HEART FAILURE CLINICAL _PROJECT

ANWAR ABU ALHUSSEIN

Required packages

HEART FAILURE CLINICAL

PROJECT LINEAR REGRESSION

###* Loading Packages ###* ——————-

```
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(caret))
suppressPackageStartupMessages(library(ggcorrplot))
suppressPackageStartupMessages(library(Metrics))
suppressPackageStartupMessages(library(corrplot))
suppressPackageStartupMessages(library(bannerCommenter))
suppressPackageStartupMessages(library(psych))
suppressPackageStartupMessages(library(WVPlots))
suppressPackageStartupMessages(library(PerformanceAnalytics))
suppressPackageStartupMessages(library(car))
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(outliers))
suppressPackageStartupMessages(library(broom))
suppressPackageStartupMessages(library(equatiomatic))
```

##======================================
## Reading in the data == ##====================================

```
download.file("https://raw.githubusercontent.com/ABUALHUSSEIN/test/main/data/heart.csv",
              destfile = "heart.csv")
```

##source(destfile) ##ls() #========================

```
setwd("C:/Users/WAFA/Desktop")
heart <- read.csv("heart.csv",header=TRUE)
View(heart)
```

#txt <- "Explore the data set" #banner(txt, centre = TRUE, bandChar = "-") ##———————————————————————- ## Explore the data set – ##——————————————————————————- #txt <- "Data contents" #banner(txt, centre = TRUE, bandChar = "=") ##======================================
## Data contents == ##====================================

```
Hmisc::contents(heart)
```

```
##
```

```
## Data frame:heart 299 observations and 13 variables    Maximum # NAs:0
##
##
##                           Storage
## anaemia                   integer
## diabetes                  integer
## sex                       integer
## smoking                   integer
## high_blood_pressure       integer
## DEATH_EVENT               integer
## ejection_fraction         integer
## platelets                  double
## serum_creatinine           double
## serum_sodium              integer
## time                      integer
## age                        double
## creatinine_phosphokinase integer
```

```r
glimpse(heart)
```

```
## Rows: 299
## Columns: 13
## $ anaemia                  <int> 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, ~
## $ diabetes                 <int> 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, ~
## $ sex                      <int> 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, ~
## $ smoking                  <int> 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, ~
## $ high_blood_pressure      <int> 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, ~
## $ DEATH_EVENT              <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, ~
## $ ejection_fraction        <int> 20, 38, 20, 20, 20, 40, 15, 60, 65, 35, 38, 2~
## $ platelets                <dbl> 265000, 263358, 162000, 210000, 327000, 20400~
## $ serum_creatinine         <dbl> 1.90, 1.10, 1.30, 1.90, 2.70, 2.10, 1.20, 1.1~
## $ serum_sodium             <int> 130, 136, 129, 137, 116, 132, 137, 131, 138, ~
## $ time                     <int> 4, 6, 7, 7, 8, 8, 10, 10, 10, 10, 10, 10, 11,~
## $ age                      <dbl> 75, 55, 65, 50, 65, 90, 75, 60, 65, 80, 75, 6~
## $ creatinine_phosphokinase <int> 582, 7861, 146, 111, 160, 47, 246, 315, 157, ~
```

```r
head(heart, 10)
```

```
##    anaemia diabetes sex smoking high_blood_pressure DEATH_EVENT
## 1        0        0   1       0                   1           1
## 2        0        0   1       0                   0           1
## 3        0        0   1       1                   0           1
## 4        1        0   1       0                   0           1
## 5        1        1   0       0                   0           1
## 6        1        0   1       1                   1           1
## 7        1        0   1       0                   0           1
## 8        1        1   1       1                   0           1
## 9        0        0   0       0                   0           1
## 10       1        0   1       1                   1           1
##    ejection_fraction platelets serum_creatinine serum_sodium time age
## 1                 20    265000              1.9          130    4  75
## 2                 38    263358              1.1          136    6  55
## 3                 20    162000              1.3          129    7  65
## 4                 20    210000              1.9          137    7  50
## 5                 20    327000              2.7          116    8  65
## 6                 40    204000              2.1          132    8  90
## 7                 15    127000              1.2          137   10  75
## 8                 60    454000              1.1          131   10  60
## 9                 65    263358              1.5          138   10  65
## 10                35    388000              9.4          133   10  80
##    creatinine_phosphokinase
## 1                       582
## 2                      7861
## 3                       146
## 4                       111
## 5                       160
## 6                        47
## 7                       246
## 8                       315
## 9                       157
## 10                      123
```

```
tail(heart)
```

```
##     anaemia diabetes sex smoking high_blood_pressure DEATH_EVENT
## 294       1        1   1       1                   0           0
## 295       0        1   1       1                   1           0
## 296       0        0   0       0                   0           0
## 297       0        1   0       0                   0           0
## 298       0        0   1       1                   0           0
## 299       0        0   1       1                   0           0
##     ejection_fraction platelets serum_creatinine serum_sodium time age
## 294                35    179000              0.9          136  270  63
## 295                38    155000              1.1          143  270  62
## 296                38    270000              1.2          139  271  55
## 297                60    742000              0.8          138  278  45
## 298                38    140000              1.4          140  280  45
## 299                45    395000              1.6          136  285  50
##     creatinine_phosphokinase
## 294                      103
## 295                       61
## 296                     1820
## 297                     2060
## 298                     2413
## 299                      196
```

```
car::brief(heart)
```

```
## 299 x 13 data.frame (294 rows and 7 columns omitted)
##     anaemia diabetes sex smoking . . . age creatinine_phosphokinase
##         [i]      [i] [i]     [i]       [n]                      [i]
## 1         0        0   1       0        75                      582
## 2         0        0   1       0        55                     7861
## 3         0        0   1       1        65                      146
## . . .
## 298       0        0   1       1        45                     2413
## 299       0        0   1       1        50                      196
```

```
str(heart)
```

```
## 'data.frame':    299 obs. of  13 variables:
```

```
##  $ anaemia                 : int  0 0 0 1 1 1 1 1 0 1 ...
##  $ diabetes                : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ sex                     : int  1 1 1 1 0 1 1 1 0 1 ...
##  $ smoking                 : int  0 0 1 0 0 1 0 1 0 1 ...
##  $ high_blood_pressure     : int  1 0 0 0 0 1 0 0 0 1 ...
##  $ DEATH_EVENT             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ ejection_fraction       : int  20 38 20 20 20 40 15 60 65 35 ...
##  $ platelets               : num  265000 263358 162000 210000 327000 ...
##  $ serum_creatinine        : num  1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.4 ...
##  $ serum_sodium            : int  130 136 129 137 116 132 137 131 138 133 ...
##  $ time                    : int  4 6 7 7 8 8 10 10 10 10 ...
##  $ age                     : num  75 55 65 50 65 90 75 60 65 80 ...
##  $ creatinine_phosphokinase: int  582 7861 146 111 160 47 246 315 157 123 ...
```

We have 299 obs. of 13 variables.

##==
## Data Description == ##==

age: age of the patient (years)

anaemia: decrease of red blood cells or hemoglobin (boolean)

high blood pressure: if the patient has hypertension (boolean) creatinine phosphokinase (CPK): level of the CPK enzyme in the blood (mcg/L)

diabetes: if the patient has diabetes (boolean)

ejection fraction: percentage of blood leaving the heart at each contraction (percentage)

platelets: platelets in the blood (kiloplatelets/mL)

** sex: woman or man (binary) ## serum creatinine: level of serum creatinine in the blood (mg/dL) ## serum sodium: level of serum sodium in the blood (mEq/L) ## smoking: if the patient smokes or not (boolean) ## time: follow-up period (days) ## death event**: if the patient deceased during the follow-up period (boolean)

##==
## EDA: Exploratory Data Analysis == ##===

```r
heart_data <- heart[, 7:ncol(heart)]
```

Check the column names again,

```r
colnames(heart_data)
```

```
## [1] "ejection_fraction"        "platelets"
## [3] "serum_creatinine"         "serum_sodium"
## [5] "time"                     "age"
## [7] "creatinine_phosphokinase"
```

#txt <- “Get the summary statistics of the variables” #banner(txt, centre = TRUE, bandChar = "") ##** ## Get the summary statistics of the variables ##**

```
summary(heart_data)
```

```
##  ejection_fraction   platelets      serum_creatinine  serum_sodium
##  Min.   :14.00     Min.   : 25100   Min.   :0.500    Min.   :113.0
##  1st Qu.:30.00     1st Qu.:212500   1st Qu.:0.900    1st Qu.:134.0
##  Median :38.00     Median :262000   Median :1.100    Median :137.0
##  Mean   :38.08     Mean   :263358   Mean   :1.394    Mean   :136.6
##  3rd Qu.:45.00     3rd Qu.:303500   3rd Qu.:1.400    3rd Qu.:140.0
##  Max.   :80.00     Max.   :850000   Max.   :9.400    Max.   :148.0
##       time             age        creatinine_phosphokinase
##  Min.   :  4.0   Min.   :40.00   Min.   :  23.0
##  1st Qu.: 73.0   1st Qu.:51.00   1st Qu.: 116.5
##  Median :115.0   Median :60.00   Median : 250.0
##  Mean   :130.3   Mean   :60.83   Mean   : 581.8
##  3rd Qu.:203.0   3rd Qu.:70.00   3rd Qu.: 582.0
##  Max.   :285.0   Max.   :95.00   Max.   :7861.0
```

```
Hmisc::describe(heart_data)
```

```
## heart_data
##
##  7  Variables      299  Observations
## --------------------------------------------------------------------------------
## ejection_fraction
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0       17    0.986    38.08    13.09       20       25
##      .25      .50      .75      .90      .95
##       30       38       45       60       60
##
## lowest : 14 15 17 20 25, highest: 60 62 65 70 80
##
## Value         14    15    17    20    25    30    35    38    40    45    50
## Frequency      1     2     2    18    36    34    49    40    37    20    21
## Proportion 0.003 0.007 0.007 0.060 0.120 0.114 0.164 0.134 0.124 0.067 0.070
##
## Value         55    60    62    65    70    80
## Frequency      3    31     2     1     1     1
## Proportion 0.010 0.104 0.007 0.003 0.003 0.003
## --------------------------------------------------------------------------------
## platelets
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0      176    0.999   263358   100936   131800   153000
##      .25      .50      .75      .90      .95
##   212500   262000   303500   374600   422500
##
## lowest :  25100  47000  51000  62000  70000, highest: 533000 543000 621000 742000 850000
## --------------------------------------------------------------------------------
## serum_creatinine
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0       40    0.991    1.394    0.783      0.7      0.8
##      .25      .50      .75      .90      .95
##      0.9      1.1      1.4      2.1      3.0
##
## lowest : 0.50 0.60 0.70 0.75 0.80, highest: 5.80 6.10 6.80 9.00 9.40
## --------------------------------------------------------------------------------
## serum_sodium
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0       27    0.991    136.6    4.683    130.0    132.0
##      .25      .50      .75      .90      .95
##    134.0    137.0    140.0    141.2    144.0
##
## lowest : 113 116 121 124 125, highest: 143 144 145 146 148
## --------------------------------------------------------------------------------
## time
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0      148        1    130.3    89.31     12.9     26.8
##      .25      .50      .75      .90      .95
##     73.0    115.0    203.0    244.0    250.0
##
## lowest :   4   6   7   8  10, highest: 270 271 278 280 285
## --------------------------------------------------------------------------------
## age
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0       47    0.996    60.83    13.44     42.9     45.0
##      .25      .50      .75      .90      .95
##     51.0     60.0     70.0     75.4     82.0
##
## lowest : 40 41 42 43 44, highest: 86 87 90 94 95
## --------------------------------------------------------------------------------
## creatinine_phosphokinase
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      299        0      208    0.996    581.8    710.2     59.0     67.6
##      .25      .50      .75      .90      .95
##    116.5    250.0    582.0   1203.8   2263.0
##
## lowest :   23   30   47   52   53, highest: 4540 5209 5882 7702 7861
## --------------------------------------------------------------------------------
```

```
psych::describe(heart_data, skew = TRUE,
                IQR = TRUE)
```

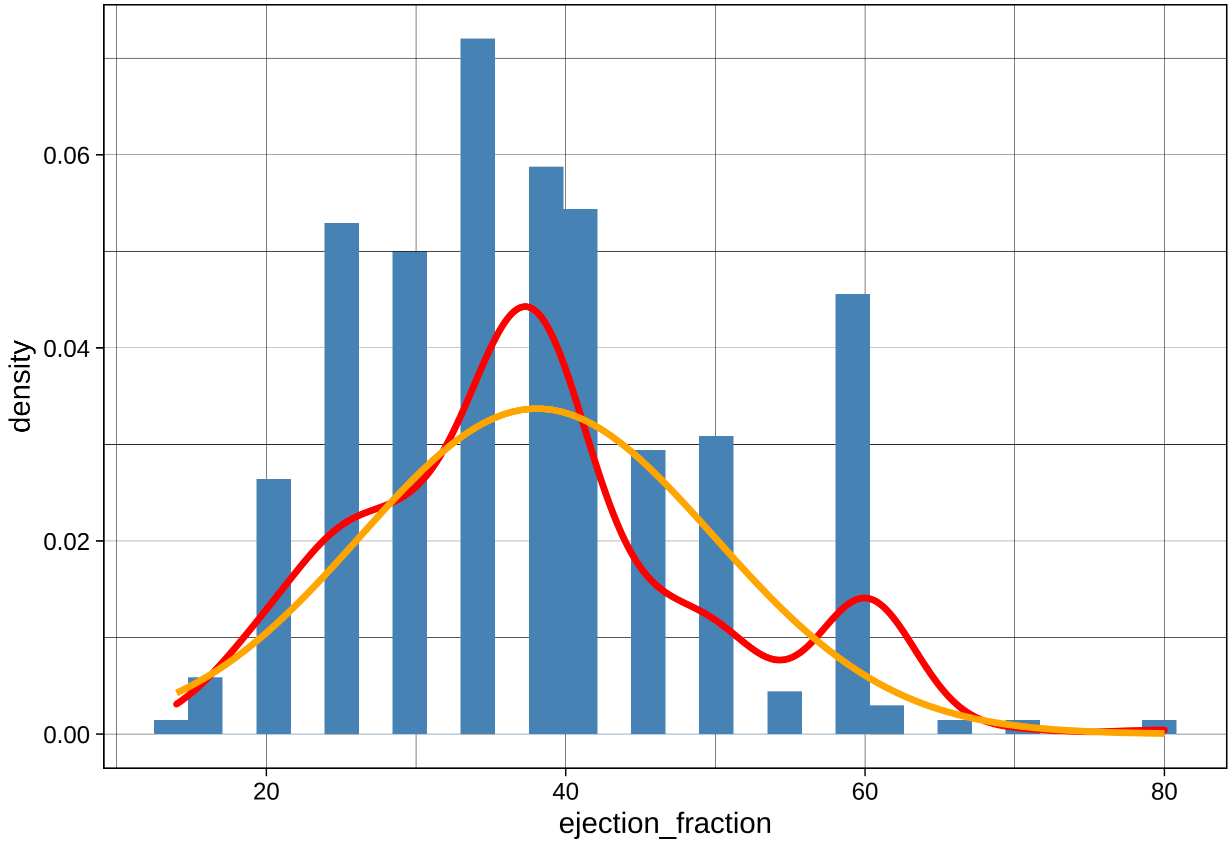
```
##                          vars   n      mean       sd   median   trimmed
## ejection_fraction           1 299     38.08    11.83     38.0     37.43
## platelets                   2 299 263358.03 97804.24 262000.0 256730.09
## serum_creatinine            3 299      1.39     1.03      1.1      1.19
## serum_sodium                4 299    136.63     4.41    137.0    136.82
## time                        5 299    130.26    77.61    115.0    129.28
## age                         6 299     60.83    11.89     60.0     60.22
## creatinine_phosphokinase    7 299    581.84   970.29    250.0    365.49
##                               mad     min      max    range  skew kurtosis
## ejection_fraction           11.86    14.0     80.0     66.0  0.55     0.00
## platelets                65234.40 25100.0 850000.0 824900.0  1.45     6.03
## serum_creatinine             0.30     0.5      9.4      8.9  4.41    25.19
## serum_sodium                 4.45   113.0    148.0     35.0 -1.04     3.98
## time                       105.26     4.0    285.0    281.0  0.13    -1.22
## age                         14.83    40.0     95.0     55.0  0.42    -0.22
## creatinine_phosphokinase   269.83    23.0   7861.0   7838.0  4.42    24.53
##                                se     IQR
## ejection_fraction             0.68    15.0
## platelets                  5656.17 91000.0
## serum_creatinine              0.06     0.5
## serum_sodium                  0.26     6.0
## time                          4.49   130.0
## age                           0.69    19.0
## creatinine_phosphokinase     56.11   465.5
```

#txt <- “The Dependent Variables in this study is ejection_fraction” #banner(txt, centre = TRUE, bandChar = “:”)

##:: ## The Dependent Variables in this study is ejection_fraction :: ##::

We are going to explore the distribution of this variable

```
ggplot(heart_data, aes(x = ejection_fraction)) +
  geom_histogram(aes(y = ..density..),
                 fill = "steelblue") +
  geom_density(color = "red", lwd = 1.2) +
  stat_function(fun = dnorm, args = list(mean = mean(heart_data$ejection_fraction),
                                         sd = sd(heart_data$ejection_fraction)),
                color = "orange", lwd = 1.2) +
  theme_linedraw()
```

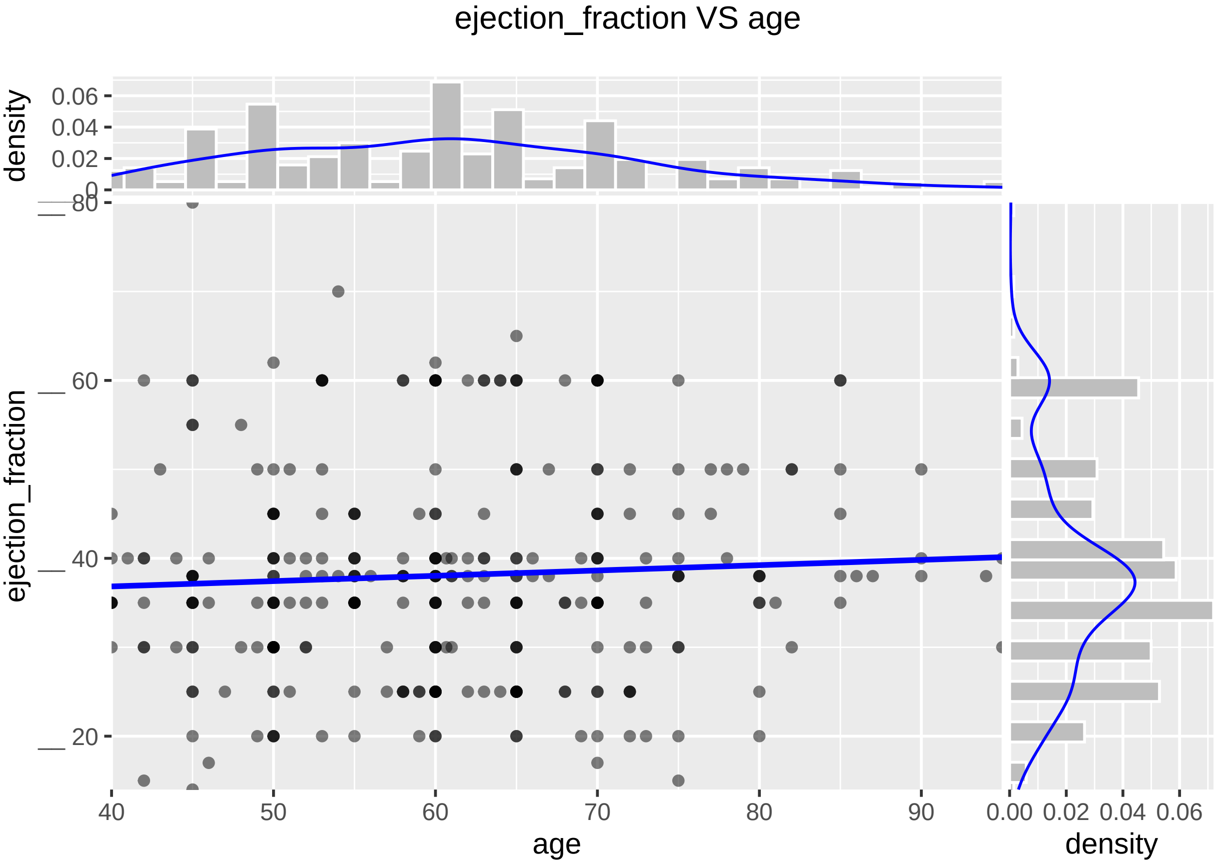


Joint Graphs

```
library(WVPlots)
```

#ejection_fraction VS age

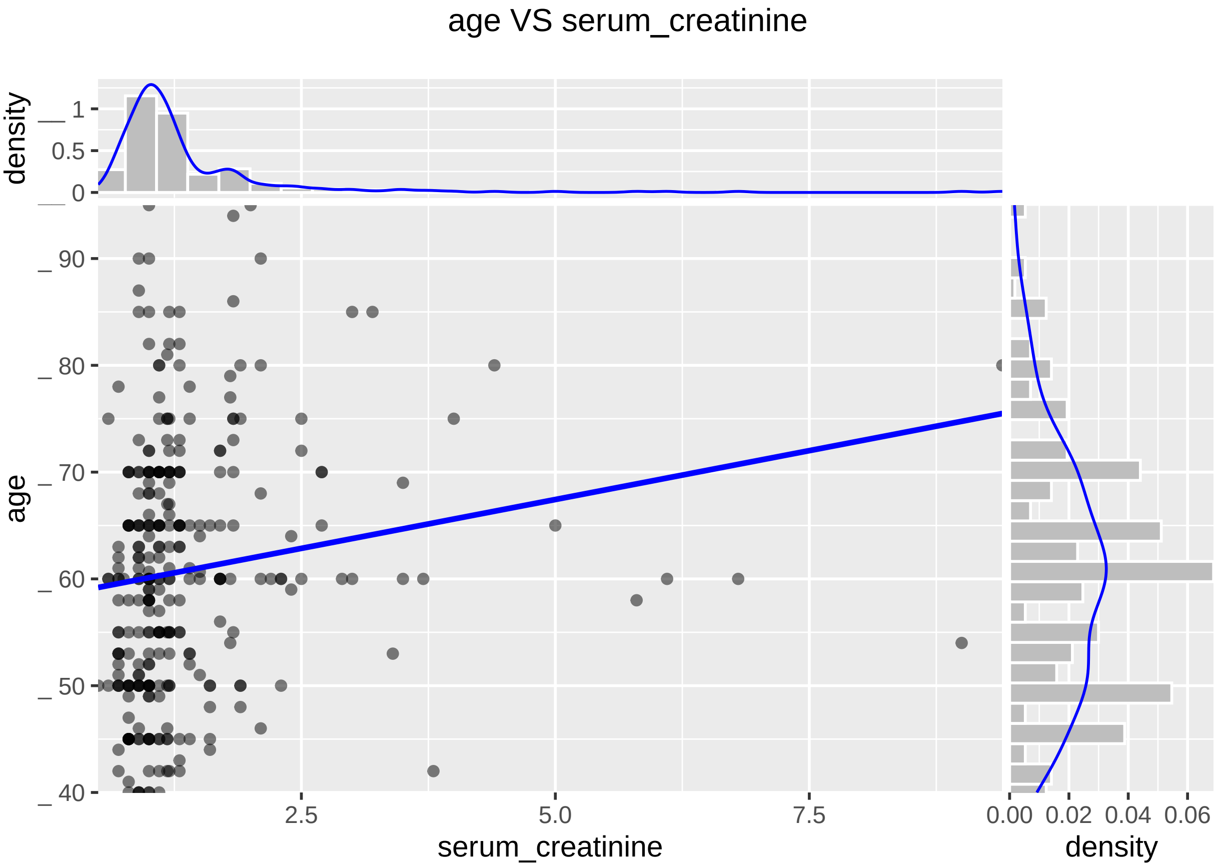
```
ScatterHist(heart_data, title = "ejection_fraction VS age",
            xvar = "age", yvar = "ejection_fraction",
            smoothmethod = "lm")
```



There is a very weak linear relationshipbetween the variables(ejection_fraction VS age)

#age VS serum_creatinine

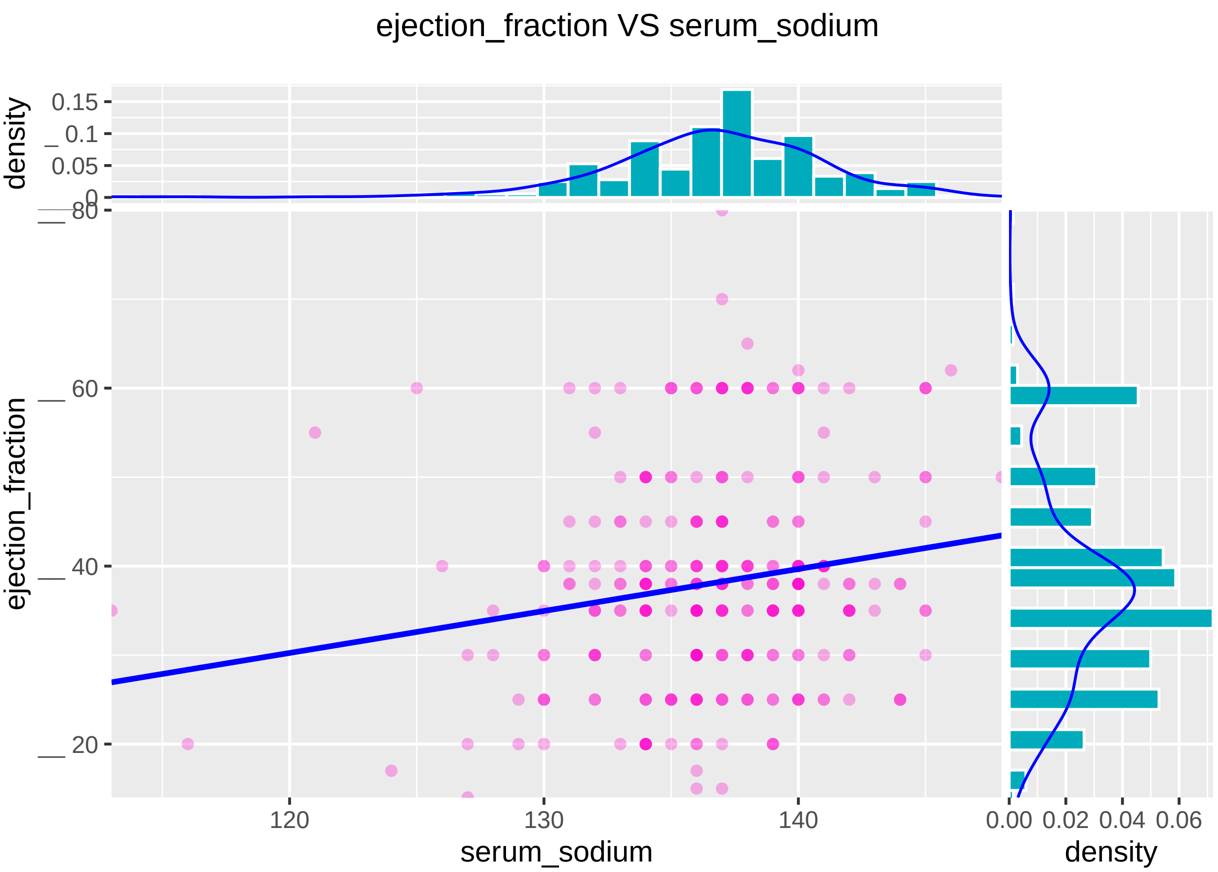
```
ScatterHist(heart_data, title = "age VS serum_creatinine",
            xvar = "serum_creatinine", yvar = "age",
            smoothmethod = "lm")
```



There is a positive linear relationship between the variables(age & serum_creatinine)

ejection_fraction VS serum_sodium

```r
ScatterHist(heart_data, title = "ejection_fraction VS serum_sodium",
            xvar = "serum_sodium", yvar = "ejection_fraction",
            smoothmethod = "lm", hist_color = "#00ACBB",
            point_alpha = 0.3,
            point_color = "#FF00CC")
```



There is a positive linear relationship between the variables (ejection__fraction & serum__sodium)

##*** ## Study the correlation ##***

```
psych::lowerCor(x = heart_data)
```

```
##                          ejct_ pltlt srm_c srm_s time  age   crtn_
## ejection_fraction         1.00
## platelets                 0.07  1.00
## serum_creatinine         -0.01 -0.04  1.00
## serum_sodium              0.18  0.06 -0.19  1.00
## time                      0.04  0.01 -0.15  0.09  1.00
## age                       0.06 -0.05  0.16 -0.05 -0.22  1.00
## creatinine_phosphokinase -0.04  0.02 -0.02  0.06 -0.01 -0.08  1.00
```

```
psych::corr.test(heart_data)$p
```

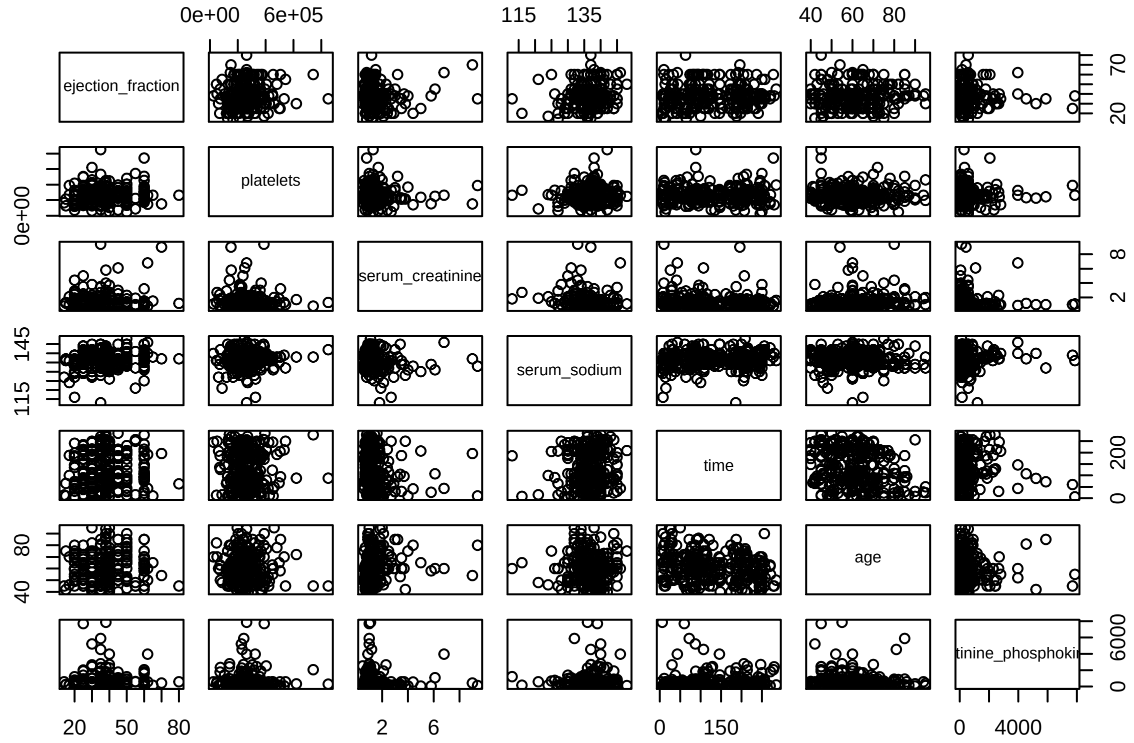
```
##                          ejection_fraction platelets serum_creatinine
## ejection_fraction              0.000000000 1.0000000      1.000000000
## platelets                      0.213329933 0.0000000      1.000000000
## serum_creatinine               0.845686017 0.4778909      0.000000000
## serum_sodium                   0.002267684 0.2842728      0.001017081
## time                           0.472229204 0.8563315      0.009720948
## age                            0.300304089 0.3669939      0.005803433
## creatinine_phosphokinase       0.447623299 0.6735340      0.777514366
##                           serum_sodium         time         age
## ejection_fraction           0.04308600 1.000000e+00 1.000000000
## platelets                   1.00000000 1.000000e+00 1.000000000
## serum_creatinine            0.02034161 1.652561e-01 0.104461793
## serum_sodium                0.00000000 1.000000e+00 1.000000000
## time                        0.13053379 0.000000e+00 0.001953672
## age                         0.42841134 9.303199e-05 0.000000000
## creatinine_phosphokinase    0.30474247 8.721501e-01 0.159381379
##                           creatinine_phosphokinase
## ejection_fraction                                1
## platelets                                        1
## serum_creatinine                                 1
## serum_sodium                                     1
## time                                             1
## age                                              1
## creatinine_phosphokinase                         0
```

Plotting the ScatterPlotMatrix

First look at the help, and the arguments

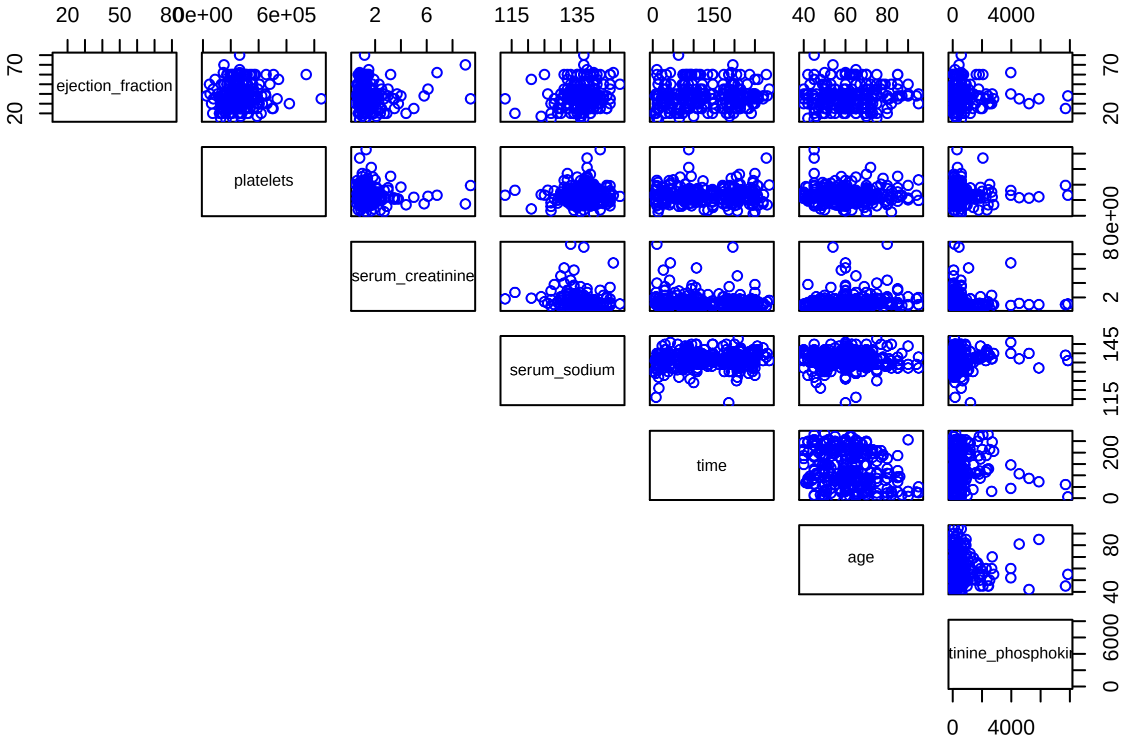
I am going to tweak the knobs a little

```
pairs(heart_data)
```



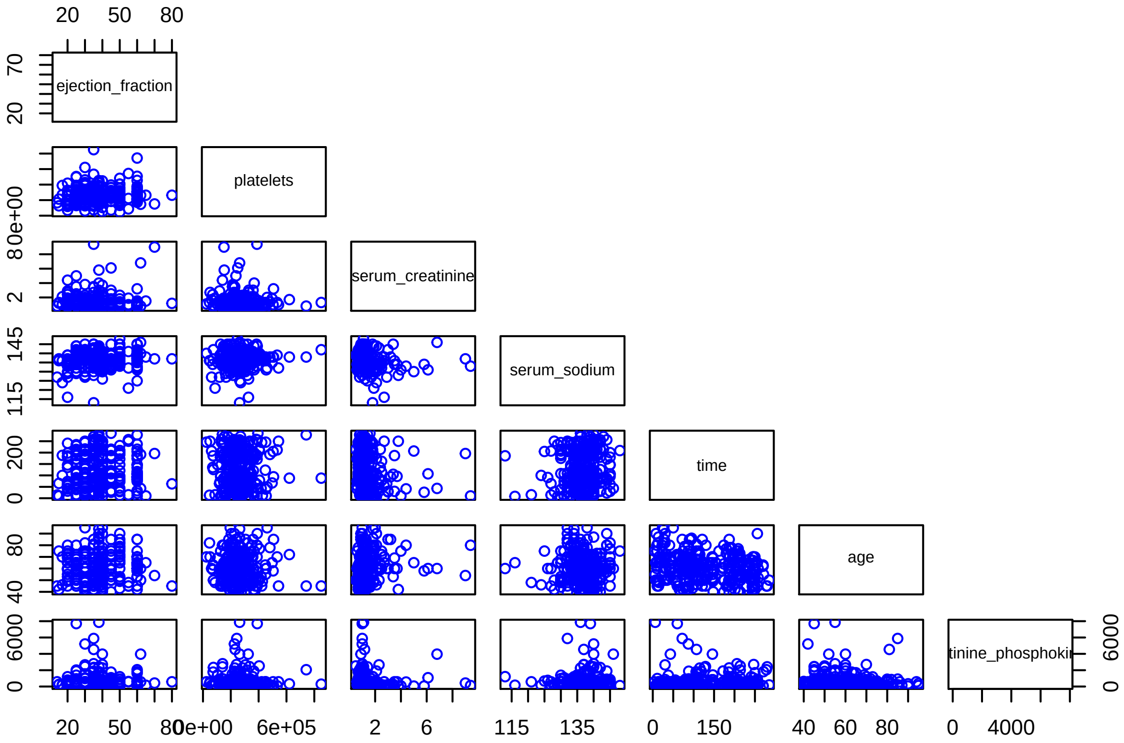
#change the color and get halp matrix

```
pairs(heart_data, lower.panel = NULL, col= "blue")
```



Or if you want only the lower part matrix

```
pairs(heart_data, upper.panel = NULL, col= "blue")
```

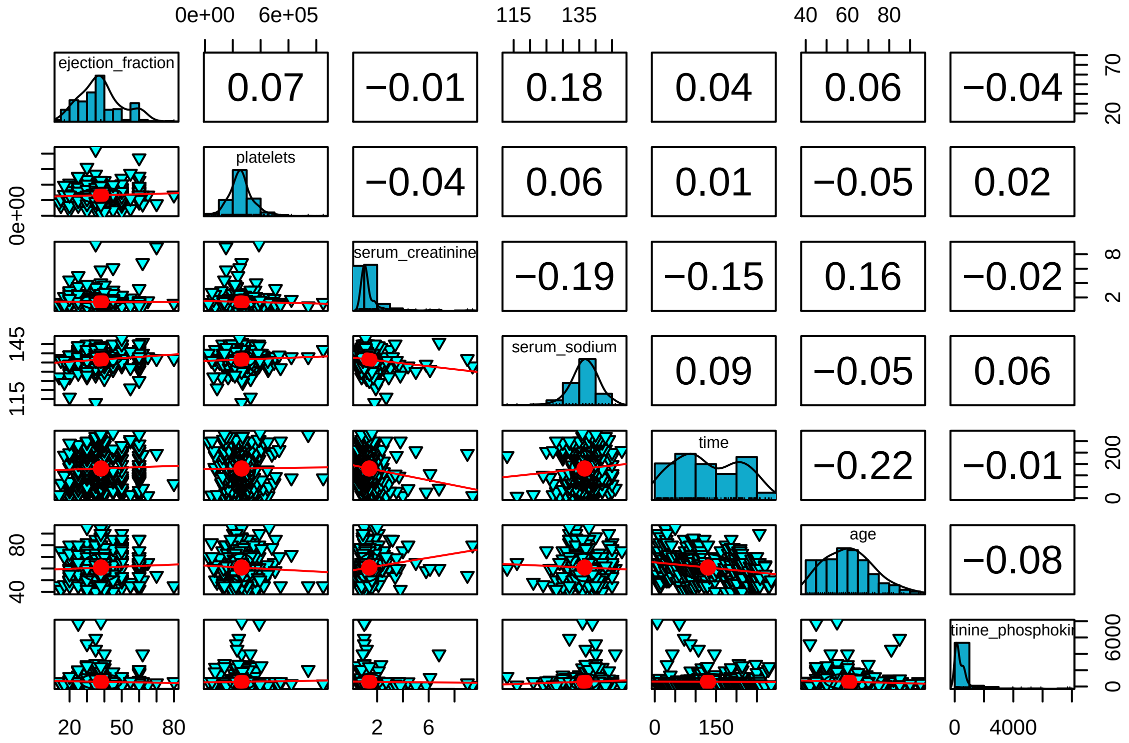


Check the documentation for more information

```
?pairs
```

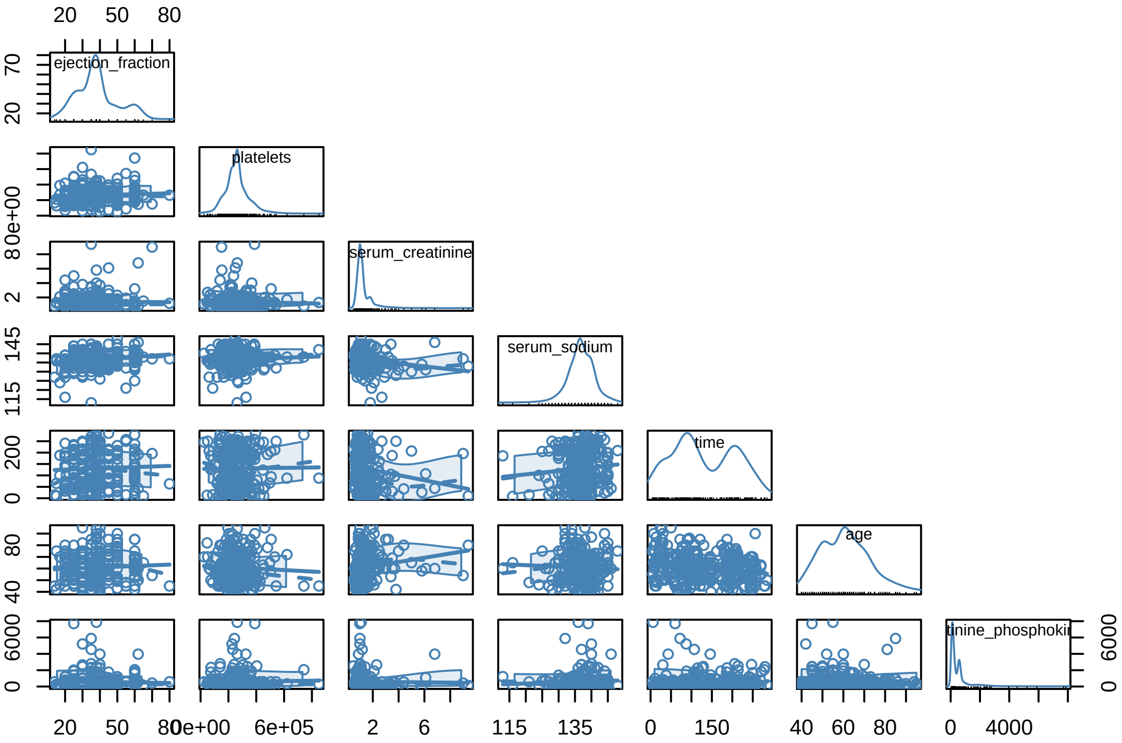
Scatter Matrix with psych package

```
library(psych)
pairs.panels(heart_data,
             method = "pearson", # Correlation method
             hist.col = "#11AACC",
             density = TRUE,
             cex.cor = 1.3,
             col = "red",
             lm = TRUE,
             pch = 25,     # point character
             bg = "cyan") # background
```



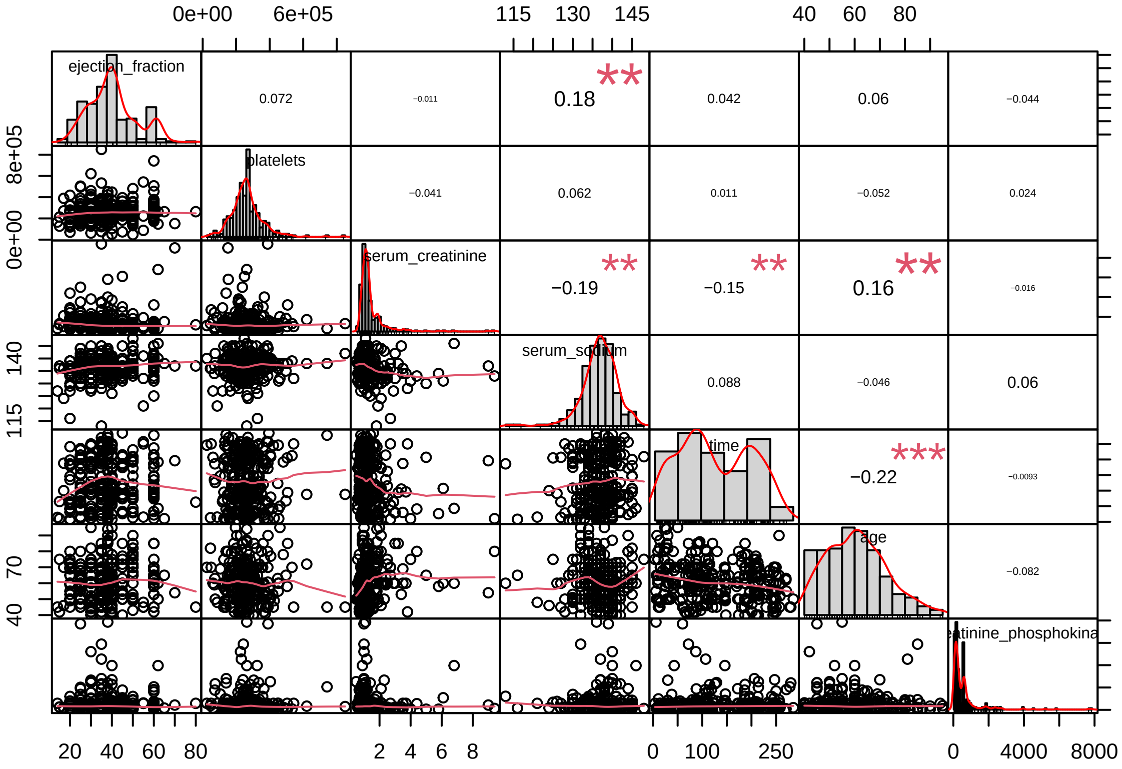
Scatter Matrix with car package

```
car::scatterplotMatrix(heart_data,
                       col = "steelblue",
                       pch = 21,
                       upper.panel = NULL)
```



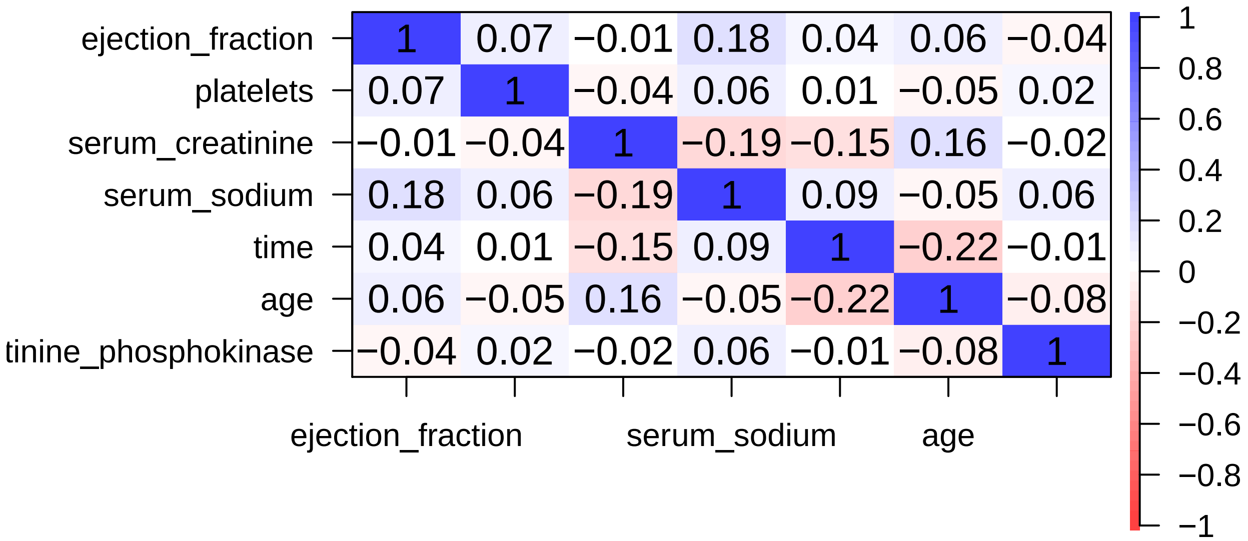
Lastly

```
library(PerformanceAnalytics)
chart.Correlation(heart_data,
                  histogram=TRUE,
                  pch=19,
                  col = "blue")
```

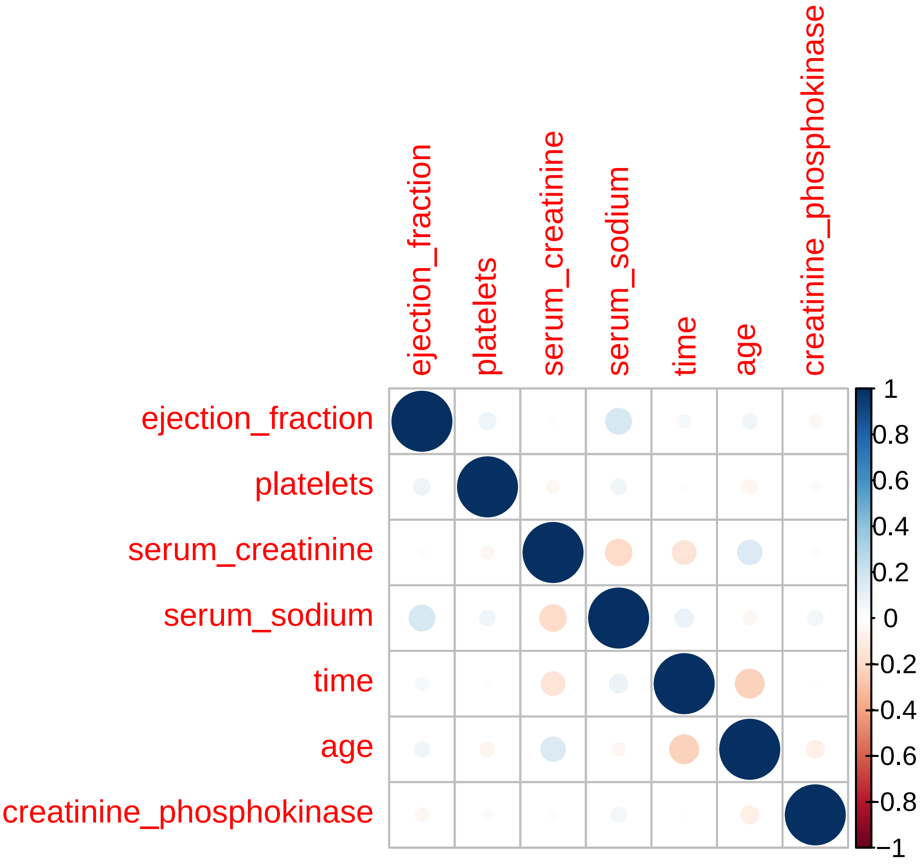


##** ## correlation plot matrices
##**

```
correl <- cor(heart_data)

psych::cor.plot(correl)
```

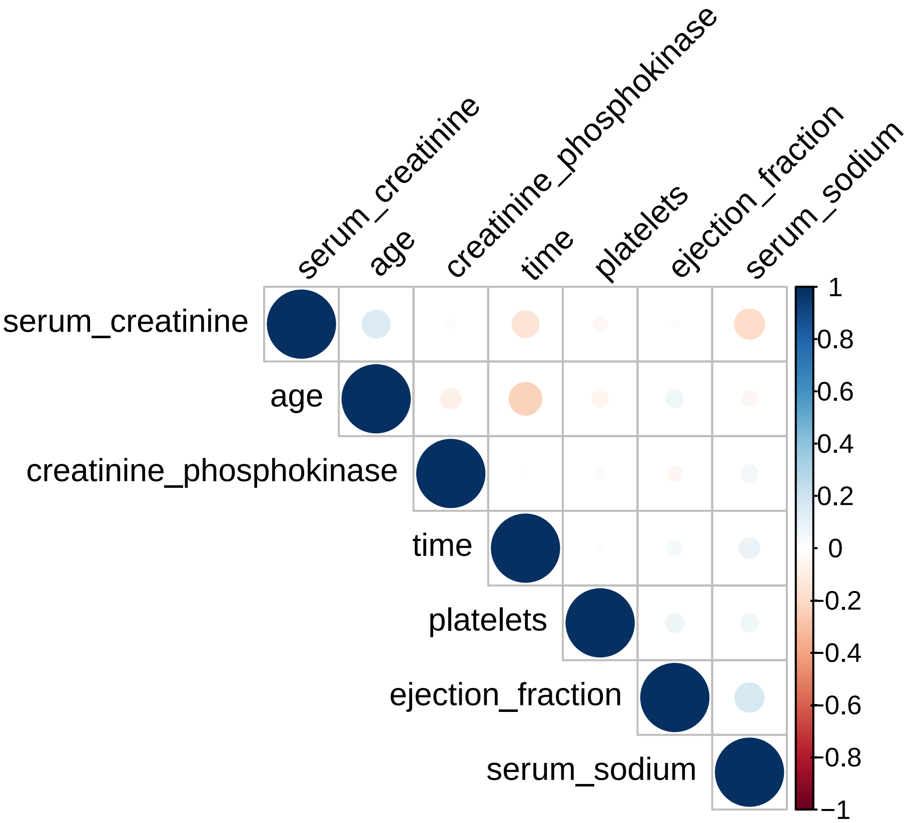


```r
corrplot::corrplot(correl)
```



Tweak the knobs

```
corrplot(correl, type = "upper",
         order = "hclust",
         tl.col = "black",
         tl.srt = 45)
```

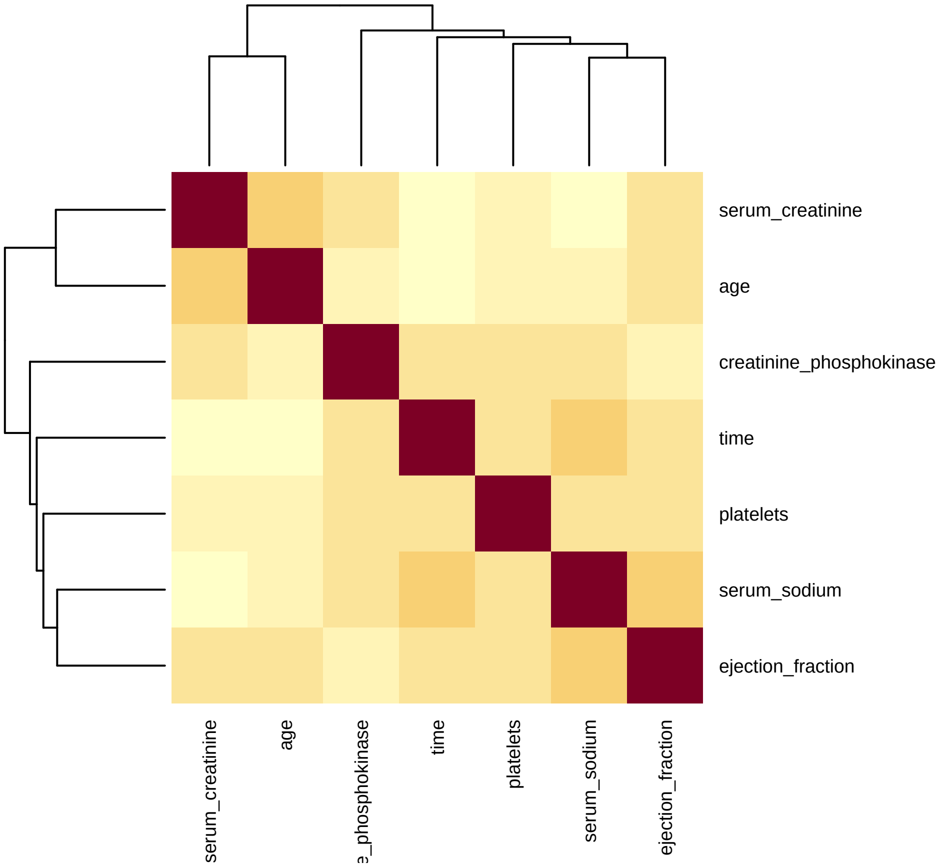


check

```
?corrplot
```

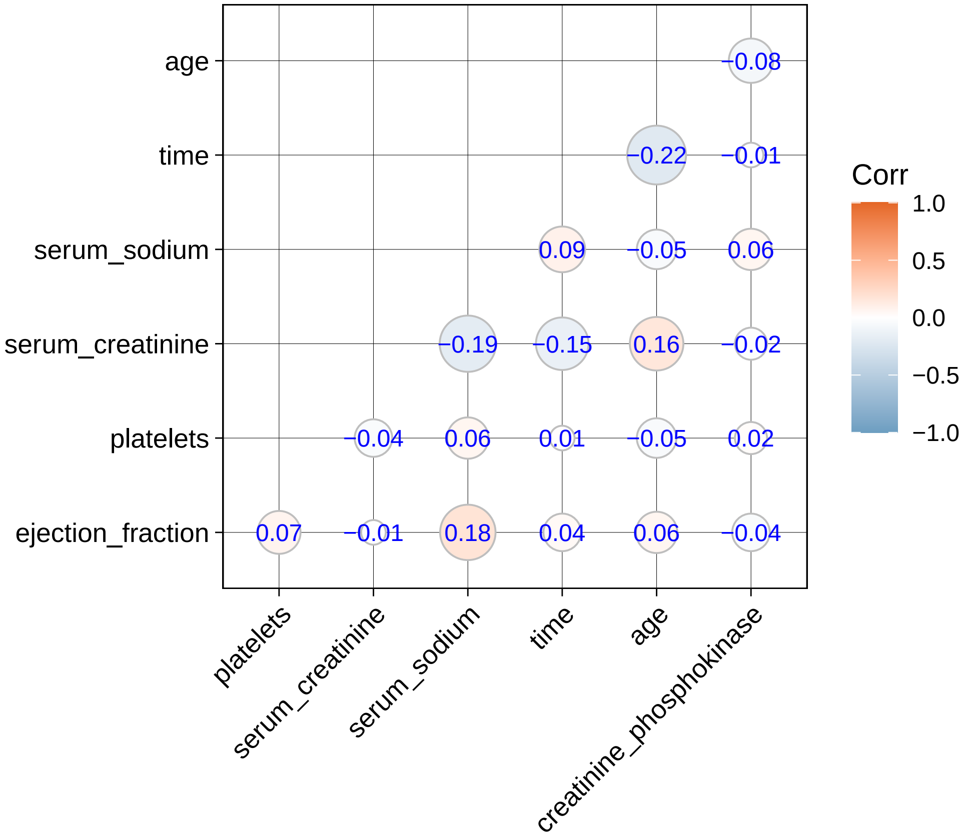
Heatmap

```
heatmap(correl, symm = TRUE,
        cexRow = 0.7,
        cexCol = 0.7)
```

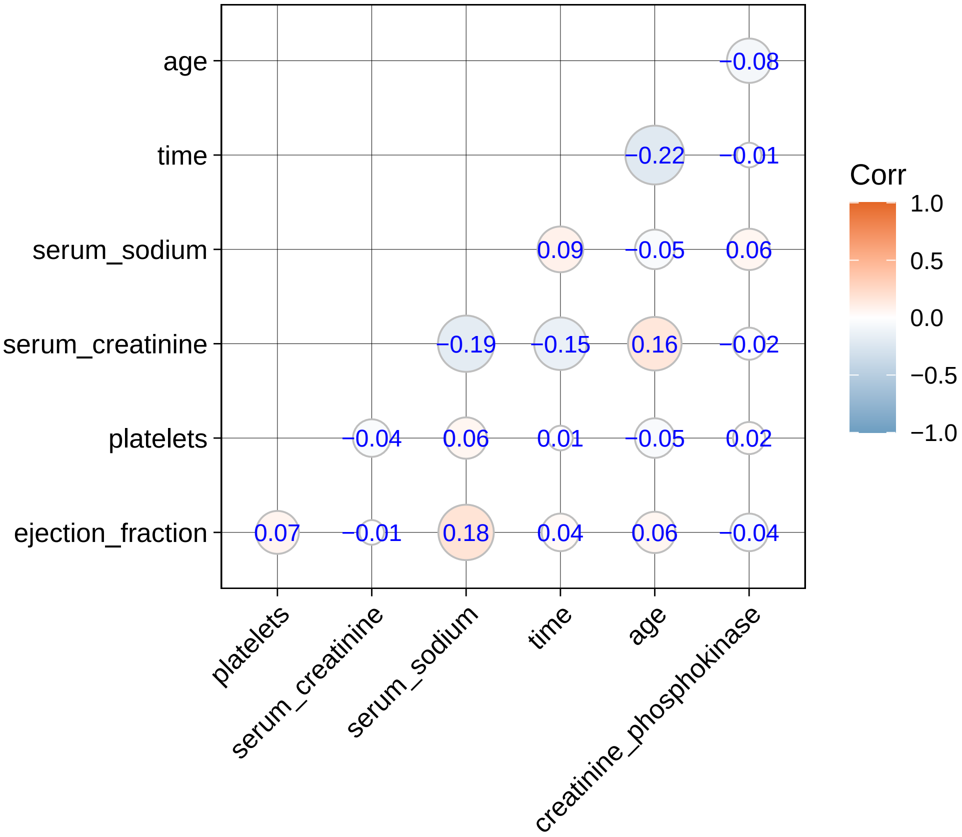


ggcorrplot

```
p <- ggcorrplot::ggcorrplot(correl,
                            method = "circle",
                            type = "lower",
                            ggtheme = ggplot2::theme_linedraw,
                            lab_col = "blue",
                            lab_size = 3,
                            tl.cex = 10,
                            lab = TRUE,
                            pch.cex = 10,
                            colors = c("#6D9EC1", "white", "#E46726"))
p
```



```r
p + guides(scale = "none")
```

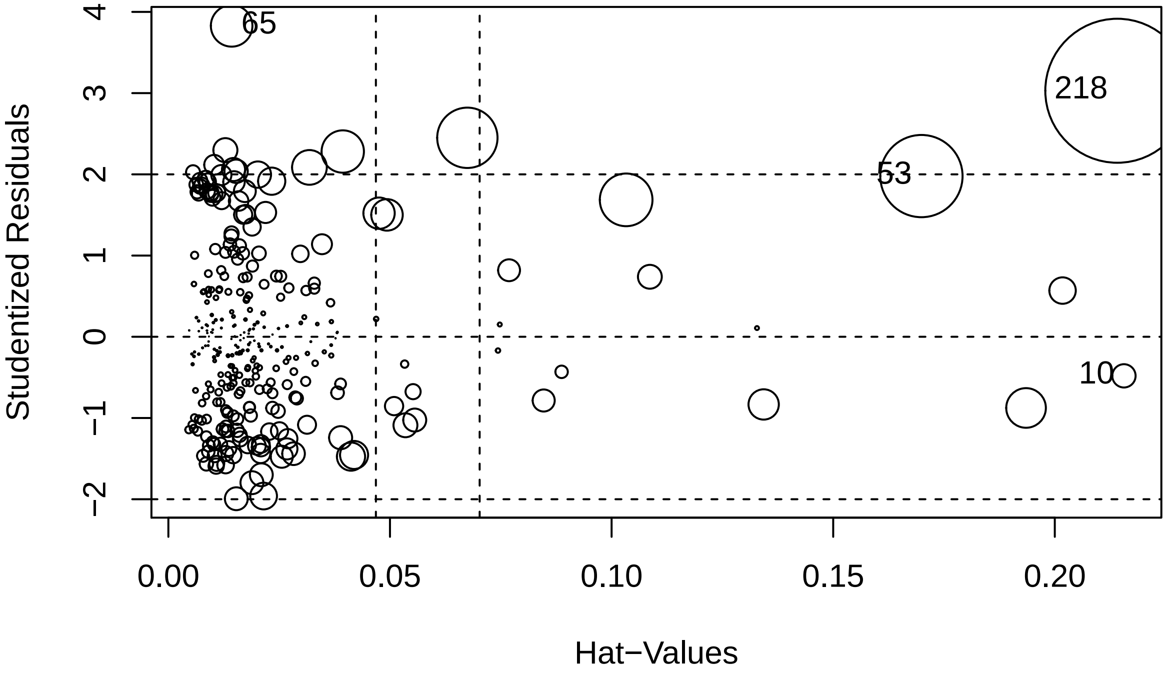


##————————————————- ## Fitting Multiple Regression – ##———————
————————————-

```
model <- lm(ejection_fraction ~ serum_sodium + age + platelets + serum_creatinine +
              time + creatinine_phosphokinase , data = heart_data)
model
```

```
##
## Call:
## lm(formula = ejection_fraction ~ serum_sodium + age + platelets +
##     serum_creatinine + time + creatinine_phosphokinase, data = heart_data)
##
## Coefficients:
##              (Intercept)              serum_sodium                       age
##               -3.456e+01                 4.770e-01                 7.430e-02
##                platelets          serum_creatinine                      time
##                8.055e-06                 2.176e-01                 6.794e-03
## creatinine_phosphokinase
##               -6.035e-04
```

```
influencePlot(model)
```



```
##        StudRes        Hat       CookD
## 10  -0.4808995 0.21560118 0.009104784
## 53   1.9785962 0.16990708 0.113341065
## 65   3.8282196 0.01427503 0.028964627
## 218  3.0292263 0.21410385 0.347401188
```

##———————————————— ## Checking lm objects – ##————————————————
———————— # Show the components of lm object

```
str(model)
```

```
## List of 12
##  $ coefficients : Named num [1:7] -3.46e+01 4.77e-01 7.43e-02 8.05e-06 2.18e-01 ...
##   ..- attr(*, "names")= chr [1:7] "(Intercept)" "serum_sodium" "age" "platelets" ...
##  $ residuals    : Named num [1:299] -15.24 5.95 -13.34 -16.58 -8.78 ...
##   ..- attr(*, "names")= chr [1:299] "1" "2" "3" "4" ...
##  $ effects      : Named num [1:299] -658.53 35.94 -13.94 13.25 -2.76 ...
##   ..- attr(*, "names")= chr [1:299] "(Intercept)" "serum_sodium" "age" "platelets" ...
##  $ rank         : int 7
##  $ fitted.values: Named num [1:299] 35.2 32.1 33.3 36.6 28.8 ...
##   ..- attr(*, "names")= chr [1:299] "1" "2" "3" "4" ...
##  $ assign       : int [1:7] 0 1 2 3 4 5 6
##  $ qr           :List of 5
##   ..$ qr   : num [1:299, 1:7] -17.2916 0.0578 0.0578 0.0578 0.0578 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : chr [1:299] "1" "2" "3" "4" ...
##   .. .. ..$ : chr [1:7] "(Intercept)" "serum_sodium" "age" "platelets" ...
##   .. ..- attr(*, "assign")= int [1:7] 0 1 2 3 4 5 6
##   ..$ qraux: num [1:7] 1.06 1 1.02 1.04 1.03 ...
##   ..$ pivot: int [1:7] 1 2 3 4 5 6 7
##   ..$ tol  : num 1e-07
##   ..$ rank : int 7
##   ..- attr(*, "class")= chr "qr"
##  $ df.residual  : int 292
##  $ xlevels      : Named list()
##  $ call         : language lm(formula = ejection_fraction ~ serum_sodium + age + platelets + serum_c
##  $ terms        :Classes 'terms', 'formula'  language ejection_fraction ~ serum_sodium + age + plate
##   .. ..- attr(*, "variables")= language list(ejection_fraction, serum_sodium, age, platelets, serum_
##   .. ..- attr(*, "factors")= int [1:7, 1:6] 0 1 0 0 0 0 0 0 0 1 ...
##   .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. ..$ : chr [1:7] "ejection_fraction" "serum_sodium" "age" "platelets" ...
##   .. .. .. ..$ : chr [1:6] "serum_sodium" "age" "platelets" "serum_creatinine" ...
##   .. ..- attr(*, "term.labels")= chr [1:6] "serum_sodium" "age" "platelets" "serum_creatinine" ...
##   .. ..- attr(*, "order")= int [1:6] 1 1 1 1 1 1
##   .. ..- attr(*, "intercept")= int 1
##   .. ..- attr(*, "response")= int 1
##   .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. ..- attr(*, "predvars")= language list(ejection_fraction, serum_sodium, age, platelets, serum_c
##   .. ..- attr(*, "dataClasses")= Named chr [1:7] "numeric" "numeric" "numeric" "numeric" ...
##   .. .. ..- attr(*, "names")= chr [1:7] "ejection_fraction" "serum_sodium" "age" "platelets" ...
##  $ model        :'data.frame':   299 obs. of  7 variables:
##   ..$ ejection_fraction       : int [1:299] 20 38 20 20 20 40 15 60 65 35 ...
##   ..$ serum_sodium            : int [1:299] 130 136 129 137 116 132 137 131 138 133 ...
##   ..$ age                     : num [1:299] 75 55 65 50 65 90 75 60 65 80 ...
##   ..$ platelets               : num [1:299] 265000 263358 162000 210000 327000 ...
##   ..$ serum_creatinine        : num [1:299] 1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.4 ...
##   ..$ time                    : int [1:299] 4 6 7 7 8 8 10 10 10 10 ...
##   ..$ creatinine_phosphokinase: int [1:299] 582 7861 146 111 160 47 246 315 157 123 ...
##   ..- attr(*, "terms")=Classes 'terms', 'formula'  language ejection_fraction ~ serum_sodium + age +
##   .. .. ..- attr(*, "variables")= language list(ejection_fraction, serum_sodium, age, platelets, ser
##   .. .. ..- attr(*, "factors")= int [1:7, 1:6] 0 1 0 0 0 0 0 0 0 1 ...
##   .. .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. .. ..$ : chr [1:7] "ejection_fraction" "serum_sodium" "age" "platelets" ...
##   .. .. .. .. ..$ : chr [1:6] "serum_sodium" "age" "platelets" "serum_creatinine" ...
##   .. .. ..- attr(*, "term.labels")= chr [1:6] "serum_sodium" "age" "platelets" "serum_creatinine" ..
##   .. .. ..- attr(*, "order")= int [1:6] 1 1 1 1 1 1
##   .. .. ..- attr(*, "intercept")= int 1
##   .. .. ..- attr(*, "response")= int 1
##   .. .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. .. ..- attr(*, "predvars")= language list(ejection_fraction, serum_sodium, age, platelets, seru
##   .. .. ..- attr(*, "dataClasses")= Named chr [1:7] "numeric" "numeric" "numeric" "numeric" ...
##   .. .. .. ..- attr(*, "names")= chr [1:7] "ejection_fraction" "serum_sodium" "age" "platelets" ...
##  - attr(*, "class")= chr "lm"
```

```
class(model)
```

```
## [1] "lm"
```

```
typeof(model)
```

```
## [1] "list"
```

```
length(model)
```

```
## [1] 12
```

```
names(model)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

Summary Function

```
summary(model)
```

```
##
## Call:
## lm(formula = ejection_fraction ~ serum_sodium + age + platelets +
##     serum_creatinine + time + creatinine_phosphokinase, data = heart_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.008  -7.838  -1.393   6.397  43.418
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -3.456e+01  2.191e+01  -1.578   0.1157
## serum_sodium              4.770e-01  1.570e-01   3.037   0.0026 **
## age                       7.430e-02  5.918e-02   1.256   0.2103
## platelets                 8.055e-06  6.947e-06   1.159   0.2472
## serum_creatinine          2.176e-01  6.784e-01   0.321   0.7486
## time                      6.794e-03  9.035e-03   0.752   0.4527
## creatinine_phosphokinase -6.035e-04  7.017e-04  -0.860   0.3905
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.69 on 292 degrees of freedom
## Multiple R-squared:  0.0444, Adjusted R-squared:  0.02476
## F-statistic: 2.261 on 6 and 292 DF,  p-value: 0.03778
```

Coefficients Function

```
model$coefficients
```

```
##              (Intercept)             serum_sodium                      age
##            -3.456302e+01             4.769850e-01             7.429994e-02
##                platelets         serum_creatinine                     time
##             8.054673e-06             2.176059e-01             6.793735e-03
## creatinine_phosphokinase
##            -6.034873e-04
```

Fitted function

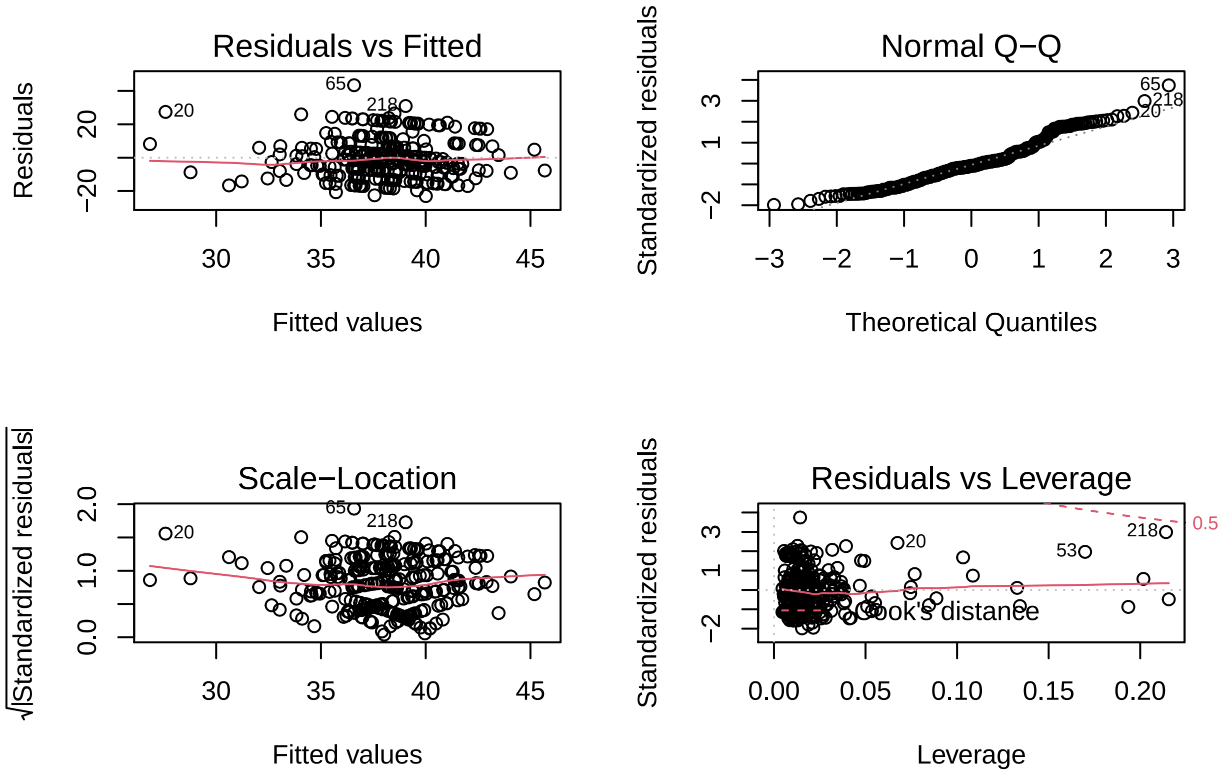
```
model$fitted
```

```
##        1        2        3        4        5        6        7        8
## 35.24141 32.05081 33.34473 36.58442 28.77594 37.21210 37.55997 36.15404
##        9       10       11       12       13       14       15       16
## 38.51127 39.98440 37.34811 38.98368 34.94493 36.93472 38.59180 36.92059
##       17       18       19       20       21       22       23       24
## 40.99034 30.61261 39.56891 27.58345 38.19671 37.93547 39.79403 37.01750
##       25       26       27       28       29       30       31       32
## 37.24997 41.74735 40.21118 38.18136 36.29727 36.49805 38.68885 38.44334
##       33       34       35       36       37       38       39       40
## 33.03306 37.77061 38.91655 36.93033 38.24553 43.18338 36.80004 40.99164
##       41       42       43       44       45       46       47       48
## 36.93282 35.65799 38.56973 36.82364 39.29816 36.34190 33.03834 39.40265
##       49       50       51       52       53       54       55       56
## 36.84991 40.05666 37.81190 39.58450 41.03567 37.85604 35.53848 39.72170
##       57       58       59       60       61       62       63       64
## 39.16723 37.58982 36.65395 38.16647 34.19988 34.09303 38.45149 41.32514
##       65       66       67       68       69       70       71       72
## 36.58223 32.45540 35.71679 40.36427 40.94754 39.44783 38.82037 37.35693
##       73       74       75       76       77       78       79       80
## 33.82882 37.45042 37.02805 38.45895 40.59051 37.94674 36.28477 39.40690
##       81       82       83       84       85       86       87       88
## 38.70577 38.30515 35.45698 37.01949 40.08661 39.91633 37.01159 40.16275
##       89       90       91       92       93       94       95       96
## 37.53827 40.52248 38.19130 35.94599 36.48841 36.81351 40.74142 39.46645
##       97       98       99      100      101      102      103      104
## 36.61633 38.34310 38.54776 39.46267 40.56801 38.97428 41.55545 34.82093
##      105      106      107      108      109      110      111      112
## 41.46703 42.38229 37.41813 36.08631 36.27785 44.06333 37.97394 36.58927
##      113      114      115      116      117      118      119      120
## 36.78863 39.62013 33.06135 37.96270 37.68075 42.93352 39.44573 38.55642
##      121      122      123      124      125      126      127      128
## 36.50647 37.33756 39.26001 40.57076 36.66494 35.65964 31.21869 38.26966
##      129      130      131      132      133      134      135      136
## 37.69904 41.65637 36.98577 35.79503 36.87278 42.55017 36.90672 39.11016
##      137      138      139      140      141      142      143      144
## 37.88599 35.75494 37.47537 36.64905 37.91230 39.19645 35.40946 39.96114
##      145      146      147      148      149      150      151      152
## 40.36491 39.35098 36.76002 37.50203 37.55245 36.21404 38.25708 39.28244
##      153      154      155      156      157      158      159      160
## 35.47813 36.57137 37.81645 39.28125 37.48657 37.01424 38.11707 40.02094
##      161      162      163      164      165      166      167      168
## 37.24571 37.40058 39.75356 37.13126 37.41369 38.01821 35.53031 35.70007
##      169      170      171      172      173      174      175      176
## 39.08267 37.18892 37.17777 37.49054 42.34648 38.07179 38.94502 39.58549
##      177      178      179      180      181      182      183      184
## 36.16233 38.09344 42.61748 39.19572 36.63235 37.60117 38.33991 38.31457
##      185      186      187      188      189      190      191      192
## 38.18585 39.37314 35.24819 42.00205 37.72848 38.63261 39.40395 38.54515
##      193      194      195      196      197      198      199      200
## 34.52245 41.22173 35.40815 43.47940 37.41107 39.51350 36.44523 26.84005
##      201      202      203      204      205      206      207      208 
## 36.40241 37.98241 39.63487 38.46897 40.19459 38.32498 38.86778 39.45894 
##      209      210      211      212      213      214      215      216 
## 39.54081 34.10230 40.00809 38.24116 42.40447 34.55707 37.92892 37.66840 
##      217      218      219      220      221      222      223      224 
## 41.57846 39.04471 37.54975 40.42912 38.29028 42.51881 39.35145 35.08057 
##      225      226      227      228      229      230      231      232 
## 40.88895 34.05710 35.87193 37.36806 36.64402 38.44964 32.64751 37.64042 
##      233      234      235      236      237      238      239      240 
## 37.34775 38.67774 38.48652 41.36869 45.18825 36.54125 38.41626 38.57331 
##      241      242      243      244      245      246      247      248 
## 42.90631 38.51398 36.95820 40.81556 36.97913 39.29107 38.35256 38.45641 
##      249      250      251      252      253      254      255      256 
## 39.86019 34.76918 38.74765 40.88250 36.32837 37.16957 39.20616 41.29249 
##      257      258      259      260      261      262      263      264 
## 40.60328 39.78563 36.74700 37.93657 38.74322 37.10203 35.13784 40.67562 
##      265      266      267      268      269      270      271      272 
## 39.92821 40.49217 36.87266 39.39308 39.49317 34.68318 34.49010 36.42463 
##      273      274      275      276      277      278      279      280 
## 37.38288 38.63597 38.17710 39.01381 42.53460 39.17744 37.07994 39.92955 
##      281      282      283      284      285      286      287      288 
## 38.19536 37.85346 33.83012 39.13078 40.49404 38.47979 40.83581 37.68099 
##      289      290      291      292      293      294      295      296 
## 41.55986 45.68280 39.37243 39.13148 41.01109 38.39762 41.53777 39.00303 
##      297      298      299 
## 41.40053 37.43671 39.36963
```

```
extract_eq(model,use_cof=TRUE)
```

$$\operatorname{ejection\_fraction} = \alpha + \beta_{1}(\operatorname{serum\_sodium}) + \beta_{2}(\operatorname{age}) + \beta_{3}(\operatorname{platelets}) + \beta_{4}(\operatorname{serum\_creatinine}) + \beta_{5}(\operatorname{time}) + \beta_{6}(\operatorname{creatinine\_phospho}$$

```
par(mfrow = c(2, 2))
plot(model)
```



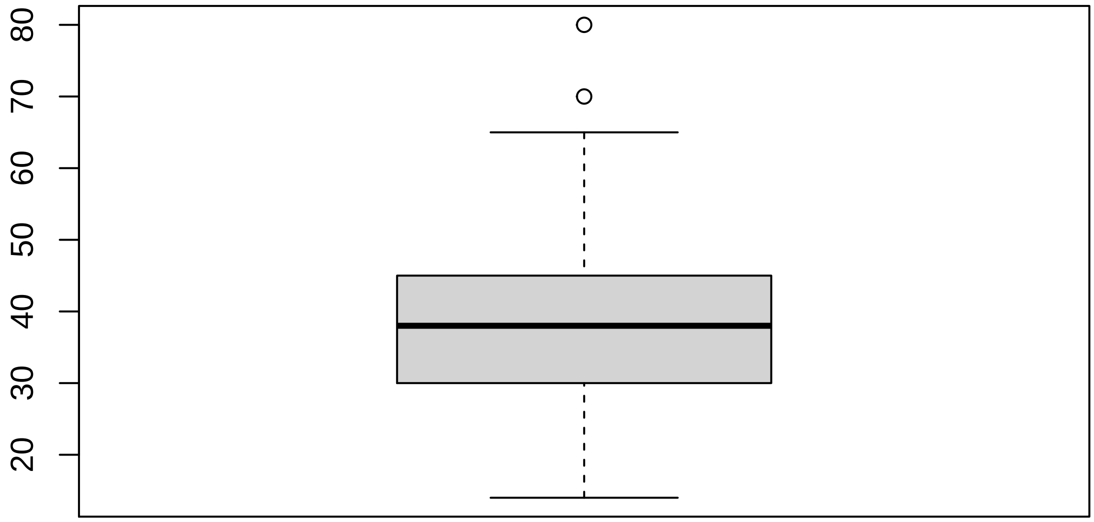
```r
vif(model)
```

```
##             serum_sodium                      age                platelets 
##                 1.047593                 1.080921                 1.007235 
##         serum_creatinine                     time creatinine_phosphokinase 
##                 1.074673                 1.072699                 1.011278
```

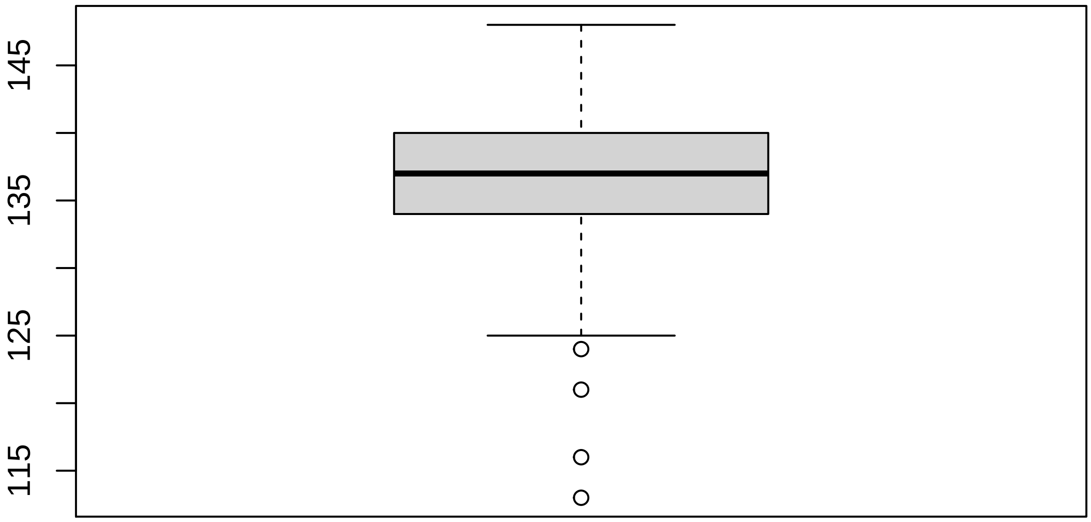
We will improve our result by removing the outlier

##create a boxplot to identify outliers

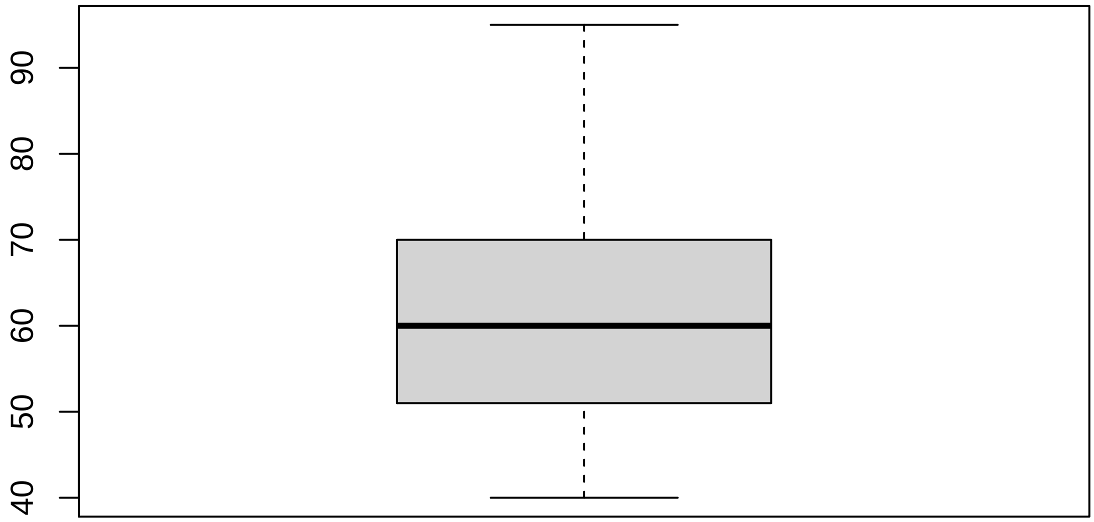
```r
boxplot(heart_data$ejection_fraction)
```



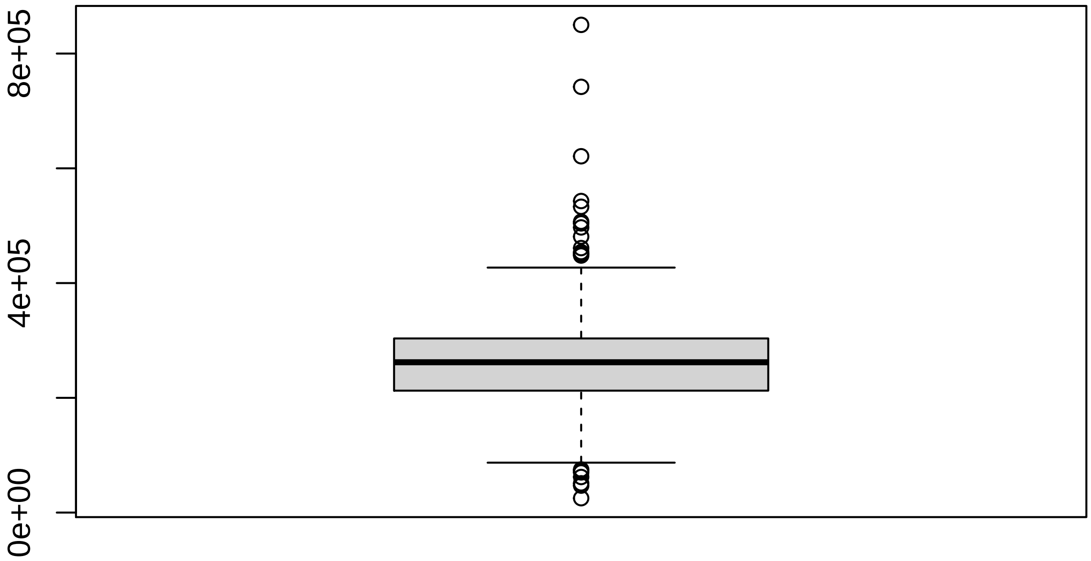
```
boxplot(heart_data$serum_sodium)
```



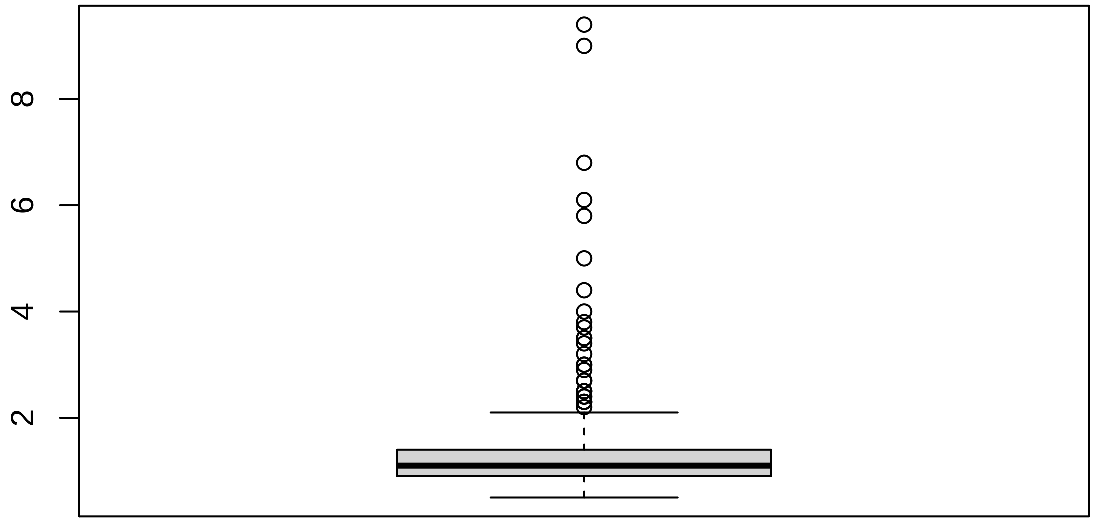
```
boxplot(heart_data$age)
```



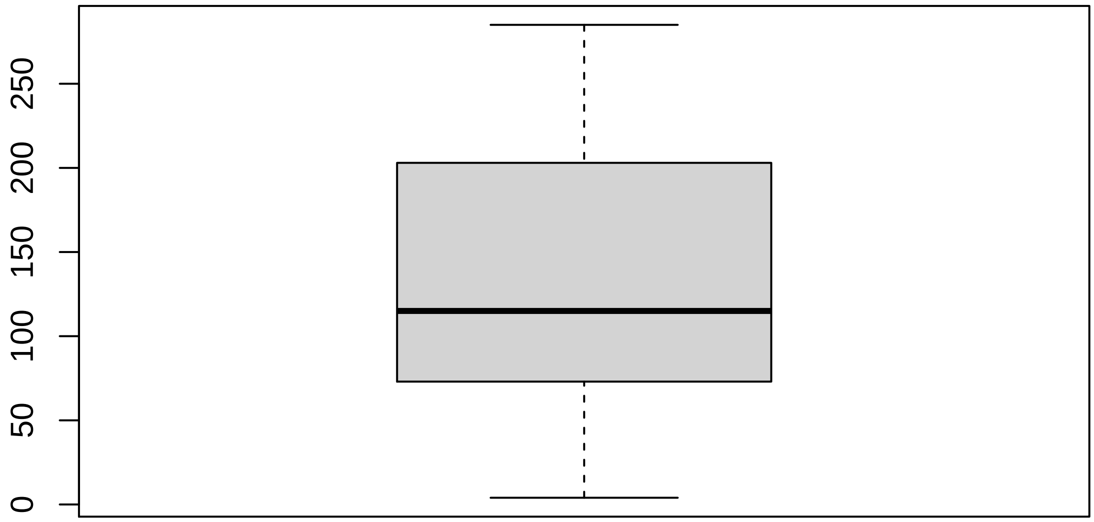
```
boxplot(heart_data$platelets)
```



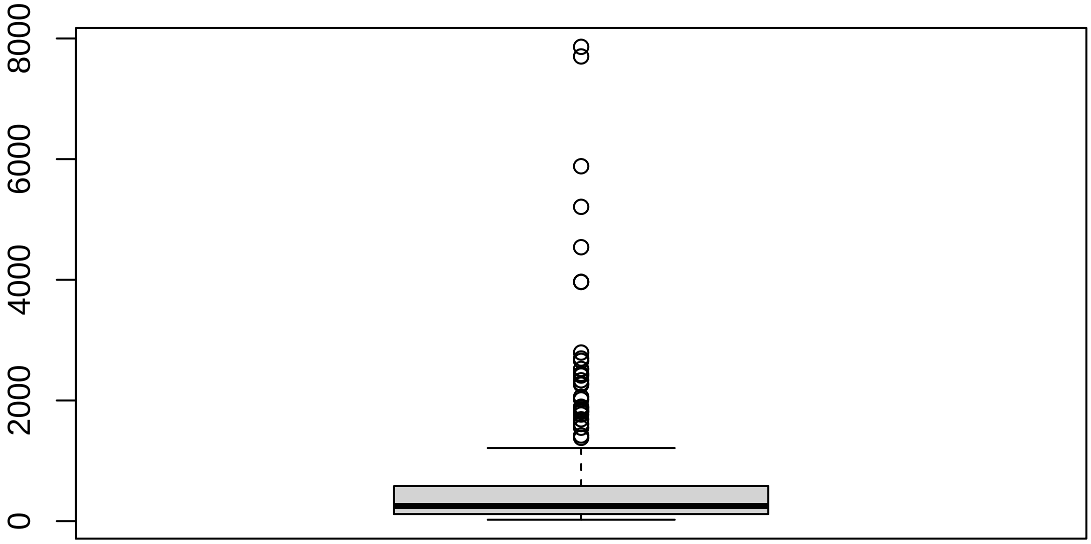
```r
boxplot(heart_data$serum_creatinine)
```



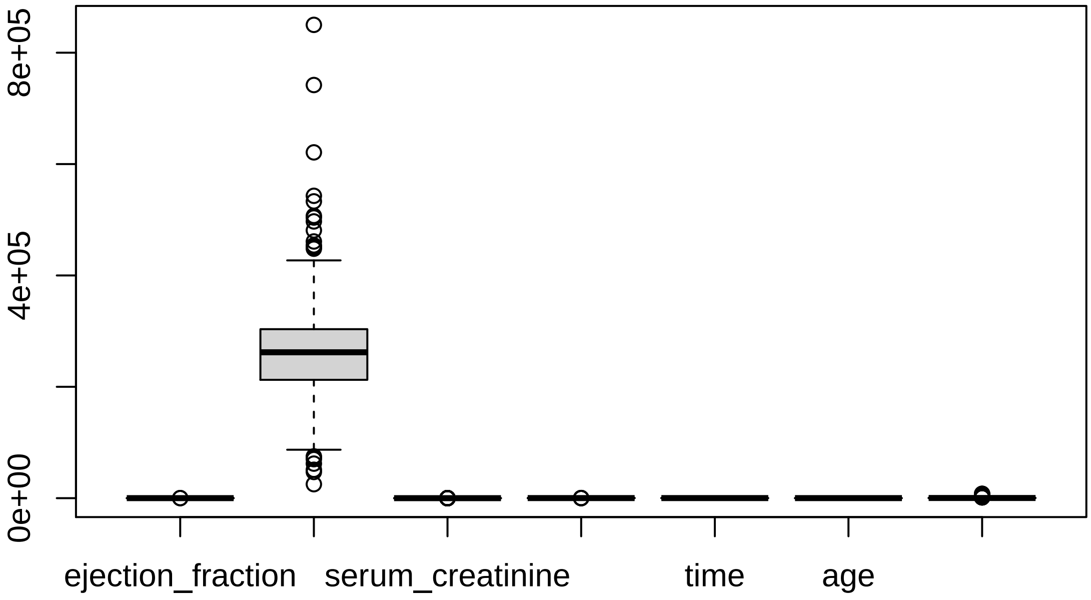
```
boxplot(heart_data$time)
```



```r
boxplot(heart_data$creatinine_phosphokinase)
```



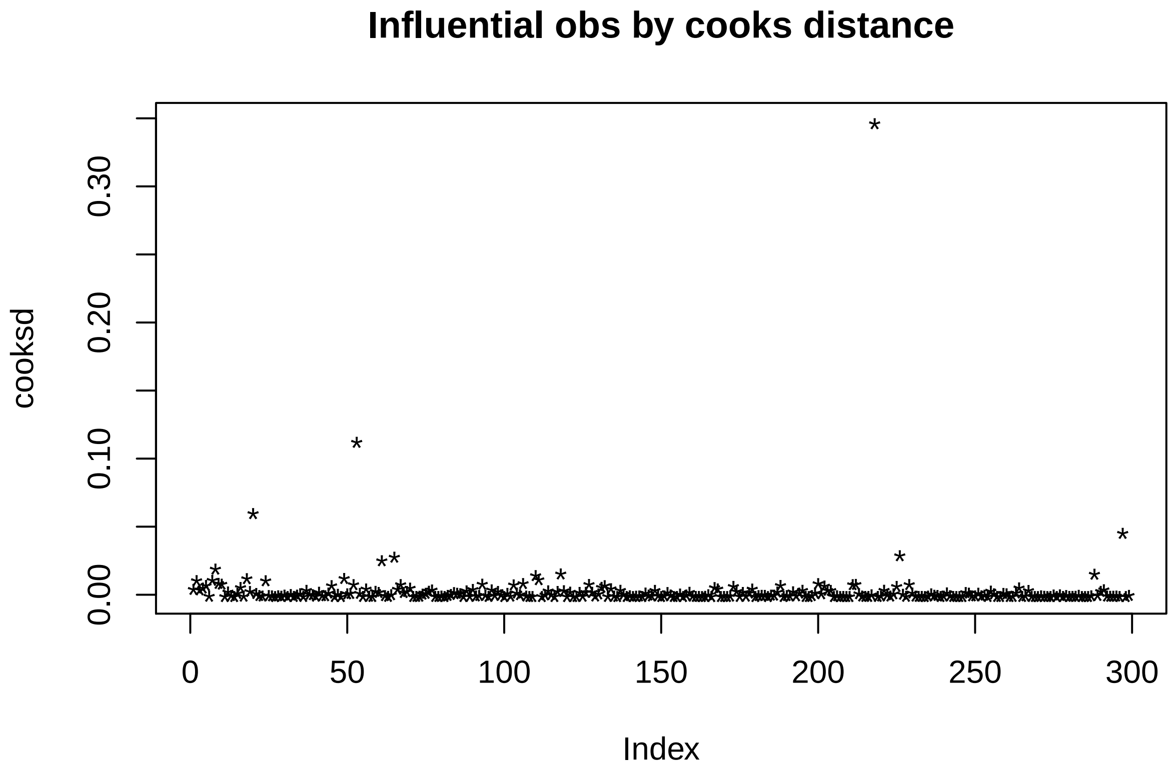
```
boxplot(heart_data)
boxplot(heart_data)$out
```



```
##  [1]     80.0     70.0 454000.0  47000.0 451000.0 461000.0 497000.0 621000.0
##  [9] 850000.0 507000.0 448000.0  75000.0  70000.0  73000.0 481000.0 504000.0
## [17]  62000.0 533000.0  25100.0 451000.0  51000.0 543000.0 742000.0      2.7
## [25]      9.4      4.0      5.8      3.0      3.5      2.3      3.0      4.4
## [33]      6.8      2.2      2.7      2.3      2.9      2.5      2.3      3.2
## [41]      3.7      3.4      6.1      2.5      2.4      2.5      3.5      9.0
## [49]      5.0      2.4      2.7      3.8    116.0    121.0    124.0    113.0
## [57]   7861.0   2656.0   1380.0   3964.0   7702.0   5882.0   5209.0   1876.0
## [65]   1808.0   4540.0   1548.0   1610.0   2261.0   1846.0   2334.0   2442.0
## [73]   3966.0   1419.0   1896.0   1767.0   2281.0   2794.0   2017.0   2522.0
## [81]   2695.0   1688.0   1820.0   2060.0   2413.0
```

###How to Remove Outliers from Multiple Columns in R

```
cooksd<-cooks.distance((model))

plot(cooksd,pch="*",cex=1,main="Influential obs by cooks distance")
```



```
cooksd
```

```
##            1            2            3            4            5            6 
## 5.284681e-03 1.171972e-02 4.912216e-03 6.245319e-03 8.141469e-03 2.653447e-04 
##            7            8            9           10           11           12 
## 1.194381e-02 2.019379e-02 9.687037e-03 9.104784e-03 1.831718e-05 3.394552e-03 
##           13           14           15           16           17           18 
## 7.667166e-04 2.265372e-05 2.342132e-03 6.653790e-03 2.857165e-04 1.317423e-02 
##           19           20           21           22           23           24 
## 3.724945e-03 6.097901e-02 2.279553e-03 7.651668e-04 3.730553e-04 1.161865e-02 
##           25           26           27           28           29           30 
## 7.472761e-04 5.201301e-04 1.900456e-06 5.728536e-04 2.648348e-04 1.068289e-03 
##           31           32           33           34           35           36 
## 1.702930e-05 1.487766e-03 1.284833e-04 1.061639e-03 2.072173e-03 8.540088e-05 
##           37           38           39           40           41           42 
## 4.572457e-03 1.710926e-03 1.369558e-03 2.682449e-04 3.203227e-03 6.881869e-04 
##           43           44           45           46           47           48 
## 6.016504e-04 2.594720e-03 8.008238e-03 4.448462e-05 1.664264e-03 4.984561e-05 
##           49           50           51           52           53           54 
## 1.330929e-02 2.012071e-03 2.290776e-03 8.774700e-03 1.133411e-01 2.354134e-03 
##           55           56           57           58           59           60 
## 7.839482e-05 5.588178e-03 2.623447e-04 6.646141e-05 3.850300e-03 3.011220e-03 
##           61           62           63           64           65           66 
## 2.633563e-02 6.881912e-04 1.329521e-04 1.380565e-03 2.896463e-02 5.409143e-03 
##           67           68           69           70           71           72 
## 8.790110e-03 2.973094e-03 3.731132e-03 6.211757e-03 3.803257e-05 9.246817e-05 
##           73           74           75           76           77           78
## 2.533409e-04 1.780941e-03 2.409340e-03 3.926756e-03 4.850425e-03 9.242458e-05
##           79           80           81           82           83           84
## 6.217289e-05 3.590252e-04 1.355204e-05 8.580346e-04 1.520130e-03 2.917386e-03
##           85           86           87           88           89           90
## 2.433875e-03 2.096440e-03 4.741065e-05 4.178946e-03 1.142178e-04 5.240252e-03
##           91           92           93           94           95           96
## 4.479522e-04 7.868025e-04 9.111827e-03 1.012419e-03 1.087684e-04 4.921409e-03
##           97           98           99          100          101          102
## 8.407937e-04 3.753156e-03 1.290173e-03 2.705104e-06 2.358255e-03 3.521267e-04
##          103          104          105          106          107          108
## 8.572764e-03 2.596878e-03 1.402845e-03 9.573433e-03 3.510242e-04 2.314998e-05
##          109          110          111          112          113          114
## 2.702955e-05 1.539698e-02 1.240126e-02 2.071835e-05 1.275086e-03 4.206122e-03
##          115          116          117          118          119          120
## 1.446532e-03 4.503556e-05 3.740700e-03 1.662530e-02 4.294496e-03 6.537544e-06
##          121          122          123          124          125          126
## 3.247024e-03 4.535887e-06 1.518915e-05 3.041305e-03 2.599806e-04 3.142591e-03
##          127          128          129          130          131          132
## 8.895245e-03 3.149970e-03 3.844198e-04 1.949019e-03 6.783257e-03 7.993333e-03
##          133          134          135          136          137          138
## 1.022391e-04 5.751694e-03 3.300314e-04 7.347130e-06 4.610002e-03 1.662575e-03
##          139          140          141          142          143          144
## 3.444767e-05 5.687146e-04 1.242432e-07 2.726775e-04 4.229062e-04 1.472794e-08
##          145          146          147          148          149          150
## 2.593485e-03 1.100202e-03 4.393990e-04 4.551796e-03 5.800149e-04 2.415411e-05
##          151          152          153          154          155          156
## 4.716964e-04 2.953297e-03 1.158468e-03 3.074422e-05 4.862289e-05 1.859283e-03
##          157          158          159          160          161          162
## 4.507423e-05 1.155256e-03 3.145229e-03 2.300947e-04 5.101938e-05 9.903583e-05
##          163          164          165          166          167          168
## 1.071340e-06 1.795456e-04 1.341500e-03 5.423067e-09 6.606950e-03 5.613007e-03
##          169          170          171          172          173          174
## 4.180093e-06 5.974847e-05 2.966692e-05 3.398817e-04 7.471013e-03 3.807119e-03
##          175          176          177          178          179          180
## 9.003808e-05 3.013658e-03 2.943979e-04 2.577543e-03 5.516997e-03 4.721195e-06
##          181          182          183          184          185          186
## 6.950263e-04 9.006640e-04 8.674883e-04 1.516047e-06 1.056627e-03 1.033790e-03
##          187          188          189          190          191          192
## 3.339629e-03 8.146482e-03 3.736027e-05 7.102899e-04 4.347388e-04 3.508900e-03
##          193          194          195          196          197          198
## 5.763194e-04 2.542098e-03 4.607915e-03 6.840248e-05 3.675433e-06 7.318874e-04
##          199          200          201          202          203          204
## 2.592985e-03 9.521235e-03 2.062839e-03 7.733730e-03 4.573590e-03 4.525863e-03
##          205          206          207          208          209          210
## 6.974020e-07 5.452967e-04 2.675198e-05 6.007895e-05 4.148692e-06 1.591460e-05
##          211          212          213          214          215          216
## 8.750738e-03 8.936827e-03 2.123926e-03 4.019204e-04 9.448173e-05 1.029148e-04
##          217          218          219          220          221          222
## 1.295557e-03 3.474012e-01 7.750182e-05 3.729277e-04 4.637201e-03 1.320985e-03
##          223          224          225          226          227          228
## 3.683757e-04 2.014246e-03 7.453644e-03 3.000988e-02 1.684352e-03 1.961317e-04
##          229          230          231          232          233          234
## 8.837211e-03 2.615989e-03 2.904187e-04 1.081326e-04 9.737810e-05 4.447544e-06
##          235          236          237          238          239          240
## 3.607094e-04 1.994600e-03 9.546182e-04 8.098831e-04 2.355419e-05 3.435480e-04
##          241          242          243          244          245          246
## 2.697206e-03 6.566114e-04 6.968474e-05 1.331616e-05 1.114431e-05 1.486776e-05
##          247          248          249          250          251          252
## 2.984275e-03 2.502847e-03 5.043576e-04 5.216583e-04 2.461028e-03 5.369566e-04
##          253          254          255          256          257          258
## 1.018217e-03 8.212610e-05 4.023192e-03 2.011645e-03 8.080747e-05 3.535551e-05
##          259          260          261          262          263          264
## 2.316954e-03 2.013333e-03 2.024340e-05 1.329005e-04 2.646723e-03 6.392683e-03
##          265          266          267          268          269          270
## 7.220088e-05 5.219092e-04 4.482996e-03 4.342987e-05 3.775377e-05 2.587794e-06
##          271          272          273          274          275          276
## 5.387748e-04 1.966122e-04 7.054131e-06 4.394479e-05 1.148183e-03 2.543075e-05
##          277          278          279          280          281          282
## 1.271522e-03 5.740894e-05 7.601651e-04 1.003344e-04 1.225563e-04 1.910696e-04
##          283          284          285          286          287          288
## 9.126335e-04 2.516511e-05 4.289123e-06 3.668249e-05 5.405292e-04 1.643697e-02
##          289          290          291          292          293          294
## 8.598465e-04 3.818555e-03 5.012986e-03 3.705568e-04 1.907882e-04 2.387322e-04
##          295          296          297          298          299
## 3.666463e-04 2.235451e-05 4.647102e-02 1.365131e-05 8.839829e-04
```

```
influential<-cooksd[(cooksd>(3*mean(cooksd,na.rm=TRUE)))]

names_of_influential<-names(influential)
names_of_influential
```

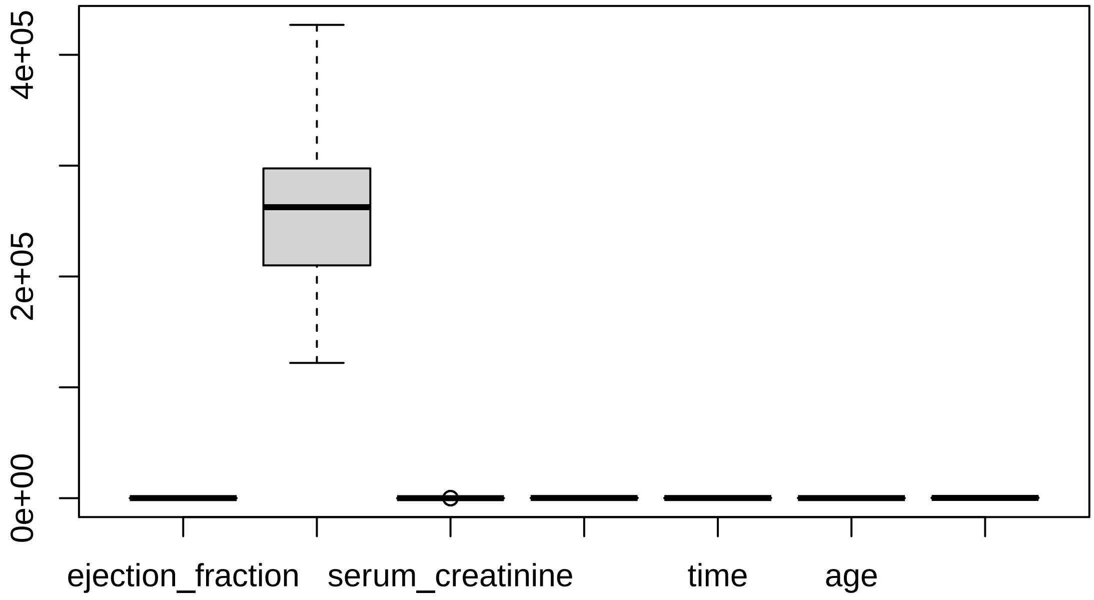
```
##  [1] "8"   "20"  "53"  "61"  "65"  "110" "118" "218" "226" "288" "297"
```

```
outliers <- heart_data[names_of_influential,]
heart_data_without_outliers <-heart_data %>% anti_join(outliers)
heart_data_without_outliers
```

```
##     ejection_fraction platelets serum_creatinine serum_sodium time    age
## 1                  20    265000             1.90          130    4 75.000
## 2                  38    263358             1.10          136    6 55.000
## 3                  20    162000             1.30          129    7 65.000
## 4                  20    210000             1.90          137    7 50.000
## 5                  20    327000             2.70          116    8 65.000
## 6                  40    204000             2.10          132    8 90.000
## 7                  15    127000             1.20          137   10 75.000
## 8                  65    263358             1.50          138   10 65.000
## 9                  35    388000             9.40          133   10 80.000
## 10                 38    368000             4.00          131   10 75.000
## 11                 25    253000             0.90          140   10 62.000
## 12                 30    136000             1.10          137   11 45.000
## 13                 38    276000             1.10          137   11 50.000
## 14                 30    427000             1.00          138   12 49.000
## 15                 50     47000             1.30          136   13 82.000
## 16                 38    262000             0.90          140   14 87.000
## 17                 14    166000             0.80          127   14 45.000
## 18                 25    237000             1.00          140   15 70.000
## 19                 25    276000             1.30          137   16 65.000
## 20                 30    297000             1.60          136   20 65.000
## 21                 35    289000             0.90          140   20 68.000
## 22                 60    368000             0.80           135   22 53.000
## 23                 30    263358             1.83           134   23 75.000
## 24                 38    149000             1.90           144   23 80.000
## 25                 40    196000             1.00           138   24 95.000
## 26                 45    284000             1.30           136   26 70.000
## 27                 38    153000             5.80           134   26 58.000
## 28                 30    200000             1.20           132   26 82.000
## 29                 38    263358             1.83           134   27 94.000
## 30                 45    360000             3.00           132   28 85.000
## 31                 35    319000             1.00           128   28 50.000
## 32                 30    302000             1.20           138   29 50.000
## 33                 50    188000             1.00           140   29 65.000
## 34                 35    228000             3.50           134   30 69.000
## 35                 50    226000             1.00           134   30 90.000
## 36                 50    321000             1.00           145   30 82.000
## 37                 30    305000             2.30           137   30 60.000
## 38                 38    329000             3.00           142   30 60.000
## 39                 20    263358             1.83           134   31 70.000
## 40                 30    153000             1.20           136   32 50.000
## 41                 45    185000             1.20           139   33 70.000
## 42                 50    218000             1.00           134   33 72.000
## 43                 60    194000             1.10           142   33 60.000
## 44                 38    310000             1.90           135   35 50.000
## 45                 25    271000             0.90           130   38 51.000
## 46                 38    451000             0.60           138   40 60.000
## 47                 20    140000             4.40           133   41 80.000
## 48                 30    395000             1.00           140   42 57.000
## 49                 25    166000             1.00           138   43 68.000
## 50                 20    418000             1.40           139   43 53.000
## 51                 50    351000             1.00           134   44 70.000
## 52                 38    255000             2.20           132   45 60.000
## 53                 30    461000             2.00           132   50 95.000
## 54                 35    223000             2.70           138   54 70.000
## 55                 40    216000             0.60           138   54 60.000
## 56                 20    319000             1.10           136   55 49.000
## 57                 20    254000             1.30           136   59 72.000
## 58                 40    216000             2.30           131   60 50.000
## 59                 35    254000             1.10           139   60 55.000
## 60                 35    385000             1.00           145   61 45.000
## 61                 20    119000             2.90           127   64 60.000
## 62                 15    213000             1.30           136   65 42.000
## 63                 25    274000             1.00           140   65 72.000
## 64                 25    244000             1.20           142   66 70.000
## 65                 25    497000             1.83           135   67 65.000
## 66                 40    374000             0.80           140   68 41.000
## 67                 35    122000             0.90           139   71 58.000
## 68                 35    243000             1.00           132   72 85.000
## 69                 50    149000             1.30           137   72 65.000
## 70                 20    266000             1.20           134   73 69.000
## 71                 20    204000             0.70           139   73 60.000
## 72                 60    317000             0.80           140   74 70.000
## 73                 40    237000             1.20           140   74 42.000
## 74                 38    283000             0.60           131   74 75.000
## 75                 45    324000             0.90           140   74 55.000
## 76                 40    293000             1.70          136   75 70.000
## 77                 50    263358             1.18          137   76 67.000
## 78                 25    196000             2.50          132   77 60.000
## 79                 50    172000             1.80          133   78 79.000
## 80                 25    302000             1.00          141   78 59.000
## 81                 50    406000             0.70          140   79 51.000
## 82                 35    173000             1.10          137   79 55.000
## 83                 60    304000             0.80          140   79 65.000
## 84                 40    235000             0.70          139   79 44.000
## 85                 25    181000             1.10          144   79 57.000
## 86                 45    249000             0.80          136   80 70.000
## 87                 45    297000             1.00          133   80 60.000
## 88                 60    263358             1.18          137   82 42.000
## 89                 25    210000             1.70          135   82 60.000
## 90                 38    327000             0.70          142   83 58.000
## 91                 60    219000             1.00          141   83 58.000
## 92                 25    254000             1.30          134   83 63.000
## 93                 60    255000             1.10          136   85 70.000
## 94                 25    318000             1.20          137   85 60.000
## 95                 40    221000             1.10          140   86 63.000
## 96                 25    298000             1.10          141   87 65.000
## 97                 45    263358             1.18          137   87 75.000
## 98                 25    149000             1.10          144   87 80.000
## 99                 30    226000             1.00          140   87 42.000
## 100                50    286000             2.30          143   87 60.000
## 101                30    621000             1.70          138   88 72.000
## 102                45    263000             1.30          137   88 55.000
## 103                35    226000             0.90          138   88 45.000
## 104                38    304000             1.10          133   88 63.000
## 105                60    306000             1.20          132   90 85.000
## 106                35    228000             1.20          135   90 55.000
## 107                25    252000             1.60          136   90 50.000
## 108                60    351000             1.30          137   90 70.000
## 109                40    328000             1.20          126   91 60.000
## 110                40    164000             1.00          139   91 58.000
## 111                60    271000             0.70          136   94 60.000
## 112                60    203000             0.90          140   94 65.000
## 113                38    263358             1.83          134   95 86.000
## 114                60    210000             1.50          135   95 60.000
## 115                38    162000             1.00          136   95 66.000
## 116                38    228000             0.75          140   95 60.000
## 117                30    127000             0.90          145   95 60.000
## 118                40    217000             3.70          134   96 60.000
## 119                50    237000             1.30          135   97 43.000
## 120                17    271000             2.10          124  100 46.000
## 121                60    300000             0.80          137  104 58.000
## 122                30    267000             0.70          136  104 61.000
## 123                35    227000             3.40          145  105 53.000
## 124                60    249000             0.70          138  106 53.000
## 125                45    250000             6.10          131  107 60.000
## 126                40    263358             1.18          137  107 46.000
## 127                60    295000             1.30          145  107 63.000
## 128                35    231000             1.18          137  107 81.000
## 129                40    263358             1.18          137  107 75.000
## 130                60    172000              0.90          137  107 65.000
## 131                25    305000              2.10          130  108 68.000
## 132                35    221000              1.00          136  108 62.000
## 133                30    211000              0.80          138  108 50.000
## 134                38    263358              1.10          134  109 80.000
## 135                35    348000              0.90          140  109 46.000
## 136                30    329000              0.90          132  109 50.000
## 137                40    229000              0.90          141  110 61.000
## 138                25    338000              1.70          139  111 72.000
## 139                30    266000              0.70          141  112 50.000
## 140                30    218000              0.70          136  112 52.000
## 141                60    242000              1.00          137  113 64.000
## 142                30    225000              1.83          134  113 75.000
## 143                35    228000              0.90          136  115 60.000
## 144                45    235000              2.50          135  115 72.000
## 145                60    244000              0.90          139  117 62.000
## 146                45    184000              0.90          134  118 50.000
## 147                35    263358              1.18          137  119 50.000
## 148                35    235000              0.80          136  120 65.000
## 149                25    194000              1.70          140  120 60.000
## 150                35    277000              1.40          136  120 52.000
## 151                25    262000              1.00          136  120 50.000
## 152                50    235000              1.30          134  121 85.000
## 153                45    362000              1.10          139  121 59.000
## 154                40    242000              1.20          134  121 66.000
## 155                35    174000              0.80          139  121 45.000
## 156                40    448000              0.90          137  123 63.000
## 157                35     75000              0.90          142  126 50.000
## 158                30    334000              1.10          139  129 45.000
## 159                38    192000              1.30          135  130 80.000
## 160                60    220000              0.70          133  134 53.000
## 161                20     70000              2.40          134  135 59.000
## 162                40    270000              1.00          138  140 65.000
## 163                35    305000              0.80          133  145 70.000
## 164                35    263358              1.50          136  145 51.000
## 165                40    325000              0.90          140  146 52.000
## 166                60    176000              1.10          145  146 70.000
## 167                20    189000              0.80          139  146 50.000
## 168                35    281000              0.90          137  146 65.000
## 169                60    337000              1.00          138  146 60.000
## 170                40    105000              1.00          135  147 69.000
## 171                50    132000              1.00          140  147 49.000
## 172                60    267000              1.20          145  147 63.000
## 173                40    279000              0.70          140  147 55.000
## 174                30    303000              0.90          136  148 40.000
## 175                25    221000              1.00          136  150 59.000
## 176                25    265000              1.20          136  154 65.000
## 177                38    224000              2.50          134  162 75.000
## 178                25    219000              1.20          137  170 58.000
## 179                30    389000              1.50          136  171 60.667
## 180                50    153000              0.60          134  172 50.000
## 181                25    365000              2.10          144  172 60.000
## 182                40    201000              1.00          136  172 60.667
## 183                45    275000              0.90          140  174 40.000
## 184                35    350000             2.10          134  174 80.000
## 185                60    309000             1.50          135  174 64.000
## 186                40    260000             0.70          130  175 50.000
## 187                30    160000             1.18          142  180 73.000
## 188                20    126000             1.60          135  180 45.000
## 189                45    223000             1.80          145  180 77.000
## 190                38    263358             1.18          137  185 45.000
## 191                30    259000             0.80          138  186 65.000
## 192                20    279000             1.00          134  186 50.000
## 193                35    263358             1.80          113  186 60.000
## 194                45     73000             0.70          137  186 63.000
## 195                60    377000             1.00          136  186 45.000
## 196                60    220000             0.90          138  186 70.000
## 197                25    212000             3.50          136  187 60.000
## 198                40    277000             0.70          137  187 78.000
## 199                45    362000             1.00          136  187 50.000
## 200                40    226000             0.80          141  187 40.000
## 201                38    186000             0.90          136  187 85.000
## 202                40    283000             1.00          141  187 60.000
## 203                35    268000             0.80          130  187 49.000
## 204                17    389000             1.00          136  188 70.000
## 205                62    147000             0.80          140  192 50.000
## 206                50    481000             1.40          138  192 78.000
## 207                30    244000             1.60          130  193 48.000
## 208                35    290000             0.80          134  194 65.000
## 209                35    203000             1.30          134  195 73.000
## 210                50    358000             0.90          141  196 70.000
## 211                35    271000             1.10          134  197 68.000
## 212                35    371000             0.70          140  197 55.000
## 213                20    263358             1.83          134  198 73.000
## 214                50    194000             1.10          145  200 65.000
## 215                35    365000             1.10          139  201 42.000
## 216                25    130000             0.80          134  201 47.000
## 217                25    504000             1.00          138  205 58.000
## 218                25    189000             1.30          132  205 58.000
## 219                35    141000             1.00          140  206 55.000
## 220                25    237000             5.00          130  207 65.000
## 221                25    274000             1.20          134  207 72.000
## 222                30     62000             1.70          127  207 60.000
## 223                35    185000             1.10          134  208 70.000
## 224                35    255000             0.90          137  209 40.000
## 225                38    330000             1.40          137  209 53.000
## 226                45    305000             1.10          137  209 53.000
## 227                50    406000             1.10          137  209 77.000
## 228                50    248000             1.10          148  209 75.000
## 229                30    173000             1.20          132  210 70.000
## 230                40    257000             1.00          136  210 65.000
## 231                45    263358             1.18          137  211 55.000
## 232                35    533000             1.30          139  212 70.000
## 233                30    249000             1.30          136  212 65.000
## 234                35    255000             1.10          136  212 40.000
## 235                40    220000             0.90          141  213 73.000
## 236                38    264000             1.80          134  213 54.000
## 237                38    282000             1.40          137  213 61.000
## 238                25    314000             1.10          138  214 55.000
## 239                25    246000             2.40          135  214 64.000
## 240                35    301000             1.00          142  214 40.000
## 241                40    223000             1.20          130  214 53.000
## 242                30    404000             0.50          139  214 50.000
## 243                35    231000             0.80          143  215 55.000
## 244                45    274000             1.00          133  215 50.000
## 245                35    236000             1.20          132  215 70.000
## 246                60    263358             1.00          139  215 53.000
## 247                30    334000             1.00          142  216 52.000
## 248                38    294000             1.70          139  220 65.000
## 249                38    253000             1.00          139  230 58.000
## 250                25    233000             0.80          135  230 45.000
## 251                50    308000             0.70          135  231 53.000
## 252                40    203000             1.00          138  233 55.000
## 253                40    283000             0.70          133  233 62.000
## 254                25    198000             1.40          129  235 65.000
## 255                60    208000             1.00          140  237 68.000
## 256                38    147000             1.20          141  237 61.000
## 257                35    362000             0.90          140  240 50.000
## 258                20    263358             1.83          134  241 55.000
## 259                38    133000             1.70          140  244 56.000
## 260                38    302000             0.90          140  244 45.000
## 261                35    222000             1.00          132  244 40.000
## 262                30    263358             1.60          130  244 44.000
## 263                40    221000             0.90          134  244 51.000
## 264                38    215000             1.20          133  245 67.000
## 265                40    189000             0.70          140  245 42.000
## 266                30    150000             1.00          137  245 60.000
## 267                38    422000             0.80          137  245 45.000
## 268                35    327000             1.10          142  245 70.000
## 269                38     25100             1.10          140  246 70.000
## 270                30    232000             0.70          136  246 50.000
## 271                38    451000             1.30          136  246 55.000
## 272                40    241000             1.00          137  247 70.000
## 273                40     51000             2.70          136  250 70.000
## 274                30    215000             3.80          128  250 42.000
## 275                38    263358             1.10          138  250 65.000
## 276                40    279000             0.80          141  250 50.000
## 277                40    336000             1.20          135  250 55.000
## 278                35    279000             1.70          140  250 60.000
## 279                35    263358             1.10          142  256 65.000
## 280                38    390000             0.90          144  256 90.000
## 281                55    222000             0.80          141  257 45.000
## 282                35    133000             1.40          139  258 60.000
## 283                38    382000             1.00          140  258 52.000
## 284                35    179000             0.90          136  270 63.000
## 285                38    155000             1.10          143  270 62.000
## 286                38    270000             1.20          139  271 55.000
## 287                38    140000             1.40          140  280 45.000
## 288                45    395000             1.60          136  285 50.000
##     creatinine_phosphokinase
## 1                        582
## 2                       7861
## 3                        146
## 4                        111
## 5                        160
## 6                         47
## 7                        246
## 8                        157
## 9                        123
## 10                        81
## 11                       231
## 12                       981
## 13                       168
## 14                        80
## 15                       379
## 16                       149
## 17                       582
## 18                       125
## 19                        52
## 20                       128
## 21                       220
## 22                        63
## 23                       582
## 24                       148
## 25                       112
## 26                       122
## 27                        60
## 28                        70
## 29                       582
## 30                        23
## 31                       249
## 32                       159
## 33                        94
## 34                       582
## 35                        60
## 36                       855
## 37                      2656
## 38                       235
## 39                       582
## 40                       124
## 41                       571
## 42                       127
## 43                       588
## 44                       582
## 45                      1380
## 46                       582
## 47                       553
## 48                       129
## 49                       577
## 50                        91
## 51                        69
## 52                       260
## 53                       371
## 54                        75
## 55                       607
## 56                       789
## 57                       364
## 58                       318
## 59                       109
## 60                       582
## 61                        68
## 62                       250
## 63                       110
## 64                       161
## 65                       113
## 66                       148
## 67                       582
## 68                      5882
## 69                       224
## 70                       582
## 71                        47
## 72                        92
## 73                       102
## 74                       203
## 75                       336
## 76                        69
## 77                       582
## 78                        76
## 79                        55
## 80                       280
## 81                        78
## 82                        47
## 83                        68
## 84                        84
## 85                       115
## 86                        66
## 87                       897
## 88                       582
## 89                       154
## 90                       144
## 91                       133
## 92                       514
## 93                        59
## 94                       156
## 95                        61
## 96                       305
## 97                       582
## 98                       898
## 99                      5209
## 100                       53
## 101                      328
## 102                      748
## 103                     1876
## 104                      936
## 105                      129
## 106                       60
## 107                      369
## 108                      143
## 109                      754
## 110                      400
## 111                       96
## 112                      113
## 113                      582
## 114                      737
## 115                       68
## 116                       96
## 117                      582
## 118                      582
## 119                      358
## 120                      168
## 121                      200
## 122                      248
## 123                      270
## 124                     1808
## 125                     1082
## 126                      719
## 127                      193
## 128                     4540
## 129                      582
## 130                       59
## 131                      646
## 132                      281
## 133                     1548
## 134                      805
## 135                      291
## 136                      482
## 137                       84
## 138                      943
## 139                      185
## 140                      132
## 141                     1610
## 142                      582
## 143                     2261
## 144                      233
## 145                       30
## 146                      115
## 147                     1846
## 148                      335
## 149                      231
## 150                       58
## 151                      250
## 152                      910
## 153                      129
## 154                       72
## 155                      130
## 156                      582
## 157                     2334
## 158                     2442
## 159                      776
## 160                      196
## 161                       66
## 162                      582
## 163                      835
## 164                      582
## 165                     3966
## 166                      171
## 167                      115
## 168                      198
## 169                       95
## 170                     1419
## 171                       69
## 172                      122
## 173                      835
## 174                      478
## 175                      176
## 176                      395
## 177                       99
## 178                      145
## 179                      104
## 180                      582
## 181                     1896
## 182                      151
## 183                      244
## 184                      582
## 185                       62
## 186                      121
## 187                      231
## 188                      582
## 189                      418
## 190                      582
## 191                      167
## 192                      582
## 193                     1211
## 194                     1767
## 195                      308
## 196                       97
## 197                       59
## 198                       64
## 199                      167
## 200                      101
## 201                      212
## 202                     2281
## 203                      972
## 204                      212
## 205                      582
## 206                      224
## 207                      131
## 208                      135
## 209                      582
## 210                     1202
## 211                     1021
## 212                      582
## 213                      582
## 214                      118
## 215                       86
## 216                      582
## 217                      582
## 218                       57
## 219                     2794
## 220                       56
## 221                      211
## 222                      166
## 223                       93
## 224                      129
## 225                      707
## 226                      582
## 227                      109
## 228                      119
## 229                      232
## 230                      720
## 231                      180
## 232                       81
## 233                      582
## 234                       90
## 235                     1185
## 236                      582
## 237                       80
## 238                     2017
## 239                      143
## 240                      624
## 241                      207
## 242                     2522
## 243                      572
## 244                      245
## 245                       88
## 246                      446
## 247                      191
## 248                      326
## 249                      132
## 250                       66
## 251                       56
## 252                       66
## 253                      655
## 254                      258
## 255                      157
## 256                      582
## 257                      298
## 258                     1199
## 259                      135
## 260                      582
## 261                      582
## 262                      582
## 263                      582
## 264                      213
## 265                       64
## 266                      257
## 267                      582
## 268                      618
## 269                      582
## 270                     1051
## 271                       84
## 272                     2695
## 273                          582
## 274                           64
## 275                         1688
## 276                           54
## 277                          170
## 278                          253
## 279                          892
## 280                          337
## 281                          615
## 282                          320
## 283                          190
## 284                          103
## 285                           61
## 286                         1820
## 287                         2413
## 288                          196
```

Other Ways of Removing Outliers

```
outliers <- function(x) {

  Q1 <- quantile(x, probs=.25)
  Q3 <- quantile(x, probs=.75)
  iqr = Q3-Q1

  upper_limit = Q3 + (iqr*1.5)
  lower_limit = Q1 - (iqr*1.5)

  x > upper_limit | x < lower_limit
}

remove_outliers <- function(heart_data, cols = names(heart_data)) {
  for (col in cols) {
    heart_data <- heart_data[!outliers(heart_data[[col]]),]
  }
  heart_data
}

heart_data_without_outliers <- remove_outliers(heart_data)
boxplot(heart_data_without_outliers)
```



```
mode2 <- lm(ejection_fraction ~ serum_sodium + age + platelets + serum_creatinine + time +
              creatinine_phosphokinase , data = heart_data_without_outliers)
mode2
```

```
##
## Call:
## lm(formula = ejection_fraction ~ serum_sodium + age + platelets +
##     serum_creatinine + time + creatinine_phosphokinase, data = heart_data_without_outliers)
##
## Coefficients:
##              (Intercept)              serum_sodium                       age
##               -2.268e+01                 4.174e-01                 1.429e-01
##                platelets          serum_creatinine                      time
##                1.336e-05                -7.241e+00                 8.623e-03
## creatinine_phosphokinase
##               -4.521e-03
```

```
summary(mode2)
```

```
##
## Call:
## lm(formula = ejection_fraction ~ serum_sodium + age + platelets +
##     serum_creatinine + time + creatinine_phosphokinase, data = heart_data_without_outliers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.7044  -7.3759  -0.9897   6.6674  28.7599
```

```
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -2.268e+01  3.123e+01  -0.726  0.46852
## serum_sodium              4.174e-01  2.193e-01   1.903  0.05834 .
## age                       1.429e-01  6.778e-02   2.108  0.03616 *
## platelets                 1.336e-05  1.126e-05   1.186  0.23692
## serum_creatinine         -7.241e+00  2.462e+00  -2.941  0.00363 **
## time                      8.623e-03  1.032e-02   0.835  0.40455
## creatinine_phosphokinase -4.522e-03  2.758e-03  -1.639  0.10262
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.15 on 213 degrees of freedom
## Multiple R-squared:  0.09792,    Adjusted R-squared:  0.07251
## F-statistic: 3.853 on 6 and 213 DF,  p-value: 0.001136
```

Other Ways of Removing Outliers

To detect the outlier,we can use the influencePlot

##influencePlot(model)

##mode2 <- lm(ejection_fraction ~ serum_sodium + age + platelets + serum_creatinine + time + creatinine_phosphokinase , data = heart_data[!(row.names(heart_data)==“218”&“65”)])

##mode2

Coefficients Function & Fitted function

```
mode2$coefficients
```

```
##              (Intercept)             serum_sodium                      age
##            -2.268009e+01             4.173797e-01             1.429044e-01
##                platelets         serum_creatinine                     time
##             1.335629e-05            -7.240539e+00             8.622597e-03
## creatinine_phosphokinase
##            -4.521499e-03
```

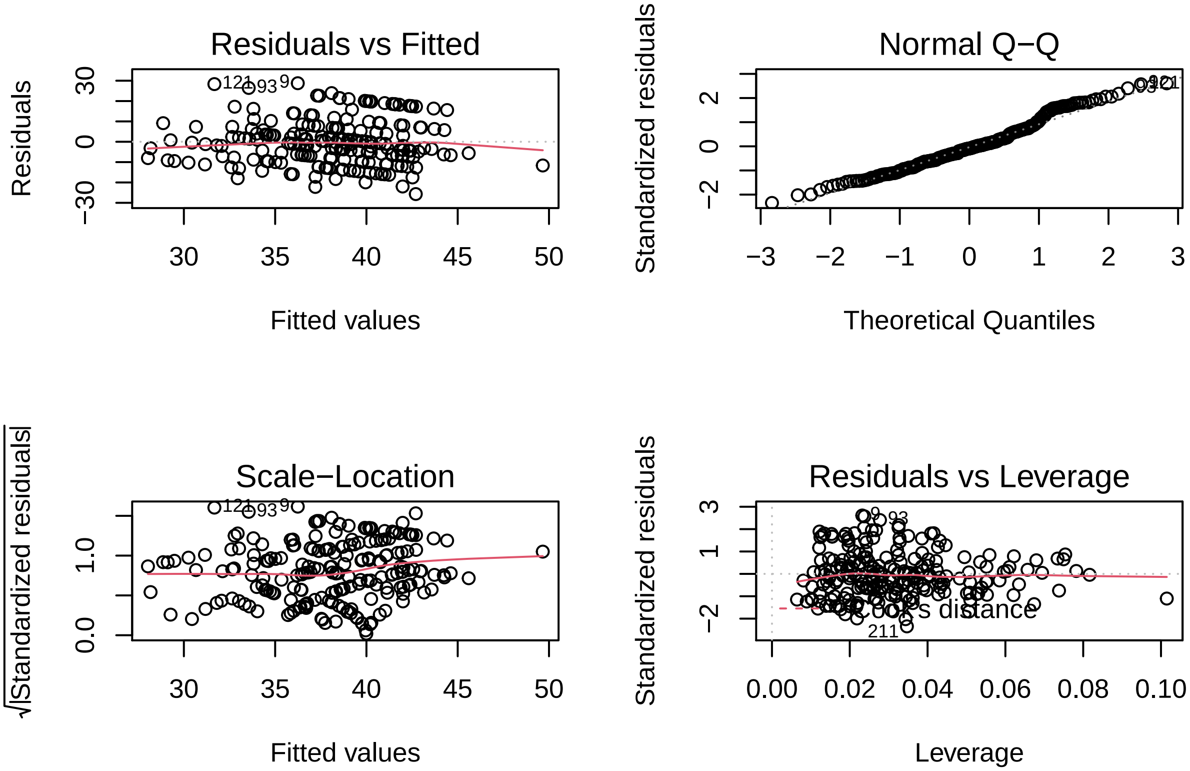
```
mode2$fitted
```

```
##        1        3        4        6        7        9       12       13
## 29.48247 32.60191 30.25242 32.65145 37.20030 36.24012 40.51756 30.44275
##       14       15       17       19       21       22       23       24
## 36.70313 40.12497 44.61563 41.24543 37.96619 35.34799 41.99177 40.26763
##       25       26       27       28       30       31       33       34
## 31.80073 36.61714 43.57205 38.13991 38.02248 34.55040 34.02543 36.93962
##       35       37       38       41       42       43       44       45
## 40.13733 41.87556 42.99775 31.15519 34.29889 36.82403 38.91935 37.41452
##       46       50       51       52       54       58       59       60
## 28.86509 41.71269 37.37428 38.31511 40.76703 39.75428 34.28871 37.21537
##       63       64       67       68       69       71       72       74
## 38.64785 40.06676 32.94782 42.52437 41.00255 40.73213 36.71782 35.97511
##       75       76       77       78       79       80       81       82
## 35.97126 41.98320 44.41998 36.40873 41.87020 40.54261 36.02604 37.07298
##       84       85       86       87       88       89       90       91
## 33.80905 40.80141 43.72398 37.17539 43.68346 39.99521 40.18225 42.01153
##       92       93       94       95       96       97       98       99
## 34.76598 33.55211 32.74708 44.23999 40.25771 34.62319 39.99427 38.66141
##      100      101      102      103      107      109      111      112
## 40.20892 40.84592 38.31106 39.57029 33.83738 34.45677 40.15204 36.38724
##      113      114      116      117      119      120      121      122
## 32.11729 39.90905 37.55009 41.58545 41.53629 33.99350 31.67125 38.95010
##      123      124      126      128      129      133      134      136
## 42.32724 39.78163 32.78149 40.99629 41.07389 33.71984 41.42032 38.48351
##      137      138      139      141      142      143      144      145
## 40.22636 28.17566 38.31552 37.53409 40.08228 36.19749 41.99840 34.52365
##      146      147      149      152      153      155      156      157
## 41.92932 39.72677 32.06444 41.81141 36.83258 40.23864 34.59978 35.84998
##      158      159      160      161      162      166      167      169
## 37.39192 36.05046 41.09749 37.94184 38.75349 35.86248 38.54456 39.14841
##      170      171      173      174      175      176      178      179
## 38.59078 32.64712 42.71511 39.95174 41.38999 41.58246 38.23341 42.43633
##      180      181      182      183      185      186      187      189
## 39.76291 36.44506 38.72372 37.76499 37.83599 38.09215 36.94477 38.99755
##      190      191      192      193      194      195      196      197
## 39.02283 33.01953 37.29833 38.09060 41.12062 29.11545 38.45117 34.86895
##      198      199      202      203      205      206      207      208
## 42.72266 35.85217 38.52021 42.50873 45.60184 39.68054 40.26847 42.85208
##      210      211      212      214      215      216      217      219
## 33.58617 42.70441 38.09326 31.18460 41.68085 36.02933 40.69401 35.70345
##      220      221      222      223      224      227      230      232
## 42.56676 33.02387 42.94626 39.59242 35.01081 35.32403 39.33972 39.13142
##      233      234      235      236      238      239      240      242
## 38.32532 34.95111 37.35455 44.27191 36.80109 38.11963 38.33982 36.48183
##      243      244      245      246      249      250      252      253
## 36.66204 39.50300 30.66382 38.32271 38.10753 34.35230 41.42538 37.14181
##      254      255      256      257      258      259      260      261
## 38.33674 39.02384 42.23819 36.66527 41.14911 39.10123 42.02408 39.95948
##      262      263      264      265      266      267      268      269
## 39.45042 33.81823 42.34182 37.57439 41.93880 28.03260 34.71670 39.17328
##      270      271      272      273      274      275      276      277
## 33.32717 29.27209 36.44457 37.73842 41.03418 38.78861 40.25657 42.31230
##      279      285      286      287      289      290      291      292
## 36.62812 43.16113 38.71197 36.75653 39.60371 49.66009 39.20910 36.32733
##      293      294      295      299
## 42.41121 40.82320 42.02320 36.49087
```

```r
extract_eq(mode2,use_cof=TRUE)
```

$$\text{ejection\_fraction} = \alpha + \beta_{1}(\text{serum\_sodium}) + \beta_{2}(\text{age}) + \beta_{3}(\text{platelets}) + \beta_{4}(\text{serum\_creatinine}) + \beta_{5}(\text{time}) + \beta_{6}(\text{creatinine\_phospho}$$

```r
par(mfrow = c(2, 2))
plot(mode2)
```



We can see here,that the coefficient is different between our new model and our previous model

Also, after removing the outlier from our dataset the residual standard error is decreasing, while the R_ squared is increasing

It means that our new model fits better to our data compared to the previous model

VIF for new model

```
vif(mode2)
```

```
##              serum_sodium                      age                platelets
##                  1.059446                 1.150795                 1.011299
##          serum_creatinine                     time creatinine_phosphokinase
##                  1.158148                 1.097319                 1.040899
```

```
glance(mode2)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1    0.0979        0.0725  11.2      3.85 0.00114     6  -839. 1694. 1722.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
tidy(mode2)
```

```
## # A tibble: 7 x 5
##   term                       estimate  std.error statistic p.value
##   <chr>                         <dbl>      <dbl>     <dbl>   <dbl>
## 1 (Intercept)              -22.7       31.2          -0.726 0.469
## 2 serum_sodium               0.417      0.219         1.90  0.0583
## 3 age                        0.143      0.0678        2.11  0.0362
## 4 platelets                  0.0000134  0.0000113     1.19  0.237
## 5 serum_creatinine          -7.24       2.46         -2.94  0.00363
## 6 time                       0.00862    0.0103        0.835 0.405
## 7 creatinine_phosphokinase  -0.00452    0.00276      -1.64  0.103
```

vif<10 for all variables the model good.

###** ###*****************************
SPLITTING THE DATA ### TRAINING AND TEST SETS *###*
###** ###*****************************
###*

Let’s say we want to split the data in 80:20 for train :test dataset

```
set.seed(299)
ind <- createDataPartition(heart_data$ ejection_fraction,
                           p = 0.8, times = 1, list = FALSE)
length(ind)
```

```
## [1] 241
```

```
train_set <- heart_data[ind, ]
test_set <- heart_data[-ind, ]
nrow(train_set); nrow(test_set)
```

```
## [1] 241
```

```
## [1] 58
```

Training the model ——————————————————————

```
lm_fit <- lm(ejection_fraction ~ . , data = train_set)
broom::tidy(lm_fit)
```

```
## # A tibble: 7 x 5
##   term                        estimate   std.error statistic  p.value
##   <chr>                          <dbl>       <dbl>     <dbl>    <dbl>
## 1 (Intercept)              -39.2        22.4           -1.75  0.0813
## 2 platelets                  0.00000350  0.00000760     0.461 0.645
## 3 serum_creatinine           0.280       0.709          0.395 0.693
## 4 serum_sodium               0.541       0.161          3.36  0.000918
## 5 time                       0.00760     0.00979        0.777 0.438
## 6 age                        0.0133      0.0646         0.205 0.838
## 7 creatinine_phosphokinase   0.000392    0.00105        0.372 0.710
```

```
broom::glance(lm_fit)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df  logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl>
## 1    0.0527        0.0284  11.5      2.17  0.0468     6  -927. 1869. 1897.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

We've built a machine learning model and trained it on train_set

###* *** Prediction *** # ——————

```r
pred <- predict(object = lm_fit, newdata = test_set, type = "response")

head(pred)
```

```
##        2        3        7        8       11       28
## 39.43998 32.47492 36.84354 34.54559 35.16405 36.88770
```

**** Model Evaluation *** # ————————-

```r
actual <- test_set$ejection_fraction
mae <- Metrics::mae(actual = actual, predicted = pred)
mse <- Metrics::mse(actual = actual, predicted = pred)
rmse <- Metrics::rmse(actual = actual, predicted = pred)
```

These common metrics are used to evaluate the model.

Table of results

```r
knitr::kable(cbind(mae, mse, rmse))
```

mae	mse	rmse
10.22927	165.1528	12.85118