Part 1. Data collection\importing and processing

1. I am interested in the issue of longevity and a healthy lifestyle, so I decided to use machine learning tools to investigate the likelihood of dying from a heart attack
2. I found a relevant dataset on the Kaggle site, which I will use to create a model that predicts the likelihood of a heart attack. Source: https://www.kaggle.com/nareshbhat/health-care-data-set-on-heart-attack-possibility

3. **Information about the dataset I use:**

   14 attributes, 303 observations

   The "target" field refers to the presence of heart disease in the patient. It is integer valued 0 = no/less chance of heart attack and 1 = more chance of heart attack

   Attribute Information:

   1) age
   2) sex
   3) cp - chest pain type (4 values)
   4) trestbps - resting blood pressure
   5) chol - serum cholestoral in mg/dl
   6) fbs - fasting blood sugar > 120 mg/dl
   7) restecg - resting electrocardiographic results (values 0,1,2)
   8) thalach - maximum heart rate achieved
   9) exang - exercise induced angina
   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 - Thalassemia: 0 = normal; 1 = fixed defect; 2 = reversable defect
   14) target: 0= less chance of heart attack 1= more chance of heart attack

4. **Preparing the data**

Changed class of target variable to factor, every other variable has appropriate class

Data structure:

```
> str(data)
data.frame':   303 obs. of  15 variables:
$ age       : int  63 37 41 56 57 57 56 44 52 57 ...
$ sex       : int  1 1 0 1 0 1 0 1 1 1 ...
$ cp        : int  3 2 1 1 0 0 1 1 2 2 ...
$ trestbps  : num  145 130 130 120 120 140 140 120 160 150 ...
$ chol      : num  233 250 204 236 327 ...
$ fbs       : int  1 0 0 0 0 0 0 0 1 0 ...
$ restecg   : int  0 1 0 1 1 1 0 1 1 1 ...
$ thalach   : num  150 182 172 178 163 ...
$ exang     : int  0 0 0 0 1 0 0 0 0 0 ...
$ oldpeak   : num  2.3 3.4 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
$ slope     : int  0 0 2 2 2 1 1 2 2 2 ...
$ ca        : int  0 0 0 0 0 0 0 0 0 0 ...
$ thal      : int  1 2 2 2 2 1 2 3 3 2 ...
$ target    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
```

There are no NAs in the data:

```
   age      sex       cp trestbps     chol      fbs  restecg  thalach
     0        0        0        0        0        0        0        0
 exang  oldpeak    slope       ca     thal   target
     0        0        0        0        0        0
```

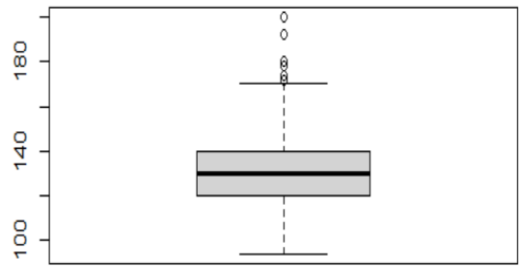
Check for imbalance in data (target variable):

```
 0   1
138 165
```

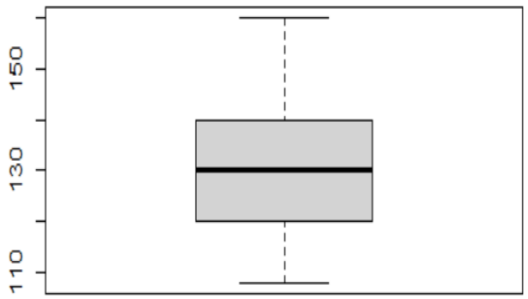
So, found that data in balanced

Checked for outliers using box plots and used Winsorize method to get rid of them:

Before:



After:



Part 2. Application of the ML techniques

My task is to classify a binary variable. In order to do it properly I decided to test different models and validate them applying varies methods of evaluating the model quality.

I am going to use:

- logistic regression, finding the optimal cutoff point to get a better quality
- random forest (both with and without cross validation)

I validate bult models using confusion matrixed and Aria Under Curve method

To avoid overfitting I randomly split the data into training set (70% for building a predictive model) and test set (30% for evaluating the model). Set seed for to make the result repeatable reproducible.

Logistic regression:

Model output:

```
Call:
glm(formula = target ~ ., family = binomial(link = "logit"),
    data = training_set)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7578  -0.2917   0.1533   0.5644   2.6138

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.293861   3.586045   0.640 0.522391
age          0.006564   0.029331   0.224 0.822907
sex         -1.484051   0.589986  -2.515 0.011890 *
cp           1.033245   0.263891   3.915 9.02e-05 ***
trestbps    -0.022559   0.015832  -1.425 0.154189
chol        -0.002585   0.005918  -0.437 0.662291
fbs         -0.454689   0.666443  -0.682 0.495072
restecg      0.239588   0.456064   0.525 0.599348
thalach      0.028315   0.014613   1.938 0.052653 .
exang       -0.857012   0.527883  -1.623 0.104485
oldpeak     -0.835458   0.288897  -2.892 0.003829 **
slope        0.544995   0.448186   1.216 0.223984
ca          -0.906069   0.237254  -3.819 0.000134 ***
thal        -0.986862   0.379433  -2.601 0.009298 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Significant variables are: sex, cp, oldpeak, ca, thal

We can see that the more heart problems and related symptoms a person has, the higher the likelihood of a heart attack.

Confusion matrixed when cutoff point is 0.3:

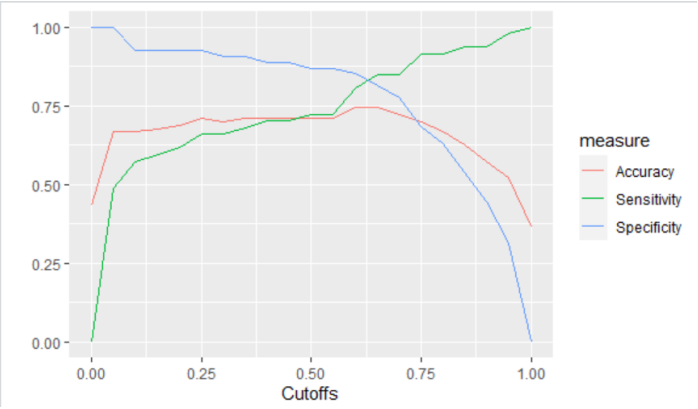
```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 31  5
         1 16 49

               Accuracy : 0.7921
```

Area under the curve: **0.7835**

Finding a better cutoff point:



According to graph, the best cutoff point is 0. 625

Confusion matrixed when cutoff point is 0. 625:

```
          Reference
Prediction  0  1
         0 40  8
         1  7 46

               Accuracy : 0.8515
```

Area under the curve: **0.8515**

So, the best result for logistic regression is 85% of accurate prediction.

Random forest without cross validation:

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 37  9
         1 10 45

               Accuracy : 0.8119
```

Area under the curve: **0.8103**

Random forest with cross validation:

(cross-validation allows to compensate for the effect of random distribution of samples)

```
          Reference
Prediction   0   1
         0 136   1
         1   2 164

               Accuracy : 0.9901
```

Area under the curve: **0.9897**

According to the metrics of accuracy and AUC the best model to predict a risk of heart attack is a random forest with cross validation, it gives almost 100% accurate predictions.