INFX 573: Problem Set 6 - Regression

TAPASVI BANSAL

Due: Tuesday, November 21, 2017

Problem Set 6

Collaborators:

Instructions:

Before beginning this assignment, please ensure you have access to R and RStudio.

1. Replace the "Insert Your Name Here" text in the `author:` field with your own name. List all collaborators on the top of your assignment.
2. Be sure to include well-documented (e.g. commented) code chucks, figures and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text.
3. Collaboration on problem sets is fun and useful but turn in an individual write-up in your own words and involving your own code. Do not just copy-and-paste from others' responses or code.
4. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the R Markdown file to `YourLastName_YourFirstName_ps6.Rmd`, knit a PDF and submit the PDF file on Canvas.

1. Housing Values in Suburbs of Boston

In this problem we will use the Boston dataset that is available in *MASS* package. This dataset contains information about median house value for 506 neighborhoods in Boston, MA.

```
#install.packages("MASS")
#install.packages("tidyverse")
library(MASS)
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.4.2
```

```
## -- Attaching packages ---------------------------------- tidyverse 1.2.1 --
```

```
## ✓ ggplot2 2.2.1     ✓ purrr   0.2.4
## ✓ tibble  1.3.4     ✓ dplyr   0.7.4
## ✓ tidyr   0.7.2     ✓ stringr 1.2.0
## ✓ readr   1.1.1     ✓ forcats 0.2.0
```

```
## Warning: package 'tidyr' was built under R version 3.4.2
```

```
## Warning: package 'purrr' was built under R version 3.4.2
```

```
## Warning: package 'dplyr' was built under R version 3.4.2
```

```
## -- Conflicts ------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
```

1.1 Describe data

Describe the data and variables that are part of the *Boston* dataset. Tidy data as necessary.

```
MBoston <- MASS::Boston
#?MASS::Boston
#View(MBoston)
#MBoston
#names(MBoston)
```

The data is in tabular form with 506 rows and 14 columns and is describing about housing Values in Suburbs of Boston with 14 attributes/varaibles ("crim", "zn", "indus", "chas", "nox", "rm", "age", "dis", "rad", "tax", "ptratio", "black", "lstat", "medv"). The rows represent suburbs of Boston. The columns represent attributes of suburb which can be used to predict the housing values. Th

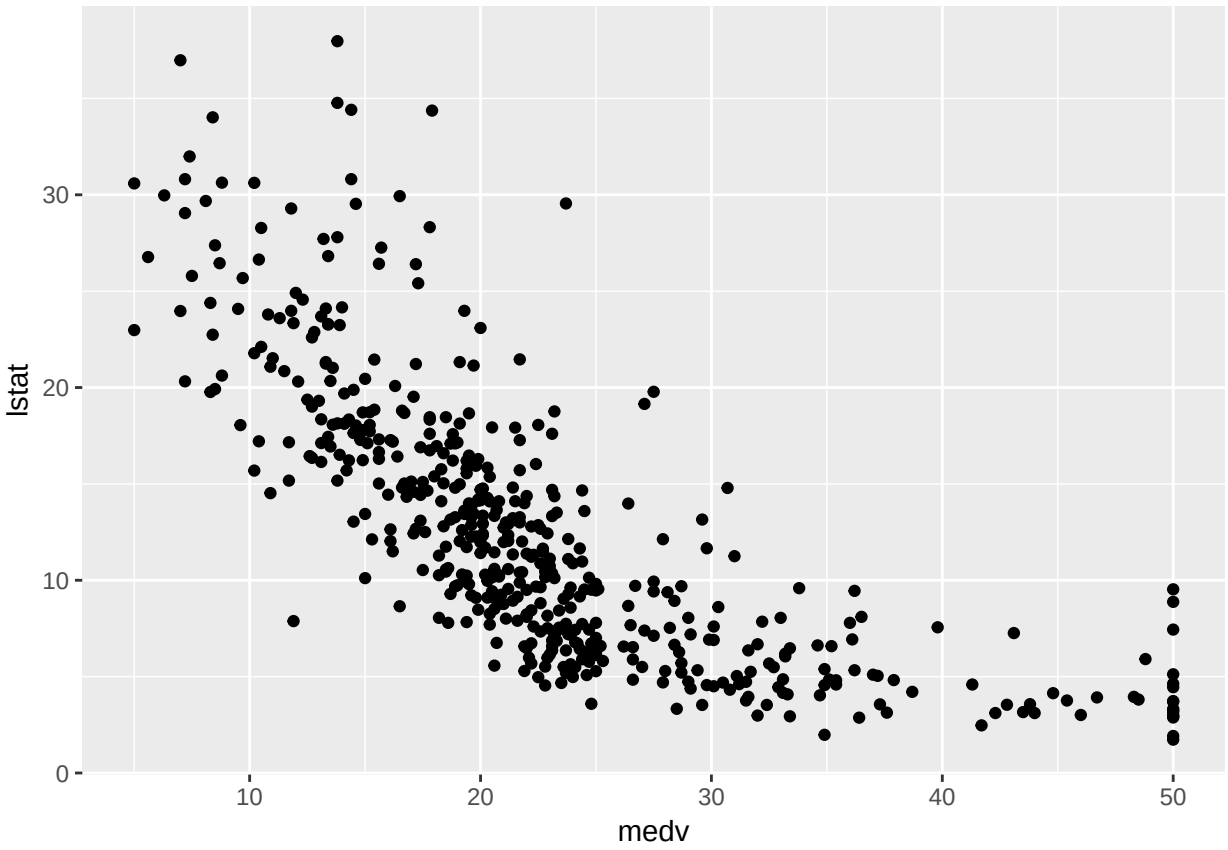
The Description of the variables are as follows:

1. crim- 1per capita crime rate by town.
2. zn- proportion of residential land zoned for lots over 25,000 sq.ft.
3. indus- proportion of non-retail business acres per town.
4. chas- Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
5. nox- nitrogen oxides concentration (parts per 10 million).
6. rm- average number of rooms per dwelling. age proportion of owner-occupied units built prior to 1940.
7. age- proportion of owner-occupied units built prior to 1940.
8. dis- weighted mean of distances to five Boston employment centres.
9. rad- index of accessibility to radial highways.
10. tax- full-value property-tax rate per $10,000.
11. ptratio- pupil-teacher ratio by town.
12. black- 1000(Bk − 0.63)2 where Bk is the proportion of blacks by town.
13. lstat- lower status of the population (percent).
14. medv- median value of owner-occupied homes in $1000s.

1.2 Variable of interest

Consider this data in context, what is the response variable of interest? Discuss how you think some of the possible predictor variables might be associated with this response.

```
#ggplot(MBoston) + geom_point(aes(x=medv, y=tax))
ggplot(MBoston) + geom_point(aes(x=medv, y=lstat))
```




```
#ggplot(MBoston) + geom_point(aes(x=medv, y=rm))
```

The response variable of interest is median value of owner-occupied homes in $1000s ("medv"). With the attributes/variables provided (14), describing the housing values in Suburbs of Boston, almost all of them can be said to have an association with the response variable (crim, zn, indus,nox,ptratio, dis,ptratio). Some attributes(rad & tax) reflects almost similar association with the variable medv.

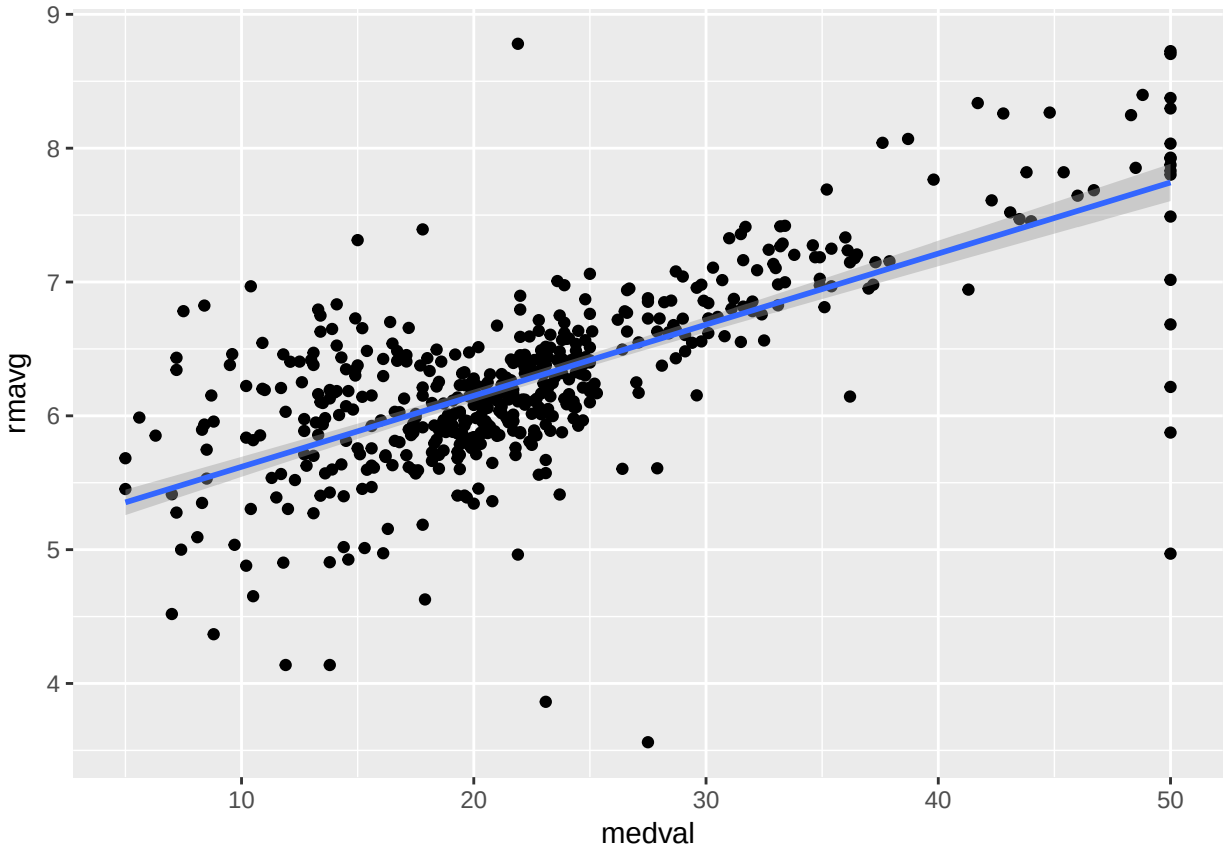
The attributes "lstat" and "rm" shows continuous association with the variable of interest.

1.3 Simple Regression

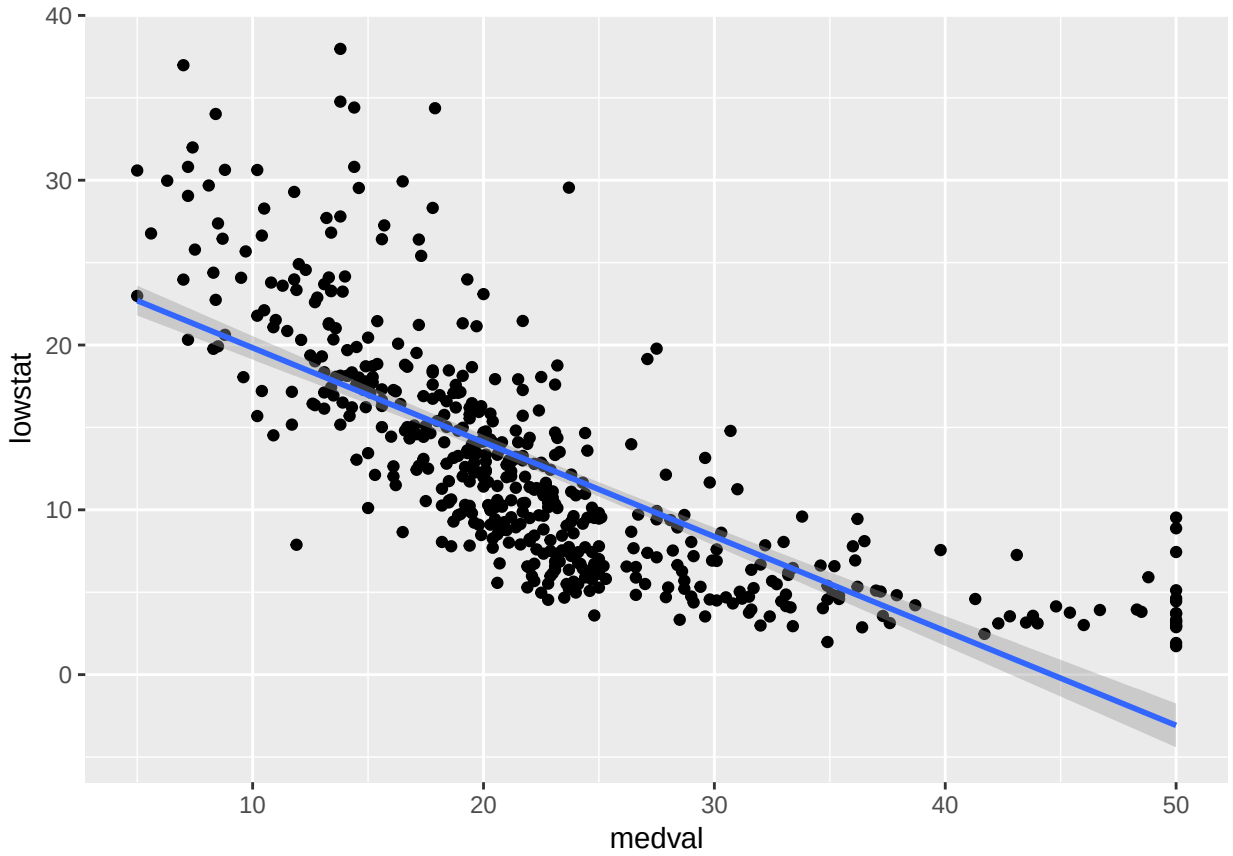
For each predictor, fit a simple linear regression model to predict the response. In which of the models is there a statistically significant association between the predictor and the response? Create some plots to back up your assertions.

```
mc <- lm(medv~crim, data = MBoston) # adj.R-squared = 0.1491
#summary(mc)
mz <- lm(medv~zn, data = MBoston)   # adj.R-squared = 0.1282
#summary(mz)
mi <- lm(medv~indus, data = MBoston)   # adj.R-squared = 0.2325
#summary(mi)
mch <- lm(medv~chas, data = MBoston)   # adj.R-squared = 0.2879
#summary(mch)
mn <- lm(medv~nox, data = MBoston)   # adj.R-squared = 0.181
#summary(mn)
mr <- lm(medv~rm, data = MBoston)   # adj.R-squared = 0.4825  Second Highest
#summary(mr)
```

```r
ma <- lm(medv~age, data = MBoston)    # adj.R-squared = 0.1404
#summary(ma)
md <- lm(medv~dis, data = MBoston)    # adj.R-squared = 0.0606
#summary(md)
mra <- lm(medv~rad, data = MBoston)    # adj.R-squared = 0.1439
#summary(mra)
mt <- lm(medv~tax, data = MBoston)    # adj.R-squared = 0.218
#summary(mt)
mp <- lm(medv~ptratio, data = MBoston)    # adj.R-squared = 0.2564
#summary(mp)
mb <- lm(medv~black, data = MBoston)    # adj.R-squared = 0.1094
#summary(mb)
ml <- lm(medv~lstat, data = MBoston)    # adj.R-squared = 0.5432 Highest
#summary(ml)

# FOR PLOTTING

medval <- MBoston$medv
rmavg <- MBoston$rm
lowstat <- MBoston$lstat

## PLOTS

qplot(medval,rmavg,geom ="point") + geom_smooth(method ="lm")
```



```
qplot(medval,lowstat,geom ="point") + geom_smooth(method ="lm")
```



The models with "rm" and "stat" as predictor variable and "medv" as response variable shows there's a statistically significant association between the predictor individually and the response.

1.4 Multiple Regression

Make sure you are familiar with multiple regression (Openintro Statistics, Ch 8.1-8.3).

Fit a multiple regression model to predict the response using all of the predictors. Describe your results. For which predictors can we reject the null hypothesis $H_0 : \beta_j = 0$?

```
MR <- lm(medv~., data=MBoston)
#MR
#summary(MR)


bmodel <- lm(medv~crim+zn+indus+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bmodel)

bmodel1 <- lm(medv~crim+zn+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bmodel1) ## Removing "indus" increases the adjusted R-Squared Value

bmodel3 <- lm(medv~crim+zn+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bmodel3) ## Removing "chas" decreases the adjusted R-Squared Value

bmodel4 <- lm(medv~zn+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
```

```
#summary(bmodel4) ## Removing "crim" decreases the adjusted R-Squared Value

bmodel5 <- lm(medv~crim+zn+chas+nox+rm+dis+rad+ptratio+black+lstat,data=MBoston)
#summary(bmodel5) ## Removing "tax" decreases the adjusted R-Squared Value



# Highest adjusted R-Squared Value
bmodel2 <- lm(medv~crim+zn+chas+nox+rm+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bmodel2) ## Removing "age" increases the adjusted R-Squared Value more
```

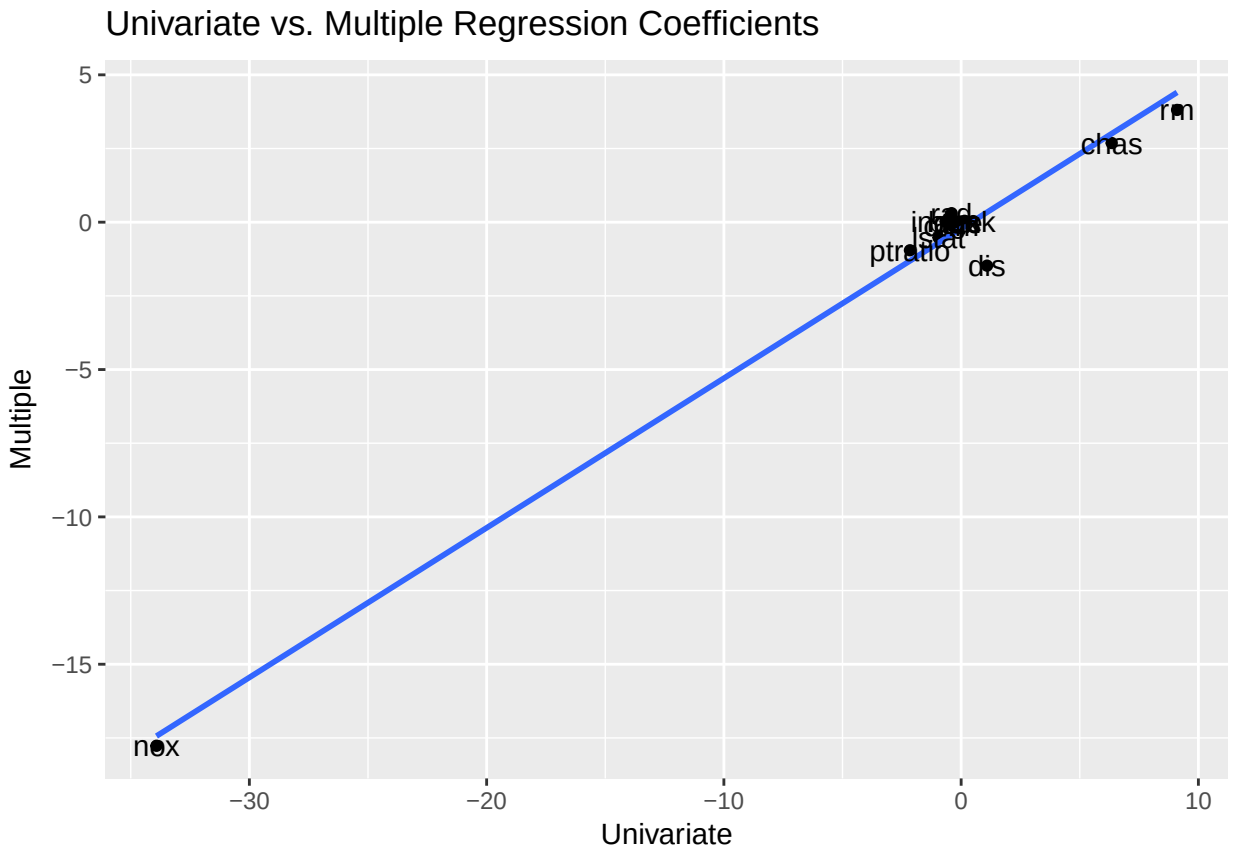
For "indus" and "age" predictors we can reject the null hypothesis as their p-value is much higher than the considered p-value (0.05). Hence, removing the insignificant predictors (indus and age) definitely improved the model.

The Backward Elimination model is the best fit model with the highest evaluated adjusted R-squared value of 0.7348, bringing the correlation value close to +1 and the farthest from 0.

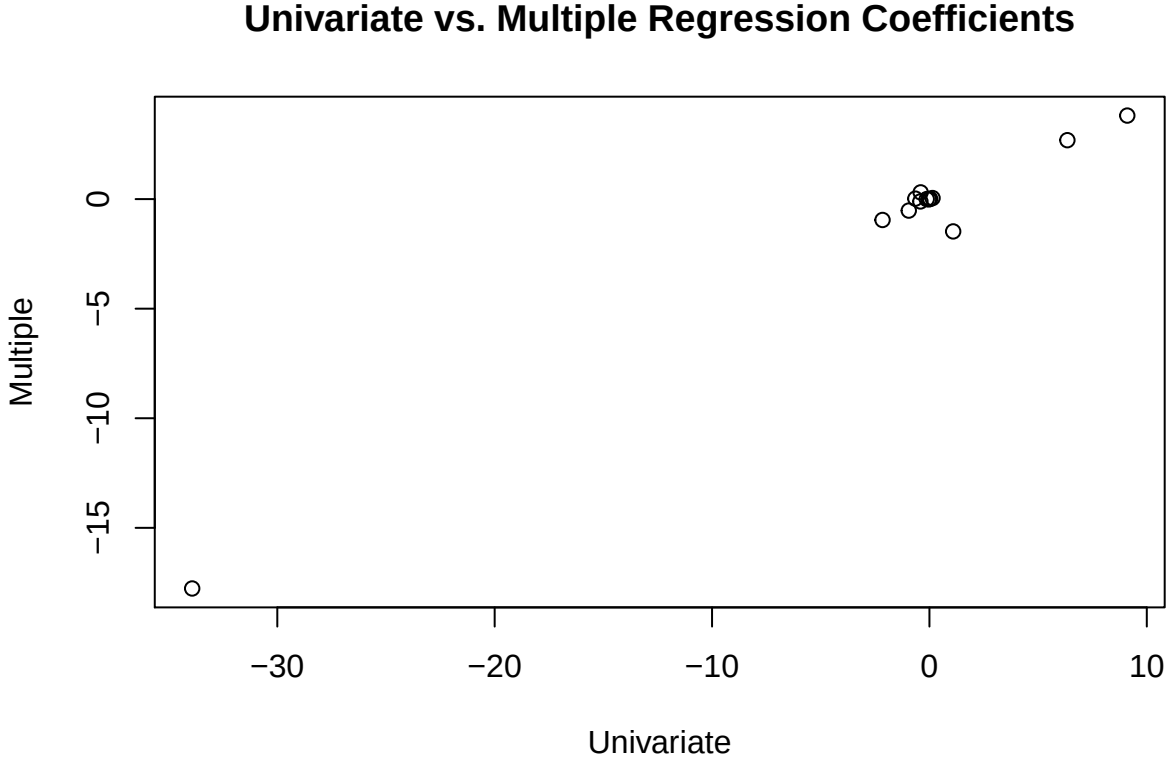
1.5 Compare Regressions

How do your results from (3) compare to your results from (4)? Create a plot displaying the univariate regression coefficients from (3) on the x-axis and the multiple regression coefficients from part (4) on the y-axis. Use this visualization to support your response.

```
# length(MBoston)
# v = list()
# for (i in 1:length(MBoston))
#   v[i] <- MBoston[i]

univariate <- c(mc$coefficients[2], mz$coefficients[2] , mi$coefficients[2], mch$coefficients[2], mn$coe
#univariate

multiplereg <- bmodel$coefficients[2:14]
#multiplereg


## Plotting

qplot(univariate,bmodel$coefficients[2:14],geom="point") + geom_smooth(method ="lm", se= FALSE) + geom_t
```




```
plot(univariate, multiplereg, main = "Univariate vs. Multiple Regression Coefficients",
    xlab = "Univariate", ylab = "Multiple")
```



The univariate regression coefficients points change drastically in camparison to the multiple regression coefficients. This difference is beacuse in the linear regression, the slope represents the average effect of an increase (or decrease) in the predictor, ignoring other predictors. Whereas, in multiple regression, the slope represents the average effect of an increase (or decrease) in the predictor, while considering other predictor.

1.6 Non-linearities

Is there evidence of a non-linear association between any of the predictors and the response? To answer this question, for each predictor X fit a model of the form:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon$$

```
mcpoly <- lm(medv~poly(crim,3), data = MBoston) # adj.R-squared = 0.214 (0.1491)
#summary(mcpoly)
mzpoly <- lm(medv~poly(zn,3), data = MBoston)   # adj.R-squared = 0.1599 (0.1282)
#summary(mzpoly)
mipoly <- lm(medv~poly(indus,3), data = MBoston)   # adj.R-squared = 0.2725 (0.2325)
#summary(mipoly)

##  for  "chas" has outcomes in just 0 or 1, not applicable
#mchpoly <- lm(medv~poly(chas,3), data = MBoston)   # adj.R-squared = 0.2879


mnpoly <- lm(medv~poly(nox,3), data = MBoston)   # adj.R-squared = 0.189 (0.181)
#summary(mnpoly)
mrpoly <- lm(medv~poly(rm,3), data = MBoston)   # adj.R-squared = 0.5586 (0.4825)  Second Highest
#summary(mrpoly)
mapoly <- lm(medv~poly(age,3), data = MBoston)   # adj.R-squared = 0.1515 (0.1404)
#summary(mapoly)
mdpoly <- lm(medv~poly(dis,3), data = MBoston)   # adj.R-squared = 0.09968 (0.0606)
#summary(mdpoly)
mrapoly <- lm(medv~poly(rad,3), data = MBoston)   # adj.R-squared = 0.1718 (0.1439)
#summary(mrapoly)
mtpoly <- lm(medv~poly(tax,3), data = MBoston)   # adj.R-squared = 0.2215 (0.218)
#summary(mtpoly)
mppoly <- lm(medv~poly(ptratio,3), data = MBoston)   # adj.R-squared = 0.2625 (0.2564)
#summary(mppoly)
mbpoly <- lm(medv~poly(black,3), data = MBoston)   # adj.R-squared = 0.1082 (0.1094)
#summary(mbpoly)
mlpoly <- lm(medv~poly(lstat,3), data = MBoston)   # adj.R-squared = 0.6558 (0.5432) Highest
#summary(mlpoly)


## PLOT
#qplot(MBoston$medv,MBoston$nox,geom ="point") + geom_smooth(method ="lm")
#qplot(MBoston$medv,MBoston$tax,geom ="point") + geom_smooth(method ="lm")
#qplot(MBoston$medv,MBoston$ptratio,geom ="point") + geom_smooth(method ="lm")
#qplot(MBoston$medv,MBoston$black,geom ="point") + geom_smooth(method ="lm")
```

The cubic coefficient for “crim”, “indus”,“nox”,“age”,“dis”,“tax”,“ptratio”,“black” doesn’t have statistically significant value considering statistically significance of p-value at 0.05.

HOWEVER, for “nox”, “tax”, “ptratio”, “black” the the p-values suggest that for both quadratic and cubic coefficients are not statistically significant, so in these cases no non-linear effect is visible.

1.7 Stepwise Model Selection

Consider performing a stepwise model selection procedure to determine the best fit model (consult Openintro Statistics, 8.2.2). Discuss your results. How is this model different from the model in (4)?

```
## BACKWARD ELIMINATION MODEL
bemodel <- lm(medv~crim+zn+indus+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bemodel) ## The adjusted R-Squared Value = 0.7338


##
bemodel1 <- lm(medv~crim+zn+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bemodel1)
## Removing "indus" increases the adjusted R-Squared Value
## NEW adjusted R-Squared Value = 0.7343

##
bemodel2 <- lm(medv~crim+zn+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bemodel2)
## Removing "chas" decreases the adjusted R-Squared Value ==> NO CHANGE

##
bemodel3 <- lm(medv~zn+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bemodel3)
## Removing "crim" decreases the adjusted R-Squared Value ==> NO CHANGE

##
bemodel4 <- lm(medv~crim+zn+chas+nox+rm+dis+rad+ptratio+black+lstat,data=MBoston)
#summary(bemodel4)
## Removing "tax" decreases the adjusted R-Squared Value ==> NO CHANGE

##
bemodel5 <- lm(medv~crim+zn+chas+nox+rm+dis+rad+tax+ptratio+black+lstat,data=MBoston)
#summary(bemodel5)
## Removing "age" increases the adjusted R-Squared Value more
## NEW adjusted R-Squared Value = 0.7348


## USING AIC
fitmodel <- lm(medv~crim+zn+indus+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat,data=MBoston)
stff <- stepAIC(fitmodel, direction = "forward")
```

```
## Start:  AIC=1589.64
## medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat
```

```
stfb<- stepAIC(fitmodel, direction = "backward")
```

```
## Start:  AIC=1589.64
## medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - age      1      0.06 11079 1587.7
## - indus    1      2.52 11081 1587.8
```

```
## <none>                 11079 1589.6
## - chas     1    218.97 11298 1597.5
## - tax      1    242.26 11321 1598.6
## - crim     1    243.22 11322 1598.6
## - zn       1    257.49 11336 1599.3
## - black    1    270.63 11349 1599.8
## - rad      1    479.15 11558 1609.1
## - nox      1    487.16 11566 1609.4
## - ptratio  1   1194.23 12273 1639.4
## - dis      1   1232.41 12311 1641.0
## - rm       1   1871.32 12950 1666.6
## - lstat    1   2410.84 13490 1687.3
##
## Step:  AIC=1587.65
## medv ~ crim + zn + indus + chas + nox + rm + dis + rad + tax +
##     ptratio + black + lstat
##
##           Df Sum of Sq   RSS    AIC
## - indus    1      2.52 11081 1585.8
## <none>                 11079 1587.7
## - chas     1    219.91 11299 1595.6
## - tax      1    242.24 11321 1596.6
## - crim     1    243.20 11322 1596.6
## - zn       1    260.32 11339 1597.4
## - black    1    272.26 11351 1597.9
## - rad      1    481.09 11560 1607.2
## - nox      1    520.87 11600 1608.9
## - ptratio  1   1200.23 12279 1637.7
## - dis      1   1352.26 12431 1643.9
## - rm       1   1959.55 13038 1668.0
## - lstat    1   2718.88 13798 1696.7
##
## Step:  AIC=1585.76
## medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio +
##     black + lstat
##
##           Df Sum of Sq   RSS    AIC
## <none>                 11081 1585.8
## - chas     1    227.21 11309 1594.0
## - crim     1    245.37 11327 1594.8
## - zn       1    257.82 11339 1595.4
## - black    1    270.82 11352 1596.0
## - tax      1    273.62 11355 1596.1
## - rad      1    500.92 11582 1606.1
## - nox      1    541.91 11623 1607.9
## - ptratio  1   1206.45 12288 1636.0
## - dis      1   1448.94 12530 1645.9
## - rm       1   1963.66 13045 1666.3
## - lstat    1   2723.48 13805 1695.0
```

```
#summary(stf)
#stff$anova
#stfb$anova
```

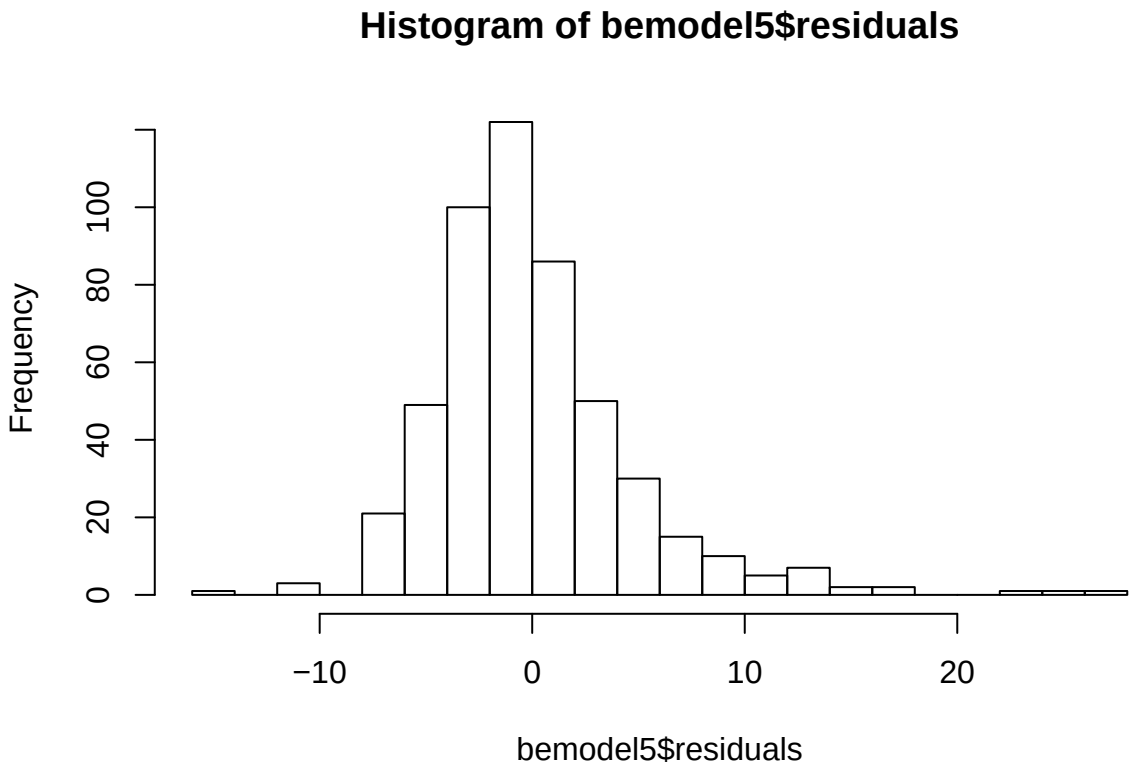
The best fit model is Backward Elimination Model. Removing “indus” and “age” increases the adjusted

R-Squared Value to highest possible value i.e. 0.7348.

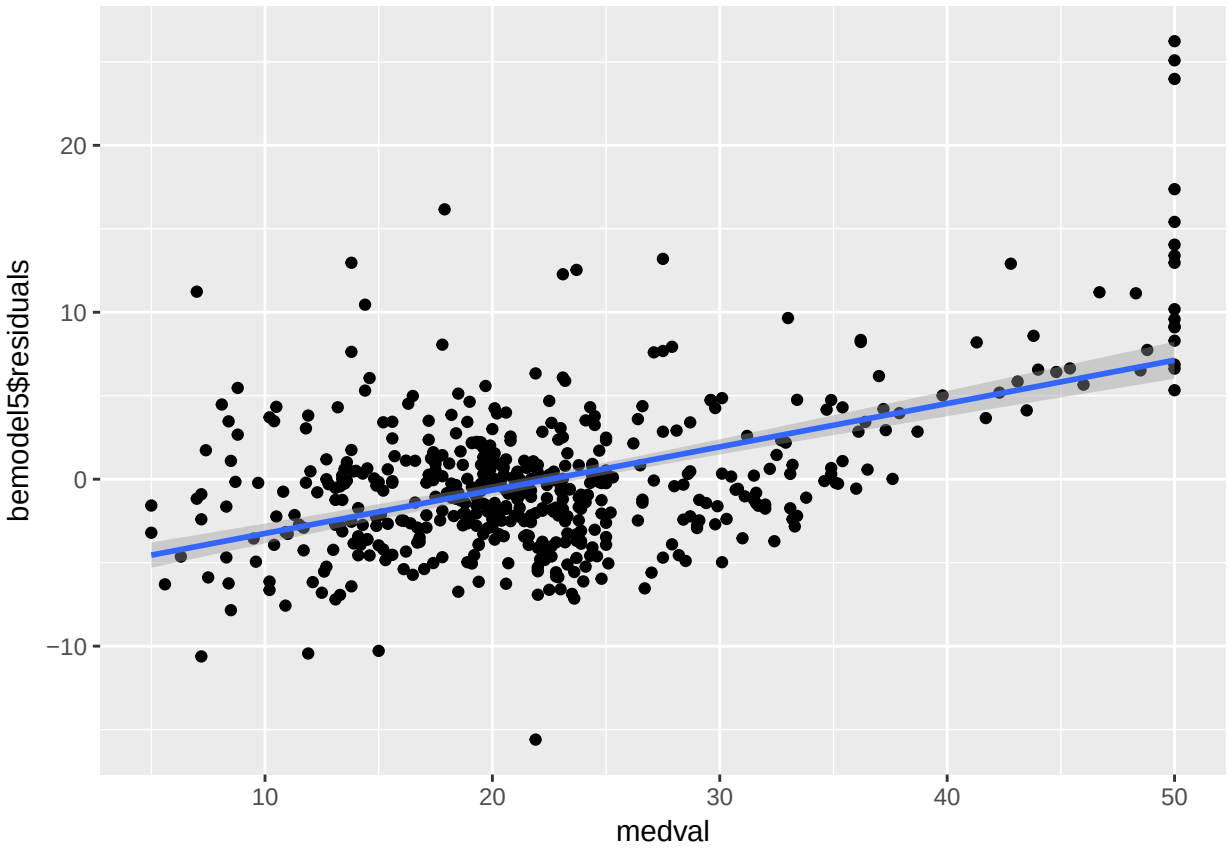
1.8 Do Assumptions Hold?

Evaluate the statistical assumptions in your regression analysis from (1.7) by performing a basic analysis of model residuals and any unusual observations (consult Openintro Statistics 7.2). Discuss any concerns you have about your model.

```
hist(bemodel5$residuals, breaks = 20)
```




```
qplot(medval, bemodel5$residuals, geom ="point") + geom_smooth(method ="lm")
```



The residuals are nearly normal, the least square line reflects linearity. However, the main concern is there exist a number of outliers in the data.

2. Diamonds' Price

Let's look at the *diamonds* dataset from *ggplot2* package. Your task is to find which parameters influence the price of diamonds.

I recommend to transform the ordered factors (such as *cut*, *clarity*) to unordered factors with a command like `factor(cut, ordered=FALSE)` in order to give more easily interpretable results.

```
data("diamonds")
#?diamonds
diamonds$cut <- factor(diamonds$cut, ordered = FALSE)
diamonds$clarity <- factor(diamonds$clarity, ordered = FALSE)
diamonds$color <- factor(diamonds$color, ordered = FALSE)
```

2.1 Describe the variables.

What do you think, which variables are relevant in determining the price? Describe your thought before you do any formal analysis.

The 3 factor variables Cut, Color and Clarity, define the quality of diamond cut, the color of diamond (from J (worst) to D (best)) and how clear the diamond is(I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best)). I believe these might influence the price of the diamonds. Also, since price by weight is relevant in consumer market, carat should definetely guide prices upto certain level.

2.2 Multiple regression

Select a number of variables you consider the most relevant. Estimate a multiple regression model in the form

$$\text{price}_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \epsilon_i.$$

Interpret the coefficient values.

- if you are able to, give the literal interpretation of the numeric value
- if there is no easy literal interpretation, broadly explain what it means, and interpret at least the sign.

```
dmds <- lm(price~cut+clarity+color, data = diamonds)
#summary(dmds)
```

Interpretation - For every rise in unit price of diamonds there will be an increase in the following varibales:

Premium Cut by 414.170 Clarity SI2 by 1354.896 Clarity SI1 by 316.532
Clarity VS2 by 293.973
Clarity VS1 by 92.755 Color F by 689.236
Color G by 1059.873 Color H by 1371.528 Color I by 2000.190 Color J by 2126.119

2.3 Other specifications

Select 2-3 different sets of explanatory variables or change the model specification in other ways, for instance by using log of the outcome or explanatory variables, adding interactions and squares, cubes of the variables, normalizing variables, or something else.

Which specification gives you the highest R^2? Comment your results.

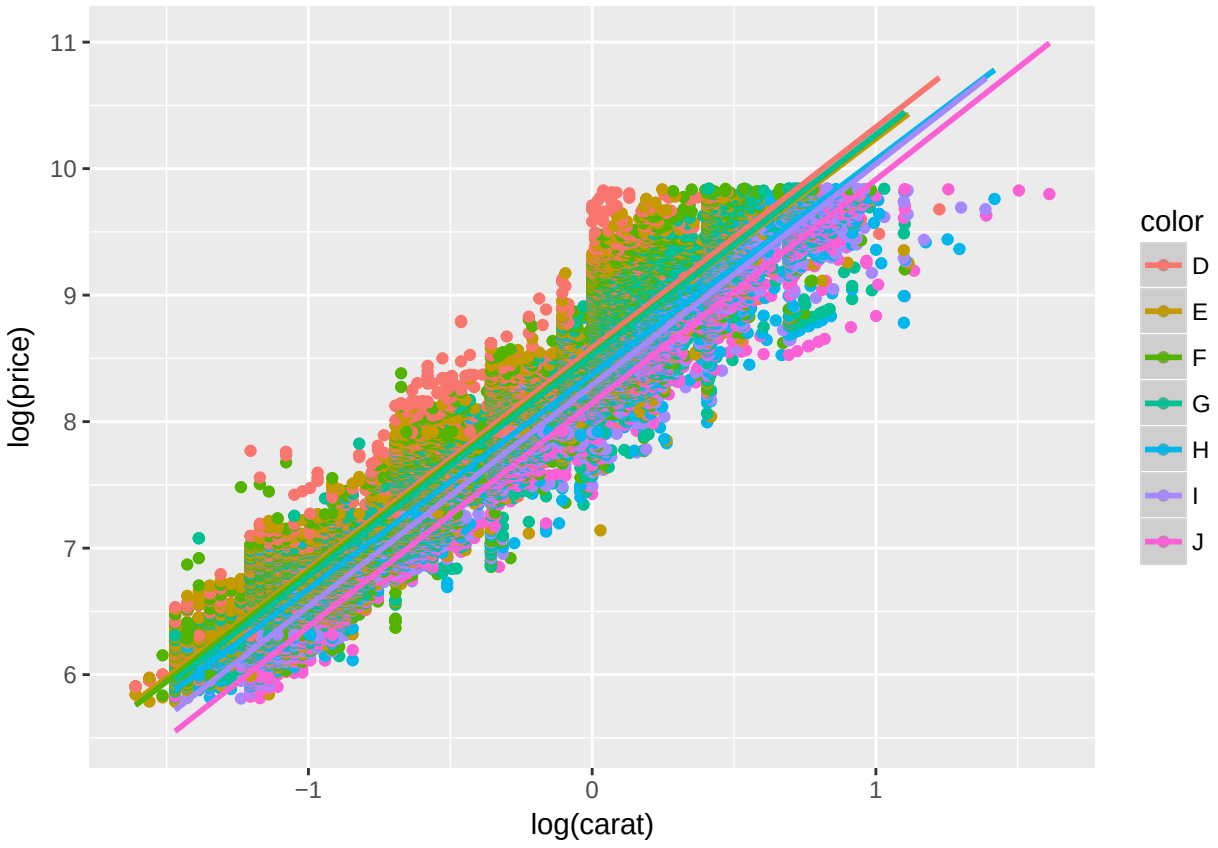
```
logdmds <- lm(log(price)~ cut+clarity+color, data = diamonds)
#summary(logdmds)

dimdmds <- lm(log(price)~ cut+clarity+color + (x*y*z), data = diamonds)
#summary(dimdmds)
```

On adding the dimensions of the diamonds to the regression analysis, I get the highest value of adjusted R-squared i.e. R = 0.9777

2.4 Visualize your best model

Visualize your best and your worst model's predictions on a true-predicted price scatterplot. Explain the differences.

```
ggplot(diamonds, aes(x = log(carat), y = log(price), color = color)) +
  geom_point() + geom_smooth(method="lm")
```



2.5 Residuals

- Show the distribution of residuals (difference between the actual and predicted price). Does it look normal?
- Analyze a few largest outliers. Anything special with those diamonds?

3. How much work?

Tell us, roughly how many hours did you spend on this homework.

For the amount of questions attempted, 9 hours to finish till Question 2.3.