**Subject:** Big data

**Student:** Oleksandr Romanchenko

**ID:** 83459

**Project:** “Classification of heart diseases using different types of machine learning models

with R programming language”

**Data:** this database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. In particular, the Cleveland database is the only one that has been used by ML researchers to this date. The "goal" field (target) refers to the presence of heart disease in the patient. It is binary value (0 or 1). Data was taken from [Kaggle](https://www.kaggle.com/ronitf/heart-disease-uci).

Attribute Information:   
> 1. **age**   
> 2. **sex** (1 – male; 0 - female)  
> 3. **cp** - chest pain type (4 values)   
> 4. **trestbps** - resting blood pressure (in Hg on admission to the hospital)   
> 5. **chol** - serum cholestoral in mg/dl   
> 6. **fbs** - fasting blood sugar > 120 mg/dl (1 = true; 0 = false)  
> 7. **restecg** - resting electrocardiographic results (values 0, 1, 2)  
> 8. **thalach** - maximum heart rate achieved   
> 9. **exang** - exercise induced angina (1 = yes; 0 = no)  
> 10. **oldpeak** - ST depression induced by exercise relative to rest   
> 11. **slope** - the slope of the peak exercise ST segment   
> 12. **ca** - number of major vessels (0-3) colored by flourosopy   
> 13. **thal** - 3 = normal; 6 = fixed defect; 7 = reversable defect

> 14. **target** – dependent variable (1 = heart disease; 0 = no heart disease)

A circuit board

Description automatically generated

**Doing some preparations of data and visualizations of main aspects:**

A screenshot of a cell phone

Description automatically generated

Number of observations in each category is almost equal, it is good for training the model.

A close up of text on a white background

Description automatically generated

Interesting that according to this density plot the most people with heart disease are between 50 and 55 years old whereas those without illness are significantly concentrated near the age of 60. Is it because heart disease often leads to a fast death (in 2-3 years)? – It is possible. Unfortunately for now it’s difficult to conclude more because it requires more domain knowledge in this field.

Checking correlation plot  
A screenshot of a cell phone

Description automatically generated

Considering correlation between variables to avoid collinearity and multicollinearity. Deleting variables: *slope*, *oldpeak*, *exang*, *thalach*.

Transforming some columns to factor

A close up of a gauge

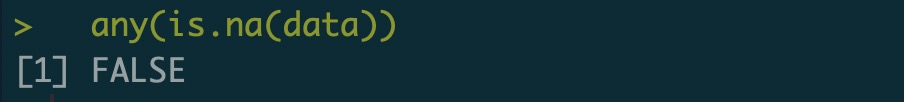
Description automatically generated

Dividing data on training and validation datasets

A picture containing text

Description automatically generated

And of course, checking missing values



Finally, it’s time to start modelling part! I’m going to predict target variable (whether the person is going to have a heart disease or not).

**Logistic regression**

A screenshot of a cell phone

Description automatically generated

Then making prediction and assessing accuracy of the model simply dividing number of true positive/negative over number of observations in dataset.

A close up of a logo

Description automatically generated

\*cm – confusion matrix

Not that bad! Let’s try the next model.

**Random forest**

**A screenshot of a cell phone

Description automatically generated**

A picture containing text

Description automatically generated

Random forest gives a bit less accuracy than logistic regression: 0.8681

Let’s see what other models can show.

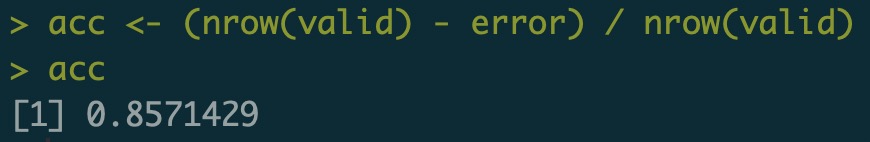
**Bagging**

Building model and making prediction

A picture containing device

Description automatically generated

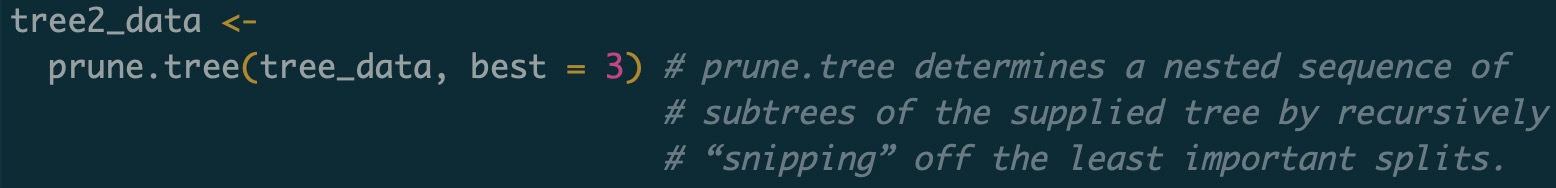
Calculating accuracy by subtracting number of incorrect predictions from number of rows of validation dataset and dividing it by number of rows.



Almost 86%. The worst result so far but not significantly smaller than 2 previous models.

**Decision tree**

****

****

Accuracy and error were calculated the same way as in bagging model.

A close up of a logo

Description automatically generated

78%. Even few percent’s worse than bagging.

**Rpart**

Just one more type of decision tree which allows to create some fancy plots for illustrating the logic of dividing variables at some thresholds inside the tree.

A close up of a logo

Description automatically generated

low cp (complexity parameter) for more sophisticated model

Change of error with decrease of complexity parameter and increase in size of the tree

A close up of a piece of paper

Description automatically generated

Plot of variables inside the tree

**A picture containing text, map

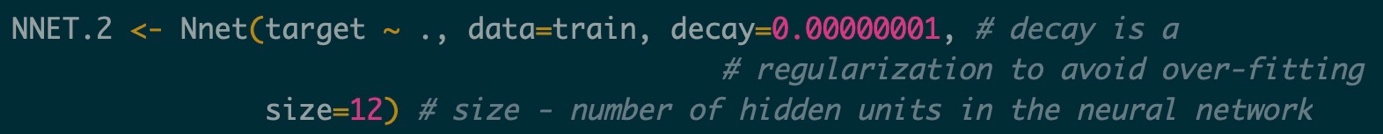
Description automatically generated**

Finding accuracy as in previous models

A close up of a logo

Description automatically generated

**Neural Network**



Setting low decay parameter and making 12 neurons in the hidden layer to make more precise model

Assessing accuracy as always

A close up of a logo

Description automatically generated

Comparison table of accuracy of all models

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| Logistic Regression | 0.8791 |
| Random Forest | 0.8681 |
| Bagging | 0.8571 |
| Decision Tree | 0.7802 |
| Rpart | 0.8791 |
| Neural Network | 0.9011 |

**Conclusion:** the worst model is Decision Tree with only 78% of correctly predicted heart diseases. A bit better are Bagging (0.8571) and Random Forest (0.8681). The 2nd place is hold by Logistic Regression and Rpart which both showed the accuracy of 0.8791. The best model is a Neural Network with 90% of correct predictions.

**Remarks**

* in seek of improvements through the feature engineering I’ve tried to

double the data by itself to give model more observations in order to make it more accurate without changing pattern



but results were poor. All models decreased their accuracy on 4-5% on the average, here is the example of glm model:

A close up of a logo

Description automatically generated

before doubling the data it was 0.87.