INFX 573: Problem Set 6 - Regression

Chinmay Tatwawadi

Due: Tuesday, November 15, 2016

Collaborators:

Instructions:

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

1. Download the `problemset6.Rmd` file from Canvas. Open `problemset6.Rmd` in RStudio and supply your solutions to the assignment by editing `problemset6.Rmd`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. 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.
4. Collaboration on problem sets is acceptable, and even encouraged, but each student must turn in an individual write-up in his or her own words and his or her own work. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. 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.

Setup:

In this problem set you will need, at minimum, the following R packages.

```
# Load standard libraries
library(tidyverse)
library(dplyr) # added dplyr
library(car)
```

```
## Warning: package 'car' was built under R version 3.3.2
```

```
library(MASS) # Modern applied statistics functions
```

Housing Values in Suburbs of Boston

In this problem we will use the Boston dataset that is available in the `MASS` package. This dataset contains information about median house value for 506 neighborhoods in Boston, MA. Load this data and use it to answer the following questions.

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

```
#describe(Boston)
```

- This is a data frame with 14 Variables and 506 observations.
- It describes the hosuing values of the different suburbs of Boston
- The different variables are:

(a) crim per capita crime rate by town.
(b) zn proportion of residential land zoned for lots over 25,000 sq.ft.

(c) indus proportion of non-retail business acres per town.
(d) chas Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
(e) nox nitrogen oxides concentration (parts per 10 million).
(f) rm average number of rooms per dwelling.
(g) age proportion of owner-occupied units built prior to 1940.
(h) dis weighted mean of distances to five Boston employment centres.
(i) rad index of accessibility to radial highways.
(j) tax full-value property-tax rate per $10,000.
(k) ptratio pupil-teacher ratio by town.
(l) black 1000(Bk ??? 0.63)2 where Bk is the proportion of blacks by town.
(m) lstat lower status of the population (percent).
(n) medv median value of owner-occupied homes in $1000s.

2. 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.

   Answer: - medv i.e. the median value of homes is the response variable.

   - Many factors could affect the median value. Some of which could be:

   (a) crim per capita crime rate by town- less the crime, higher the value
   (b) nox nitrogen oxides concentration, less the conc, better the area
   (c) rm average number of rooms per dwelling, more the rooms, higher the value
   (d) dis weighted mean of distances to five Boston employment centres- closer the better
   (e) lstat lower status of the population- higher the status, better the value
   (f) black proportion of blacks by town - it will be interesting to see if there is a racial angle to this.
   (g) ptratio pupil-teacher ratio by town- lower the ratio, higher should be the value of the area.

```r
cor(Boston$medv,Boston)
```

```
##            crim        zn      indus      chas        nox        rm
## [1,] -0.3883046 0.3604453 -0.4837252 0.1752602 -0.4273208 0.6953599
##             age       dis        rad        tax    ptratio     black
## [1,] -0.3769546 0.2499287 -0.3816262 -0.4685359 -0.5077867 0.3334608
##           lstat medv
## [1,] -0.7376627    1
```

   So we can see that rm has the highest positive correlation. Whereas, lstat and ptratio have the highest negative correlation.

3. 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.

```r
# 1. medv and crim
lm.crim <- lm(data=Boston, medv~crim)
summary(lm.crim)
```

```
##
## Call:
## lm(formula = medv ~ crim, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.957  -5.449  -2.007   2.512  29.800
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 24.03311    0.40914   58.74   <2e-16 ***
## crim        -0.41519    0.04389   -9.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.484 on 504 degrees of freedom
## Multiple R-squared:  0.1508, Adjusted R-squared:  0.1491
## F-statistic: 89.49 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
# 2. medv and zn
lm.zn <- lm(data=Boston, medv~zn)
summary(lm.zn)
```

```
##
## Call:
## lm(formula = medv ~ zn, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.918  -5.518  -1.006   2.757  29.082
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 20.91758    0.42474  49.248   <2e-16 ***
## zn           0.14214    0.01638   8.675   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.587 on 504 degrees of freedom
## Multiple R-squared:  0.1299, Adjusted R-squared:  0.1282
## F-statistic: 75.26 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
# 3. medv and indus
lm.indus <- lm(data=Boston, medv~indus)
summary(lm.indus)
```

```
##
## Call:
## lm(formula = medv ~ indus, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.017  -4.917  -1.457   3.180  32.943
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29.75490    0.68345   43.54   <2e-16 ***
## indus       -0.64849    0.05226  -12.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.057 on 504 degrees of freedom
## Multiple R-squared:  0.234,  Adjusted R-squared:  0.2325
## F-statistic:   154 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 4. medv and chas
lm.chas <- lm(data=Boston, medv~chas)
summary(lm.chas)
```

```
##
## Call:
## lm(formula = medv ~ chas, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.094  -5.894  -1.417   2.856  27.906
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.0938     0.4176  52.902  < 2e-16 ***
## chas          6.3462     1.5880   3.996 7.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.064 on 504 degrees of freedom
## Multiple R-squared:  0.03072,    Adjusted R-squared:  0.02879
## F-statistic: 15.97 on 1 and 504 DF,  p-value: 7.391e-05
```

```r
# 5. medv and nox
lm.nox <- lm(data=Boston, medv~nox)
summary(lm.nox)
```

```
##
## Call:
## lm(formula = medv ~ nox, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.691  -5.121  -2.161   2.959  31.310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   41.346      1.811   22.83   <2e-16 ***
## nox          -33.916      3.196  -10.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.323 on 504 degrees of freedom
## Multiple R-squared:  0.1826, Adjusted R-squared:  0.181
## F-statistic: 112.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 6. medv and rm
lm.rm <- lm(data=Boston, medv~rm)
summary(lm.rm)
```

```
##
## Call:
## lm(formula = medv ~ rm, data = Boston)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -23.346  -2.547   0.090   2.986  39.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -34.671      2.650  -13.08   <2e-16 ***
## rm             9.102      0.419   21.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.616 on 504 degrees of freedom
## Multiple R-squared:  0.4835, Adjusted R-squared:  0.4825
## F-statistic: 471.8 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 7. medv and age
lm.age <- lm(data=Boston, medv~age)
summary(lm.age)
```

```
##
## Call:
## lm(formula = medv ~ age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.097  -5.138  -1.958   2.397  31.338
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 30.97868    0.99911  31.006   <2e-16 ***
## age         -0.12316    0.01348  -9.137   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.527 on 504 degrees of freedom
## Multiple R-squared:  0.1421, Adjusted R-squared:  0.1404
## F-statistic: 83.48 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 8. medv and dis
lm.dis <- lm(data=Boston, medv~dis)
summary(lm.dis)
```

```
##
## Call:
## lm(formula = medv ~ dis, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.016  -5.556  -1.865   2.288  30.377
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  18.3901     0.8174  22.499  < 2e-16 ***
## dis           1.0916     0.1884   5.795 1.21e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 8.914 on 504 degrees of freedom
## Multiple R-squared:  0.06246,    Adjusted R-squared:  0.0606
## F-statistic: 33.58 on 1 and 504 DF,  p-value: 1.207e-08
```

```r
# 9. medv and rad
lm.rad <- lm(data=Boston, medv~rad)
summary(lm.rad)
```

```
##
## Call:
## lm(formula = medv ~ rad, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.770  -5.199  -1.967   3.321  33.292
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 26.38213    0.56176  46.964   <2e-16 ***
## rad         -0.40310    0.04349  -9.269   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.509 on 504 degrees of freedom
## Multiple R-squared:  0.1456, Adjusted R-squared:  0.1439
## F-statistic: 85.91 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 10. medv and tax
lm.tax <- lm(data=Boston, medv~tax)
summary(lm.tax)
```

```
##
## Call:
## lm(formula = medv ~ tax, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.091  -5.173  -2.085   3.158  34.058
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 32.970654   0.948296   34.77   <2e-16 ***
## tax         -0.025568   0.002147  -11.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.133 on 504 degrees of freedom
## Multiple R-squared:  0.2195, Adjusted R-squared:  0.218
## F-statistic: 141.8 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
# 11. medv and ptratio
lm.ptratio <- lm(data=Boston, medv~ptratio)
summary(lm.ptratio)
```

```
##
## Call:
```

```
## lm(formula = medv ~ ptratio, data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.8342  -4.8262  -0.6426   3.1571  31.2303
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   62.345      3.029   20.58   <2e-16 ***
## ptratio       -2.157      0.163  -13.23   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.931 on 504 degrees of freedom
## Multiple R-squared:  0.2578, Adjusted R-squared:  0.2564
## F-statistic: 175.1 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
# 12. medv and black
lm.black <- lm(data=Boston, medv~black)
summary(lm.black)
```

```
##
## Call:
## lm(formula = medv ~ black, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.884  -4.862  -1.684   2.932  27.763
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.551034   1.557463   6.775 3.49e-11 ***
## black        0.033593   0.004231   7.941 1.32e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.679 on 504 degrees of freedom
## Multiple R-squared:  0.1112, Adjusted R-squared:  0.1094
## F-statistic: 63.05 on 1 and 504 DF,  p-value: 1.318e-14
```

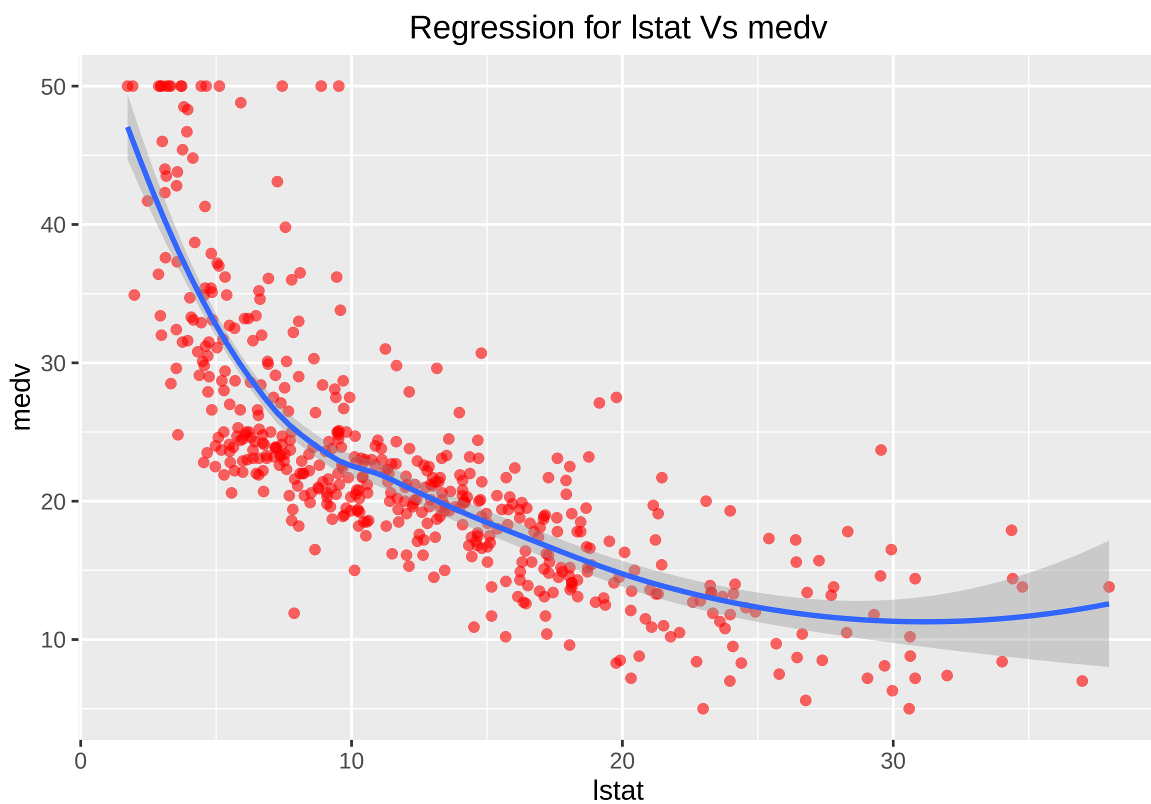
```
# 13. medv and lstat
lm.lstat <- lm(data=Boston, medv~lstat)
summary(lm.lstat)
```

```
##
## Call:
## lm(formula = medv ~ lstat, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.168  -3.990  -1.318   2.034  24.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.55384    0.56263   61.41   <2e-16 ***
## lstat       -0.95005    0.03873  -24.53   <2e-16 ***
```

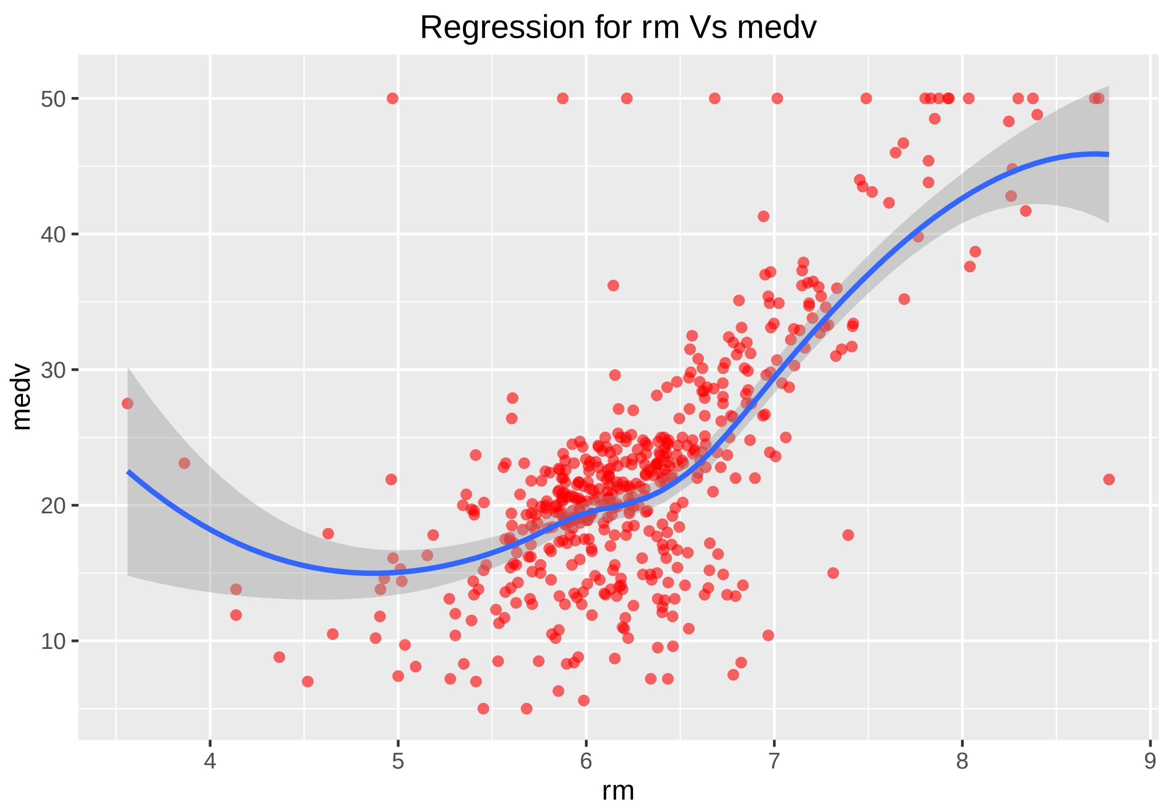
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441, Adjusted R-squared:  0.5432
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

Analysis: So we can see that rm and lstat have the highest significance: 1. Their p value are the almost 0. 2. Their r squared values are among the highest.

```r
#plot 1: lstat Vs Medv
ggplot(data=Boston, aes(x=lstat, y=medv)) +
  geom_point(col="red",alpha=0.6 ) +
    geom_smooth()+
  labs(title="Regression for lstat Vs medv", x="lstat", y="medv")
```




```r
#plot 2: rm Vs Medv
ggplot(data=Boston, aes(x=rm, y=medv)) +
  geom_point(col="red",alpha=0.6 ) +
    geom_smooth()+
  labs(title="Regression for rm Vs medv", x="rm", y="medv")
```



4. 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 $H_0 : \beta_j = 0$?

```
lm.allPred <- lm(data=Boston, medv~.)
summary(lm.allPred)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas         2.687e+00  8.616e-01   3.118 0.001925 **
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
```

```
## black        9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

Analysis: 1. We can see that p value for F-stat is significant. So there are definitely relationships between the variables. 2. Based on the p values, we can reject null hypothesis for: - crim, indus,chas,age,tax.

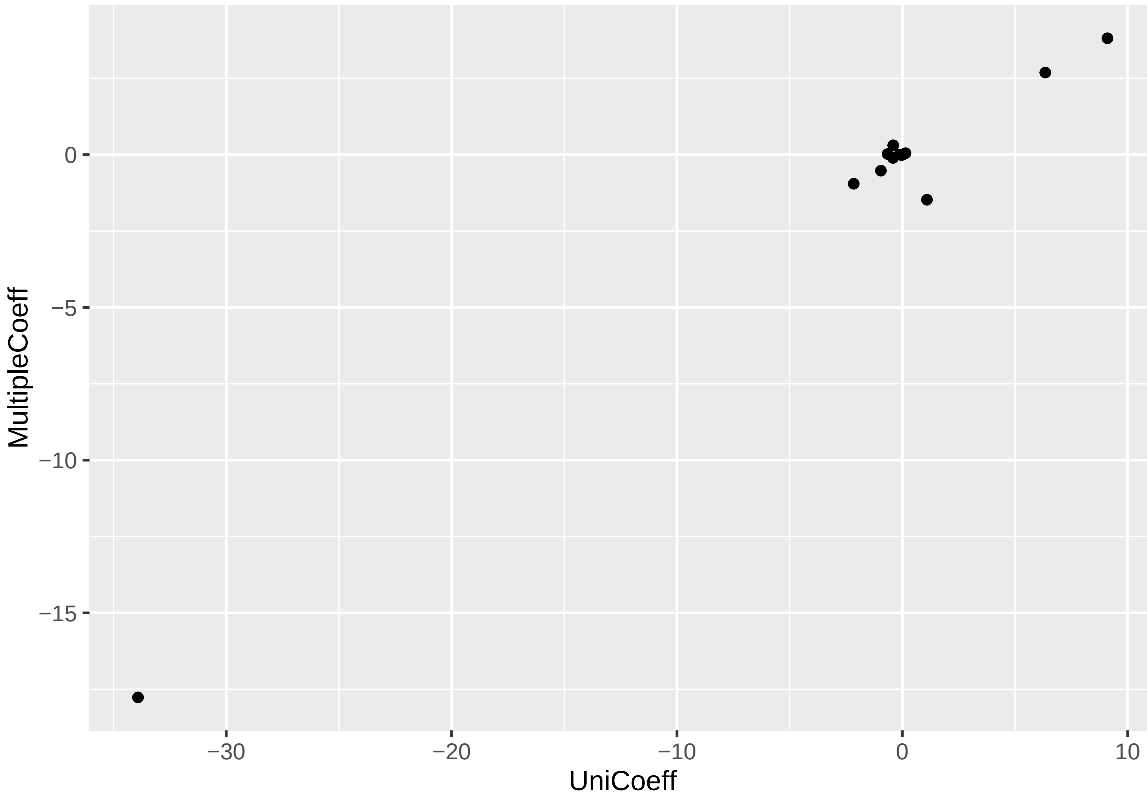
5. How do your results from (3) compare to your results from (4)? Create a plot displaying the univariate regression coefficients from (3) on the x-axis and the multiple regression coefficients from part (4) on the y-axis. Use this visualization to support your response.

```r
#Creating Data frame for multiple regression coeff:
df_allPred <- data.frame(lm.allPred$coefficients)

#Clean the data:
df_allPred <- data.frame(df_allPred[2:14,])

#Naming the cols
names(df_allPred) <- c("MultipleCoeff")

#Creating Data frame for simple regression coeff:
df_uniPred <- data.frame(c(lm.crim$coefficients[2], lm.zn$coefficients[2],
                          lm.indus$coefficients[2],lm.chas$coefficients[2],
                          lm.nox$coefficients[2],lm.rm$coefficients[2],lm.age$coefficients[2],
                          lm.dis$coefficients[2],lm.rad$coefficients[2],lm.tax$coefficients[2],
                          lm.ptratio$coefficients[2],lm.black$coefficients[2],
                          lm.lstat$coefficients[2] ))


#Naming the cols
names(df_uniPred) <- c("UniCoeff")


#Creating combined table:
ComTB <- data.frame(c(df_uniPred, df_allPred))


#displaying the table
ComTB
```

```
##        UniCoeff MultipleCoeff
## 1   -0.41519028 -1.080114e-01
## 2    0.14213999  4.642046e-02
## 3   -0.64849005  2.055863e-02
## 4    6.34615711  2.686734e+00
## 5  -33.91605501 -1.776661e+01
## 6    9.10210898  3.809865e+00
## 7   -0.12316272  6.922246e-04
## 8    1.09161302 -1.475567e+00
## 9   -0.40309540  3.060495e-01
## 10  -0.02556810 -1.233459e-02
```

```
## 11  -2.15717530 -9.527472e-01
## 12   0.03359306  9.311683e-03
## 13  -0.95004935 -5.247584e-01
```

```r
#plotting linear regression coefs on x axis & multiple regression coefs on y axis
ggplot(ComTB, aes(x = UniCoeff, y = MultipleCoeff)) +
  geom_point()
```



Analysis: 1. rm and chas are the 2 significant outliers. We already knew that rm was significant but the impact of chas is a new revelation. 2. lstat has little impact on response despite high correlation

6. 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:

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

```r
#fitting a 3rd degree exponential model for each predictor variable
poly_crim <- lm(medv~poly(crim,3), data = Boston)
poly_zn <- lm(medv~poly(zn,3), data = Boston)
poly_indus <- lm(medv~poly(indus,3), data = Boston)
poly_nox <- lm(medv~poly(nox,3), data = Boston)
poly_rm <- lm(medv~poly(rm,3), data = Boston)
poly_age <- lm(medv~poly(age,3), data = Boston)
poly_dis <- lm(medv~poly(dis,3), data = Boston)
poly_rad <- lm(medv~poly(rad,3), data = Boston)
poly_tax <- lm(medv~poly(tax,3), data = Boston)
poly_ptratio <- lm(medv~poly(ptratio,3), data = Boston)
poly_black <- lm(medv~poly(black,3), data = Boston)
poly_lstat <- lm(medv~poly(lstat,3), data = Boston)
```

```r
#checking summaries of each model to check the p value for the 3rd degree term
summary(poly_crim)
```

```
##
## Call:
## lm(formula = medv ~ poly(crim, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.983  -4.975  -1.940   2.881  33.391
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     22.5328     0.3627  62.124  < 2e-16 ***
## poly(crim, 3)1 -80.2545     8.1589  -9.836  < 2e-16 ***
## poly(crim, 3)2  50.2416     8.1589   6.158 1.51e-09 ***
## poly(crim, 3)3 -18.2905     8.1589  -2.242   0.0254 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.159 on 502 degrees of freedom
## Multiple R-squared:  0.2177, Adjusted R-squared:  0.213
## F-statistic: 46.57 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_zn)
```

```
##
## Call:
## lm(formula = medv ~ poly(zn, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.449  -5.549  -1.049   3.225  29.551
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   22.5328     0.3747  60.129  < 2e-16 ***
## poly(zn, 3)1  74.4966     8.4296   8.837  < 2e-16 ***
## poly(zn, 3)2 -19.2591     8.4296  -2.285   0.0227 *
## poly(zn, 3)3  33.5309     8.4296   3.978 7.98e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.43 on 502 degrees of freedom
## Multiple R-squared:  0.1649, Adjusted R-squared:  0.1599
## F-statistic: 33.05 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_indus)
```

```
##
## Call:
## lm(formula = medv ~ poly(indus, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -15.760  -4.725  -1.009   2.932  32.038
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        22.5328     0.3487  64.614  < 2e-16 ***
## poly(indus, 3)1 -99.9759     7.8445 -12.745  < 2e-16 ***
## poly(indus, 3)2  38.5184     7.8445   4.910 1.23e-06 ***
## poly(indus, 3)3 -18.6140     7.8445  -2.373    0.018 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.844 on 502 degrees of freedom
## Multiple R-squared:  0.2768, Adjusted R-squared:  0.2725
## F-statistic: 64.06 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_nox)
```

```
##
## Call:
## lm(formula = medv ~ poly(nox, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.104  -5.020  -2.144   2.747  32.416
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     22.5328     0.3682  61.199   <2e-16 ***
## poly(nox, 3)1 -88.3183     8.2823 -10.664   <2e-16 ***
## poly(nox, 3)2  13.8989     8.2823   1.678   0.0939 .
## poly(nox, 3)3  16.9686     8.2823   2.049   0.0410 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.282 on 502 degrees of freedom
## Multiple R-squared:  0.1939, Adjusted R-squared:  0.189
## F-statistic: 40.24 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_rm)
```

```
##
## Call:
## lm(formula = medv ~ poly(rm, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.102  -2.674   0.569   3.011  35.911
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.5328     0.2716  82.952  < 2e-16 ***
## poly(rm, 3)1 143.7164     6.1103  23.520  < 2e-16 ***
## poly(rm, 3)2  52.6526     6.1103   8.617  < 2e-16 ***
## poly(rm, 3)3 -23.3832     6.1103  -3.827 0.000146 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.11 on 502 degrees of freedom
## Multiple R-squared:  0.5612, Adjusted R-squared:  0.5586
## F-statistic:   214 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_age)
```

```
##
## Call:
## lm(formula = medv ~ poly(age, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.443  -4.909  -2.234   2.185  32.944
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.5328     0.3766  59.830   <2e-16 ***
## poly(age, 3)1 -77.9087     8.4717  -9.196   <2e-16 ***
## poly(age, 3)2 -23.3290     8.4717  -2.754   0.0061 **
## poly(age, 3)3  -8.6148     8.4717  -1.017   0.3097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.472 on 502 degrees of freedom
## Multiple R-squared:  0.1566, Adjusted R-squared:  0.1515
## F-statistic: 31.06 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_dis)
```

```
##
## Call:
## lm(formula = medv ~ poly(dis, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.571  -5.242  -2.037   2.397  34.769
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.5328     0.3879  58.082  < 2e-16 ***
## poly(dis, 3)1  51.6551     8.7267   5.919 6.00e-09 ***
## poly(dis, 3)2 -37.5859     8.7267  -4.307 1.99e-05 ***
## poly(dis, 3)3  20.1322     8.7267   2.307   0.0215 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.727 on 502 degrees of freedom
## Multiple R-squared:  0.105,  Adjusted R-squared:  0.09968
## F-statistic: 19.64 on 3 and 502 DF,  p-value: 4.736e-12
```

```r
summary(poly_rad)
```

```
##
## Call:
```

```
## lm(formula = medv ~ poly(rad, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.630  -5.151  -2.017   3.169  33.594
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.5328     0.3721  60.557  < 2e-16 ***
## poly(rad, 3)1 -78.8742     8.3700  -9.423  < 2e-16 ***
## poly(rad, 3)2 -21.4799     8.3700  -2.566 0.010568 *
## poly(rad, 3)3 -29.4095     8.3700  -3.514 0.000482 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.37 on 502 degrees of freedom
## Multiple R-squared:  0.1767, Adjusted R-squared:  0.1718
## F-statistic: 35.91 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
summary(poly_tax)
```

```
##
## Call:
## lm(formula = medv ~ poly(tax, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.109  -4.952  -1.878   2.957  33.694
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.5328     0.3608  62.460   <2e-16 ***
## poly(tax, 3)1 -96.8366     8.1150 -11.933   <2e-16 ***
## poly(tax, 3)2  14.9703     8.1150   1.845   0.0657 .
## poly(tax, 3)3  -7.5431     8.1150  -0.930   0.3531
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.115 on 502 degrees of freedom
## Multiple R-squared:  0.2261, Adjusted R-squared:  0.2215
## F-statistic: 48.89 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
summary(poly_ptratio)
```

```
##
## Call:
## lm(formula = medv ~ poly(ptratio, 3), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.7795  -5.0364  -0.9778   3.4766  31.1636
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          22.5328     0.3511  64.173   <2e-16 ***
```

```
## poly(ptratio, 3)1 -104.9490     7.8984 -13.287   <2e-16 ***
## poly(ptratio, 3)2  -12.6952     7.8984  -1.607    0.109
## poly(ptratio, 3)3  -14.9472     7.8984  -1.892    0.059 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.898 on 502 degrees of freedom
## Multiple R-squared:  0.2669, Adjusted R-squared:  0.2625
## F-statistic: 60.91 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(poly_black)
```

```
##
## Call:
## lm(formula = medv ~ poly(black, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.005  -4.802  -1.613   2.852  28.051
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      22.5328     0.3861  58.360  < 2e-16 ***
## poly(black, 3)1  68.9194     8.6851   7.935 1.38e-14 ***
## poly(black, 3)2   9.1467     8.6851   1.053    0.293
## poly(black, 3)3  -4.0541     8.6851  -0.467    0.641
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.685 on 502 degrees of freedom
## Multiple R-squared:  0.1135, Adjusted R-squared:  0.1082
## F-statistic: 21.43 on 3 and 502 DF,  p-value: 4.463e-13
```

```r
summary(poly_lstat)
```

```
##
## Call:
## lm(formula = medv ~ poly(lstat, 3), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.5441  -3.7122  -0.5145   2.4846  26.4153
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       22.5328     0.2399  93.937  < 2e-16 ***
## poly(lstat, 3)1 -152.4595     5.3958 -28.255  < 2e-16 ***
## poly(lstat, 3)2   64.2272     5.3958  11.903  < 2e-16 ***
## poly(lstat, 3)3  -27.0511     5.3958  -5.013 7.43e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.396 on 502 degrees of freedom
## Multiple R-squared:  0.6578, Adjusted R-squared:  0.6558
## F-statistic: 321.7 on 3 and 502 DF,  p-value: < 2.2e-16
```

Analysis: Variables age, tax, ptratio, black have insignificant p values so we can conclude that there is no non-linear correlation. For the other variables, p value is high so there is a non linear correlation.

7. Consider performing a stepwise model selection procedure to determine the bets fit model. Discuss your results. How is this model different from the model in (4)?

```
#stepwise model selection:
lm_stepwiseSel <- stepAIC(lm.allPred ,direction="both")
```

```
## 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
## + age      1      0.06 11079 1589.6
## - 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
```

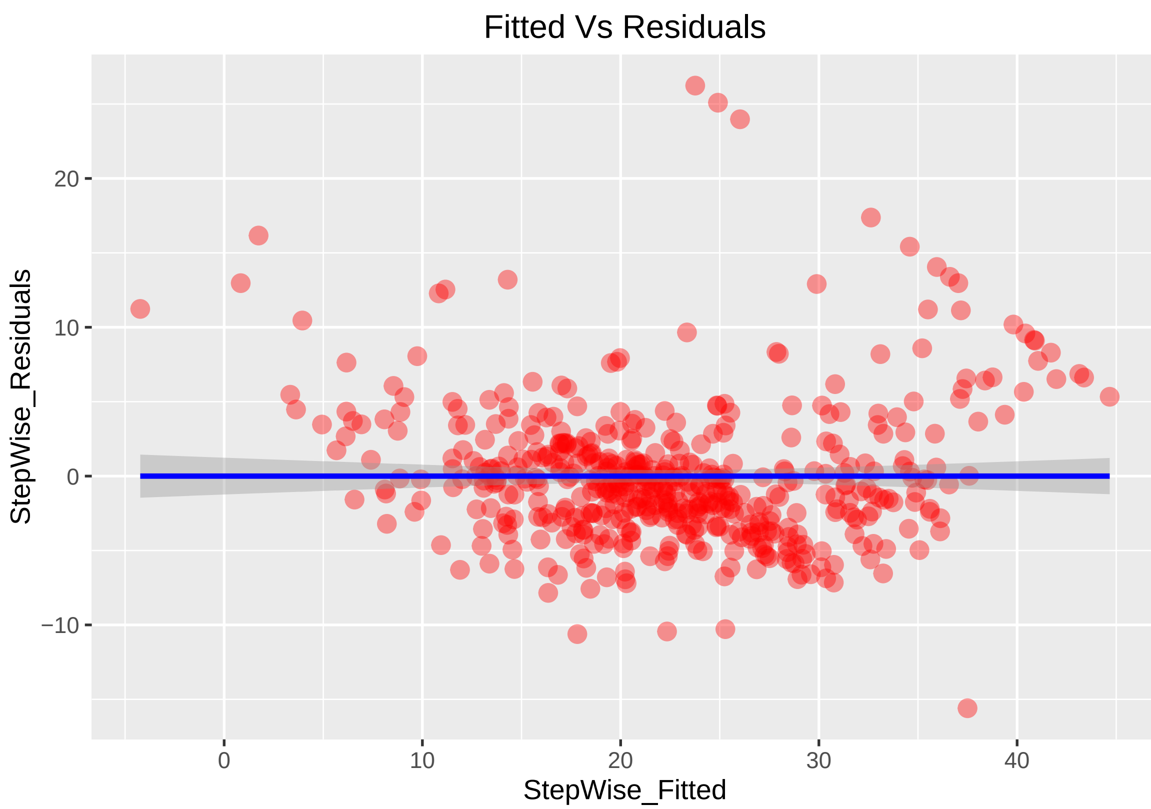
```
## + indus    1      2.52 11079 1587.7
## + age      1      0.06 11081 1587.8
## - chas     1    227.21 11309 1594.0
## - crim     1    245.37 11327 1594.8
## - zn       1    257.82 11339 1595.4
## - black    1    270.82 11352 1596.0
## - tax      1    273.62 11355 1596.1
## - rad      1    500.92 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
```

```
lm_stepwiseSel$anova
```

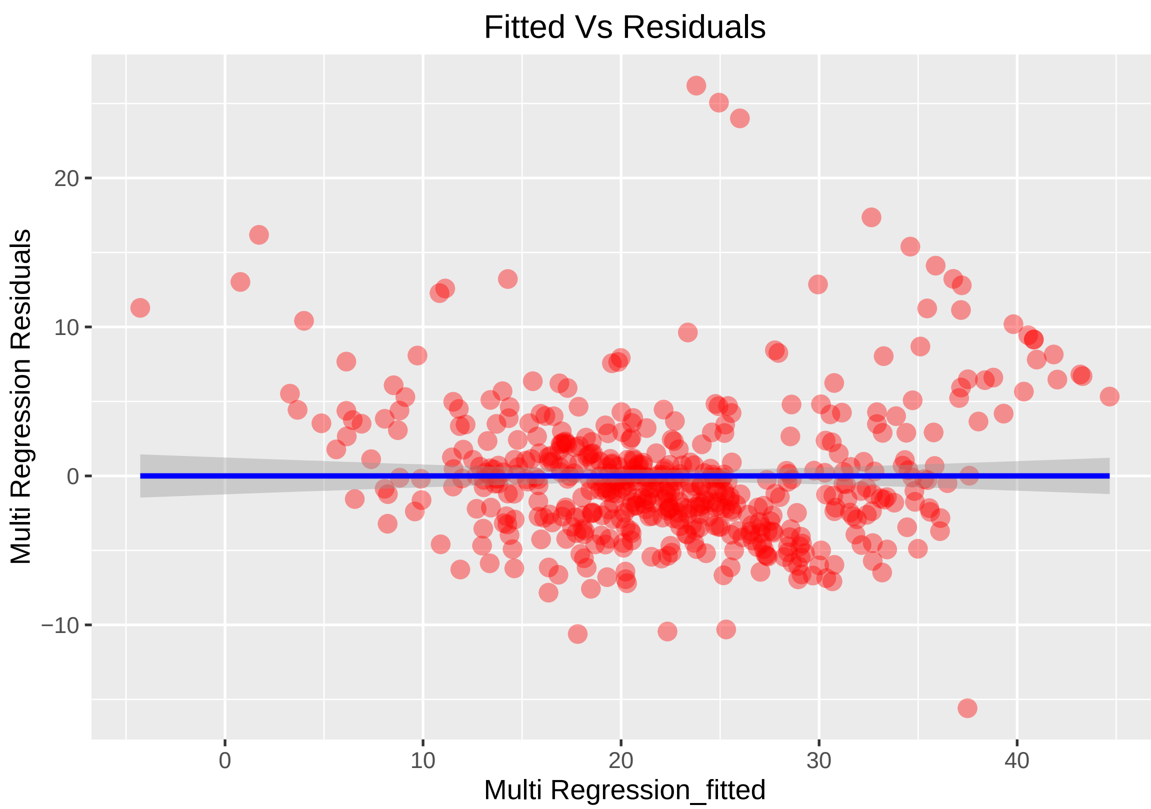
```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat
##
## Final Model:
## medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio +
##     black + lstat
##
##
##      Step Df   Deviance Resid. Df Resid. Dev      AIC
## 1                             492   11078.78 1589.643
## 2   - age  1 0.06183435       493   11078.85 1587.646
## 3 - indus  1 2.51754013       494   11081.36 1585.761
```

Analysis: In Stepwise selection model, we remove the variables age,indus as this gives us the least AIC value. Earlier we rejected null hypothesis for: crim, indus,chas,age,tax. age,indus has the highest p values in the group as well. This model has 2 less variables. Let us try to plot residuals Vs Fitted for both the models and see which one is better:

```
ggplot(data = lm_stepwiseSel, aes(x=lm_stepwiseSel$fitted.values, y=lm_stepwiseSel$residuals) )+
  geom_point(alpha=0.4,size=3,col="red") +
  geom_smooth(method="lm",col="blue")+
  labs(title="Fitted Vs Residuals", x="StepWise_Fitted", y="StepWise_Residuals")
```



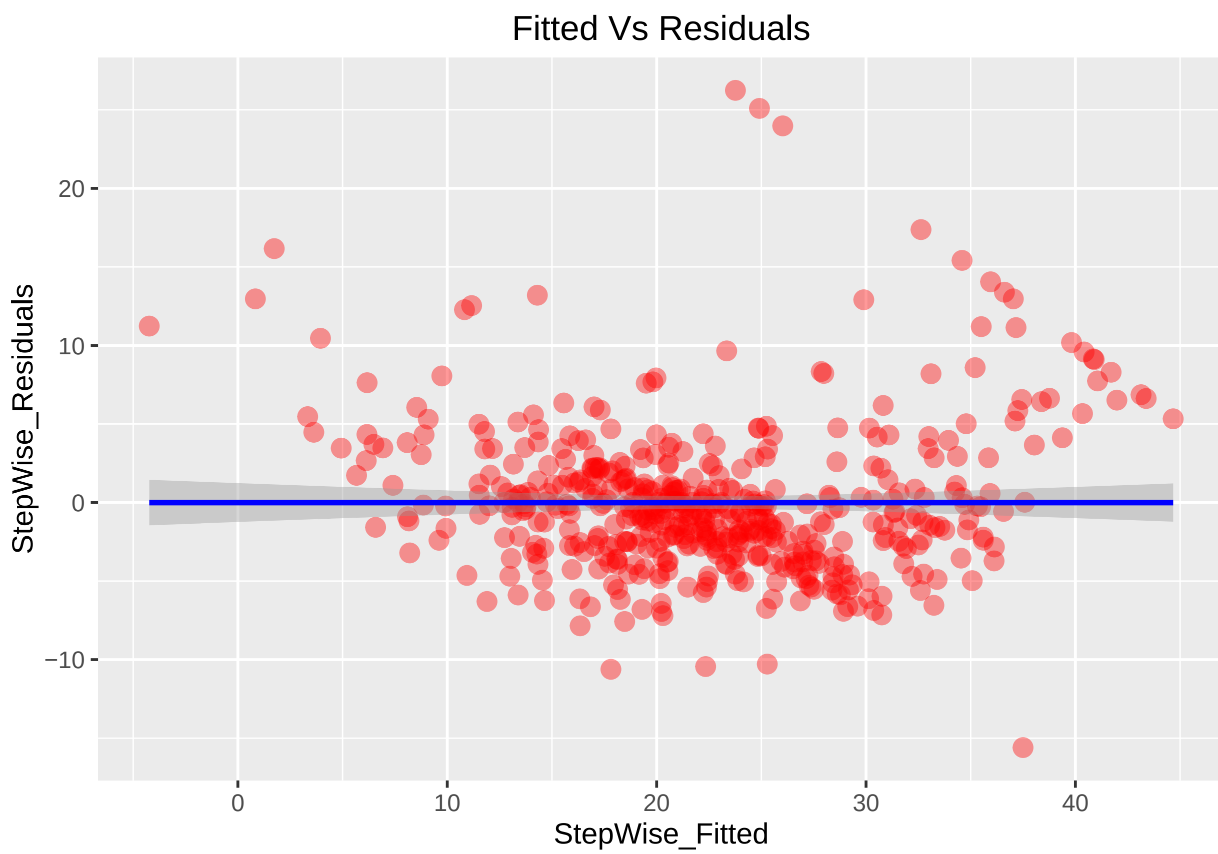

```
ggplot(data=lm.allPred, aes(x=lm.allPred$fitted.values, y=lm.allPred$residuals ) )+
  geom_point(alpha=0.4,size=3,col="red") +
  geom_smooth(method="lm",col="blue")+
  labs(title="Fitted Vs Residuals", x="Multi Regression_fitted", y="Multi Regression Residuals")
```



So, it seems that the 2 models are almost the same. There is hardly any difference in plots here.

8. Evaluate the statistical assumptions in your regression analysis from (7) by performing a basic analysis of model residuals and any unusual observations. Discuss any concerns you have about your model.

```
ggplot(data = lm_stepwiseSel, aes(x=lm_stepwiseSel$fitted.values, y=lm_stepwiseSel$residuals) )+
  geom_point(alpha=0.4,size=3,col="red") +
  geom_smooth(method="lm",col="blue")+
  labs(title="Fitted Vs Residuals", x="StepWise_Fitted", y="StepWise_Residuals")
```



Analysis:

1. From the model's Fitted Vs residuals curve, we see that most of the points are concentrated near the mean line.
2. We can also see that most value between 10 and 30 seem to have negative residual whereas values less than 10 and greater than 30 tend to have positive residual values. This means that the variation is not constant throughout the plot and violates the homoscedasticity rule.
3. We cann also see a few outliers on the top right. They seem to form a pattern. This pattern can be observed on the similar model for simple linear regression as well.
4. The overall model seems to be non linear and has a certain amount of curvature.

Concerns: 1. I have my reservations for this model as it seems to be violating a few rules- nonlinearity, homoscedasticity. 2. Also, the outliers in the top right might suggest that regression model might not be the best model. 3. A counter can be argued by saying that the model doesn't seem to be overfitting and the model is robust.