Body Fat prediction and Feature Engineering in R

Rohan Harchandani, Sourav Saha, Sobin varghese

April 06 2022

Table of Contents

Objective

This is a comprehensive dataset that lists estimates of the percentage of body fat determined by underwater weighing and various body circumference measurements for 252 men. These data were generously supplied by Dr. A. Garth Fisher who gave permission to freely distribute the data and use for non-commercial purposes.

In this presentation we will use Akaike information criterion(AIC) or Bayesan information criterion(BIC) to generate new optimized subset models. We will check how new optimized model and old model how much they are correlated to each other and reliability of new model.

Importing necessary libraries

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.1.3

library(ggplot2)

##
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':
##
##     %+%, alpha

library(DataExplorer)

## Warning: package 'DataExplorer' was built under R version 4.1.3

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.1.3

## corrplot 0.92 loaded

library(rpart)
library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 4.1.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.1.3

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(caTools)
library(RColorBrewer)
library(rattle)

## Warning: package 'rattle' was built under R version 4.1.3

## Loading required package: tibble

## Loading required package: bitops

## Rattle: A free graphical interface for data science with R.
## Version 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
library(psych)
```

Importing Dataset

```
bodyData = read.csv(file.choose())
```

Exploratory data analysis using different methods

Manual method

```
# First five rows from the dataset.
head(bodyData,5)
```

```
##   Density BodyFat Age Weight Height Neck Chest Abdomen   Hip Thigh Knee
Ankle
## 1  1.0708    12.3  23 154.25  67.75 36.2  93.1    85.2  94.5  59.0 37.3
21.9
## 2  1.0853     6.1  22 173.25  72.25 38.5  93.6    83.0  98.7  58.7 37.3
23.4
## 3  1.0414    25.3  22 154.00  66.25 34.0  95.8    87.9  99.2  59.6 38.9
24.0
## 4  1.0751    10.4  26 184.75  72.25 37.4 101.8    86.4 101.2  60.1 37.3
22.8
## 5  1.0340    28.7  24 184.25  71.25 34.4  97.3   100.0 101.9  63.2 42.2
24.0
##   Biceps Forearm Wrist
## 1   32.0    27.4  17.1
## 2   30.5    28.9  18.2
## 3   28.8    25.2  16.6
## 4   32.4    29.4  18.2
## 5   32.2    27.7  17.7
```

```
# Last five rows from the dataset.
tail(bodyData,5)
```

```
##     Density BodyFat Age Weight Height Neck Chest Abdomen   Hip Thigh Knee
Ankle
## 248  1.0736    11.0  70 134.25  67.00 34.9  89.2    83.6  88.8  49.6 34.8
21.5
## 249  1.0236    33.6  72 201.00  69.75 40.9 108.5   105.0 104.5  59.6 40.8
23.2
## 250  1.0328    29.3  72 186.75  66.00 38.9 111.1   111.5 101.7  60.3 37.3
21.5
## 251  1.0399    26.0  72 190.75  70.50 38.9 108.3   101.3  97.8  56.0 41.6
22.7
## 252  1.0271    31.9  74 207.50  70.00 40.8 112.4   108.5 107.1  59.3 42.2
24.6
##     Biceps Forearm Wrist
## 248   25.6    25.7  18.5
## 249   35.2    28.6  20.1
```

```
## 250   31.3    27.2  18.0
## 251   30.5    29.4  19.8
## 252   33.7    30.0  20.9
```

We can see the data is not consistent and we can conclude that data is well seperated.

```
summary(bodyData)
```

```
##     Density         BodyFat          Age            Weight
##  Min.   :0.995   Min.   : 0.00   Min.   :22.00   Min.   :118.5
##  1st Qu.:1.041   1st Qu.:12.47   1st Qu.:35.75   1st Qu.:159.0
##  Median :1.055   Median :19.20   Median :43.00   Median :176.5
##  Mean   :1.056   Mean   :19.15   Mean   :44.88   Mean   :178.9
##  3rd Qu.:1.070   3rd Qu.:25.30   3rd Qu.:54.00   3rd Qu.:197.0
##  Max.   :1.109   Max.   :47.50   Max.   :81.00   Max.   :363.1
##      Height           Neck           Chest           Abdomen
##  Min.   :29.50   Min.   :31.10   Min.   : 79.30   Min.   : 69.40
##  1st Qu.:68.25   1st Qu.:36.40   1st Qu.: 94.35   1st Qu.: 84.58
##  Median :70.00   Median :38.00   Median : 99.65   Median : 90.95
##  Mean   :70.15   Mean   :37.99   Mean   :100.82   Mean   : 92.56
##  3rd Qu.:72.25   3rd Qu.:39.42   3rd Qu.:105.38   3rd Qu.: 99.33
##  Max.   :77.75   Max.   :51.20   Max.   :136.20   Max.   :148.10
##       Hip             Thigh            Knee            Ankle          Biceps
##  Min.   : 85.0   Min.   :47.20   Min.   :33.00   Min.   :19.1   Min.
:24.80
##  1st Qu.: 95.5   1st Qu.:56.00   1st Qu.:36.98   1st Qu.:22.0   1st
Qu.:30.20
##  Median : 99.3   Median :59.00   Median :38.50   Median :22.8   Median
:32.05
##  Mean   : 99.9   Mean   :59.41   Mean   :38.59   Mean   :23.1   Mean
:32.27
##  3rd Qu.:103.5   3rd Qu.:62.35   3rd Qu.:39.92   3rd Qu.:24.0   3rd
Qu.:34.33
##  Max.   :147.7   Max.   :87.30   Max.   :49.10   Max.   :33.9   Max.
:45.00
##     Forearm          Wrist
##  Min.   :21.00   Min.   :15.80
##  1st Qu.:27.30   1st Qu.:17.60
##  Median :28.70   Median :18.30
##  Mean   :28.66   Mean   :18.23
##  3rd Qu.:30.00   3rd Qu.:18.80
##  Max.   :34.90   Max.   :21.40
```

Summary of the dataset. It gives five summary statistic of each variable.

```
dim(bodyData)
```

```
## [1] 252  15
```

Dimension of dataset. Number of rows: 252 and Number of columns: 15.

```
# Structure of dataset.
describe(bodyData)
```

```
##          vars   n   mean    sd median trimmed   mad   min    max  range
skew
## Density     1 252   1.06  0.02   1.05    1.06  0.02   1.0   1.11   0.11 -
0.02
## BodyFat     2 252  19.15  8.37  19.20   19.05  9.27   0.0  47.50  47.50
0.14
## Age         3 252  44.88 12.60  43.00   44.44 11.86  22.0  81.00  59.00
0.28
## Weight      4 252 178.92 29.39 176.50  177.41 28.73 118.5 363.15 244.65
1.19
## Height      5 252  70.15  3.66  70.00   70.27  2.97  29.5  77.75  48.25 -
5.32
## Neck        6 252  37.99  2.43  38.00   37.96  2.37  31.1  51.20  20.10
0.55
## Chest       7 252 100.82  8.43  99.65  100.28  8.38  79.3 136.20  56.90
0.67
## Abdomen     8 252  92.56 10.78  90.95   92.00 10.90  69.4 148.10  78.70
0.83
## Hip         9 252  99.90  7.16  99.30   99.49  5.78  85.0 147.70  62.70
1.48
## Thigh      10 252  59.41  5.25  59.00   59.17  4.60  47.2  87.30  40.10
0.81
## Knee       11 252  38.59  2.41  38.50   38.50  2.22  33.0  49.10  16.10
0.51
## Ankle      12 252  23.10  1.69  22.80   22.98  1.33  19.1  33.90  14.80
2.23
## Biceps     13 252  32.27  3.02  32.05   32.24  2.89  24.8  45.00  20.20
0.28
## Forearm    14 252  28.66  2.02  28.70   28.68  2.08  21.0  34.90  13.90 -
0.22
## Wrist      15 252  18.23  0.93  18.30   18.21  0.89  15.8  21.40   5.60
0.28
```

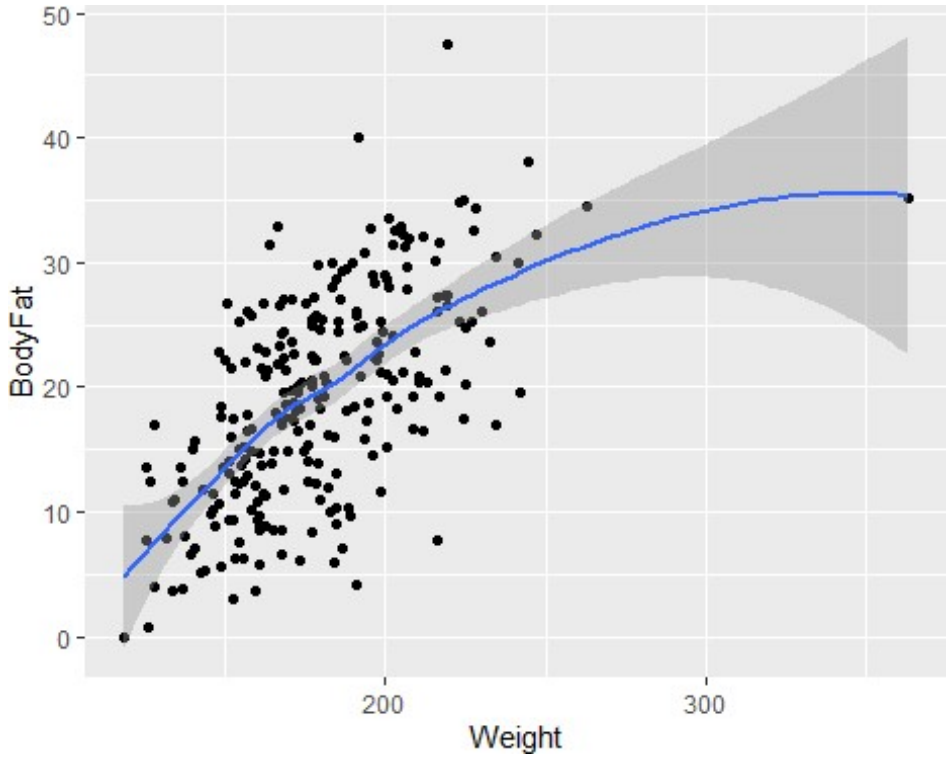
```
##          kurtosis   se
## Density     -0.35 0.00
## BodyFat     -0.37 0.53
## Age         -0.45 0.79
## Weight       5.08 1.85
## Height      57.86 0.23
## Neck         2.60 0.15
## Chest        0.91 0.53
## Abdomen      2.14 0.68
## Hip          7.22 0.45
## Thigh        2.55 0.33
## Knee         0.99 0.15
## Ankle       11.57 0.11
## Biceps       0.44 0.19
```

```
## Forearm     0.80 0.13
## Wrist       0.34 0.06

colnames(bodyData)

##  [1] "Density" "BodyFat"  "Age"     "Weight"   "Height"   "Neck"     "Chest"
##  [8] "Abdomen" "Hip"      "Thigh"   "Knee"     "Ankle"    "Biceps"   "Forearm"
## [15] "Wrist"
```

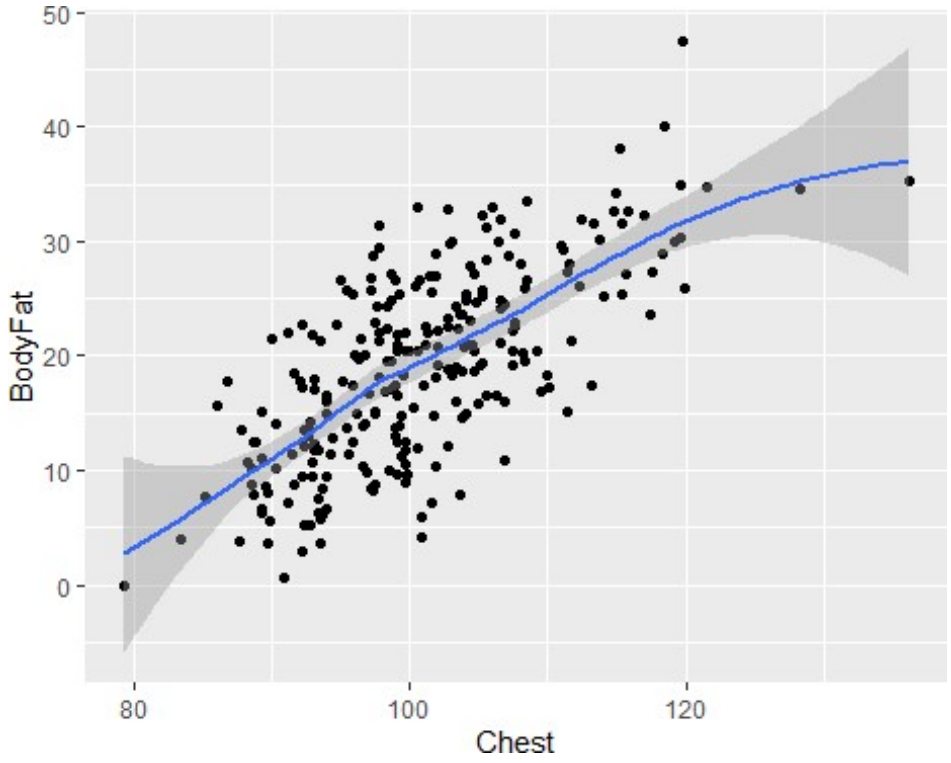
Column names of dataset. “Density” “BodyFat” “Age” “Weight” “Height” “Neck” “Chest” “Abdomen” “Hip” “Thigh” “Knee” “Ankle” “Biceps” “Forearm” “Wrist” “WeightGroup”

Relation graphs of variable.

```
ggplot(bodyData,aes(x = Weight,y = BodyFat))+geom_point()+geom_smooth()

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

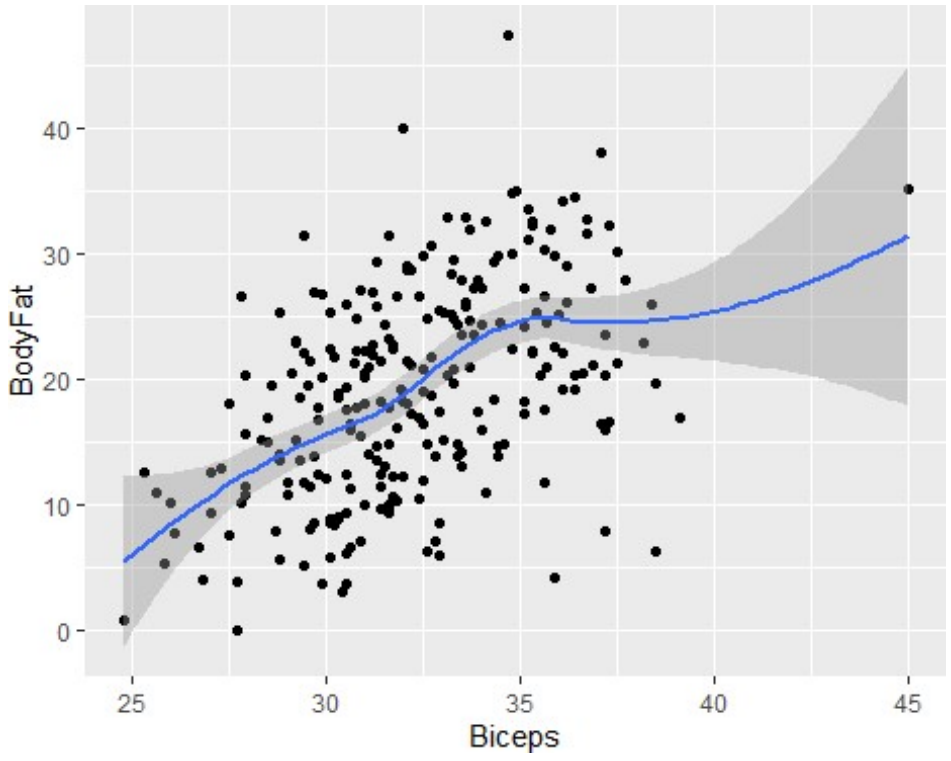



```
ggplot(bodyData,aes(x = Chest,y = BodyFat))+geom_point()+geom_smooth()

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



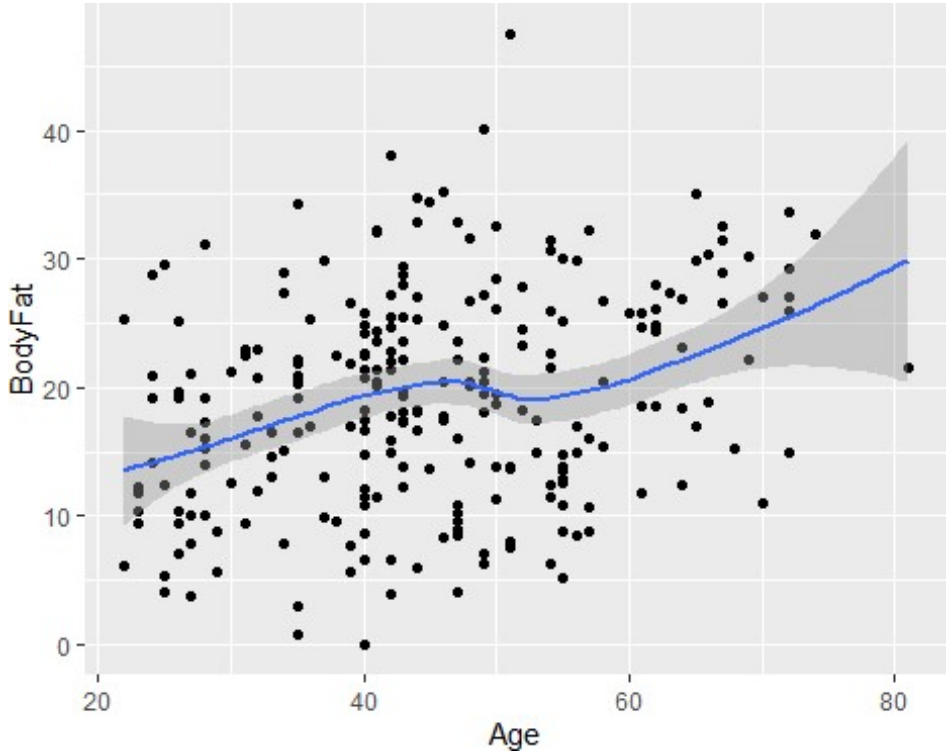

```
ggplot(bodyData,aes(x = Biceps,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



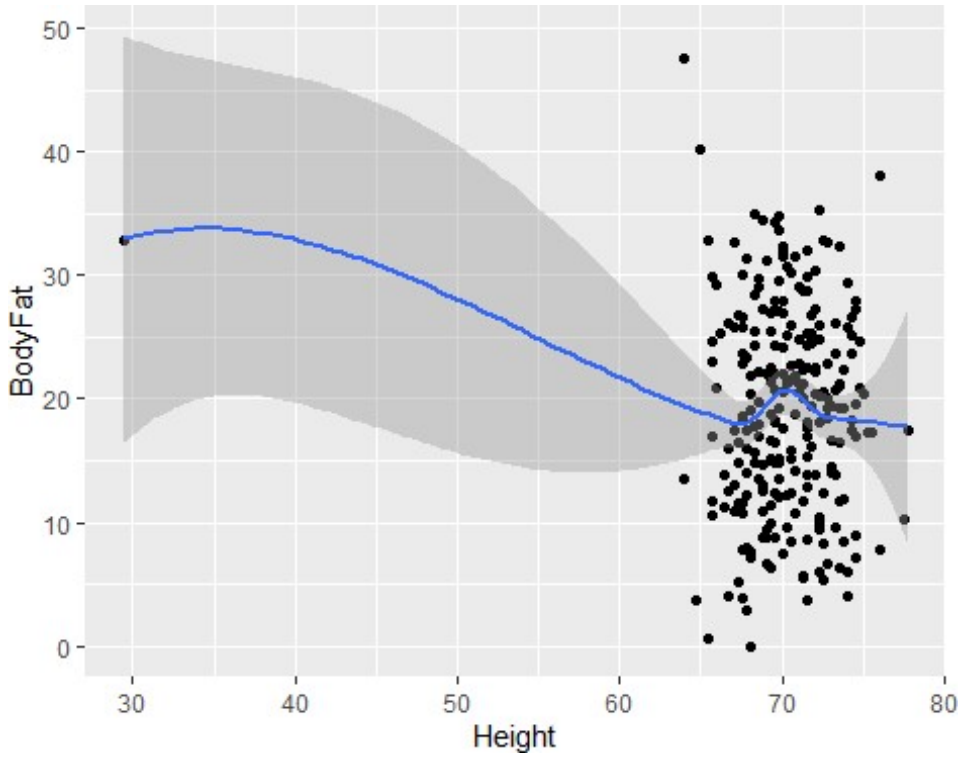
```
ggplot(bodyData,aes(x = Age,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



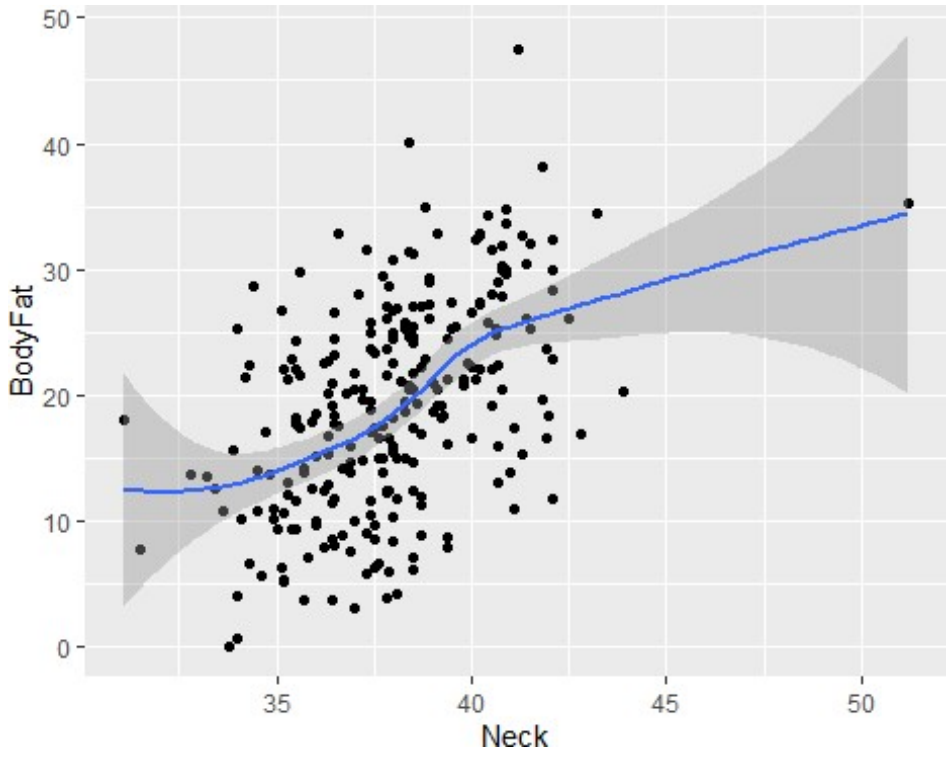

```
ggplot(bodyData,aes(x = Height,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



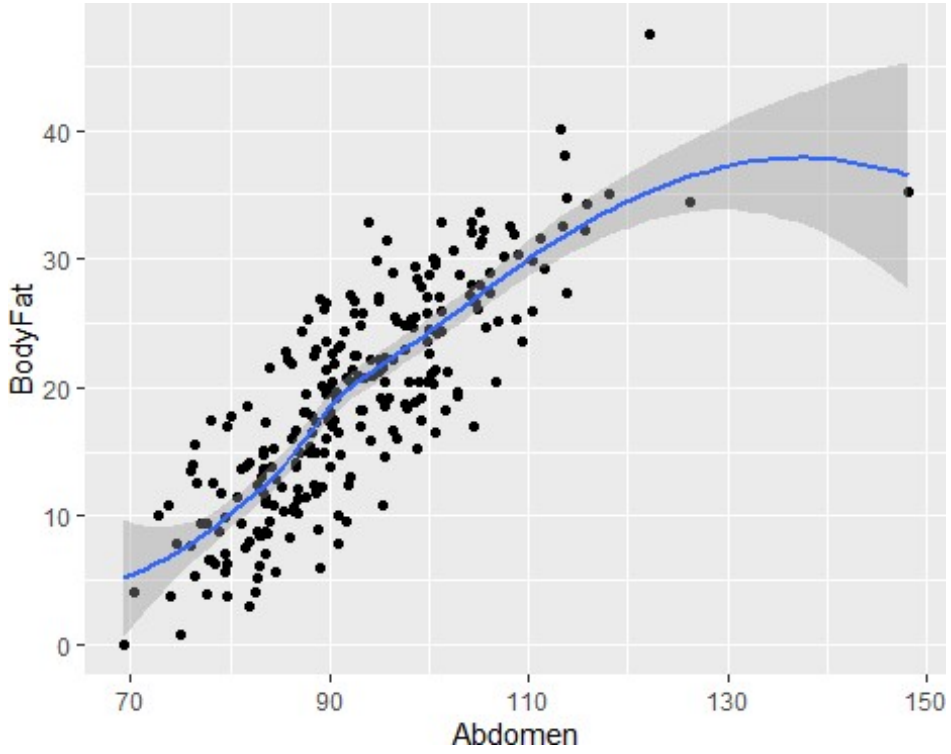

```
ggplot(bodyData,aes(x = Neck,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



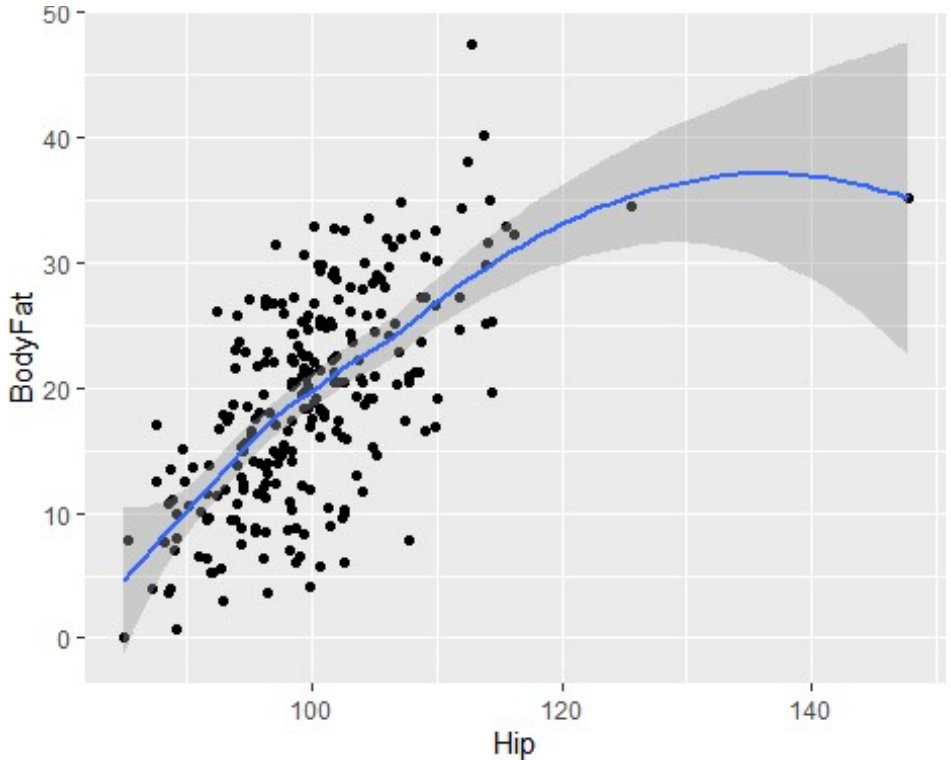
```
ggplot(bodyData,aes(x = Abdomen,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



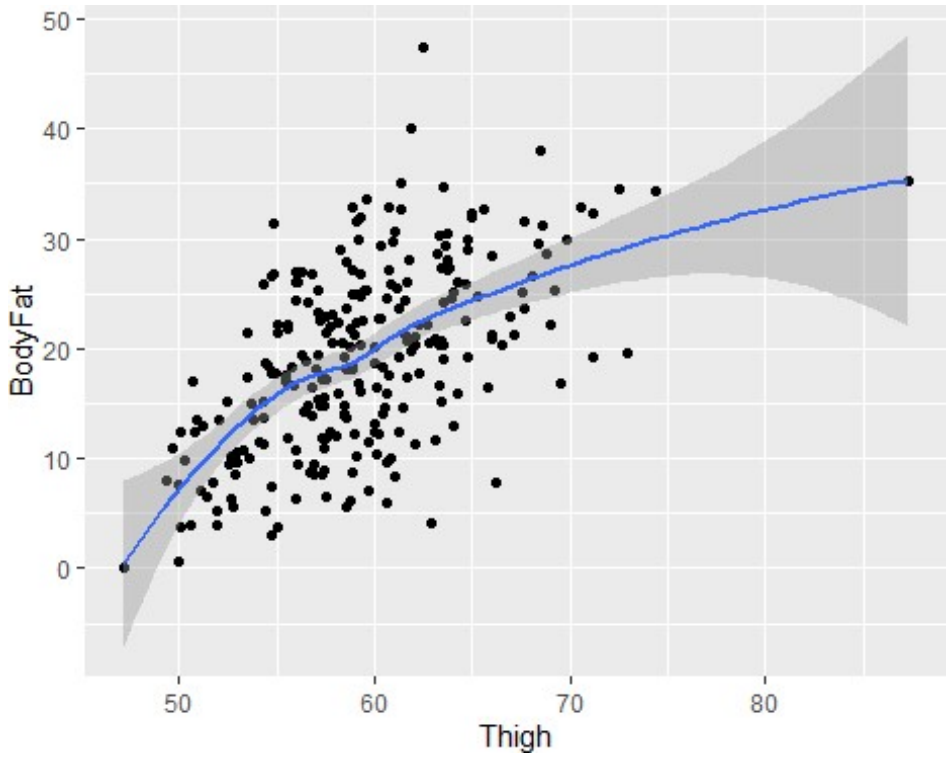

```
ggplot(bodyData,aes(x = Hip,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



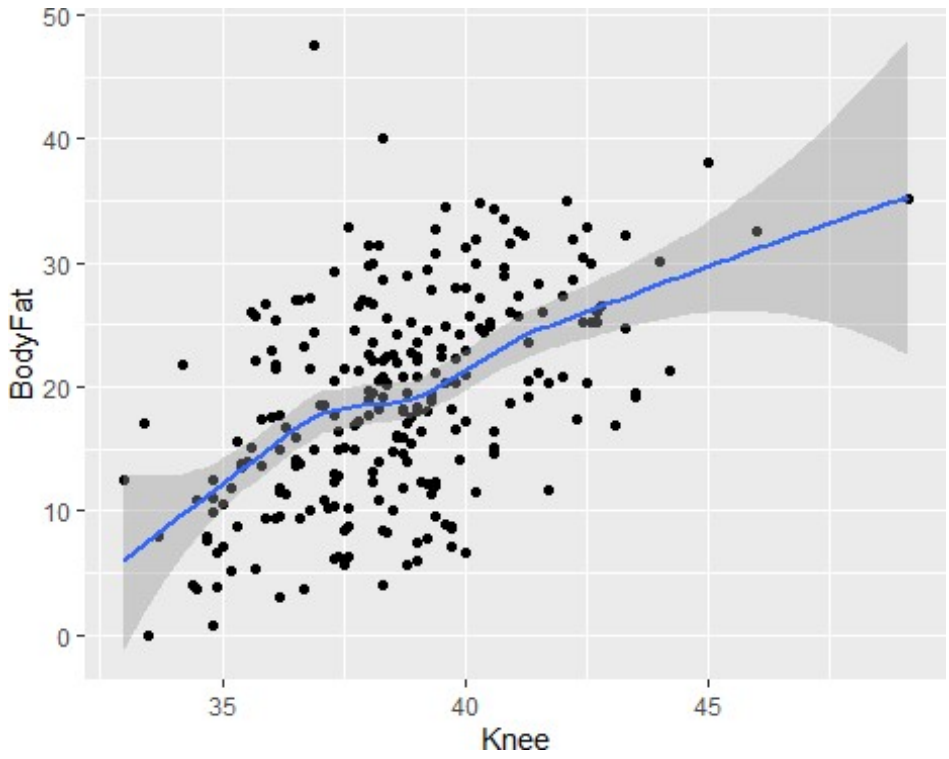

```
ggplot(bodyData,aes(x = Thigh,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



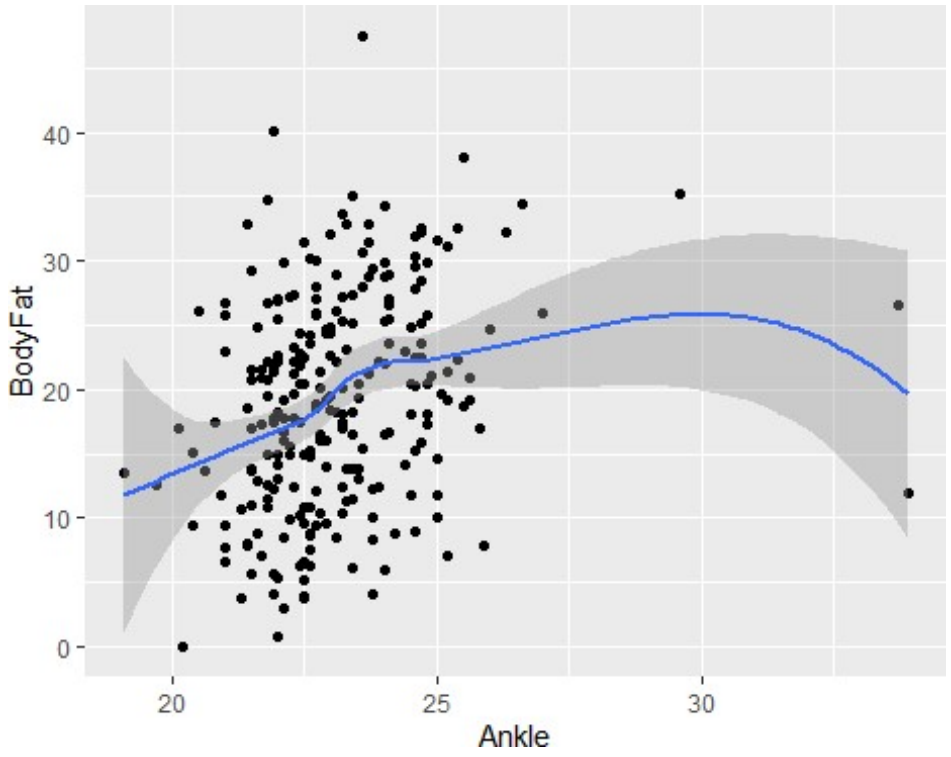
```
ggplot(bodyData,aes(x = Knee,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



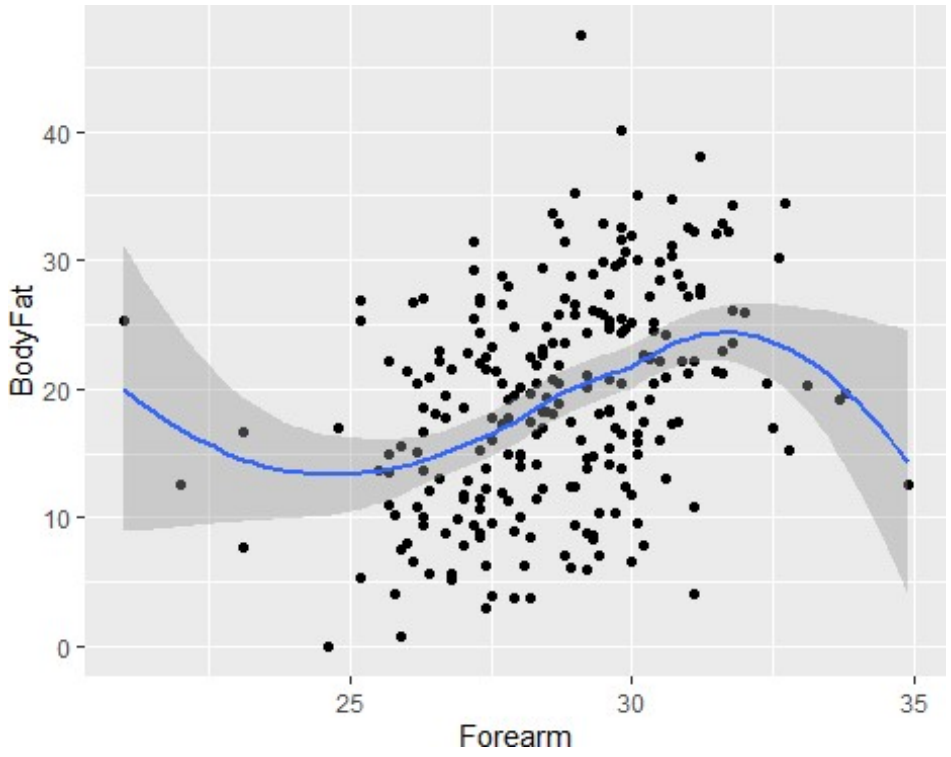

```
ggplot(bodyData,aes(x = Ankle,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



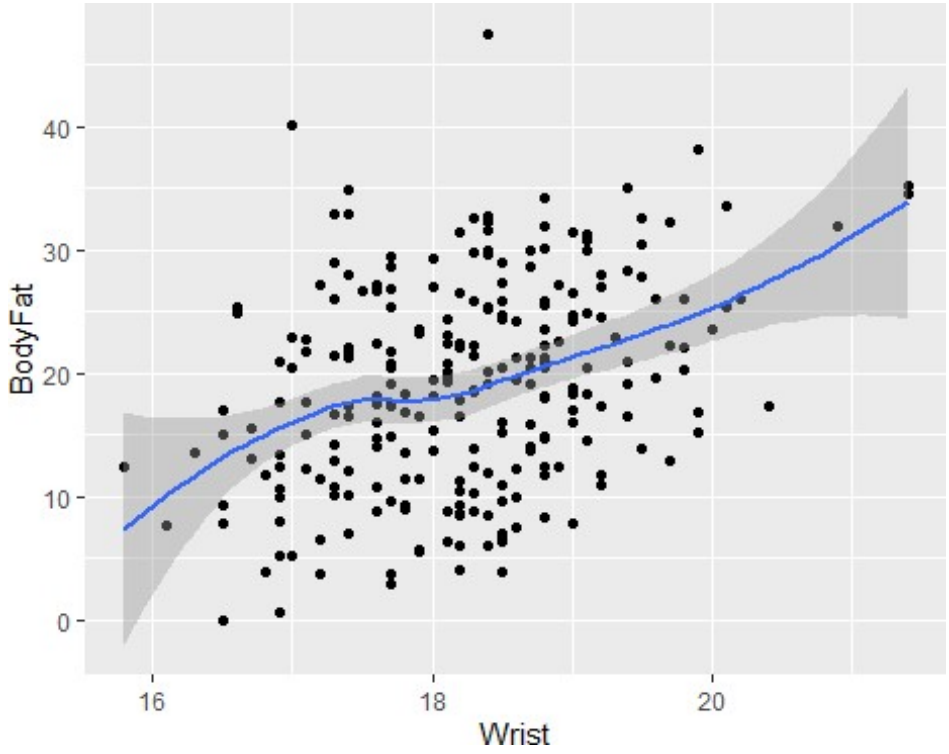

```
ggplot(bodyData,aes(x = Forearm,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



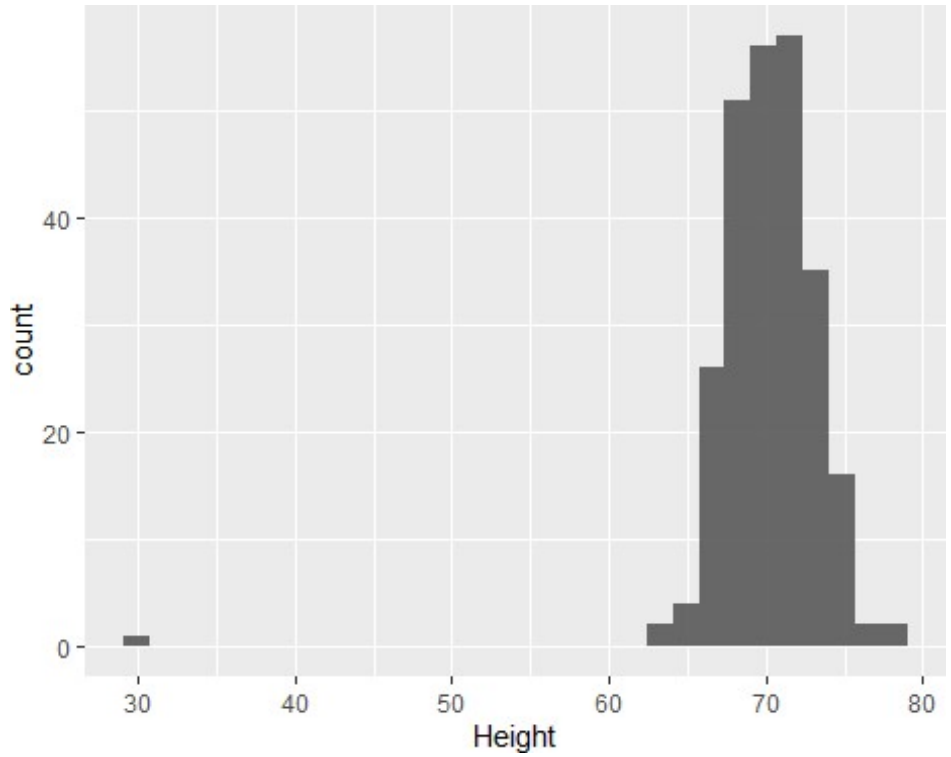
```
ggplot(bodyData,aes(x = Wrist,y = BodyFat))+geom_point()+geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
ggplot(bodyData,aes(x=Height))+geom_histogram(alpha=0.9)+theme(plot.title=ele
ment_text(size=3))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



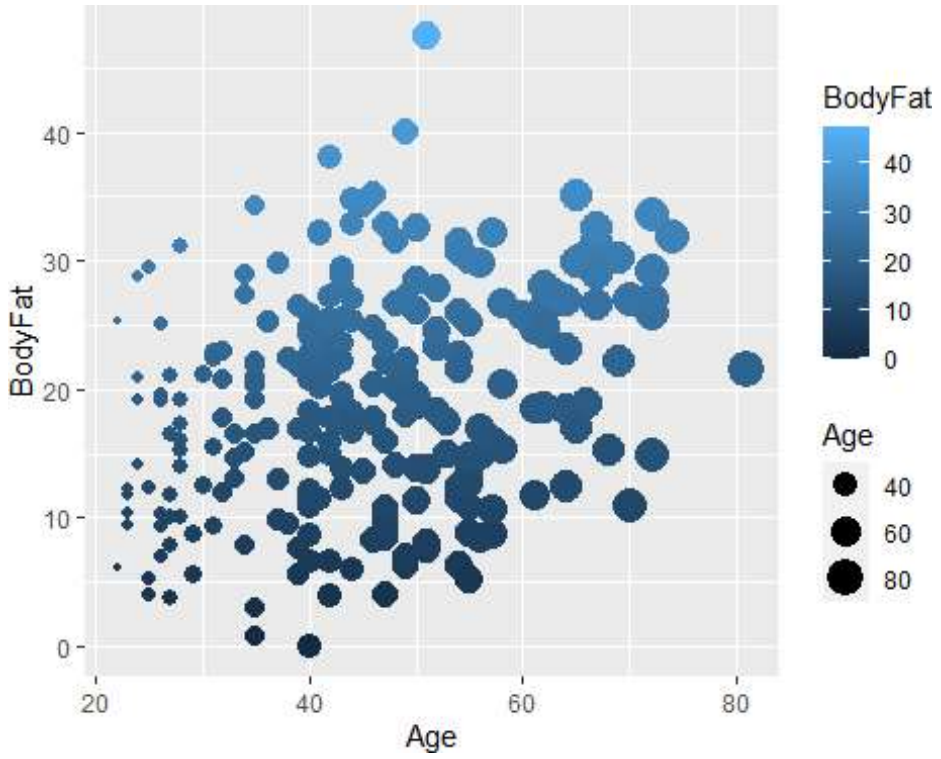

```
length(unique(bodyData$Age))
```

```
## [1] 51
```

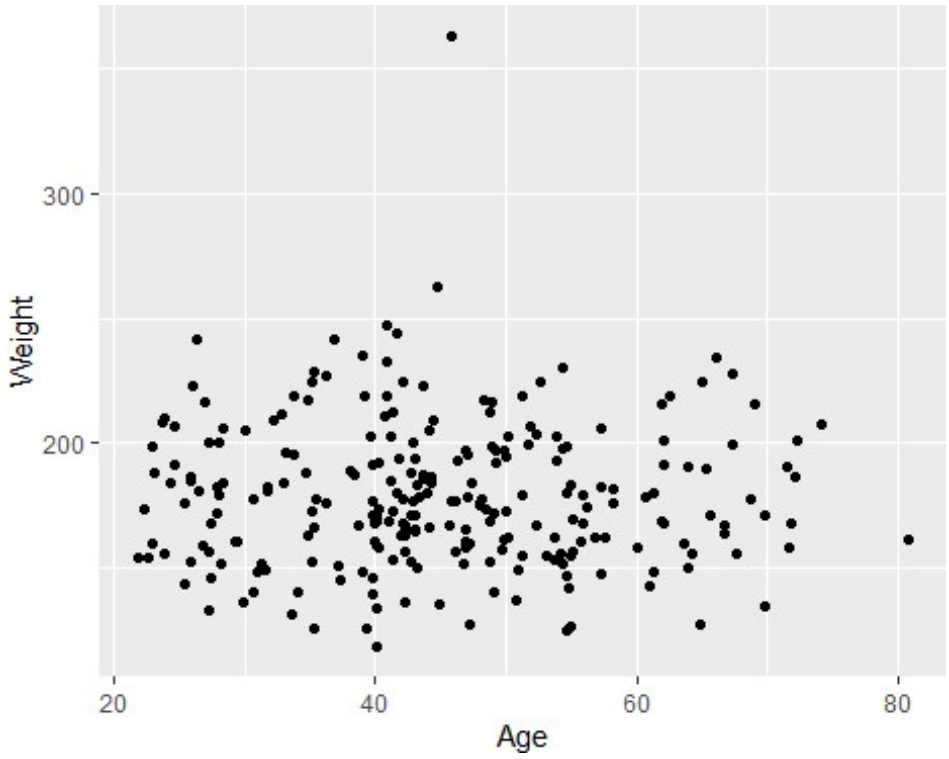
```
unique(bodyData$Age)
```

```
##  [1] 23 22 26 24 25 27 32 30 35 34 28 33 31 29 41 49 40 50 46 45 44 48 39
43 47
## [26] 51 42 54 58 62 61 56 57 55 69 81 66 67 64 70 72 53 38 52 36 37 60 63
65 68
## [51] 74
```

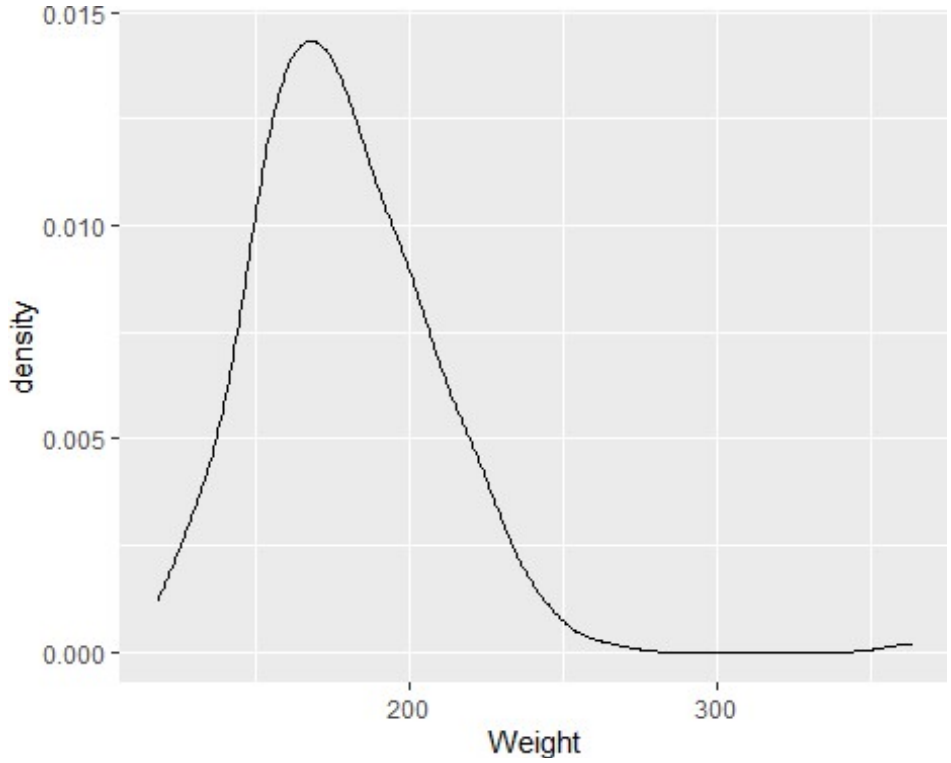
```
ggplot(bodyData,aes(Age,BodyFat))+geom_point(aes(x=Age,y=BodyFat,color=BodyFa
t,size=Age))
```



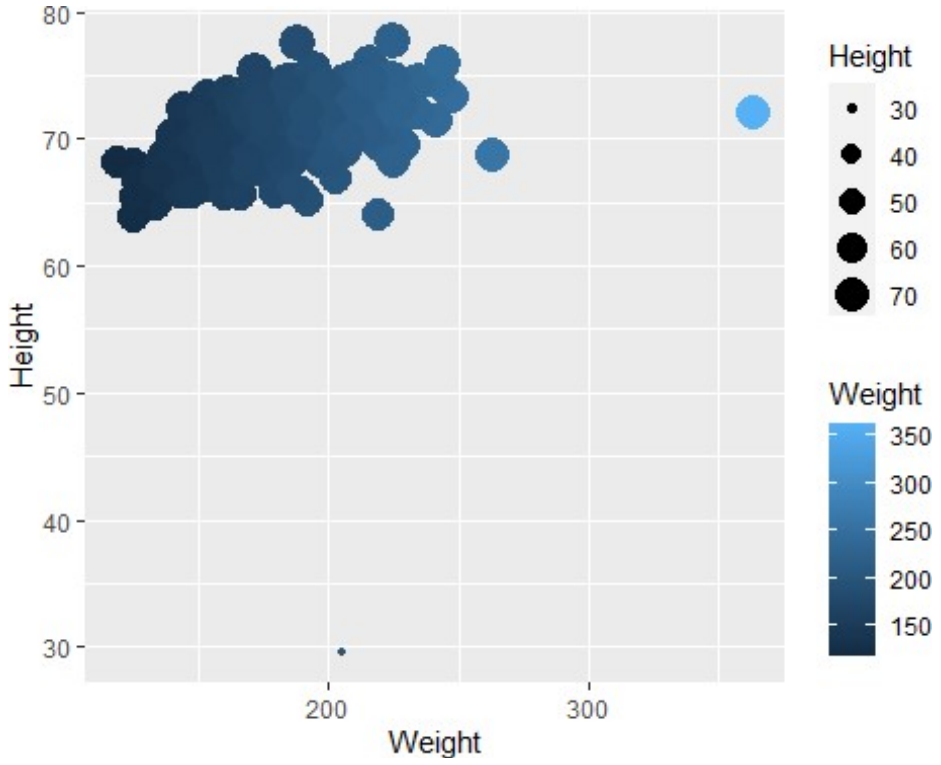

```
ggplot(bodyData,aes(Age,Weight))+geom_jitter() #No correlation b/t
```



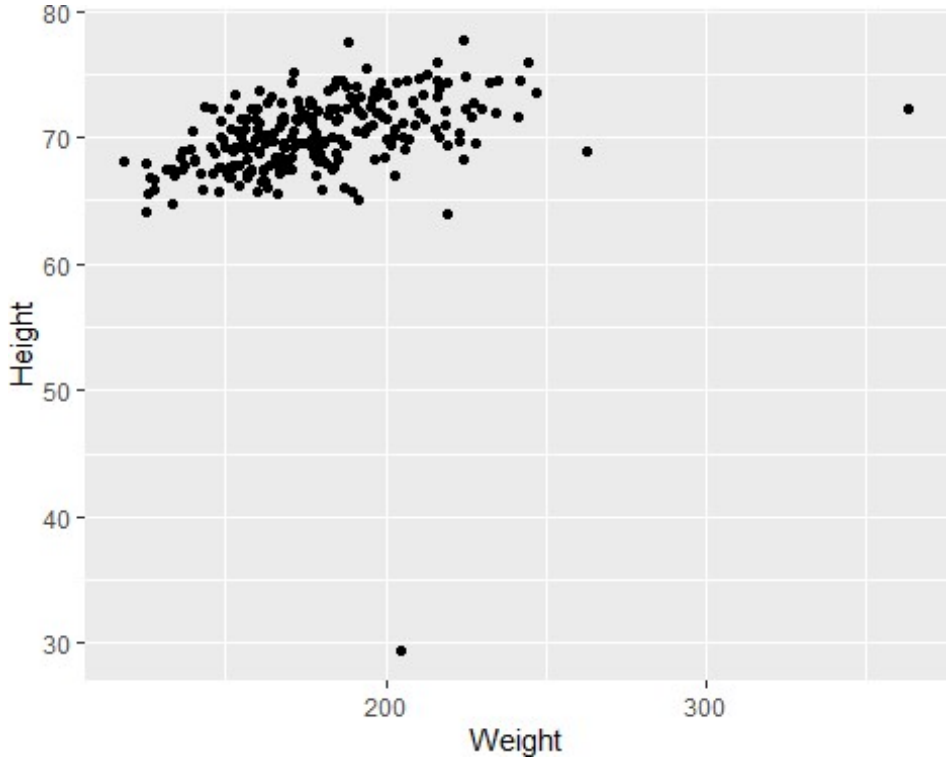

```
ggplot(bodyData) + geom_density(aes(Weight))
```




```
ggplot(bodyData,aes(Weight,Height))+geom_jitter(aes(X=Weight,y=Height,color=W
eight,size=Height))

## Warning: Ignoring unknown aesthetics: X
```




```
ggplot(bodyData,aes(Weight,Height))+geom_jitter()
```



Which patients have the highest and lowest bodyfat. Top5/bottom5

```
bodyData[order(bodyData$BodyFat,decreasing=T)[1:5],]
```

```
##     Density BodyFat Age Weight  Height Neck Chest Abdomen   Hip Thigh Knee
Ankle
## 216  0.9950    47.5  51 219.00  64.00 41.2 119.8   122.1 112.8  62.5 36.9
23.6
## 36   1.0101    40.1  49 191.75  65.00 38.4 118.5   113.1 113.8  61.9 38.3
21.9
## 192  1.0140    38.1  42 244.25  76.00 41.8 115.2   113.7 112.4  68.5 45.0
25.5
## 39   1.0202    35.2  46 363.15  72.25 51.2 136.2   148.1 147.7  87.3 49.1
29.6
## 242  1.0207    35.0  65 224.50  68.25 38.8 119.6   118.0 114.3  61.3 42.1
23.4
##     Biceps Forearm Wrist
## 216   34.7    29.1  18.4
## 36    32.0    29.8  17.0
## 192   37.1    31.2  19.9
## 39    45.0    29.0  21.4
## 242   34.9    30.1  19.4
```

```
bodyData[order(bodyData$BodyFat,decreasing=F)[1:5],]
```

```
##     Density BodyFat Age Weight  Height Neck Chest Abdomen  Hip Thigh Knee
Ankle
## 182  1.1089     0.0  40 118.50  68.00 33.8  79.3    69.4 85.0  47.2 33.5
20.2
## 172  1.0983     0.7  35 125.75  65.50 34.0  90.8    75.0 89.2  50.0 34.8
22.0
## 171  1.0926     3.0  35 152.25  67.75 37.0  92.2    81.9 92.8  54.7 36.2
22.1
## 26   1.0911     3.7  27 159.25  71.50 35.7  89.6    79.7 96.5  55.0 36.7
22.5
## 29   1.0910     3.7  27 133.25  64.75 36.4  93.5    73.9 88.5  50.1 34.5
21.3
##     Biceps Forearm Wrist
## 182   27.7    24.6  16.5
## 172   24.8    25.9  16.9
## 171   30.4    27.4  17.7
## 26    29.9    28.2  17.7
## 29    30.5    27.9  17.2
```

checking if there is any null values in dataset

```
which(is.null(bodyData))
```

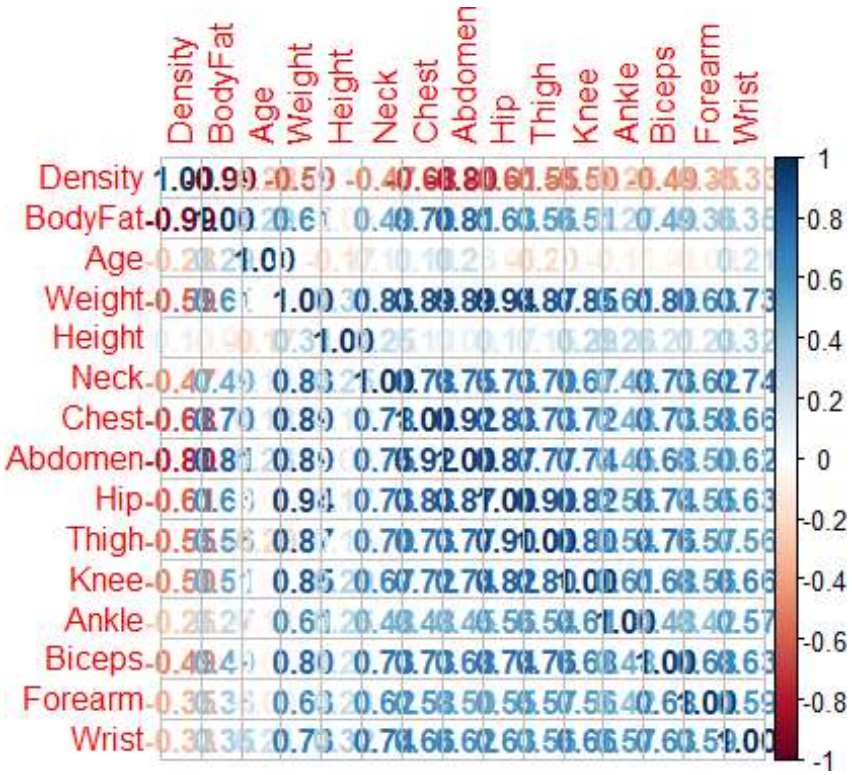
```
## integer(0)
```

there is no null value n the dataset

Correlational matrix and plot

```
cor(bodyData)

##             Density     BodyFat          Age      Weight      Height
Neck
## Density  1.00000000 -0.98778240 -0.27763721 -0.59406188  0.09788114 -
0.4729664
## BodyFat -0.98778240  1.00000000  0.29145844  0.61241400 -0.08949538
0.4905919
## Age     -0.27763721  0.29145844  1.00000000 -0.01274609 -0.17164514
0.1135052
## Weight  -0.59406188  0.61241400 -0.01274609  1.00000000  0.30827854
0.8307162
## Height   0.09788114 -0.08949538 -0.17164514  0.30827854  1.00000000
0.2537099
## Neck    -0.47296636  0.49059185  0.11350519  0.83071622  0.25370988
1.0000000
## Chest   -0.68259865  0.70262034  0.17644968  0.89419052  0.13489181
0.7848350
## Abdomen -0.79895463  0.81343228  0.23040942  0.88799494  0.08781291
0.7540774
## Hip     -0.60933143  0.62520092 -0.05033212  0.94088412  0.17039426
0.7349579
## Thigh   -0.55309098  0.55960753 -0.20009576  0.86869354  0.14843561
0.6956973
## Knee    -0.49504035  0.50866524  0.01751569  0.85316739  0.28605321
0.6724050
## Ankle   -0.26489003  0.26596977 -0.10505810  0.61368542  0.26474369
0.4778924
## Biceps  -0.48710872  0.49327113 -0.04116212  0.80041593  0.20781557
0.7311459
## Forearm -0.35164842  0.36138690 -0.08505555  0.63030143  0.22864922
0.6236603
## Wrist   -0.32571598  0.34657486  0.21353062  0.72977489  0.32206533
0.7448264
##              Chest     Abdomen          Hip      Thigh         Knee
Ankle
## Density -0.6825987 -0.79895463 -0.60933143 -0.5530910 -0.49504035 -
0.2648900
## BodyFat  0.7026203  0.81343228  0.62520092  0.5596075  0.50866524
0.2659698
## Age      0.1764497  0.23040942 -0.05033212 -0.2000958  0.01751569 -
0.1050581
## Weight   0.8941905  0.88799494  0.94088412  0.8686935  0.85316739
0.6136854
## Height   0.1348918  0.08781291  0.17039426  0.1484356  0.28605321
0.2647437
## Neck     0.7848350  0.75407737  0.73495788  0.6956973  0.67240498
0.4778924
## Chest    1.0000000  0.91582767  0.82941992  0.7298586  0.71949640
```

```
0.4829879
## Abdomen  0.9158277  1.00000000  0.87406618  0.7666239  0.73717888
0.4532227
## Hip      0.8294199  0.87406618  1.00000000  0.8964098  0.82347262
0.5583868
## Thigh    0.7298586  0.76662393  0.89640979  1.0000000  0.79917030
0.5397971
## Knee     0.7194964  0.73717888  0.82347262  0.7991703  1.00000000
0.6116082
## Ankle    0.4829879  0.45322269  0.55838682  0.5397971  0.61160820
1.0000000
## Biceps   0.7279075  0.68498272  0.73927252  0.7614774  0.67870883
0.4848545
## Forearm  0.5801727  0.50331609  0.54501412  0.5668422  0.55589819
0.4190500
## Wrist    0.6601623  0.61983243  0.63008954  0.5586848  0.66450729
0.5661946
##              Biceps     Forearm      Wrist
## Density  -0.48710872 -0.35164842 -0.3257160
## BodyFat   0.49327113  0.36138690  0.3465749
## Age      -0.04116212 -0.08505555  0.2135306
## Weight    0.80041593  0.63030143  0.7297749
## Height    0.20781557  0.22864922  0.3220653
## Neck      0.73114592  0.62366027  0.7448264
## Chest     0.72790748  0.58017273  0.6601623
## Abdomen   0.68498272  0.50331609  0.6198324
## Hip       0.73927252  0.54501412  0.6300895
## Thigh     0.76147745  0.56684218  0.5586848
## Knee      0.67870883  0.55589819  0.6645073
## Ankle     0.48485454  0.41904999  0.5661946
## Biceps    1.00000000  0.67825513  0.6321264
## Forearm   0.67825513  1.00000000  0.5855883
## Wrist     0.63212642  0.58558825  1.0000000

corrplot(cor(bodyData),method="number")
```



Creating EDA report using "DataExplorer" library

.

```
create_report(bodyData)

##
##
## processing file: report.rmd

##   |
|                                                                      |   0%
|
|..                                                                    |   2%
##    inline R code fragments
##
##   |
|...                                                                   |   5%
## label: global_options (with options)
## List of 1
##  $ include: logi FALSE
##
##   |
```

```
|.....                                                                 |   7%
##   ordinary text without R code
##
##   |
|.......                                                               |  10%
## label: introduce
##   |
|........                                                              |  12%
##   ordinary text without R code
##
##   |
|..........                                                            |  14%
## label: plot_intro

##   |
|............                                                          |  17%
##   ordinary text without R code
##
##   |
|.............                                                         |  19%
## label: data_structure
##   |
|...............                                                       |  21%
##   ordinary text without R code
##
##   |
|.................                                                     |  24%
## label: missing_profile

##   |
|...................                                                   |  26%
##   ordinary text without  R code
##
##   |
|.....................                                                 |  29%
## label: univariate_distribution_header
##   |
|.......................                                               |  31%
##   ordinary text without  R code
##
##   |
|........................                                              |  33%
## label: plot_histogram

##   |
|..........................                                            |  36%
##   ordinary text without  R code
##
##   |
|............................                                          |  38%
```

```
## label: plot_density
##   |
|.............................                                         |  40%
##   ordinary text without R code
##
##   |
|...............................                                       |  43%
## label: plot_frequency_bar
##   |
|.................................                                     |  45%
##   ordinary text without R code
##
##   |
|...................................                                   |  48%
## label: plot_response_bar
##   |
|.....................................                                 |  50%
##   ordinary text without R code
##
##   |
|.......................................                               |  52%
## label: plot_with_bar
##   |
|.........................................                             |  55%
##   ordinary text without R code
##
##   |
|...........................................                           |  57%
## label: plot_normal_qq

##   |
|.............................................                         |  60%
##   ordinary text without R code
##
##   |
|...............................................                       |  62%
## label: plot_response_qq
##   |
|.................................................                     |  64%
##   ordinary text without R code
##
##   |
|...................................................                   |  67%
## label: plot_by_qq
##   |
|.....................................................                 |  69%
##   ordinary text without R code
##
##   |
```

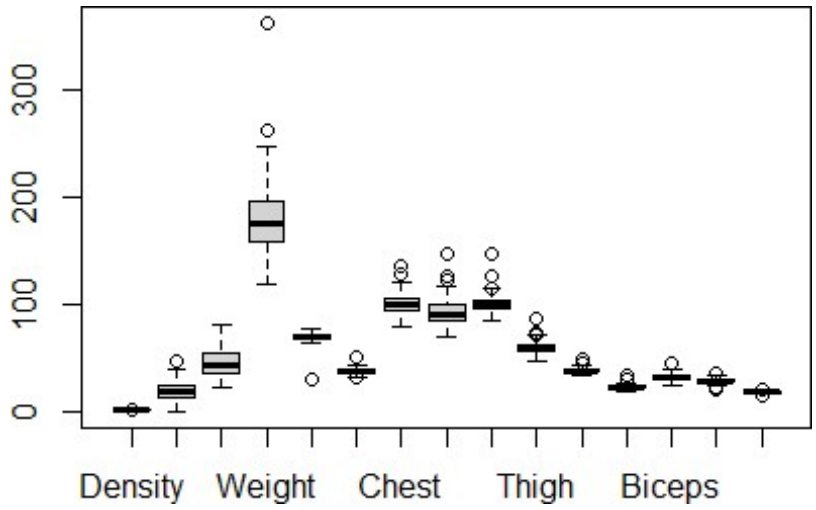
```
|.................................................                     |  71%
## label: correlation_analysis

##   |
|.....................................................                 |  74%
##   ordinary text without R code
##
##   |
|.....................................................                 |  76%
## label: principal_component_analysis

##   |
|.....................................................                 |  79%
##   ordinary text without R code
##
##   |
|.........................................................             |  81%
## label: bivariate_distribution_header
##   |
|.........................................................             |  83%
##   ordinary text without R code
##
##   |
|...........................................................           |  86%
## label: plot_response_boxplot
##   |
|.............................................................         |  88%
##   ordinary text without R code
##
##   |
|...............................................................       |  90%
## label: plot_by_boxplot
##   |
|.................................................................     |  93%
##   ordinary text without R code
##
##   |
|...................................................................   |  95%
## label: plot_response_scatterplot
##   |
|..................................................................... |  98%
##   ordinary text without R code
##
##   |
|.....................................................................| 100%
## label: plot_by_scatterplot

## output file: D:/College Notes/Christ University/sem 2/Data Science using
R/Lab work/report.knit.md
```

```
## "C:/Program Files/RStudio/bin/pandoc/pandoc" +RTS -K512m -RTS "D:/College
Notes/Christ University/sem 2/Data Science using R/Lab work/report.knit.md" -
-to html4 --from markdown+autolink_bare_uris+tex_math_single_backslash --
output pandoc1ee4200d80f.html --lua-filter "C:\Users\Rohan\Documents\R\win-
library\4.1\rmarkdown\rmarkdown\lua\pagebreak.lua" --lua-filter
"C:\Users\Rohan\Documents\R\win-library\4.1\rmarkdown\rmarkdown\lua\latex-
div.lua" --self-contained --variable bs3=TRUE --standalone --section-divs --
table-of-contents --toc-depth 6 --template "C:\Users\Rohan\Documents\R\win-
library\4.1\rmarkdown\rmd\h\default.html" --no-highlight --variable
highlightjs=1 --variable theme=yeti --include-in-header
"C:\Users\Rohan\AppData\Local\Temp\Rtmp00ORpv\rmarkdown-str1ee4440b1d0f.html"
--mathjax --variable "mathjax-
url:https://mathjax.rstudio.com/latest/MathJax.js?config=TeX-AMS-
MML_HTMLorMML"

##
## Output created: report.html
```

Pre-processing of data.

Checking and removing of outliers using box plot method.

```
boxplot(bodyData)
```

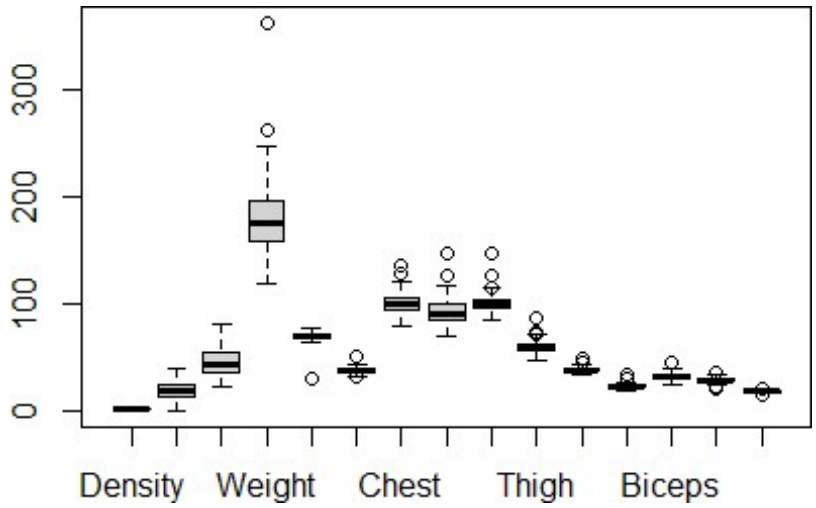


As we can see there are alot of outliers in the every variables of the dataset. “Density” “BodyFat” “Age” “Weight” “Height” “Neck” “Chest” “Abdomen” “Hip” “Thigh” “Knee” “Ankle” “Biceps” “Forearm” “Wrist” “WeightGroup”.

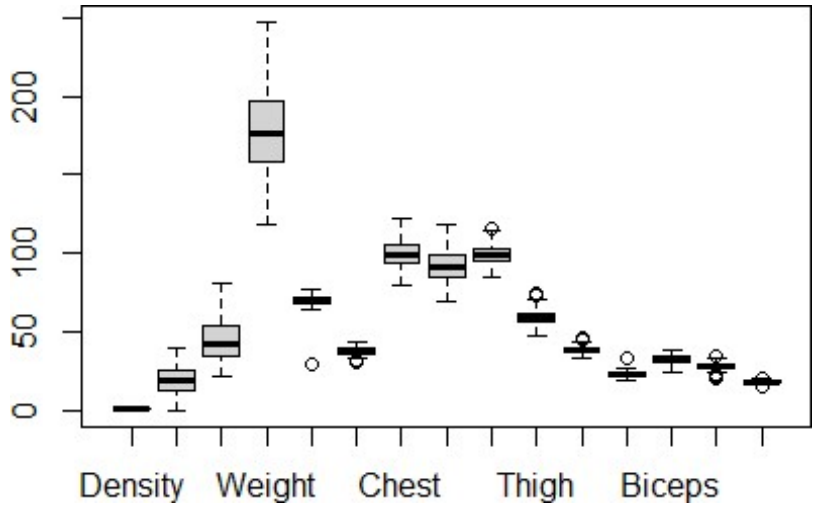
Terminilogies of the box plots.

-The interquartile range (IQR) is a measure of variability, based on dividing a data set into quartiles. -The interquartile range (IQR) is a measurement of the spread of values in the middle 50%. -Q1 is the "middle" value in the first half of the rank-ordered data set. -Q3 is the "middle" value in the second half of the rank-ordered data set.

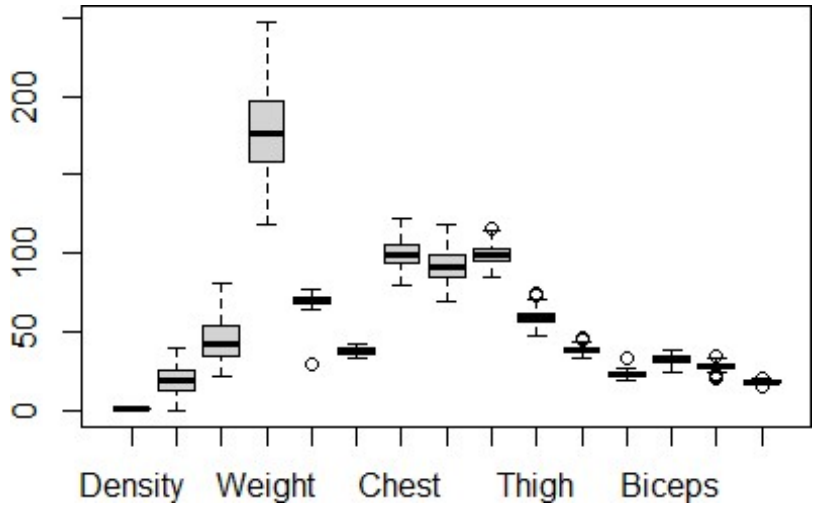
```
#find Q1, Q3, and interquartile range for values in column Bodyfat
Q1 <- quantile(bodyData$BodyFat, .25)
Q3 <- quantile(bodyData$BodyFat, .75)
IQR <- IQR(bodyData$BodyFat)
#only keep rows in dataframe that have values within 1.5*IQR of Q1 and Q3
df1<- subset(bodyData, bodyData$BodyFat> (Q1 - 1.5*IQR) & bodyData$BodyFat<
(Q3 + 1.5*IQR))
boxplot(df1)
```



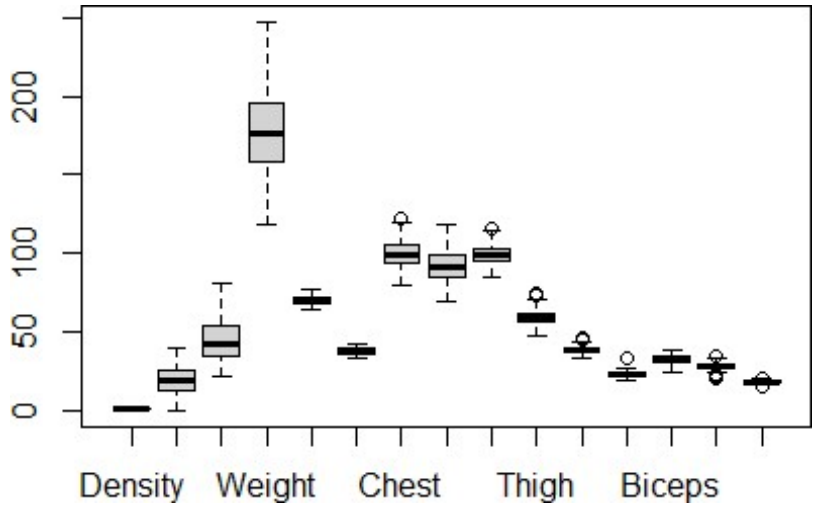

```
Q1 <- quantile(df1$Weight, .25)
Q3 <- quantile(df1$Weight, .75)
IQR <- IQR(df1$Weight)
df2<- subset(df1, df1$Weight> (Q1 - 1.5*IQR) & df1$Weight< (Q3 + 1.5*IQR))
boxplot(df2)
```



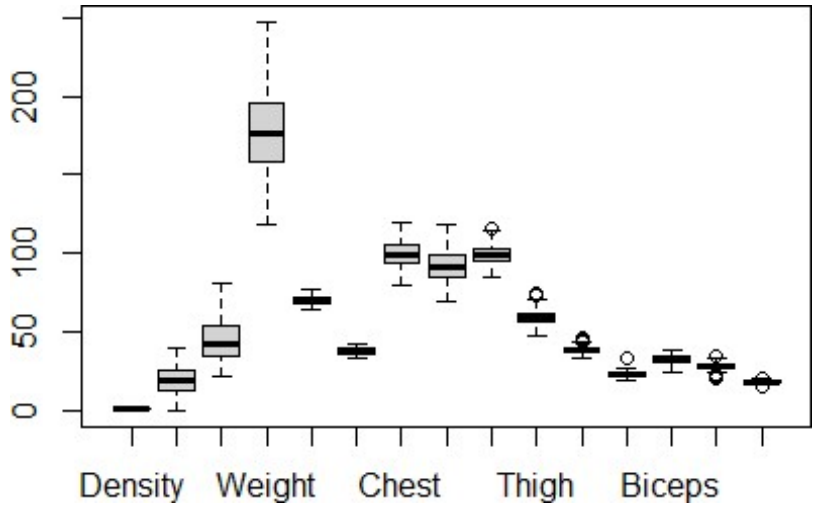

```
Q1 <- quantile(df2$Neck, .25)
Q3 <- quantile(df2$Neck, .75)
IQR <- IQR(df2$Neck)
df3 <- subset(df2, df2$Neck> (Q1 - 1.5*IQR) & df2$Neck< (Q3 + 1.5*IQR))
boxplot(df3)
```



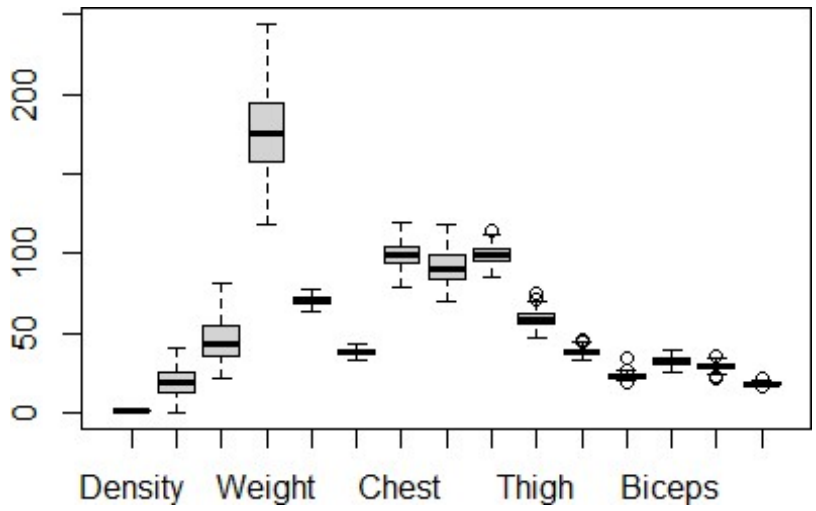

```
Q1 <- quantile(df3$Height, .25)
Q3 <- quantile(df3$Height, .75)
IQR <- IQR(df3$Height)
df4 <- subset(df3, df3$Height> (Q1 - 1.5*IQR) & df3$Height< (Q3 + 1.5*IQR))
boxplot(df4)
```



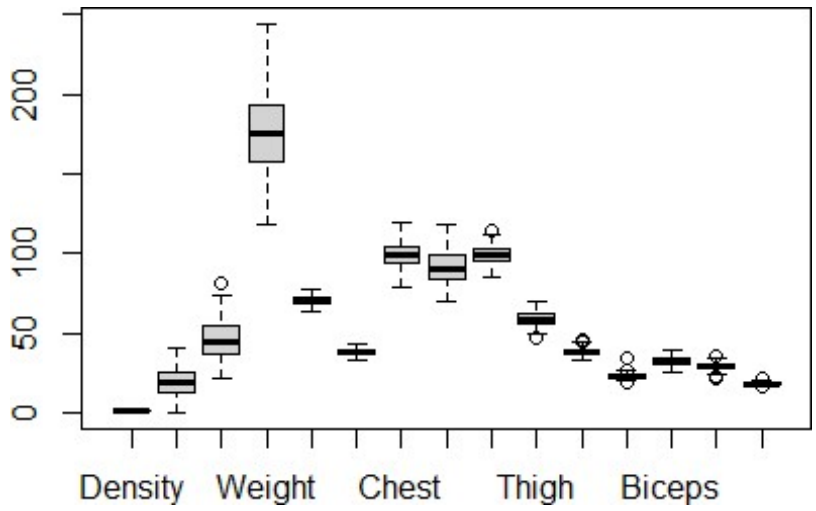

```
Q1 <- quantile(df4$Chest, .25)
Q3 <- quantile(df4$Chest, .75)
IQR <- IQR(df4$Chest)
df5 <- subset(df4, df4$Chest> (Q1 - 1.5*IQR) & df4$Chest< (Q3 + 1.5*IQR))
boxplot(df5)
```



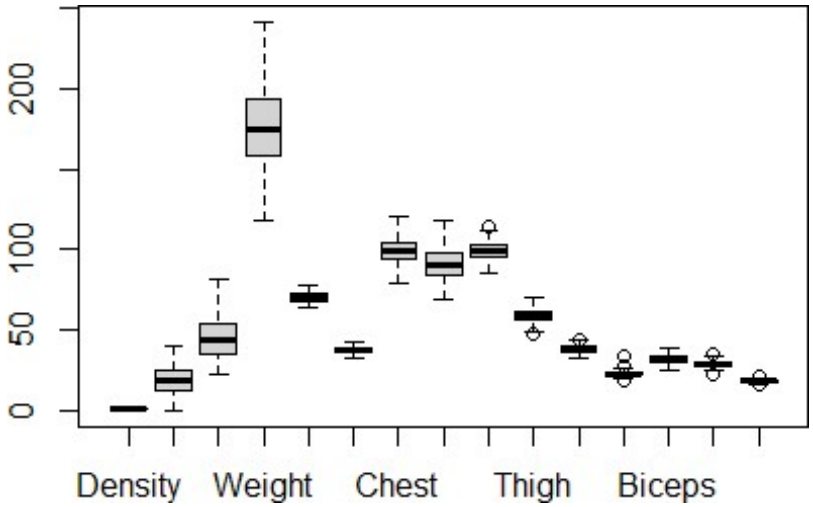

```
Q1 <- quantile(df5$Hip, .25)
Q3 <- quantile(df5$Hip, .75)
IQR <- IQR(df5$Hip)
df6 <- subset(df5, df5$Hip> (Q1 - 1.5*IQR) & df5$Hip< (Q3 + 1.5*IQR))
boxplot(df6)
```



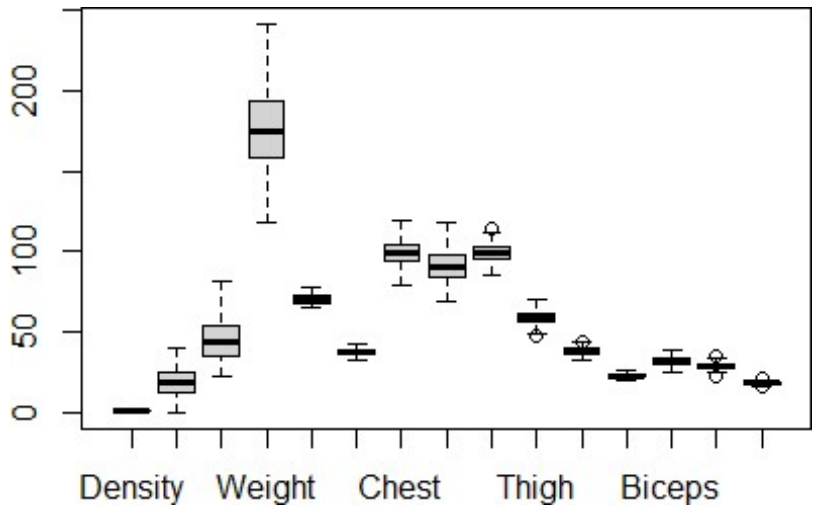

```
Q1 <- quantile(df6$Thigh, .25)
Q3 <- quantile(df6$Thigh, .75)
IQR <- IQR(df6$Thigh)
df7 <- subset(df6,df6$Thigh> (Q1 - 1.5*IQR) & df6$Thigh< (Q3 + 1.5*IQR))
boxplot(df7)
```



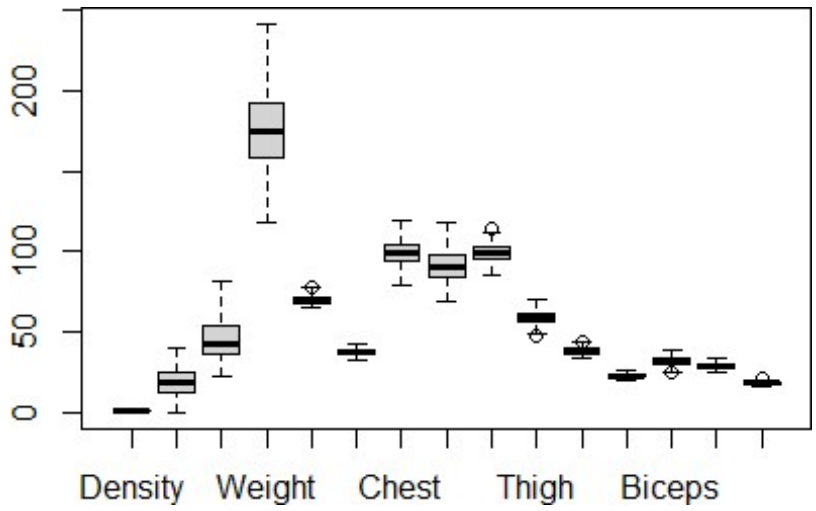

```
Q1 <- quantile(df7$Knee, .25)
Q3 <- quantile(df7$Knee, .75)
IQR <- IQR(df7$Knee)
df8 <- subset(df7,df7$Knee> (Q1 - 1.5*IQR) & df7$Knee< (Q3 + 1.5*IQR))
boxplot(df8)
```



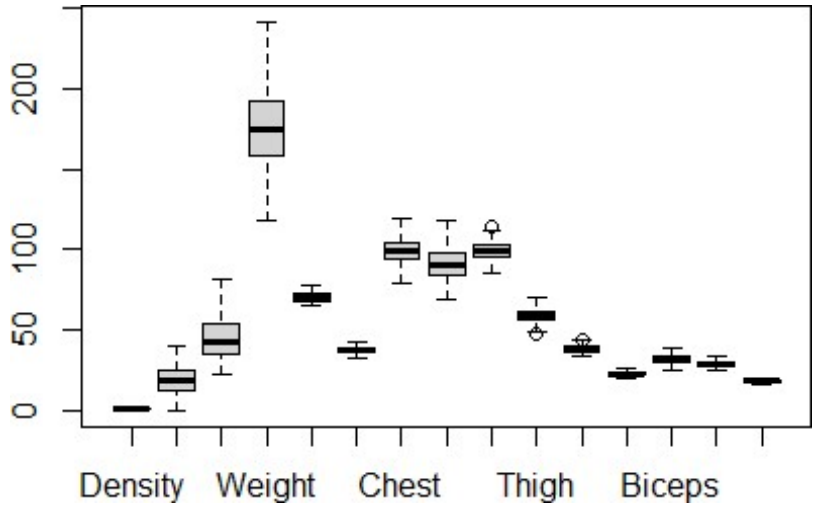

```
Q1 <- quantile(df8$Ankle, .25)
Q3 <- quantile(df8$Ankle, .75)
IQR <- IQR(df8$Ankle)
df9 <- subset(df8,df8$Ankle> (Q1 - 1.5*IQR) & df8$Ankle< (Q3 + 1.5*IQR))
boxplot(df9)
```




```
Q1 <- quantile(df9$Forearm, .25)
Q3 <- quantile(df9$Forearm, .75)
IQR <- IQR(df9$Forearm)
df10 <- subset(df9,df9$Forearm> (Q1 - 1.5*IQR) & df9$Forearm< (Q3 + 1.5*IQR))
boxplot(df10)
```




```
Q1 <- quantile(df10$Wrist, .25)
Q3 <- quantile(df10$Wrist, .75)
IQR <- IQR(df10$Wrist)
bodyData <- subset(df10,df10$Wrist> (Q1 - 1.5*IQR) & df10$Wrist< (Q3 +
1.5*IQR))
boxplot(bodyData)
```



We cleaned the dataset by removing the outliers. Dataset name: “bodyData”

Multi linear regression model.

```
bodyData$WeightGroup[bodyData$BodyFat < 18.5] = 1
bodyData$WeightGroup[bodyData$BodyFat >= 18.5 & bodyData$BodyFat <25] = 2
bodyData$WeightGroup[bodyData$BodyFat >= 25 & bodyData$BodyFat < 30] = 3
bodyData$WeightGroup[bodyData$BodyFat >= 30 ] = 4

weightType = c("Underweight", "Normalweight", "Overweight", "Obese")
factor(bodyData$WeightGroup, labels = weightType)
```

```
##   [1] Underweight   Underweight   Overweight    Underweight   Overweight
##   [6] Normalweight Normalweight Underweight   Underweight   Underweight
##  [11] Underweight   Underweight   Normalweight Normalweight Normalweight
##  [16] Normalweight Overweight    Normalweight Underweight   Underweight
##  [21] Normalweight Underweight   Underweight   Underweight   Underweight
##  [26] Underweight   Underweight   Normalweight Underweight   Underweight
##  [31] Underweight   Underweight   Obese         Normalweight Overweight
##  [36] Obese         Obese         Obese         Underweight   Underweight
##  [41] Underweight   Underweight   Underweight   Underweight   Underweight
##  [46] Underweight   Underweight   Underweight   Normalweight Normalweight
##  [51] Overweight    Obese         Normalweight Overweight    Overweight
##  [56] Obese         Overweight    Obese         Obese         Normalweight
##  [61] Underweight   Underweight   Underweight   Normalweight Underweight
##  [66] Underweight   Underweight   Normalweight Underweight   Normalweight
##  [71] Normalweight Normalweight Obese         Overweight    Underweight
```

```
##  [76] Overweight   Overweight   Underweight  Normalweight Underweight
##  [81] Underweight  Normalweight Underweight  Underweight  Normalweight
##  [86] Underweight  Underweight  Underweight  Underweight  Underweight
##  [91] Normalweight Normalweight Normalweight Normalweight Normalweight
##  [96] Overweight   Normalweight Underweight  Underweight  Normalweight
## [101] Normalweight Overweight   Normalweight Normalweight Overweight
## [106] Underweight  Normalweight Underweight  Overweight   Underweight
## [111] Overweight   Overweight   Underweight  Underweight  Underweight
## [116] Underweight  Overweight   Underweight  Normalweight Underweight
## [121] Underweight  Normalweight Normalweight Overweight   Normalweight
## [126] Overweight   Normalweight Overweight   Normalweight Normalweight
## [131] Normalweight Underweight  Normalweight Underweight  Underweight
## [136] Underweight  Overweight   Underweight  Overweight   Underweight
## [141] Underweight  Underweight  Normalweight Underweight  Obese
## [146] Underweight  Normalweight Underweight  Underweight  Underweight
## [151] Underweight  Overweight   Normalweight Normalweight Underweight
## [156] Underweight  Underweight  Normalweight Underweight  Underweight
## [161] Underweight  Overweight   Normalweight Underweight  Overweight
## [166] Underweight  Underweight  Underweight  Underweight  Underweight
## [171] Normalweight Normalweight Normalweight Normalweight Underweight
## [176] Underweight  Normalweight Normalweight Overweight   Normalweight
## [181] Underweight  Underweight  Normalweight Underweight  Normalweight
## [186] Overweight   Underweight  Obese        Obese        Underweight
## [191] Underweight  Underweight  Overweight   Normalweight Normalweight
## [196] Normalweight Underweight  Underweight  Normalweight Underweight
## [201] Underweight  Underweight  Underweight  Underweight  Underweight
## [206] Overweight   Underweight  Underweight  Underweight  Underweight
## [211] Underweight  Overweight   Overweight   Normalweight Normalweight
## [216] Overweight   Underweight  Overweight   Underweight  Obese
## [221] Obese        Overweight   Underweight  Obese        Underweight
## [226] Obese        Overweight   Overweight
## Levels: Underweight Normalweight Overweight Obese
```

```
fit = lm(BodyFat ~.,data = bodyData)

summary(fit)
```

```
##
## Call:
## lm(formula = BodyFat ~ ., data = bodyData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1590 -0.4113 -0.0717  0.3759 13.7618
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.059e+02  1.545e+01  26.267  < 2e-16 ***
## Density     -3.722e+02  1.162e+01 -32.030  < 2e-16 ***
## Age          1.108e-02  1.012e-02   1.095   0.2749
```

```
## Weight       1.083e-02  2.205e-02   0.491   0.6238
## Height       2.611e-02  6.325e-02   0.413   0.6802
## Neck        -3.559e-02  7.972e-02  -0.446   0.6558
## Chest        3.494e-02  3.411e-02   1.024   0.3069
## Abdomen      5.987e-03  3.384e-02   0.177   0.8597
## Hip          5.092e-03  4.699e-02   0.108   0.9138
## Thigh        3.878e-02  4.794e-02   0.809   0.4195
## Knee         7.063e-04  8.565e-02   0.008   0.9934
## Ankle       -2.468e-01  1.151e-01  -2.144   0.0332 *
## Biceps      -3.738e-02  5.669e-02  -0.659   0.5103
## Forearm     -3.089e-02  1.047e-01  -0.295   0.7682
## Wrist        1.316e-01  1.893e-01   0.695   0.4877
## WeightGroup  8.460e-01  1.912e-01   4.423 1.55e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.262 on 212 degrees of freedom
## Multiple R-squared:  0.9762, Adjusted R-squared:  0.9745
## F-statistic:   580 on 15 and 212 DF,  p-value: < 2.2e-16
```

Adjusted R-squared: 0.9745

Selection of final model using step() (Feature engineering).

The optimzed model can be obtained by selecting active predictors with Akaike information criterion(AIC) or Bayesan information criterion(BIC). Step function with backward method is used to select variables for the optimized subset models by the Akaike information criterion (AIC) for the given set of data.

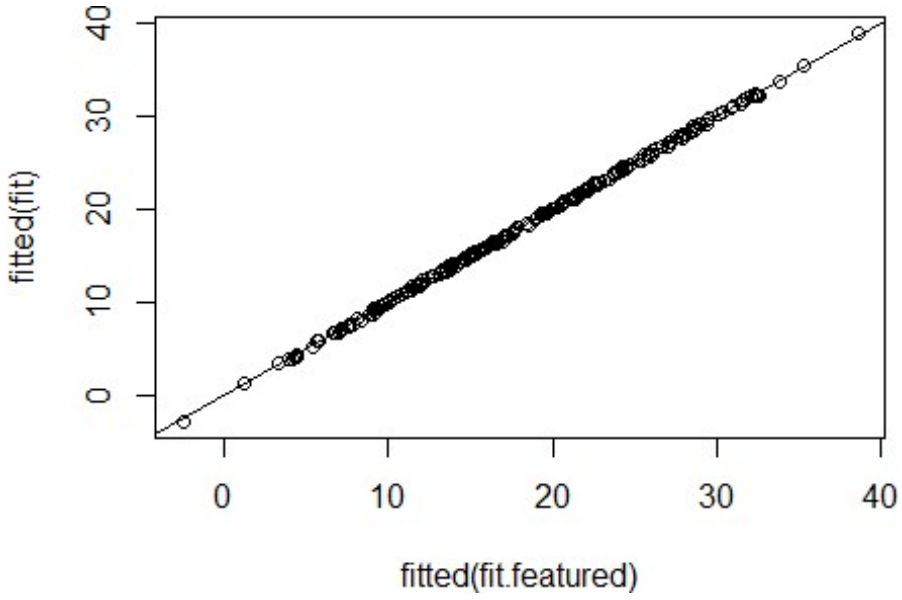
```
fit.featured = step(fit, scope = list(lower ~ Density), trace=0)

summary(fit.featured)
```

```
##
## Call:
## lm(formula = BodyFat ~ Density + Age + Weight + Ankle + WeightGroup,
##     data = bodyData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1472 -0.4193 -0.0848  0.3410 14.0963
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.133e+02  1.105e+01  37.397  < 2e-16 ***
## Density     -3.748e+02  9.962e+00 -37.621  < 2e-16 ***
## Age          1.362e-02  7.081e-03   1.924   0.0557 .
## Weight       2.465e-02  5.638e-03   4.371 1.90e-05 ***
## Ankle       -2.227e-01  9.626e-02  -2.314   0.0216 *
```

```
## WeightGroup   8.378e-01   1.825e-01    4.590 7.42e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.242 on 222 degrees of freedom
## Multiple R-squared:  0.9759, Adjusted R-squared:  0.9753
## F-statistic:  1796 on 5 and 222 DF,  p-value: < 2.2e-16
```

** Comparing the old model and new model. **

```
plot(fitted(fit) ~ fitted(fit.featured))
abline(0,1)
```




```
cor(fitted(fit),fitted(fit.featured))
```

```
## [1] 0.999827
```

The change in the fitted values is relatively small, and the two sets of fitted values have correlation .99. So we conclude that the subset model and the full model provide essentially the same information about the value of the response given predicors.