HW5

Nick Crites

2022-10-07

#1

```
library(faraway)
data('sat')
head(sat)
```

```
##            expend ratio salary takers verbal math total
## Alabama     4.405  17.2 31.144      8    491  538  1029
## Alaska      8.963  17.6 47.951     47    445  489   934
## Arizona     4.778  19.3 32.175     27    448  496   944
## Arkansas    4.459  17.1 28.934      6    482  523  1005
## California  4.992  24.0 41.078     45    417  485   902
## Colorado    5.443  18.4 34.571     29    462  518   980
```

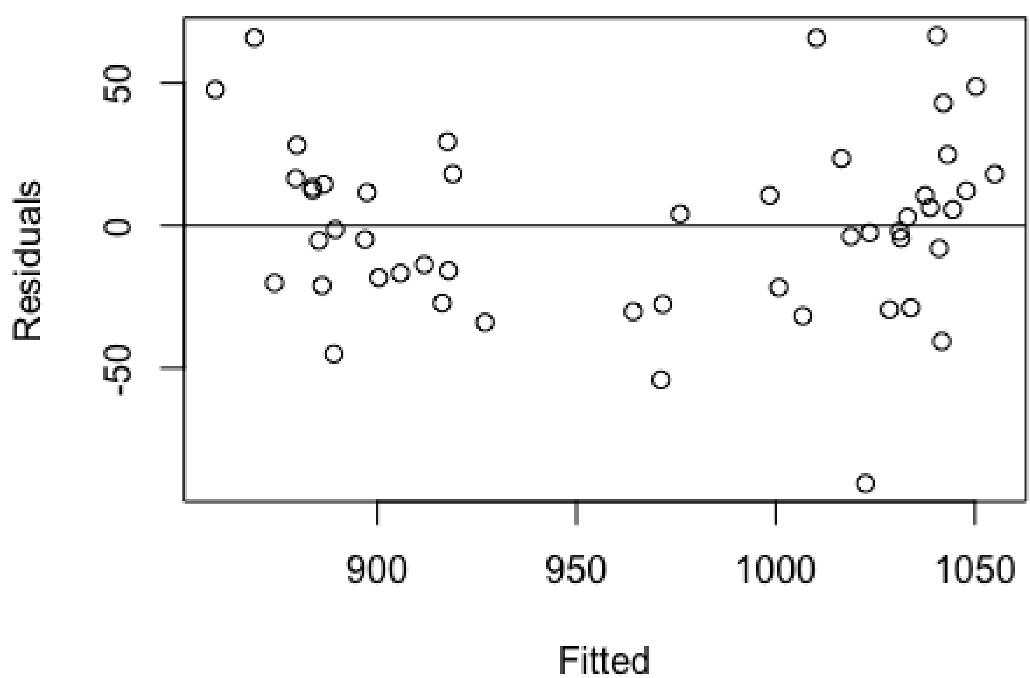
```
lm1 <- lm(total ~ expend + salary + ratio + takers, sat)
summary(lm1)
```

```
##
## Call:
## lm(formula = total ~ expend + salary + ratio + takers, data = sat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -90.531 -20.855  -1.746  15.979  66.571
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1045.9715    52.8698  19.784  < 2e-16 ***
## expend         4.4626    10.5465   0.423    0.674
## salary         1.6379     2.3872   0.686    0.496
## ratio         -3.6242     3.2154  -1.127    0.266
## takers        -2.9045     0.2313 -12.559 2.61e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.7 on 45 degrees of freedom
## Multiple R-squared:  0.8246, Adjusted R-squared:  0.809
## F-statistic: 52.88 on 4 and 45 DF,  p-value: < 2.2e-16
```

As we can see via the regression analysis, only the intercept and the takers are relevant predictors and offer any modeling value. The significance of the intercept term is likely a result of other significant predictors being left out, so to improve the model we should add other predictor, like math or salary, to improve the model fit.

#1A

```
plot(fitted(lm1), residuals(lm1), xlab = "Fitted", ylab = "Residuals")
abline(h=0)
```

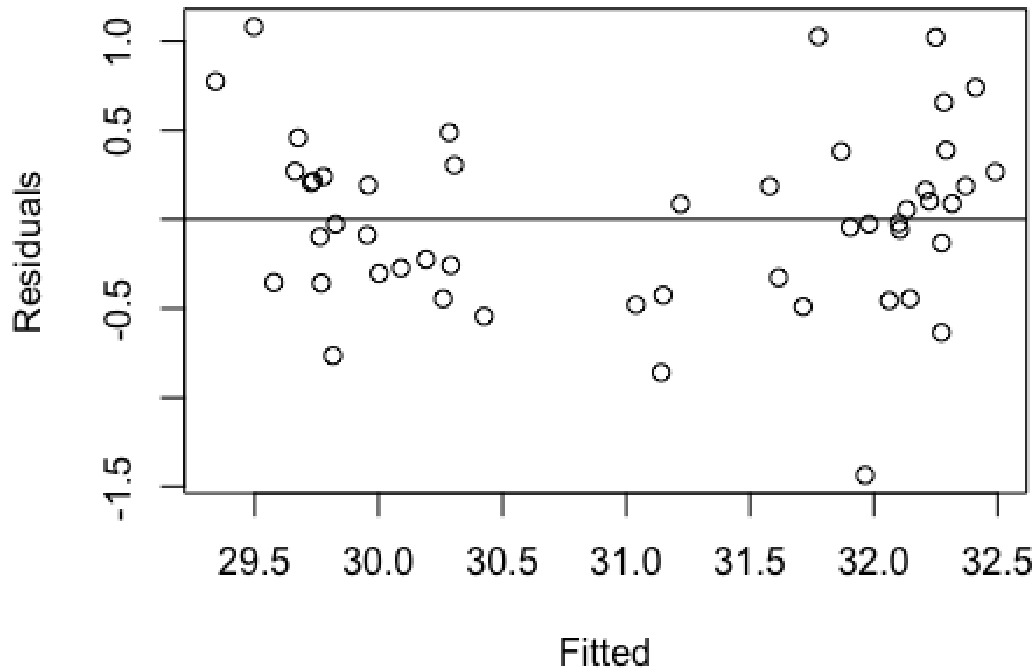


The above plot displays non-constant variance. We can experiment with a log transform to see if that effects variance:

```
lm2 <- lm(sqrt(total) ~ expend + salary + ratio + takers, data = sat)
summary(lm2)

##
## Call:
## lm(formula = sqrt(total) ~ expend + salary + ratio + takers,
##     data = sat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.43610 -0.34707 -0.02486  0.25943  1.08084
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 32.297536   0.840140  38.443   <2e-16 ***
## expend       0.075430   0.167592   0.450    0.655
## salary       0.026203   0.037935   0.691    0.493
## ratio       -0.056489   0.051095  -1.106    0.275
## takers      -0.046730   0.003675 -12.716   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5197 on 45 degrees of freedom
## Multiple R-squared:  0.8278, Adjusted R-squared:  0.8125
## F-statistic: 54.08 on 4 and 45 DF,  p-value: < 2.2e-16
```

```
plot(fitted(lm2), residuals(lm2), xlab = "Fitted", ylab = "Residuals")
abline(h=0)
```

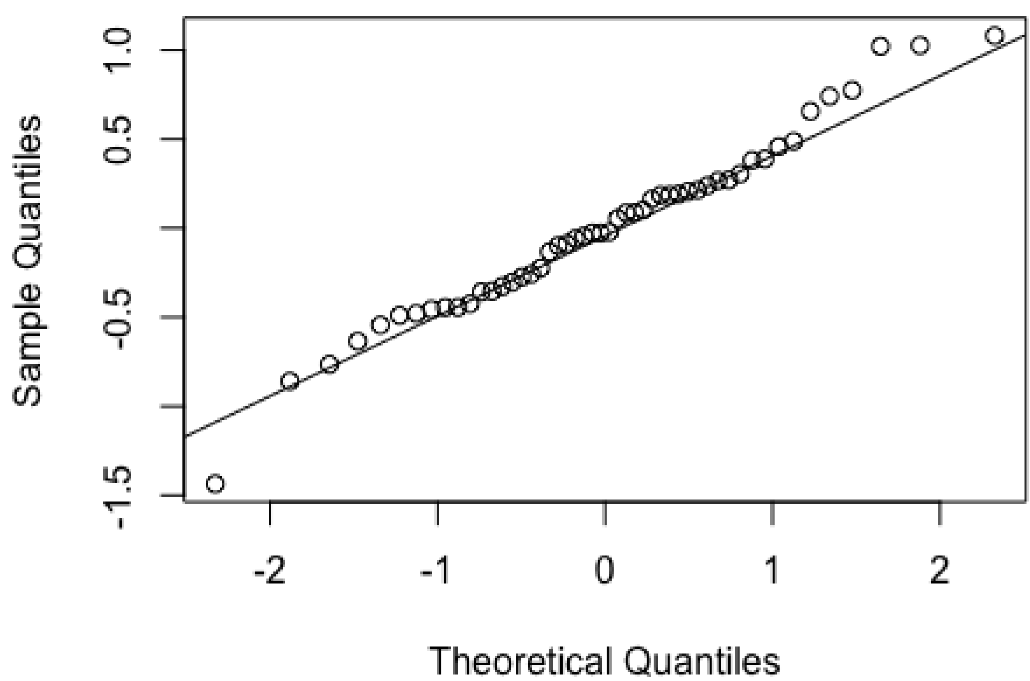


This slightly improved the variance of the model but not significantly.

1B

```
qqnorm(residuals(lm2), ylba = "Residuals", main = "")

## Warning in plot.window(...): "ylba" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "ylba" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "ylba" is not
a
## graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "ylba" is not
a
## graphical parameter

## Warning in box(...): "ylba" is not a graphical parameter

## Warning in title(...): "ylba" is not a graphical parameter

qqline(residuals(lm2))
```



From x-values 1 to 2, the plot does not look normal. Let’s try the Shapiro-Wilkes test for a definitive answer on normality:

```
shapiro.test(residuals(lm2))

##
##  Shapiro-Wilk normality test
##
```

```
## data:  residuals(lm2)
## W = 0.97976, p-value = 0.5421
```

Based on the shapiro-wilkes test, we fail to reject the null hypothesis. The normality assumption is satisfied.

1C

```
hatvalues(lm2) > 2 * mean(hatvalues(lm2))
```

```
##        Alabama         Alaska        Arizona       Arkansas     California
##          FALSE          FALSE          FALSE          FALSE           TRUE
##       Colorado    Connecticut       Delaware        Florida        Georgia
##          FALSE           TRUE          FALSE          FALSE          FALSE
##         Hawaii          Idaho       Illinois        Indiana           Iowa
##          FALSE          FALSE          FALSE          FALSE          FALSE
##         Kansas       Kentucky      Louisiana          Maine       Maryland
##          FALSE          FALSE          FALSE          FALSE          FALSE
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri
##          FALSE          FALSE          FALSE          FALSE          FALSE
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey
##          FALSE          FALSE          FALSE          FALSE           TRUE
##     New Mexico       New York North Carolina   North Dakota           Ohio
##          FALSE          FALSE          FALSE          FALSE          FALSE
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##          FALSE          FALSE          FALSE          FALSE          FALSE
##   South Dakota      Tennessee          Texas           Utah        Vermont
##          FALSE          FALSE          FALSE           TRUE          FALSE
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##          FALSE          FALSE          FALSE          FALSE          FALSE
```

As we can see from above, California, New Jersey, Connecticut, and Utah are the high leverage points.

1D

```
rstandard(lm2)[abs(rstandard(lm2)) > 2]
```

```
## New Hampshire  North Dakota          Utah West Virginia
##      2.171792      2.051525      2.348234     -2.853503
```

As we can see above, New Hampshire, North Dakota, Utah, and West Virginia are outlier states.

1E

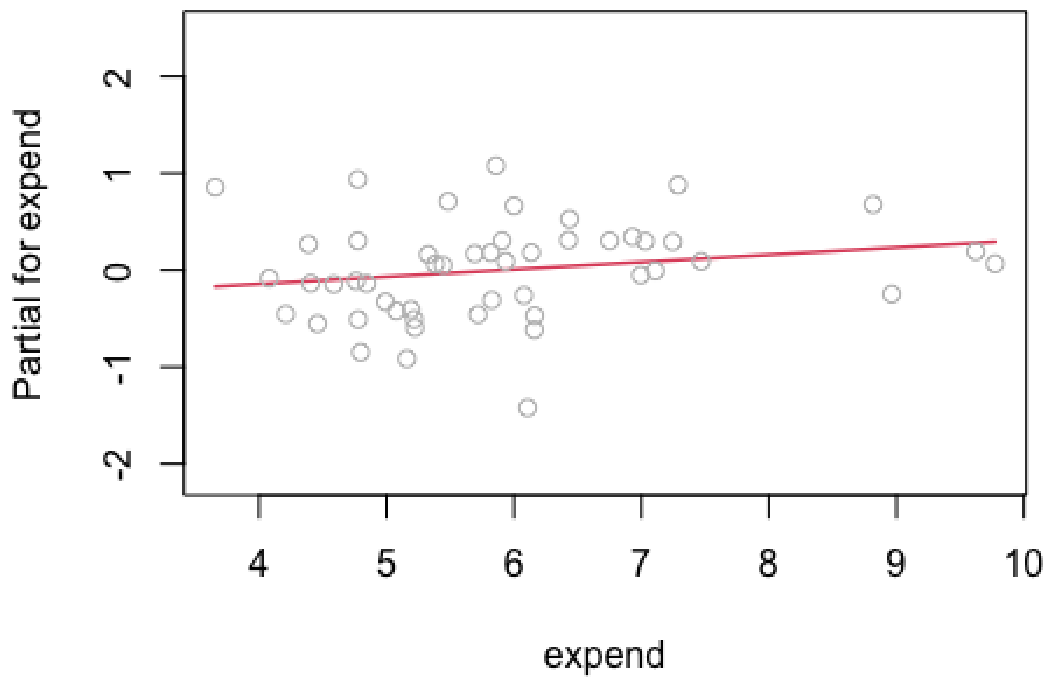
```
cooks.distance(lm2)[44] > 4 / length(cooks.distance(lm2))
```

```
## Utah
## TRUE
```

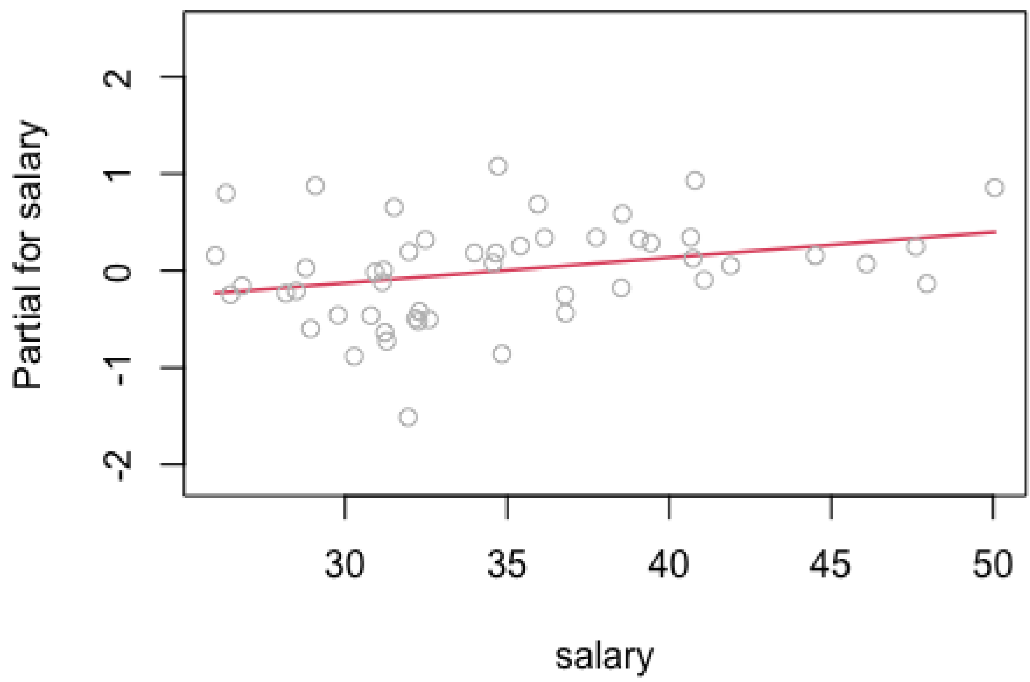
Based on the cooks distance, we can see that Utah is an influential point and the 44th observation. Because this is the only outlier AND leverage point, this can be the obly influential point we need to test.

1F

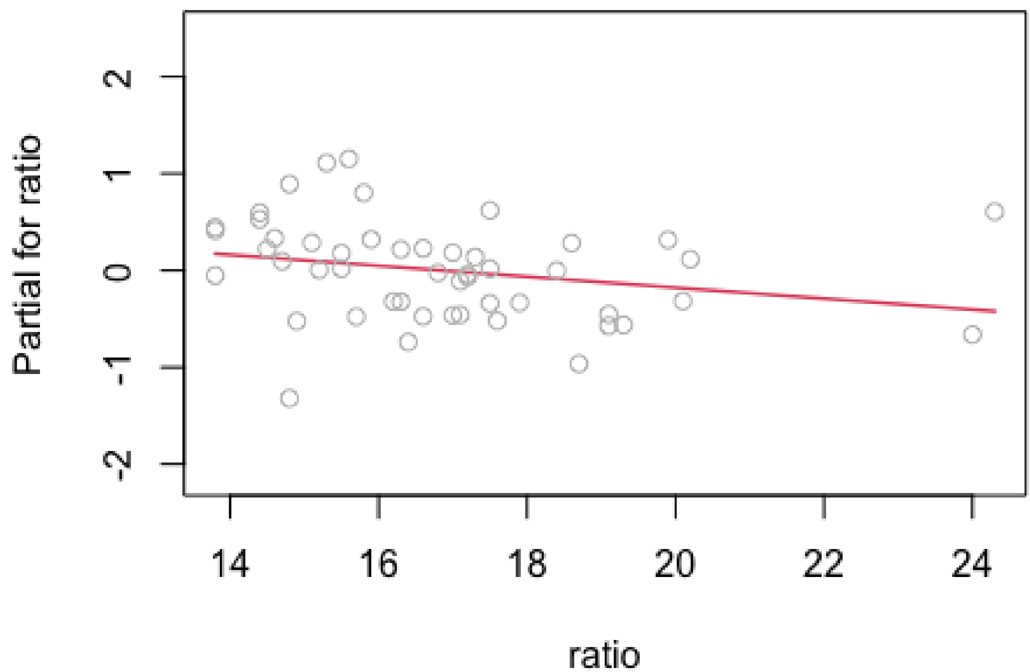
```
termplot(lm2, partial.resid = T, terms = 1:4)
```



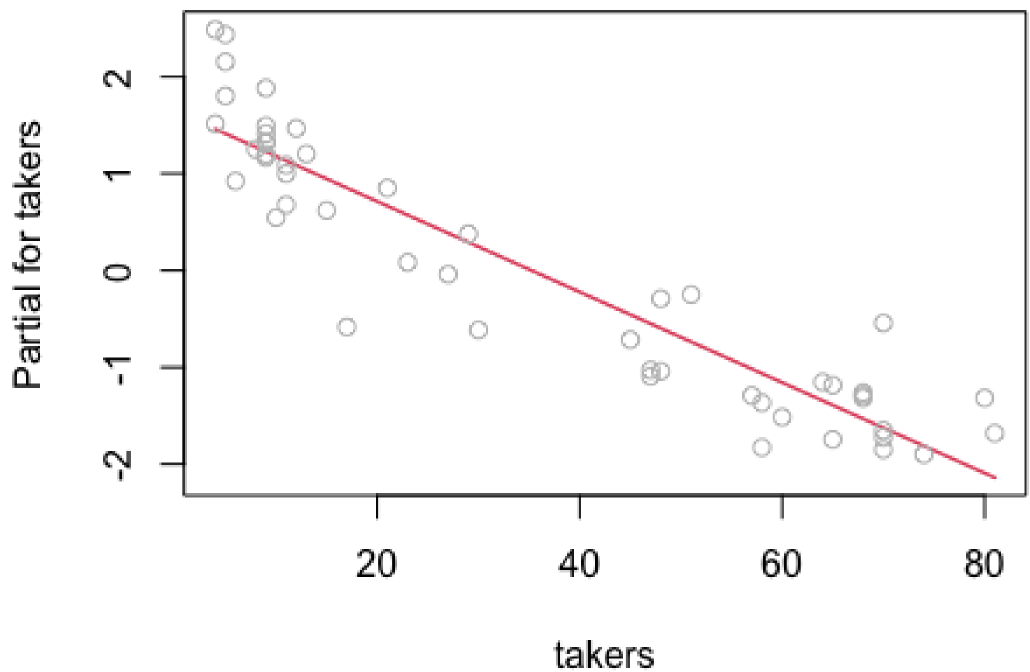
Partial for expend
2
1
0
-1
-2
4
5
6
7
8
9
10
expend



Partial for salary
2
1
0
-1
-2
30
35
40
45
50
salary



Partial for ratio
2
1
0
-1
-2
14
16
18
20
22
24
ratio



We can see from the above term plots that all predictors appear to have at least some predictive value. Expend, salary, and ratio appear to have quite little explanatory power, however - due to the flatness of the line. Takers, however, appears to have the most explanatory power; this is confirmed by our regression summary in part A. # 3

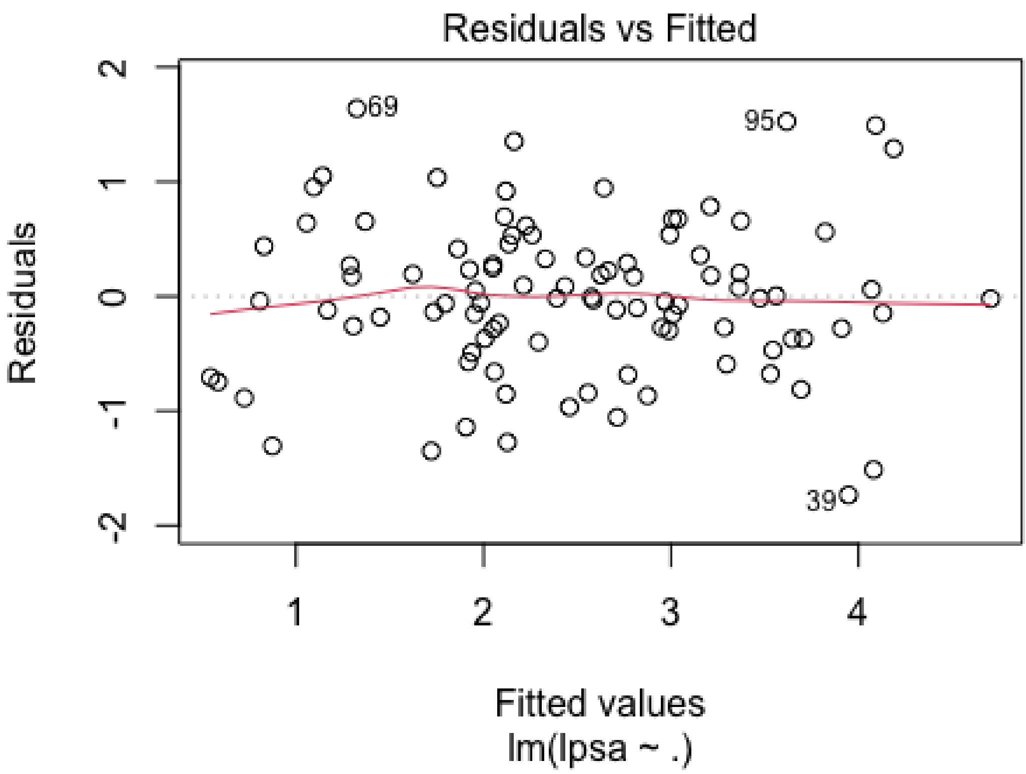
```
library(faraway)
data('prostate')
head(prostate)

##       lcavol lweight age      lbph svi      lcp gleason pgg45     lpsa
## 1 -0.5798185  2.7695  50 -1.386294   0 -1.38629       6     0 -0.43078
## 2 -0.9942523  3.3196  58 -1.386294   0 -1.38629       6     0 -0.16252
## 3 -0.5108256  2.6912  74 -1.386294   0 -1.38629       7    20 -0.16252
## 4 -1.2039728  3.2828  58 -1.386294   0 -1.38629       6     0 -0.16252
## 5  0.7514161  3.4324  62 -1.386294   0 -1.38629       6     0  0.37156
## 6 -1.0498221  3.2288  50 -1.386294   0 -1.38629       6     0  0.76547

lmod1 <- lm(lpsa ~ ., prostate)
summary(lmod1)

##
## Call:
## lm(formula = lpsa ~ ., data = prostate)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.7331 -0.3713 -0.0170  0.4141  1.6381 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.669337   1.296387   0.516  0.60693    
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 ** 
## age         -0.019637   0.011173  -1.758  0.08229 .  
## lbph         0.107054   0.058449   1.832  0.07040 .  
## svi          0.766157   0.244309   3.136  0.00233 ** 
## lcp         -0.105474   0.091013  -1.159  0.24964    
## gleason      0.045142   0.157465   0.287  0.77503    
## pgg45        0.004525   0.004421   1.024  0.30886    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234 
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

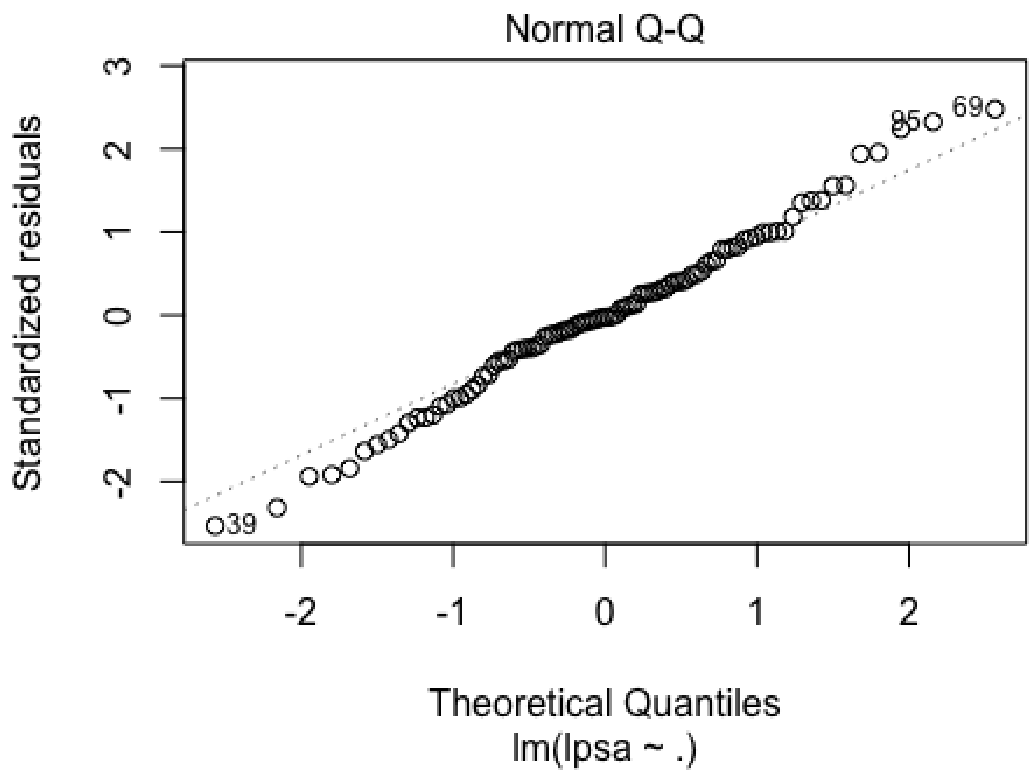
Based on the regression summary above, `lcp`, age, and `lpsa` are significant predictors.

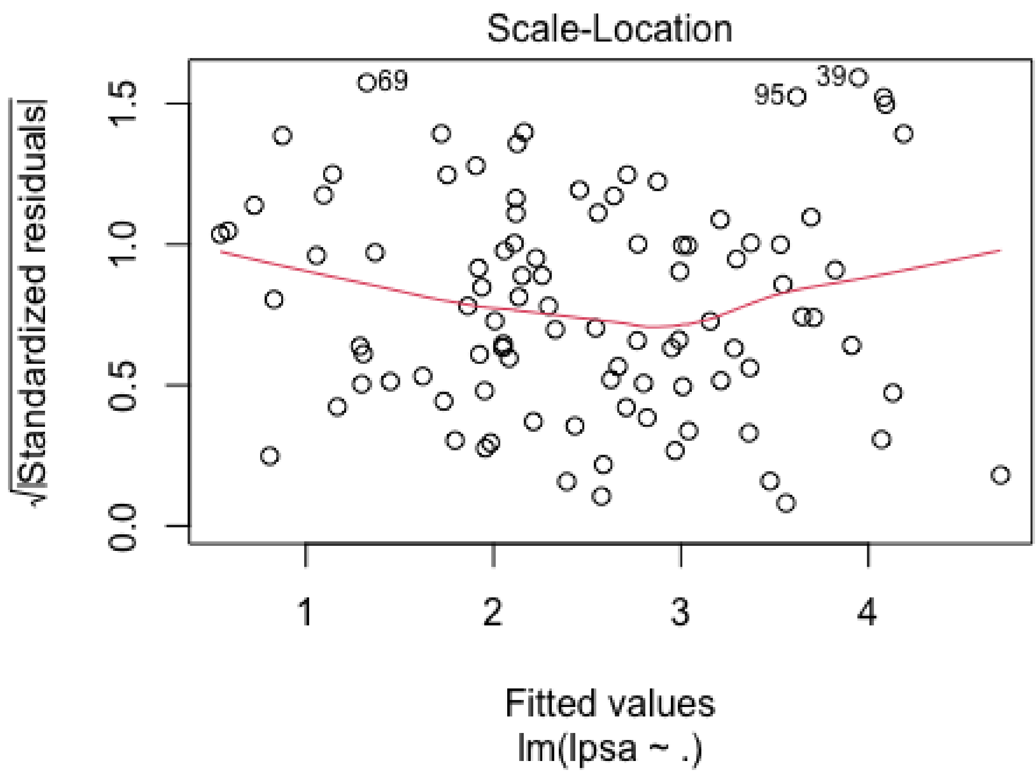
3A

```
plot(lmod1)
```

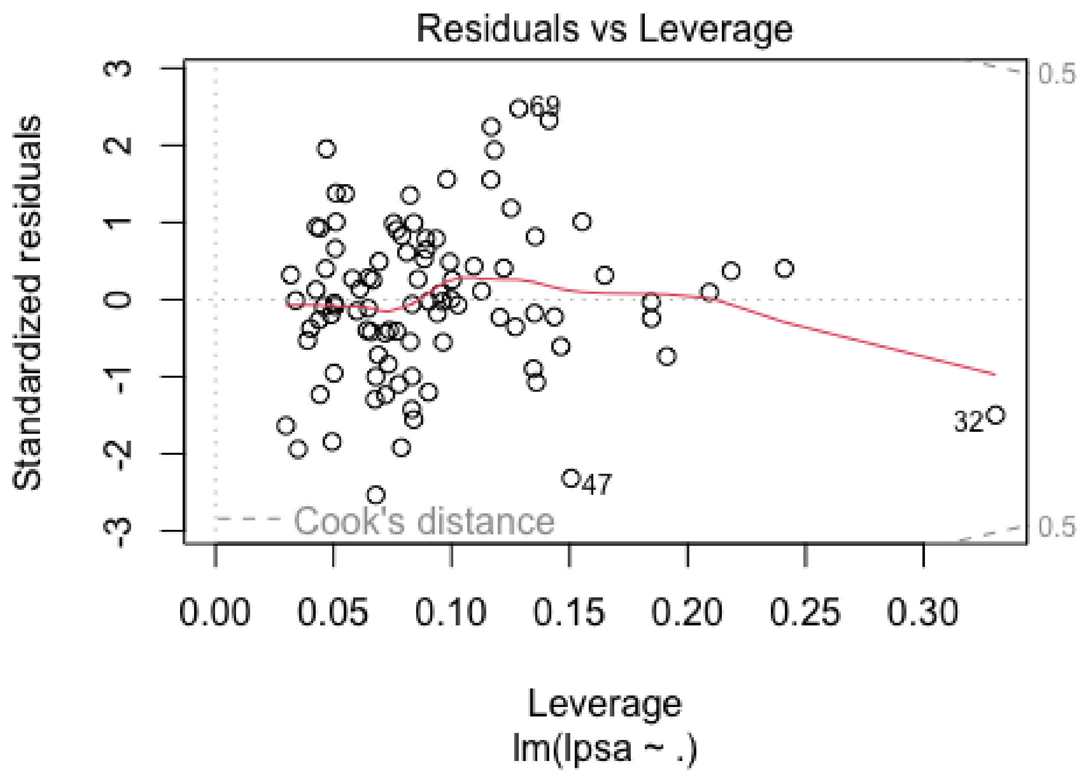


Residuals vs Fitted
Residuals
Fitted values
lm(lpsa ~ .)
69
95
39
2
1
0
-1
-2
1
2
3
4


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(lpsa ~ .)
95
69
39
3
2
1
0
-1
-2
-2
-1
0
1
2



Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
1
2
3
4
69
95
39
Fitted values
lm(lpsa ~ .)



The variance appears to beconstant - we can see this in the arc distribution of the residuals vs. fitted plot.

3B

Based on the qq plot above, some of the distribution appears to be normal, but the leverage points (-3 < x < -1) AND (1<x<3) appear to be non-normal and not follow a normal distribution. We can get a more sure answer to this with the Shapiro-Wilkes test:

```
shapiro.test(residuals(lmod1))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(lmod1)
## W = 0.99113, p-value = 0.7721
```

The P-value returned is greater than 0.5, so we fail to reject the null hypothesis: that our data was sampled from a normal distribution.

3C

```
hatvalues(lmod1)

##          1          2          3          4          5          6 
7 
## 0.07873101 0.06758053 0.13596177 0.07766218 0.03499946 0.08331908 
0.02989838 
##          8          9         10         11         12         13 
14 
## 0.04944610 0.09401490 0.04023404 0.04386826 0.08925939 0.04428928 
0.07318519 
##         15         16         17         18         19         20 
21 
## 0.05020755 0.06897432 0.06664413 0.08320122 0.12212111 0.04895576 
0.03901634 
##         22         23         24         25         26         27 
28 
## 0.08400872 0.04434074 0.07206303 0.04582684 0.06594655 0.12048487 
0.06479337 
##         29         30         31         32         33         34 
35 
## 0.12707056 0.14633177 0.05065029 0.33047574 0.09515819 0.04280678 
0.05106283 
##         36         37         38         39         40         41 
42 
## 0.06791041 0.21843920 0.09801067 0.06794996 0.08106758 0.24100789 
0.06115256 
##         43         44         45         46         47         48 
49 
## 0.04674467 0.09036588 0.04262527 0.05037151 0.15065950 0.03401242 
0.13512286 
##         50         51         52         53         54         55 
56 
## 0.05080725 0.09924342 0.06415518 0.09348895 0.07187492 0.13470108 
0.05990394 
##         57         58         59         60         61         62 
63 
## 0.11665631 0.08910835 0.05105674 0.05799578 0.07677022 0.08328592 
0.18468066 
##         64         65         66         67         68         69 
70 
## 0.09024807 0.06930978 0.03186343 0.10275238 0.06477415 0.12851989 
0.10032173 
##         71         72         73         74         75         76 
77 
## 0.07369386 0.08242713 0.10951482 0.19121086 0.09640539 0.08250756 
0.08575379 
##         78         79         80         81         82         83 
84 
## 0.11272985 0.09614805 0.08839341 0.04703294 0.13546482 0.09985996
```

```
0.16486479
##         85         86         87         88         89         90
91
## 0.05500489 0.07678173 0.08402812 0.07548214 0.14356635 0.12517373
0.15531867
##         92         93         94         95         96         97
## 0.20924207 0.07897648 0.18454695 0.14129097 0.11814056 0.11689127
```

```
hatvalues(lmod1) > 2 * mean(hatvalues(lmod1))
```

```
##     1     2     3     4     5     6     7     8     9    10    11    12
13
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    14    15    16    17    18    19    20    21    22    23    24    25
26
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    27    28    29    30    31    32    33    34    35    36    37    38
39
## FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE
FALSE
##    40    41    42    43    44    45    46    47    48    49    50    51
52
## FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    53    54    55    56    57    58    59    60    61    62    63    64
65
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    66    67    68    69    70    71    72    73    74    75    76    77
78
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
FALSE
##    79    80    81    82    83    84    85    86    87    88    89    90
91
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    92    93    94    95    96    97
##  TRUE FALSE FALSE FALSE FALSE FALSE
```

Based on the hat values above, only case 32, 37, 41, 74, and 92 are considered large leverage points.

3D

```
rstandard(lmod1)[abs(rstandard(lmod1)) > 2]
```

```
##        39        47        69        95        97
## -2.534124 -2.316280  2.477016  2.323964  2.239719
```

As we can see above, instance 39, 47, 69, 95, and 97 can be considered outliers.

3E

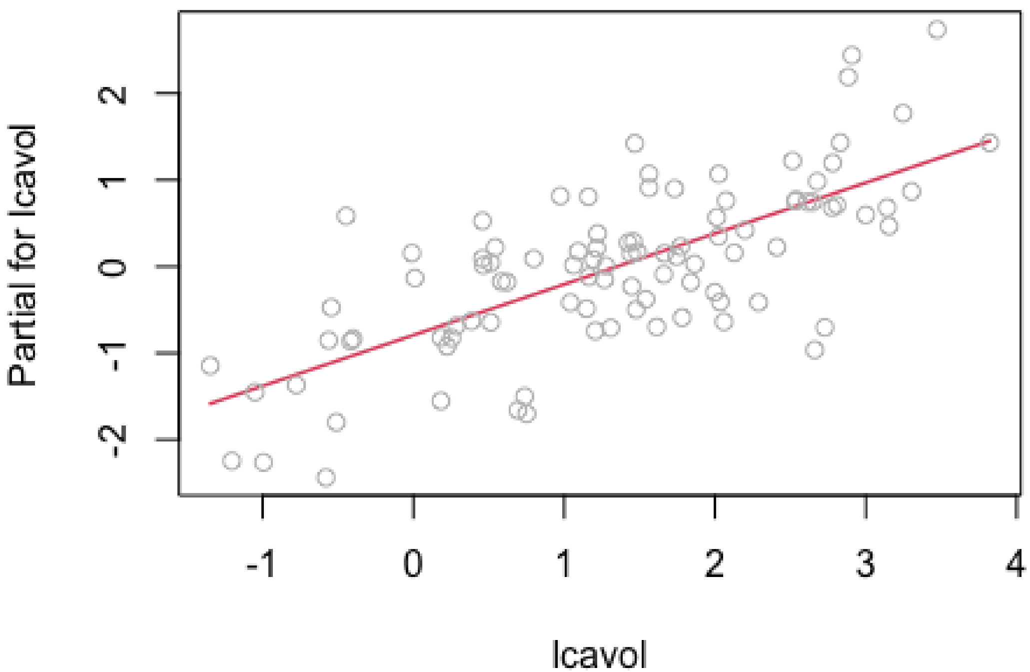
```
cooks.distance(lmod1)[39] > 4 / length(cooks.distance(lmod1))

##   39
## TRUE

cooks.distance(lmod1)[47] > 4 / length(cooks.distance(lmod1))

##   47
## TRUE

cooks.distance(lmod1)[69] > 4 / length(cooks.distance(lmod1))

##   69
## TRUE

cooks.distance(lmod1)[95] > 4 / length(cooks.distance(lmod1))

##   95
## TRUE

cooks.distance(lmod1)[97] > 4 / length(cooks.distance(lmod1))

##   97
## TRUE

cooks.distance(lmod1) > 4 / length(cooks.distance(lmod1))

##     1     2     3     4     5     6     7     8     9    10    11    12
13
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    14    15    16    17    18    19    20    21    22    23    24    25
26
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    27    28    29    30    31    32    33    34    35    36    37    38
39
## FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
TRUE
##    40    41    42    43    44    45    46    47    48    49    50    51
52
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
FALSE
##    53    54    55    56    57    58    59    60    61    62    63    64
65
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    66    67    68    69    70    71    72    73    74    75    76    77
```

```
78
## FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    79    80    81    82    83    84    85    86    87    88    89    90
91
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    92    93    94    95    96    97
## FALSE FALSE FALSE  TRUE  TRUE  TRUE
```

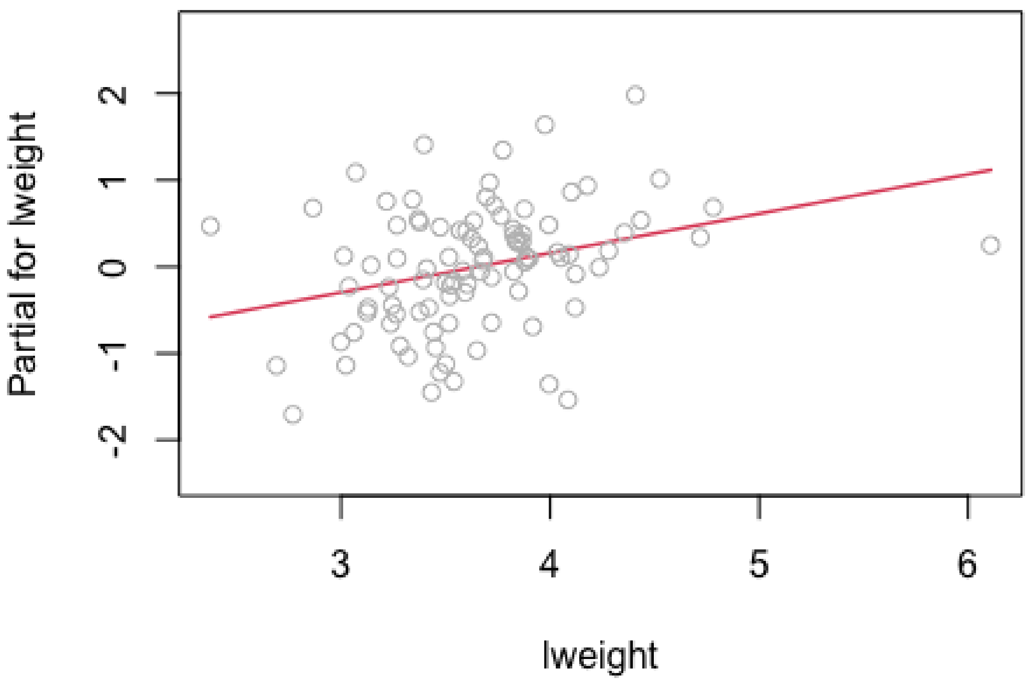
Checking the cooks distance of instance 32, 39, 47, 69, 95, 96, 97 this returned value confirms that all these instances is an influential points.

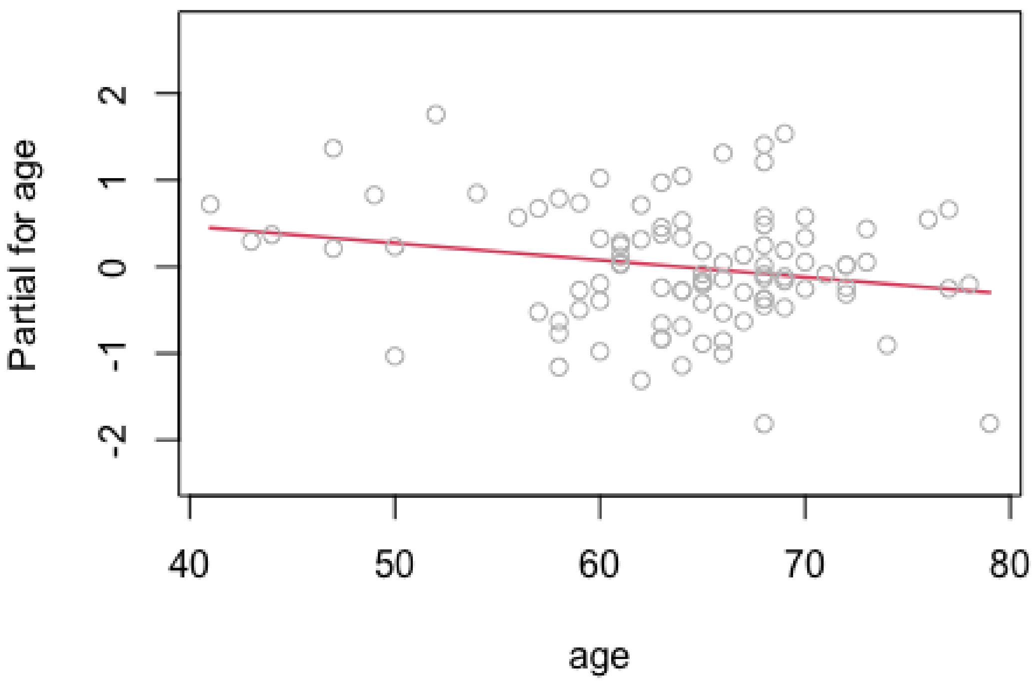
3F

```
termplot(lmod1, partial.resid = T, terms = 1:8)
```

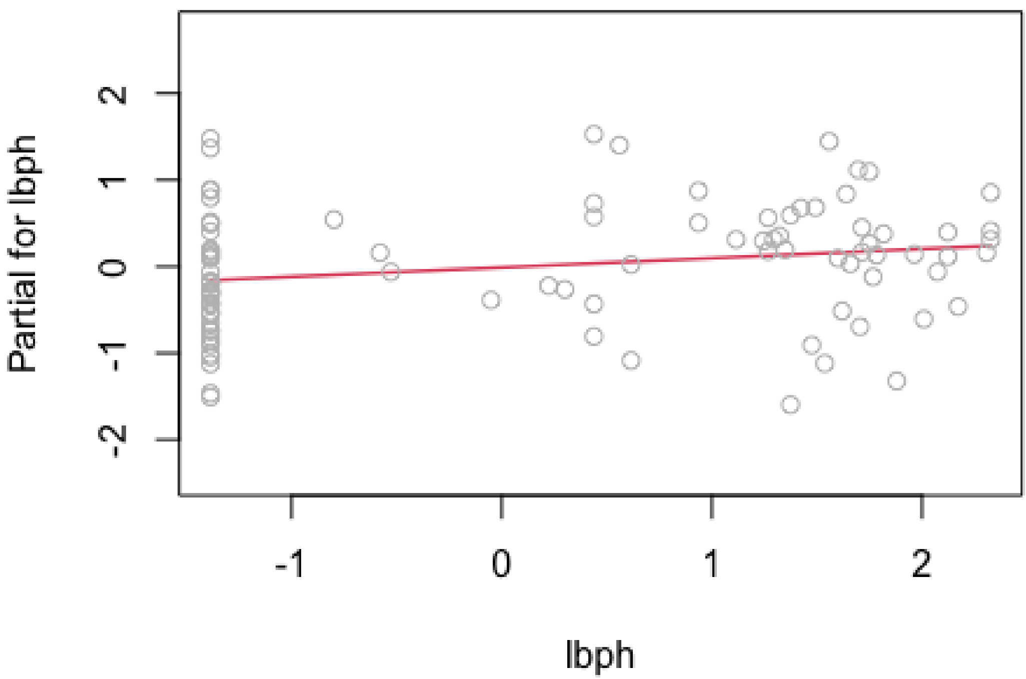


Partial for lcavol
lcavol
-2
-1
0
1
2
-1
0
1
2
3
4

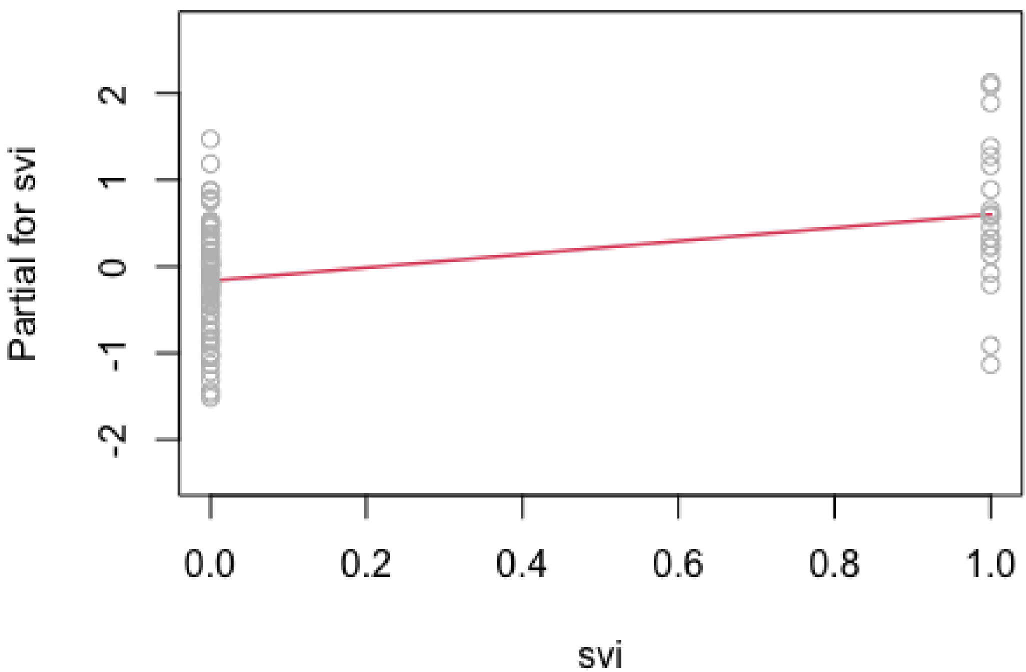

Partial for lweight
2
1
0
-1
-2
3
4
5
6
lweight

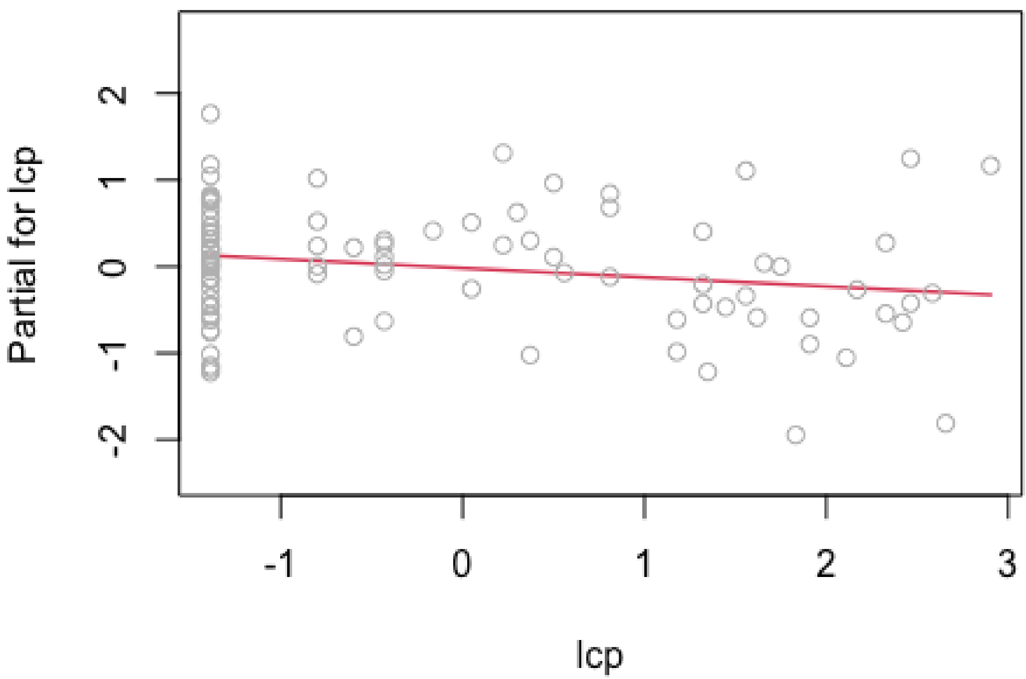


Partial for age
2
1
0
-1
-2
40
50
60
70
80
age

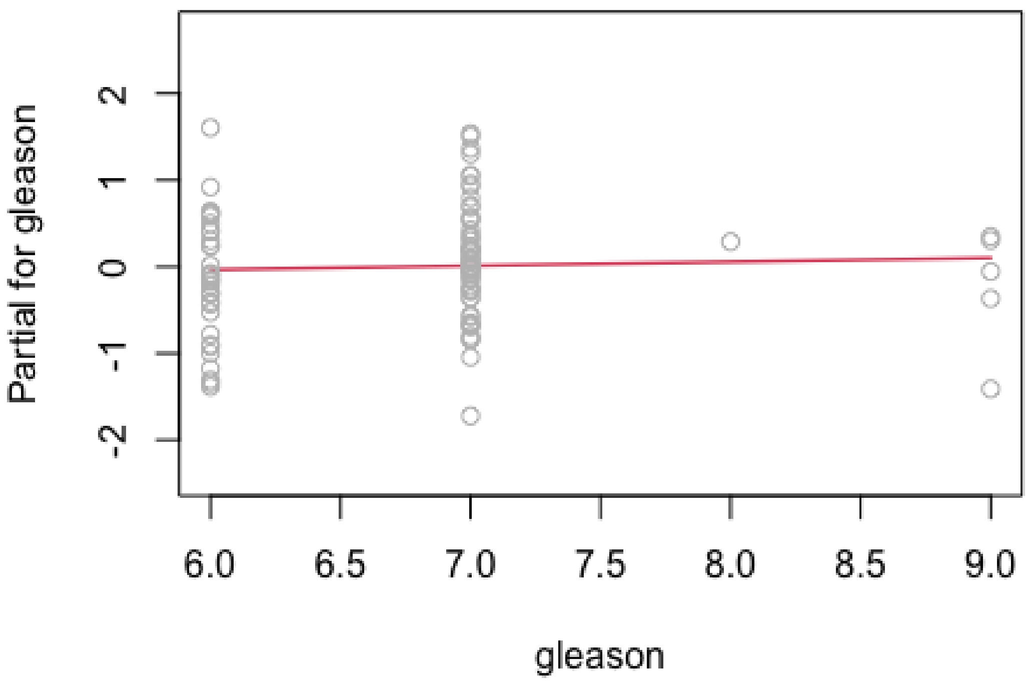


Partial for lbph
lbph
2
1
0
-1
-2
-1
0
1
2

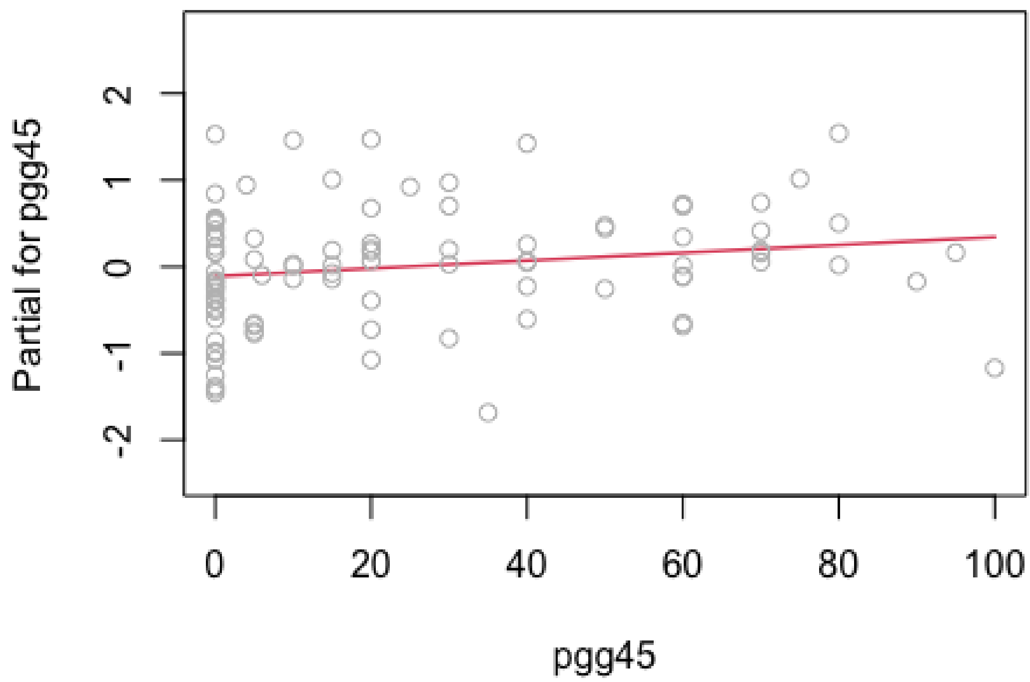

Partial for svi
-2
-1
0
1
2
0.0
0.2
0.4
0.6
0.8
1.0
svi



Partial for lcp
2
1
0
-1
-2
-1
0
1
2
3
lcp



Partial for gleason
2
1
0
-1
-2
6.0
6.5
7.0
7.5
8.0
8.5
9.0
gleason



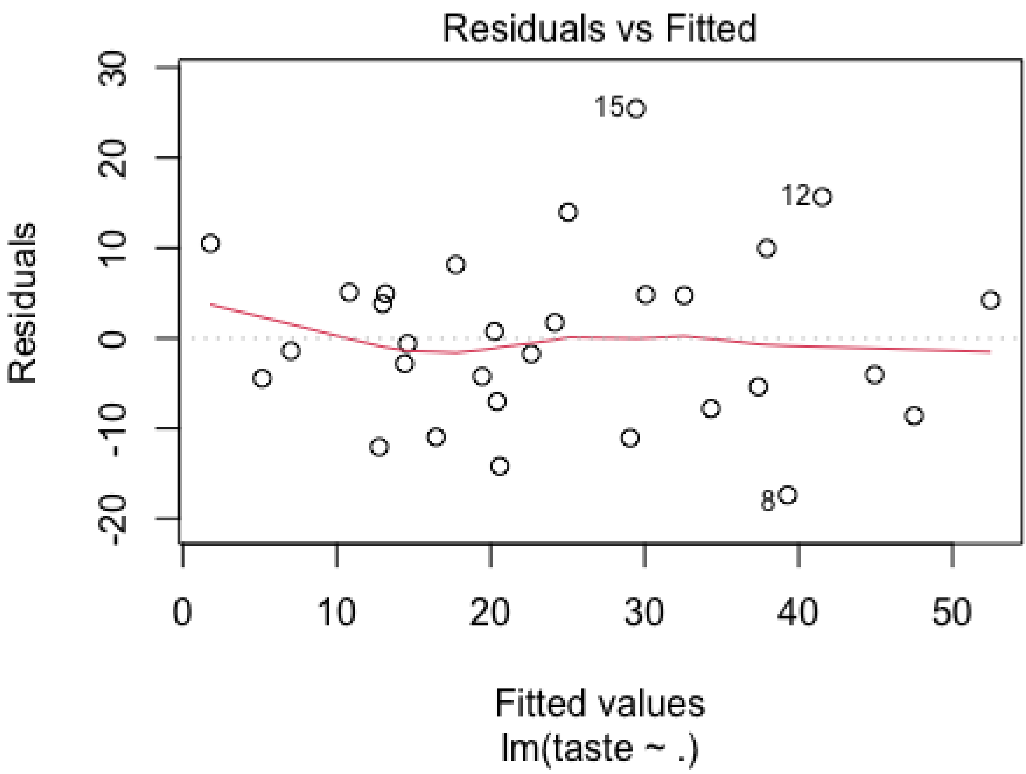
5A

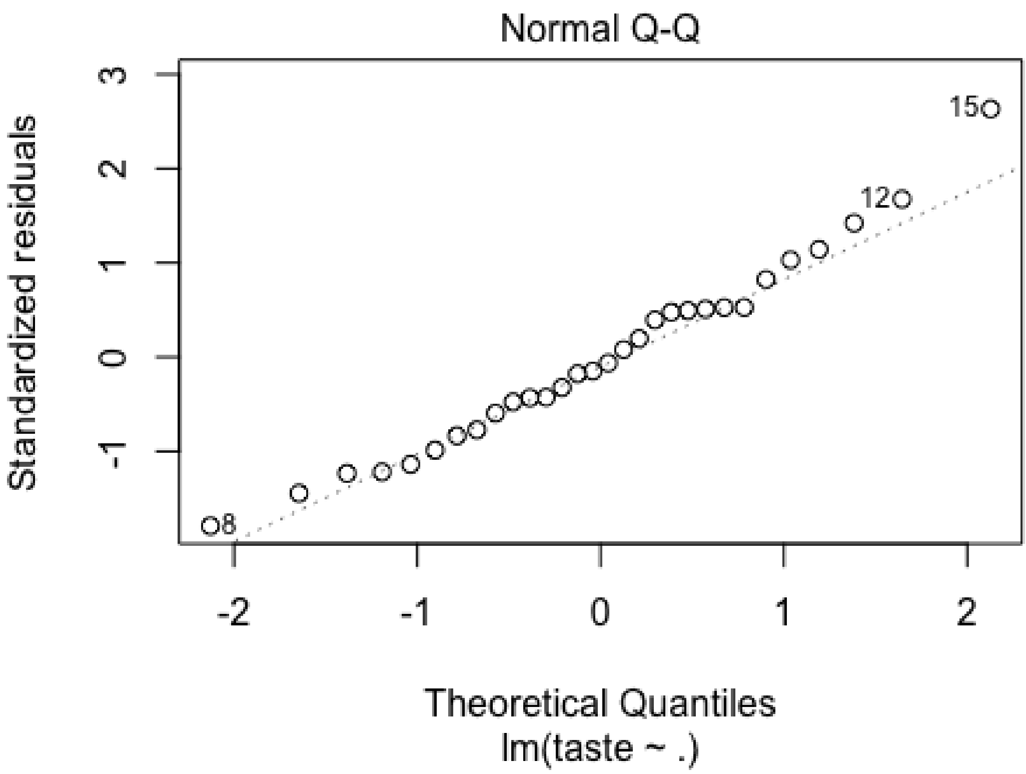
```
library(faraway)
data('cheddar')
head(cheddar)

##   taste Acetic   H2S Lactic
## 1  12.3  4.543 3.135   0.86
## 2  20.9  5.159 5.043   1.53
## 3  39.0  5.366 5.438   1.57
## 4  47.9  5.759 7.496   1.81
## 5   5.6  4.663 3.807   0.99
## 6  25.9  5.697 7.601   1.09

lmod2 <- lm(taste ~ ., cheddar)
summary(lmod2)

##
## Call:
## lm(formula = taste ~ ., data = cheddar)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

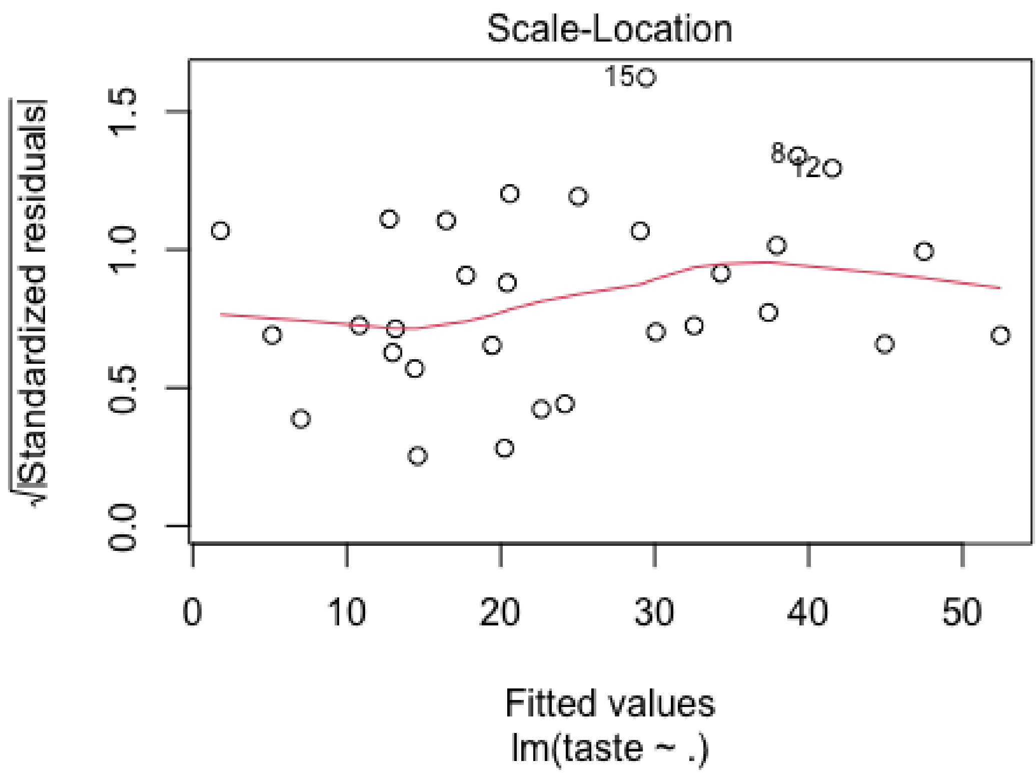
```
## -17.390  -6.612  -1.009   4.908  25.449
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -28.8768    19.7354  -1.463  0.15540
## Acetic        0.3277     4.4598   0.073  0.94198
## H2S           3.9118     1.2484   3.133  0.00425 **
## Lactic       19.6705     8.6291   2.280  0.03108 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.13 on 26 degrees of freedom
## Multiple R-squared:  0.6518, Adjusted R-squared:  0.6116
## F-statistic: 16.22 on 3 and 26 DF,  p-value: 3.81e-06
```

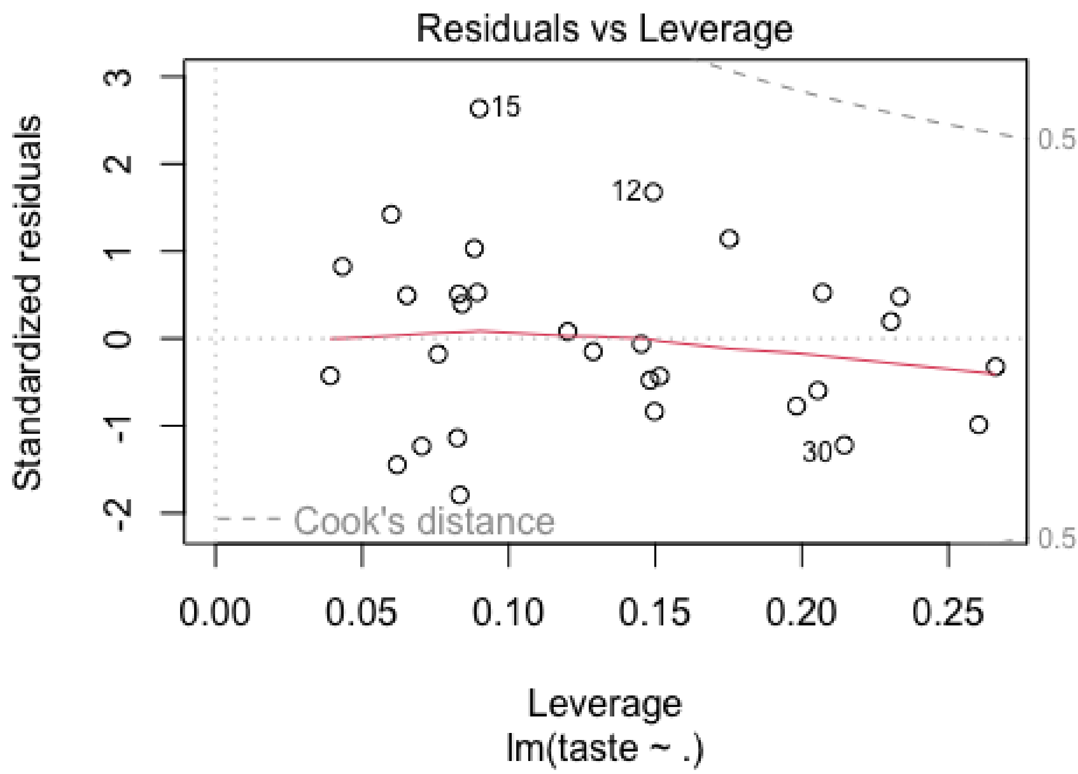
Based on the above regression summary, the Acetic prediors offers almost no predictive value and could be removed to improve the model.

```
plot(lmod2)
```


Residuals vs Fitted
Residuals
30
20
10
0
-10
-20
15
12
8
0
10
20
30
40
50
Fitted values
lm(taste ~ .)


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(taste ~ .)
15
12
8
3
2
1
0
-1
-2
-1
0
1
2


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
0
10
20
30
40
50
15
8
12
Fitted values
lm(taste ~ .)



Upon initial observation of the Residuals vs. Fitted, the residual appear to be centered around zero and the variance appears to be about constant, but there is slightly higher variance as taste increases.

5B

Based on the qq plot above, most of the distribution appears to be centered; in short, the sample appears to fit a normal distribution. Let's verify with the shapiro wilkes test:

```
shapiro.test(residuals(lmod2))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(lmod2)
## W = 0.98021, p-value = 0.8312
```

This p-value is much larger than 0.05 indicating that our value data comes from a normal distribution; we fail to reject the null hypothesis in this case.

5C

```
hatvalues(lmod2)

##          1          2          3          4          5          6
7
## 0.17525784 0.07593130 0.05994339 0.08829409 0.12879533 0.23036705
0.20709897
##          8          9         10         11         12         13
14
## 0.08333780 0.08291114 0.12013909 0.06531941 0.14929496 0.14821335
0.04332811
##         15         16         17         18         19         20
21
## 0.09000337 0.15153827 0.08934443 0.06198950 0.08249992 0.26029095
0.14521419
##         22         23         24         25         26         27
28
## 0.03912430 0.20545696 0.23343680 0.08406925 0.26606306 0.14973461
0.07036401
##         29         30
## 0.19818511 0.21445340

hatvalues(lmod2) > 2 * mean(hatvalues(lmod2))

##     1     2     3     4     5     6     7     8     9    10    11    12
13
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    14    15    16    17    18    19    20    21    22    23    24    25
26
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    27    28    29    30
## FALSE FALSE FALSE FALSE

hatvalues(lmod2) > 1.9 * mean(hatvalues(lmod2))

##     1     2     3     4     5     6     7     8     9    10    11    12
13
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
##    14    15    16    17    18    19    20    21    22    23    24    25
26
## FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
TRUE
##    27    28    29    30
## FALSE FALSE FALSE FALSE
```

Using the 2 times the mean of the hat values, we have no large leverage points. However, when looking at 1.9 times the hat values, we see that case 20 and 26 could be considered a large leverage point.

5D

```
rstandard(lmod2)[abs(rstandard(lmod2)) > 2]

##       15
## 2.633351

rstandard(lmod2)[abs(rstandard(lmod2)) > 1.5]

##         8        12        15
## -1.792952  1.675450  2.633351
```

Based on the above values, case 15 could be considered an outlier. When looking if the rstandard value is greater than 1.5 rather than 2, we see that points 8 and 12 could also be considered outliers.

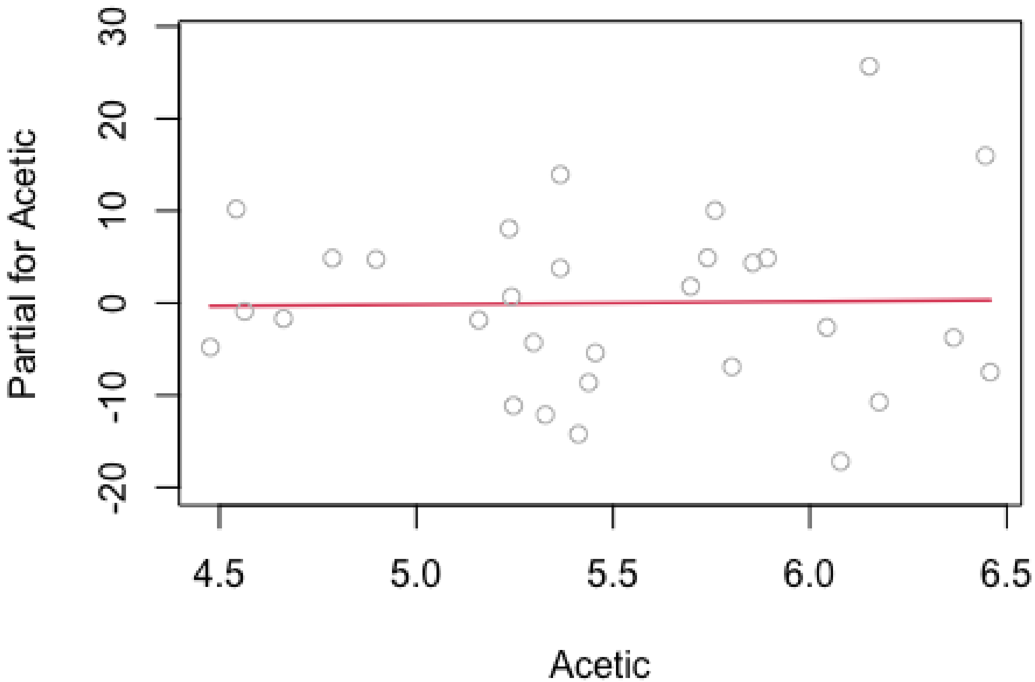
5E

```
cooks.distance(lmod2)[15] > 4 / length(cooks.distance(lmod2))

##   15
## TRUE

cooks.distance(lmod2)[8] > 4 / length(cooks.distance(lmod2))

##     8
## FALSE

cooks.distance(lmod2)[12] > 4 / length(cooks.distance(lmod2))

##    12
## FALSE

cooks.distance(lmod2)[20] > 4 / length(cooks.distance(lmod2))

##    20
## FALSE

cooks.distance(lmod2)[26] > 4 / length(cooks.distance(lmod2))

##    26
## FALSE
```

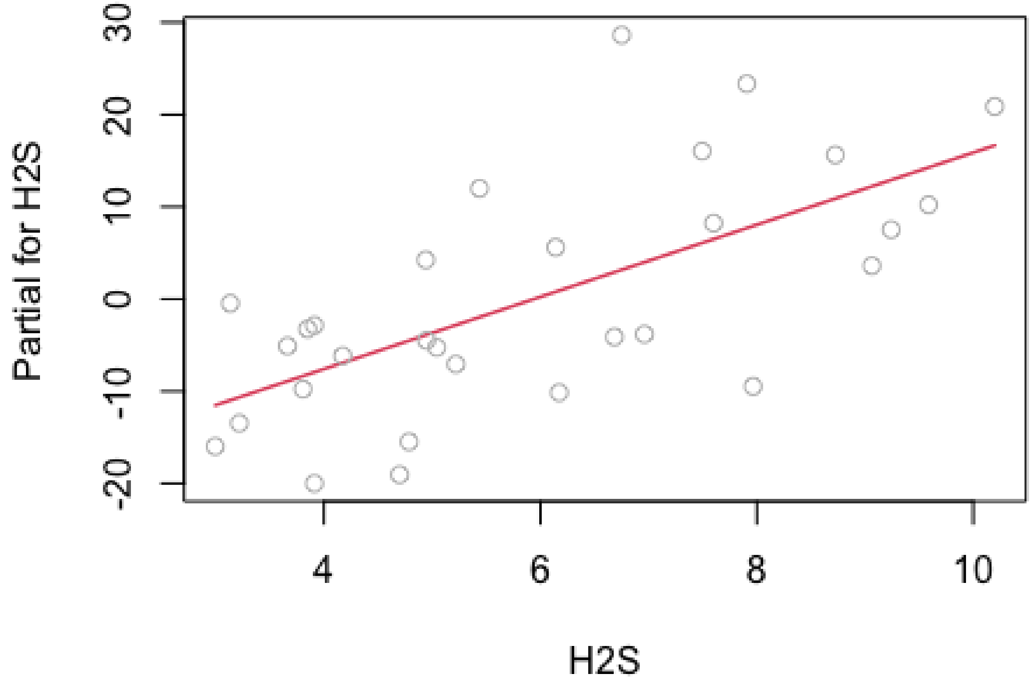
Based on the Cook’s distance for case 15, this appears the be an influential point.

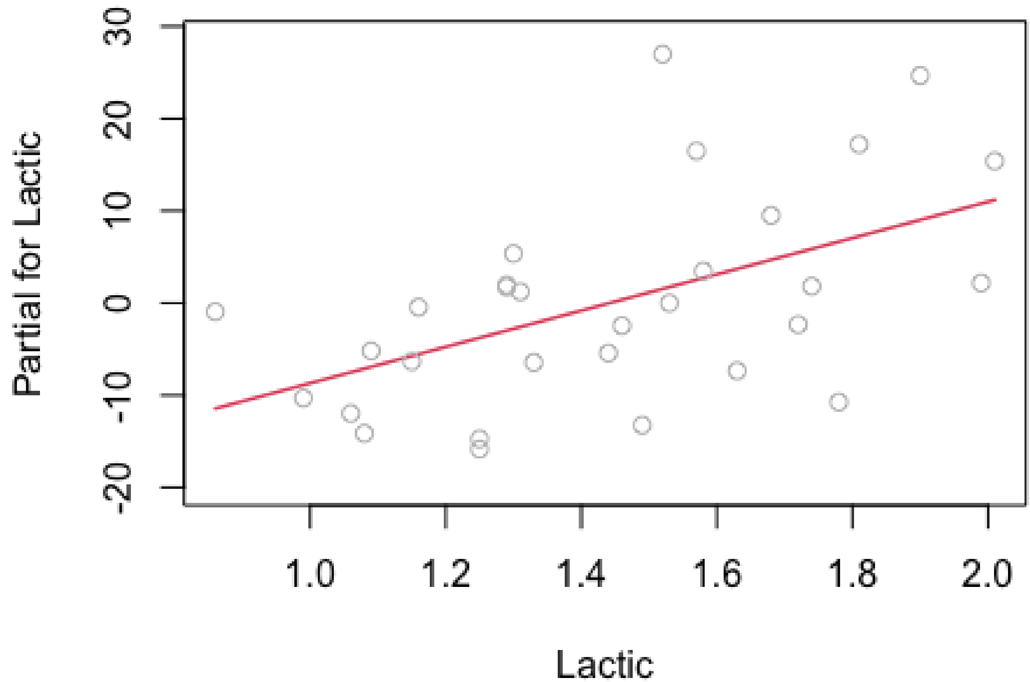
5F

```
termplot(lmod2, partial.resid = T, terms = 1:3)
```



Partial for Acetic
30
20
10
0
-10
-20
4.5
5.0
5.5
6.0
6.5
Acetic


Partial for H2S
30
20
10
0
-10
-20
4
6
8
10
H2S



Based on the above termplots, the acetic response appears to have less predictive power than the other prediors in the regression.

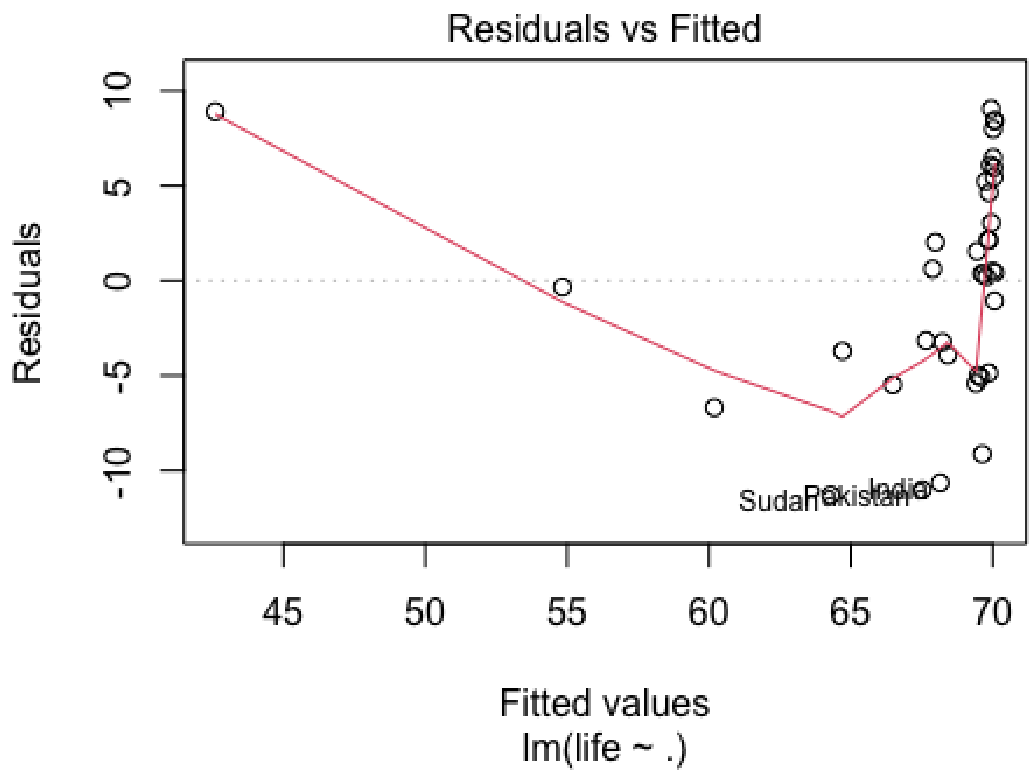
7A

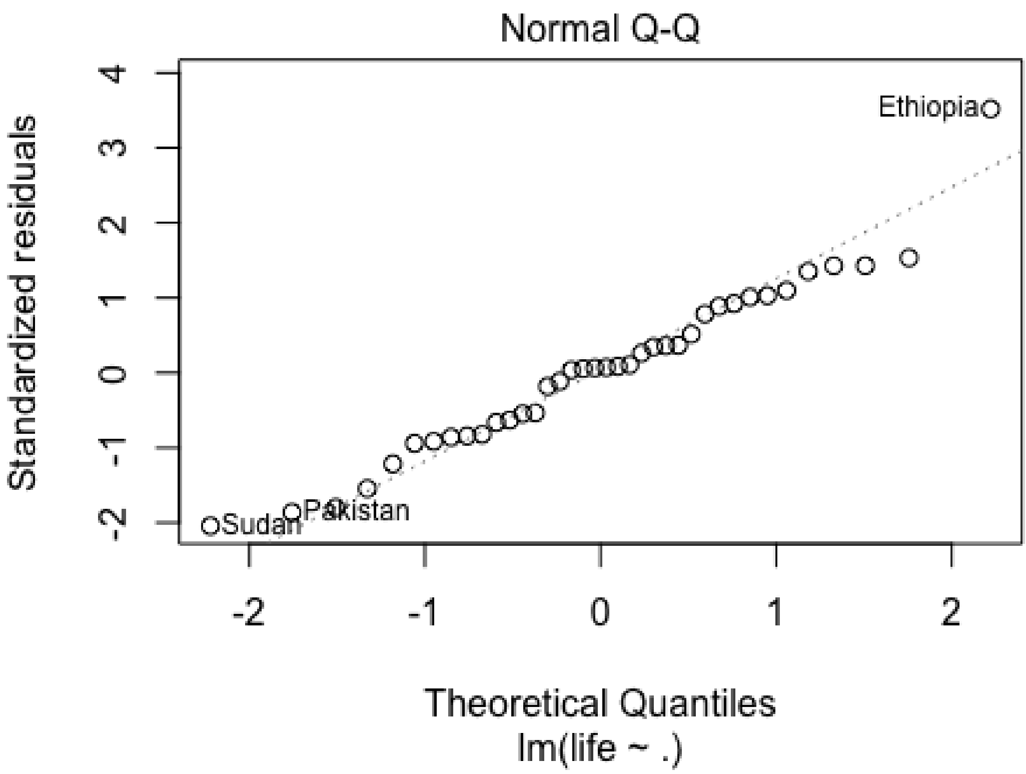
```
library(faraway)
data('tvdoctor')
head(tvdoctor)

##             life    tv doctor
## Argentina   70.5   4.0    370
## Bangladesh  53.5 315.0   6166
## Brazil      65.0   4.0    684
## Canada      76.5   1.7    449
## China       70.0   8.0    643
## Colombia    71.0   5.6   1551

lmod3 <- lm(life ~ ., tvdoctor)
summary(lmod3)

##
## Call:
## lm(formula = life ~ ., data = tvdoctor)
##
```

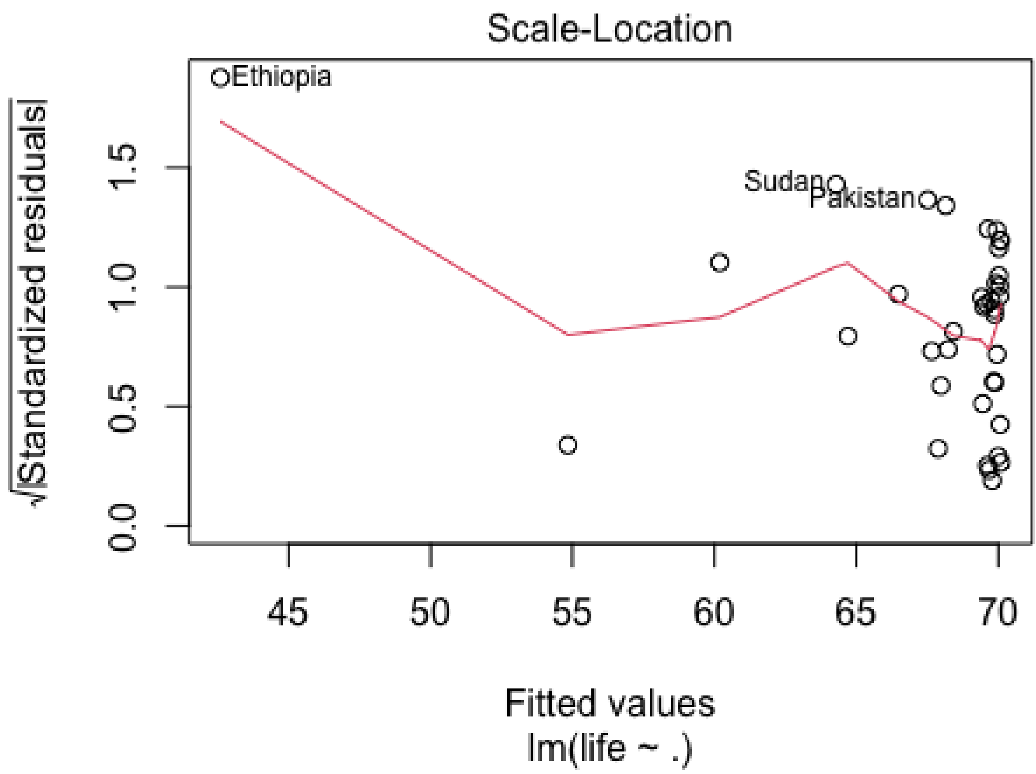
```
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -11.2894  -4.6266   0.3977   5.0872   9.0535 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 70.2519573  1.0877047  64.587   <2e-16 ***
## tv          -0.0234954  0.0096469  -2.436   0.0201 *  
## doctor      -0.0004320  0.0002023  -2.136   0.0398 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 6.003 on 35 degrees of freedom
## Multiple R-squared:   0.44,  Adjusted R-squared:  0.408 
## F-statistic: 13.75 on 2 and 35 DF,  p-value: 3.916e-05
```

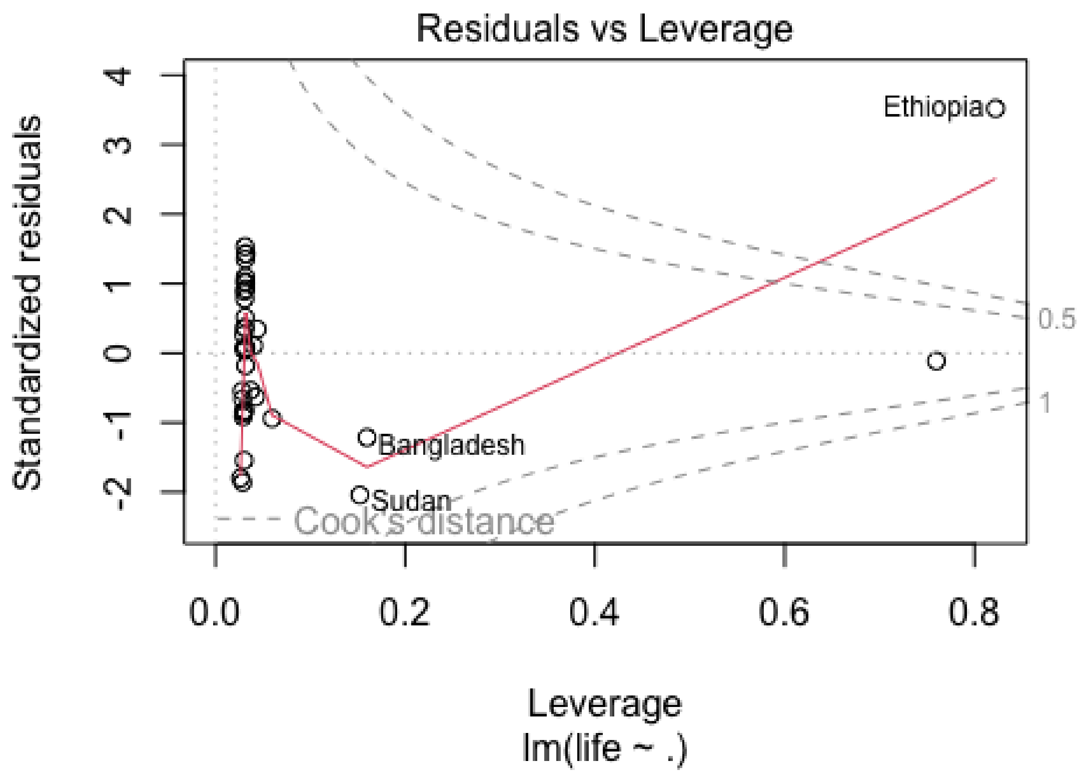
Based on the above data, the intercept has the most significance. This indicates that there might be other predictors that are left out of the regression that could offer explanations into the data.

```
plot(lmod3)
```


Residuals vs Fitted
Residuals
10
5
0
-5
-10
Sudan
Pakistan
India
45
50
55
60
65
70
Fitted values
lm(life ~ .)


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(life ~ .)
Ethiopia
Sudan
Pakistan
4
3
2
1
0
-1
-2
-2
-1
0
1
2


Scale-Location
Ethiopia
Sudan
Pakistan
√|Standardized residuals|
0.0
0.5
1.0
1.5
45
50
55
60
65
70
Fitted values
lm(life ~ .)



Looking at the Residuals vs. Fitted plot, we can see that this data is clearly not of constant variance and fails our constant variance assumption.

7B

The above qq plot shows that the data appears to be pulled from a normal distribution. We can use the shapiro-wilkes test to confirm whether this is true:

```
shapiro.test(residuals(lmod3))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(lmod3)
## W = 0.95872, p-value = 0.1725
```

The above shapiro-wilkes test confirms our assertion confirms that we fail to reject the null hypothesis - meaning that our sample data appears to have been pulled from a normally distributed dataset (because our p-value is greater than 0.05).

7C

```r
#hatvalues(lmod3)
hatvalues(lmod3) > 2 * mean(hatvalues(lmod3))
```

```
##      Argentina     Bangladesh         Brazil         Canada          China 
##          FALSE           TRUE          FALSE          FALSE          FALSE 
##       Colombia          Egypt       Ethiopia         France        Germany 
##          FALSE          FALSE           TRUE          FALSE          FALSE 
##          India      Indonesia           Iran          Italy          Japan 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##          Kenya     KoreaNorth     KoreaSouth         Mexico        Morocco 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##        Myanmar       Pakistan           Peru    Philippines         Poland 
##           TRUE          FALSE          FALSE          FALSE          FALSE 
##        Romania         Russia    SouthAfrica          Spain          Sudan 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##         Taiwan       Thailand         Turkey        Ukraine UnitedKingdom 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##   UnitedStates      Venezuela        Vietnam 
##          FALSE          FALSE          FALSE
```

```r
#hatvalues(lmod3) > 1.9 * mean(hatvalues(lmod3))
```

Based on the above hat values, Bangladesh, Ethiopia, and Myanmar appear to be large leverage points.

7D

```r
rstandard(lmod3)[abs(rstandard(lmod3)) > 2]
```

```
##  Ethiopia     Sudan 
##  3.518939 -2.042465
```

Based on our model, only Ethiopia and Sudan are outliers.

7E

```r
cooks.distance(lmod3) > 4 / length(cooks.distance(lmod3))
```

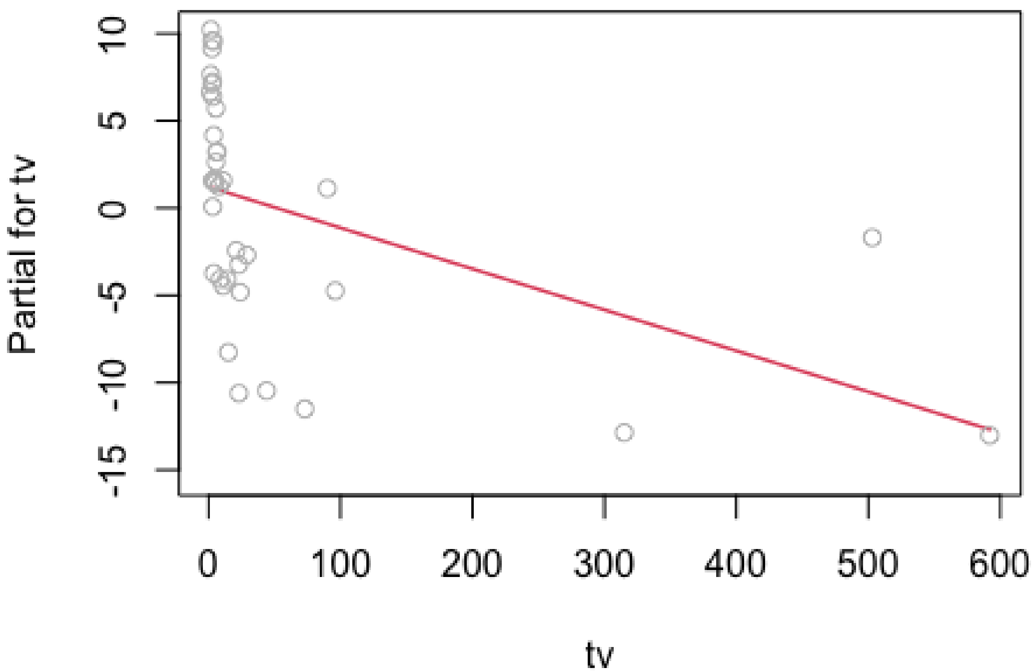
```
##      Argentina     Bangladesh         Brazil         Canada          China 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##       Colombia          Egypt       Ethiopia         France        Germany 
##          FALSE          FALSE           TRUE          FALSE          FALSE 
##          India      Indonesia           Iran          Italy          Japan 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##          Kenya     KoreaNorth     KoreaSouth         Mexico        Morocco 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##        Myanmar       Pakistan           Peru    Philippines         Poland 
##          FALSE          FALSE          FALSE          FALSE          FALSE 
##        Romania         Russia    SouthAfrica          Spain          Sudan
```

```
##          FALSE          FALSE          FALSE          FALSE           TRUE
##         Taiwan       Thailand         Turkey        Ukraine UnitedKingdom
##          FALSE          FALSE          FALSE          FALSE          FALSE
##   UnitedStates      Venezuela        Vietnam
##          FALSE          FALSE          FALSE
```

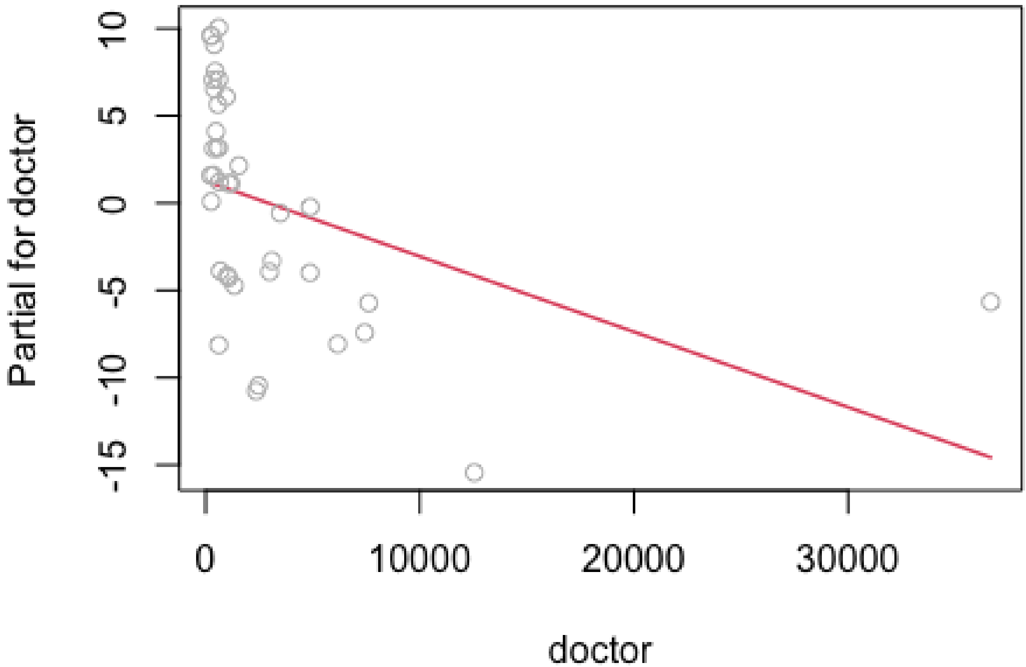
Based on the cooks distance model, only Ethiopia and Sudan qualify as influential points in our lmod3 model.

7F

```
termplot(lmod3, partial.resid = T, terms = 1:2)
```



Partial for tv
tv
-15
-10
-5
0
5
10
0
100
200
300
400
500
600



Based on the term plot, neither have much predictive power. This could indicate that we are missing predictors from the dataset that could offer explanatory power for our response values.