemphila, Logistic Regression is the most successful, followed by Decision Trees and Neural Networks. In contrast, Boonjing and Khemphila considered neural networks to be the most suitable method. Interestingly, their values for the performance measures of decision trees and neural networks happen to be fairly close to our mean values, while their estimates of the quality of logistic regressions are considerably lower than our average scores. The divergence in their and our values and the inclusion of the variance in prediction performance over various training data sets to evaluate the quality of the method illustrates how important cross validation is for a robust estimation of the quality of the different models.

# Hyperparameter Tuning

We now introduce a more advanced technique to further improve our predictive prospects. 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.

Table 7: Parameter Tuning Test data performance

|  |  |  |
| --- | --- | --- |
| Method | Performance measure | Our results (%) |
| xgboost (tuned) | **Accuracy** | **83.5** |
| Sensitivity | 92.0 |
| Specificity | 73.2 |
| Error rate | 0.176 |
| Area under the Curve | 86.6 |
| Neural Network (tuned) | **Accuracy** | **86.9** |
| Sensitivity | 94.0 |
| Specificity | 78.0 |
| Error rate | 0.143 |
| Area under the Curve | 87.9 |

# Tuned xgboost algorithm is able to outperform basic neural networks and decision trees settings. However, this is to be expected. With an average accuracy of 83.5% it matches the predictive ability of bagged trees but falls short of both simple logistic regression and a random forest. Tuned neural network then outperforms basic neural network with accuracies of 86.9 and 84.3%, respectively. Therefore, by optimizing the set of parameters available in the nnet package, we have managed to improve the results of the model. We need to mention that hyper parameter tuning does not dramatically improve the performance of the basic models.

# 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 78% and 93% for logistic regression, 71% and 86% for decision trees and 53% and 94% for neural networks in the testing samples. 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 84%, 85.2%, 83.5%, 86.9% 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 7.5%. 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