heart disease classification model

15.01.2022

1. Introduction

For my project I used four open source data bases that provided patient data on a total of 920 subjects. The data has been provided by the following institutions (and the responding author) and can be found **here**. Further description to origin and content of the data bases can be found **here**.

1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.

2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.

3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.

4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D.

The original data base included 76 attributes to each subject, but I did use the pre-processed version of the data sets. They included a column on the diagnosis of heart disease, and thirteen more columns with patient-related information:

- **age** - the patients age in years

- **sex** - the patients gender (0 for female and 1 for male)

- **cp** - the patients type of chest pain (1 for typical angina, 2 for atypical angina, 3 for non-anginal pain, 4 for asymptomatic)

- **trestbps**- the patients resting blood pressure in mm Hg on admission to the hospital

- **chol** - serum cholestoral in mg/dl

- **fbs** - if the patients fasting blood sugar was > 120 mg/dl (0 for false, 1 for true)

- **restecg** - the results from the resting electrocardiographic (0 for normal, 1 for having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV and 2 for showing probable or definite left ventricular hypertrophy by Estes' criteria)

- **thalach** - the maximum heart rate achieved by the patient in an exercise test in beats per minute ($thalach)

- **exang** - whether the patient experiences exercise induced angina (0 for no, 1 for yes)

- **oldpeak** - whether the electrogardiographic showed a ST depression induced by exercise relative to rest

- **slope** - the slope of the peak exercise ST segment (1 for upsloping, 2 for flat, 3 for downsloping)

- **ca** - the number of major vessels colored by flourosopy
- **thal** - the presence of a defect (3 for normal, 6 for fixed defect and 7 for reversable defect)

My aim in this project was to predict whether the subject had heart diease or not by using the other data I had on him or her.

2. Analysis

2.1 Download and join the data sets

First I did download the four datasets from the website of the University of California and assigned the corresponding column names. Then I used the function rbind to join the four datasets to one dataset that I named 'data'.

```r
cleveland <- read.csv(url("https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/proc
                     header=FALSE, col.names =
                       c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang",
                         "oldpeak","slope", "ca", "thal", "num"))

hungary <- read.csv(url("https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/proces
                      header=FALSE, col.names =
                        c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang",
                          "oldpeak","slope", "ca", "thal", "num"))

switzerland <- read.csv(url("https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/pr
                    header=FALSE, col.names =
                      c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang",
                        "oldpeak","slope", "ca", "thal", "num"))

va <- read.csv(url("https://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/processed.va
                        header=FALSE, col.names =
                          c("age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang
                            "oldpeak","slope", "ca", "thal", "num"))

data <- rbind(cleveland, hungary, switzerland, va)
```

2.2 Modify the outcome variable

The outcome variable $num included the angiographic results:

- the value 0 for the absence of heart disease
- the values 1-4 for the presence of heart disease whereby the numbers correspond to the disease degree For the algorithm I did mutate the value $heart_disease with the value 0 for patients without heart disease and with the value 1 for patients with heart disease. In total there are 411 patients without heart disease and 509 patients with heart disease.

2.3 Data cleaning

In total there were 1759 NAs in the dataset. The attributes with the most missing values were:

- the slope of the peak exercise ST segment $slope: 309 (33.6%) missing values
- the number of major vessels colored by flourosopy $ca: 920 (66.4%) missing values
- the presence of a defect $thal: 486 (52.8%) missing values
  I excluded the columns from the data set so that now there is one column (heart_disease) as outcome and ten columns with predictors (age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak)

2.4 Assign the proper data formats

I assigned the proper data formats: The patients age (age), the resting blood pressure (trestbps), the serum cholesterol (chol), the maximum heart rate during exercise ($thalach) and the old peak (oldpeak) are continous variables. The other, including the variable for the presence of heart disease are categorical variables.

2.5 Exploratory analysis

I tried to figure out whether there was an actual difference in the predictor variables between the group of patients with heart disease and those without. I used the Wilcoxon signed-rank test for continious variables and Fishers test for categorical values. I did also plot the variables but to save space here I won't display them. The variables were all significantly different between the two groups: Age (p=<0.001), gender (p=<0.001), the presence of chest pain (p= <0.001), resting blood pressure (p=0.002), serum cholesterol (p=<0.001), fasting blood sugar (p= <0.001), the results from the resting echocardiography (p=0.003), the maximum heart rate during exercise (p= <0.001), the presence of exercise induced angina (p= <0.001) and the presence of an old peak (p= <0.001).

```
sapply(data, function(x) if("numeric" %in% class(x) ) {
  wilcox.test (x ~ data$heart_disease)} else { fisher.test(data$heart_disease, x, simulate.p.value = TRU
```

2.8 Create train and test data set

I used the "createDataPartition" function from the caret package to part the data in two equal parts to create a train data set and test set.

```
test_index <- createDataPartition(data$heart_disease, times = 1, p = 0.5, list = FALSE)
test_set <- data[test_index, ]
train_set <- data[-test_index, ]
```

2.7 Baseline prediction

I used the sample-function from the base package to estimate accuracy if I just guessed the outcome. The accuracy would be **46.2 %**.

Method	Accuracy
Accuracy when random guessing	46.2%

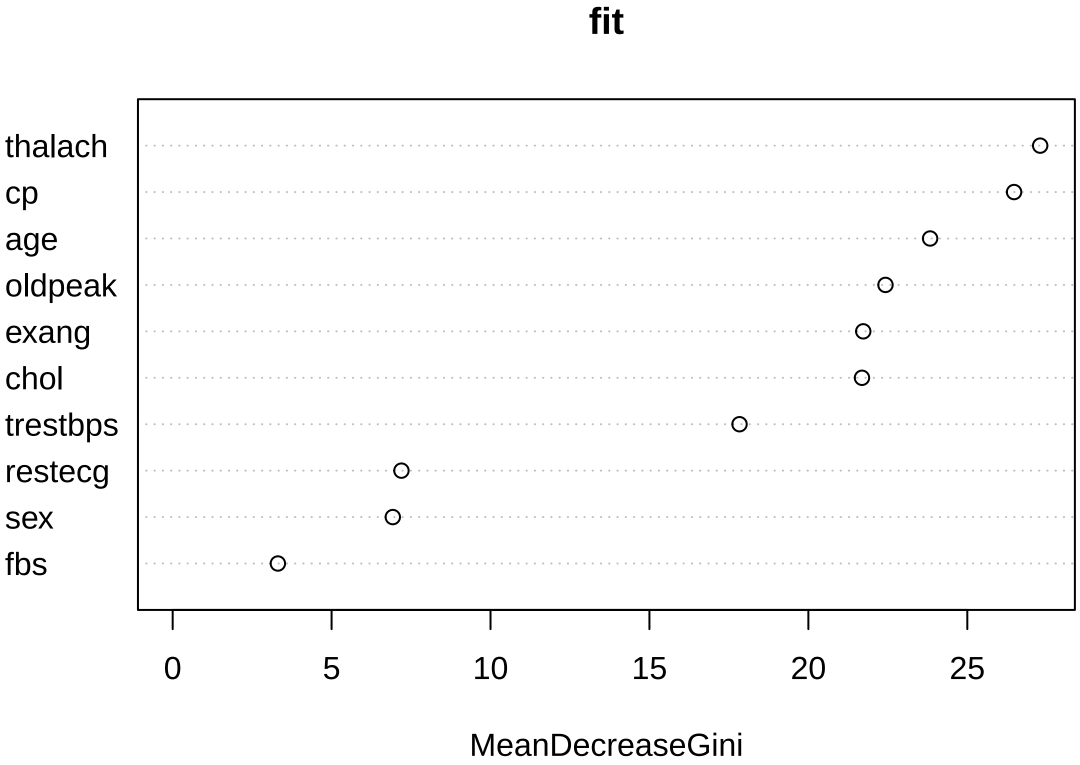
2.8 Build the random Forest

I used the randomForest-function from the randomForest-package to create a random Forest model. The binary outcome is the heart_disease variable, and all the other variables are used a predictors.

```
fit <- randomForest(heart_disease~. , data = train_set , na.action = na.omit)
```

2.9 Importance of the different predictors

The graph deplays the variable importance as measured by a Random Forest. The three most important variables (and their mean decrease in Gini coefficent) are:
- the presence of chest pain (29.709)
- the presence of an old peak (26.687)
- the maximum heart rate achieved during exercise (23.949)

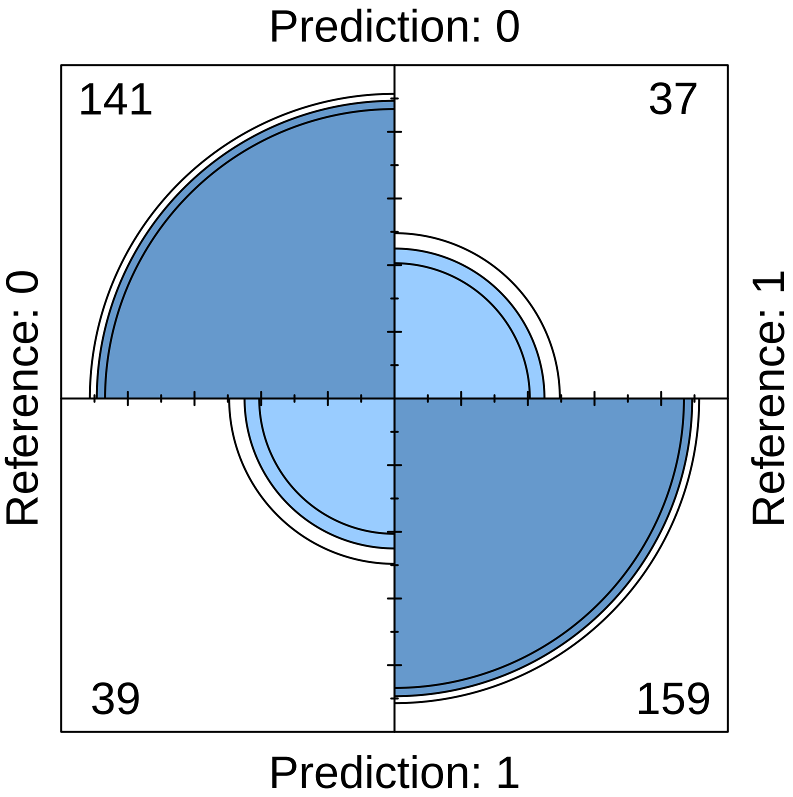


3. Results

The Random Forest model was able to achieve an accuracy of 77.9 % (95% CI 73.4%-82.1%).

Method	Accuracy
Accuracy when random guessing	46.2%
Random Forest model	77.9%

The models sensitivity is 76.3% and its specifity is 79.8%. The following graph was create by using the fourfoldplot-function from the randomForest package. It diplays the proportion of the subjects that were correctly and falsely classified by the random Forest in the test data set.



4. Conclusion

The Random Forest achieved an accuracy of **77.9%** which was way better than the accuracy when just randomly guessing (46.2%). The model surely could be improved by including more attributes from the original dataset, weighting the attributes differently, using a larger data set or applying a different model.