MATH 588

HW4

Md Ismail Hossain

2/23/2022

Question 1

a

```
library(Sleuth3)
bb = case1102
sapply(bb, class) # Simplify future typeing by changing names to lower case:
```

```
##     Brain     Liver      Time Treatment      Days       Sex    Weight      Loss
## "integer" "integer" "numeric"  "factor" "integer"  "factor" "integer" "numeric"
##     Tumor
## "integer"
```

```
names(bb) = casefold(names(bb))
names(bb)
```

```
## [1] "brain"     "liver"     "time"      "treatment" "days"      "sex"
## [7] "weight"    "loss"      "tumor"
```

```
# Make new variables
bb$logBLratio = log(bb$brain/bb$liver)
bb$logTime = log(bb$time)

# a(1)
library(dplyr)
with(bb, table(sex)) %>% prop.table()
```

```
## sex
##    Female      Male
## 0.7647059 0.2352941
```

```
# a(2)
with(bb, table(treatment,days)) %>% prop.table()
```

```
##          days
## treatment          9         10         11
##        BD 0.02941176 0.41176471 0.05882353
##        NS 0.05882353 0.38235294 0.05882353
```

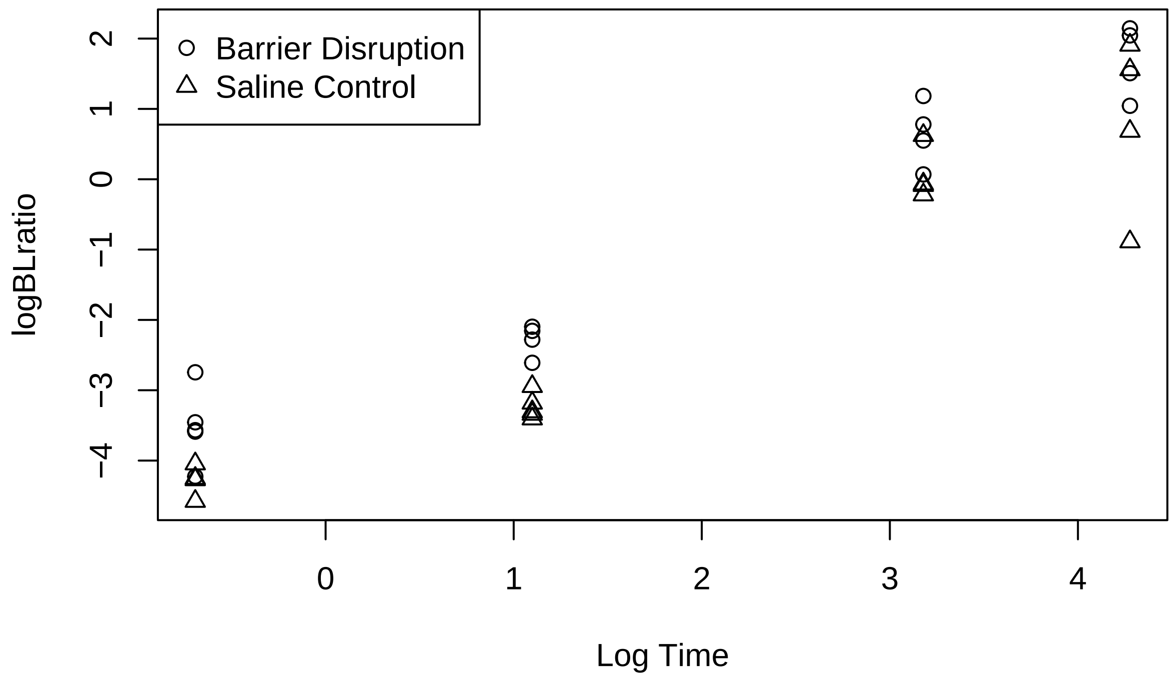
b

```
# b

# Plot key variables
with(bb, plot(logBLratio ~ logTime, pch=as.numeric(bb$treat), xlab="Log Time"))
with(bb, table(bb$treat,as.numeric(bb$treat)))
```

```
##
##       1  2
##   BD 17  0
##   NS  0 17
```

```
legend("topleft", legend=c("Barrier Disruption","Saline Control"), pch=1:2)
```



c

```r
m0 = lm(logBLratio ~ logTime + treatment, bb)
summary(m0)
```

```
##
## Call:
## lm(formula = logBLratio ~ logTime + treatment, data = bb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7280 -0.4453  0.1078  0.3556  1.0673
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.00928    0.18400 -16.355  < 2e-16 ***
## logTime      1.09784    0.05654  19.416  < 2e-16 ***
## treatmentNS -0.84579    0.21640  -3.908 0.000471 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6307 on 31 degrees of freedom
## Multiple R-squared:  0.9261, Adjusted R-squared:  0.9213
## F-statistic: 194.2 on 2 and 31 DF,  p-value: < 2.2e-16
```

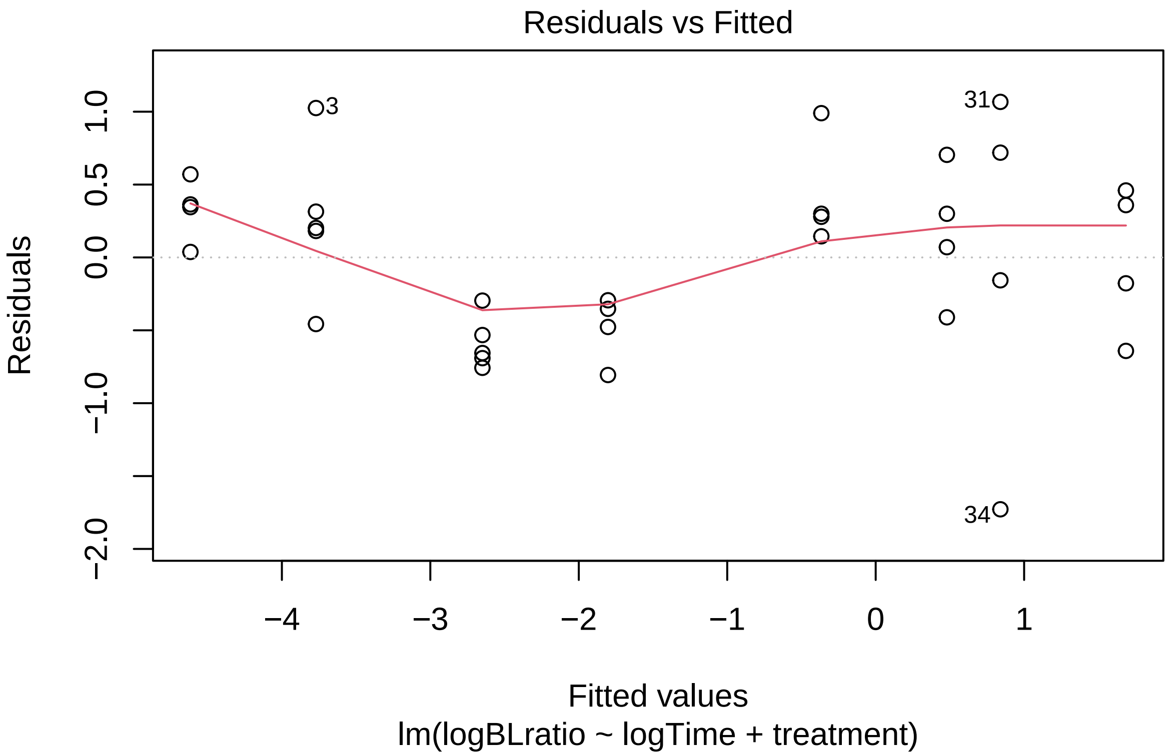
d

```
m1 = lm(logBLratio ~ logTime*treatment, bb)
summary(m1)
```

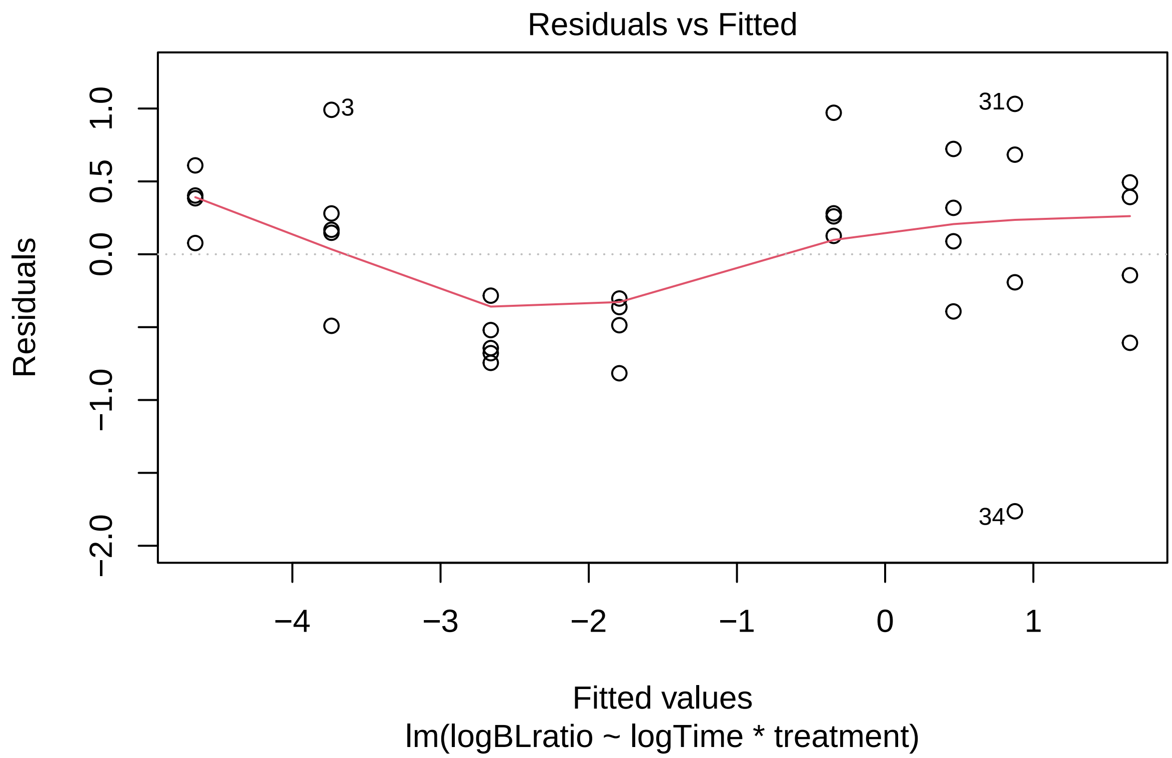
```
##
## Call:
## lm(formula = logBLratio ~ logTime * treatment, data = bb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7635 -0.4631  0.1076  0.3907  1.0318
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.98457    0.21145  -14.11 8.74e-15 ***
## logTime              1.08417    0.07933   13.67 2.02e-14 ***
## treatmentNS         -0.89928    0.30693   -2.93  0.00642 **
## logTime:treatmentNS  0.02870    0.11497    0.25  0.80458
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6404 on 30 degrees of freedom
## Multiple R-squared:  0.9262, Adjusted R-squared:  0.9189
## F-statistic: 125.6 on 3 and 30 DF,  p-value: < 2.2e-16
```

e

```
plot(m0, which=c(1,1))
```




```
plot(m1, which=c(1,1))
```



From this residual vs fitted plot we observed a non linear relationship and observation 34 seems like close look. From this plot we can not conclude that constant variance assumption violated both case. ## f

From the regression summary table presented in part (d), we found that the p-value for the interaction term does not seems statistically significant at 5% level of significance. So, we can say there is not any joint effect of the variable logTime and TreatNS. ## g

```
summary(influence.measures(m1))
```

```
## Potentially influential observations of
##   lm(formula = logBLratio ~ logTime * treatment, data = bb) :
##
##    dfb.1_ dfb.lgTm dfb.trNS dfb.lT:N dffit   cov.r   cook.d hat
## 34  0.00   0.00     0.14    -0.85    -1.49_*  0.33_*  0.40   0.15
```

```
apply(confint(m1),1,diff)
```

```
##         (Intercept)             logTime         treatmentNS logTime:treatmentNS
##           0.8636959           0.3240190           1.2536878           0.4695868
```

```
apply(confint(update(m1,subset=-34)),1,diff)
```

```
##         (Intercept)             logTime         treatmentNS logTime:treatmentNS
##           0.7368614           0.2764365           1.0704263           0.4121385
```

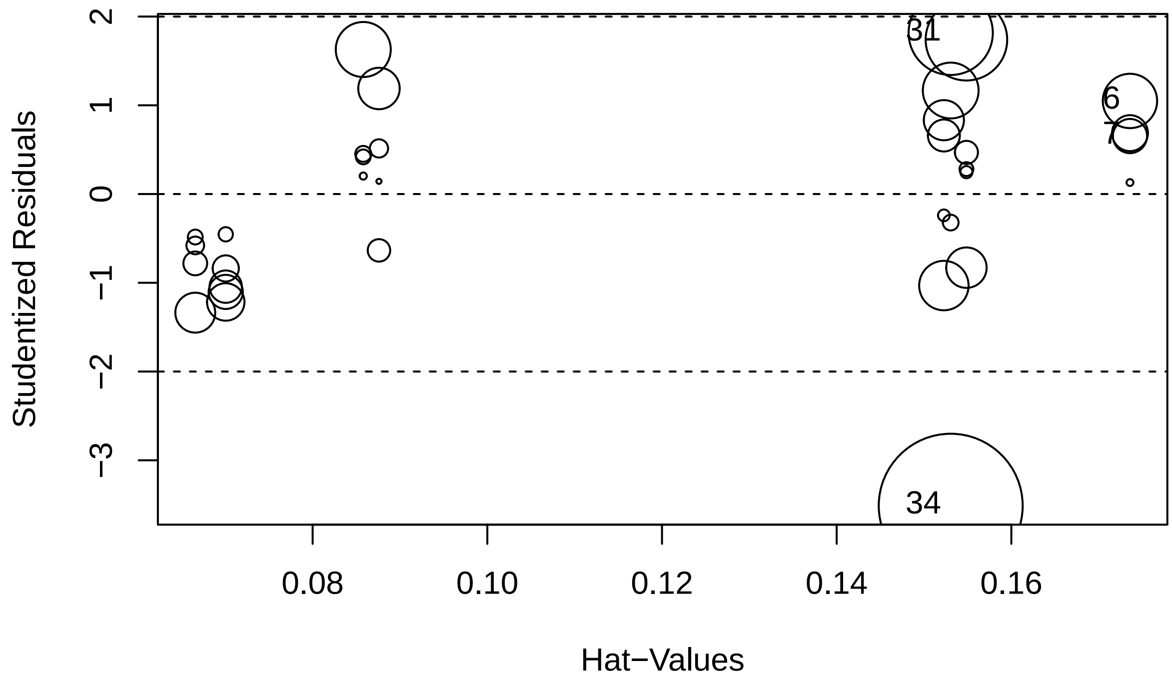
From this summary we found that observation 34 is influential according to DFFITS and cov.r. So, we tried to observed the effect of observation 34 after removing it from the data and displying the confidence interval difference between the models. ## h

```r
m2 = update(m1, subset=rownames(bb)!=34)
summary(m2)
```

```
##
## Call:
## lm(formula = logBLratio ~ logTime * treatment, data = bb, subset = rownames(bb) !=
##     34)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81638 -0.48672  0.05347  0.39283  0.99110
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.98457    0.18014 -16.568 2.52e-16 ***
## logTime              1.08417    0.06758  16.043 5.86e-16 ***
## treatmentNS         -0.93577    0.26169  -3.576  0.00125 **
## logTime:treatmentNS  0.11175    0.10076   1.109  0.27649
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5456 on 29 degrees of freedom
## Multiple R-squared:  0.9482, Adjusted R-squared:  0.9428
## F-statistic: 176.8 on 3 and 29 DF,  p-value: < 2.2e-16
```

We do not observe any drastic change in the summary after removing the 34th observation from the data set. The R squared value little increased and the Standard error of the coefficients become smaller for the adjusted data set. ## i

```r
library(car)
influencePlot(m1)
```




```
##       StudRes       Hat      CookD
## 6   1.049054 0.1735839 0.05759624
## 7   0.653651 0.1735839 0.02287253
## 31  1.816446 0.1530580 0.13845649
## 34 -3.512260 0.1530580 0.40449104
```

From this plot we observed that observation 31st and 34th have significant effect on the estimated coefficient of the model.

j

This would indicate that dropping subject 34 is lowering the estimate of the logTime slope coefficient and this indicate higher effect on estimate.

Question 2

a

```r
bost = read.csv("bost.csv")
head(bost)
```

```
##      rm tax lstat medv
## 1 6.575 296  4.98 24.0
## 2 6.421 242  9.14 21.6
## 3 7.185 242  4.03 34.7
## 4 6.998 222  2.94 33.4
```

```
## 5 7.147 222  5.33 36.2
## 6 6.430 222  5.21 28.7
```

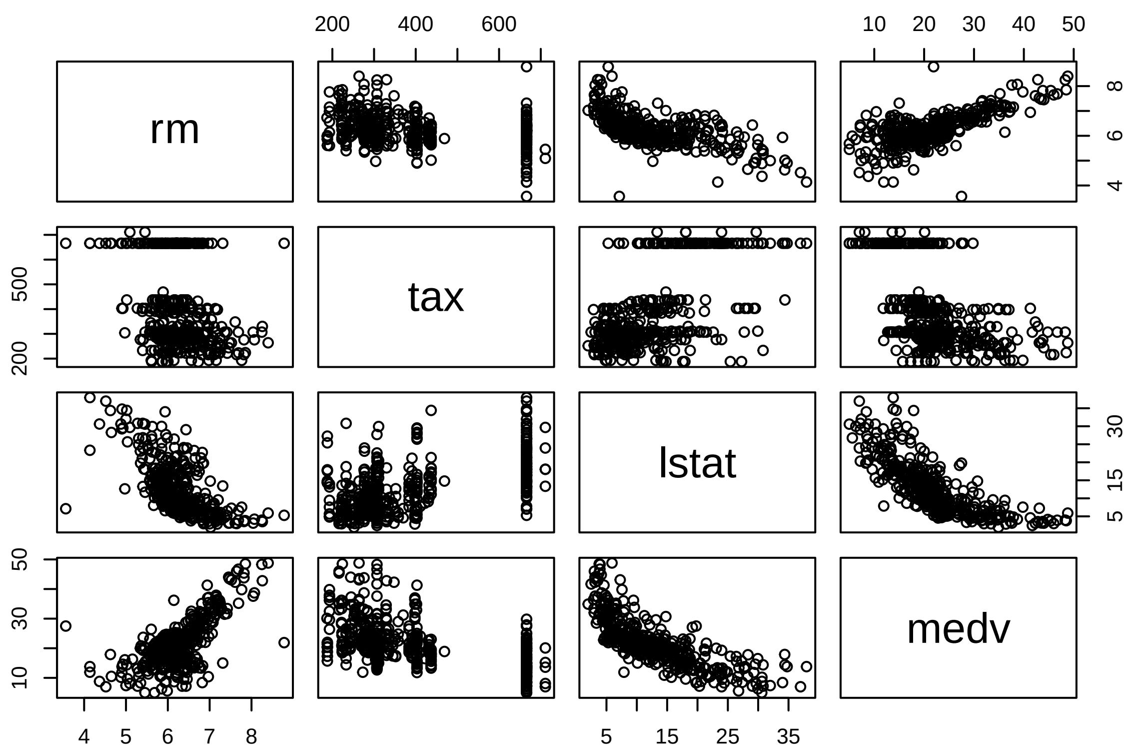
```
sapply(bost,function(x)mean(is.na(x)))
```

```
##        rm       tax     lstat      medv
## 0.1326531 0.0000000 0.1265306 0.0000000
```

We found around 13% missing values for the column “rm” and “lstat”. Other two cloumns don’t have any missing observations.

b

```
pairs(bost)
```



From this correlation plots, we can comment that that room is positively correlated with the variable “medv” and negatively with “lstat”. We can not make a clear statment about the relation between room and tax but looks like a down tren in the observation. Also lstat and medv seems like a nonlinear down trend between them and no clear pattern found between medv and tax.

c

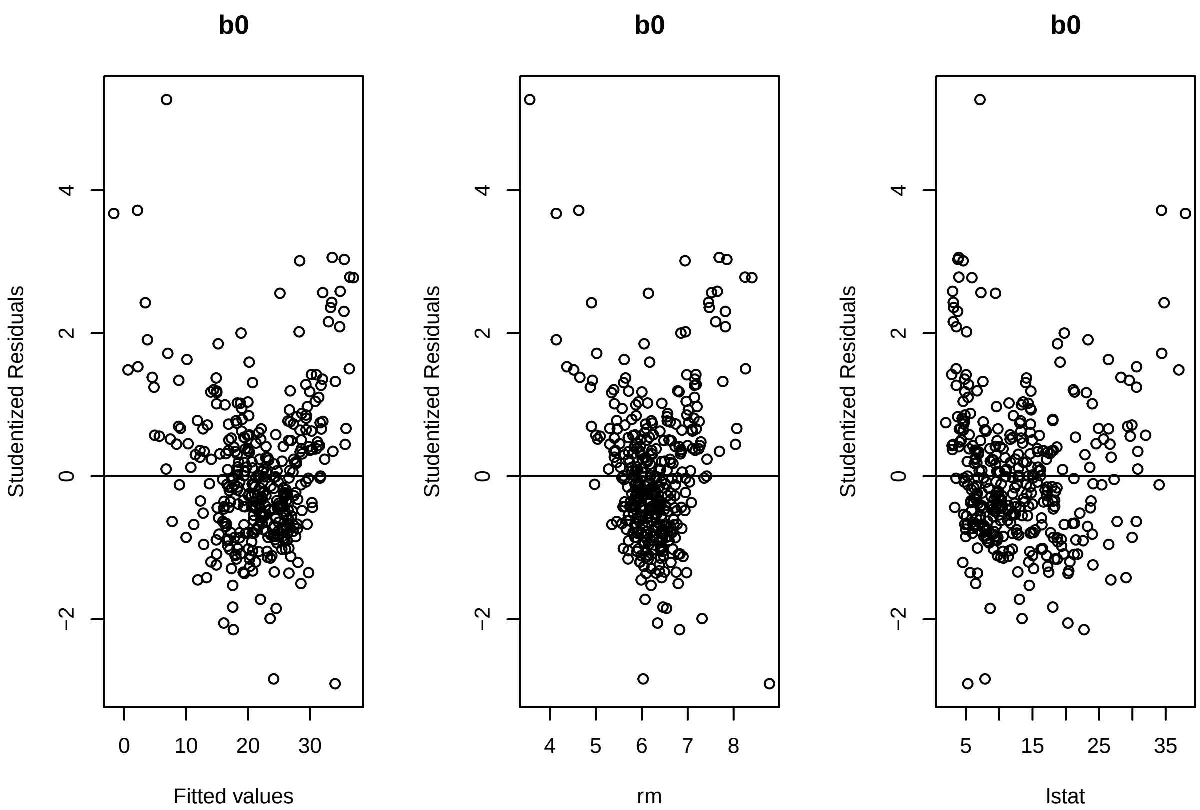
```
b0 = lm(medv ~ rm + tax + lstat, bost)
par(mfrow=c(1,3))
rp(b0,identify=TRUE)
```

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

```
rp(b0,"rm",identify=TRUE)
```

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

```
rp(b0,"lstat",identify=TRUE)
```




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

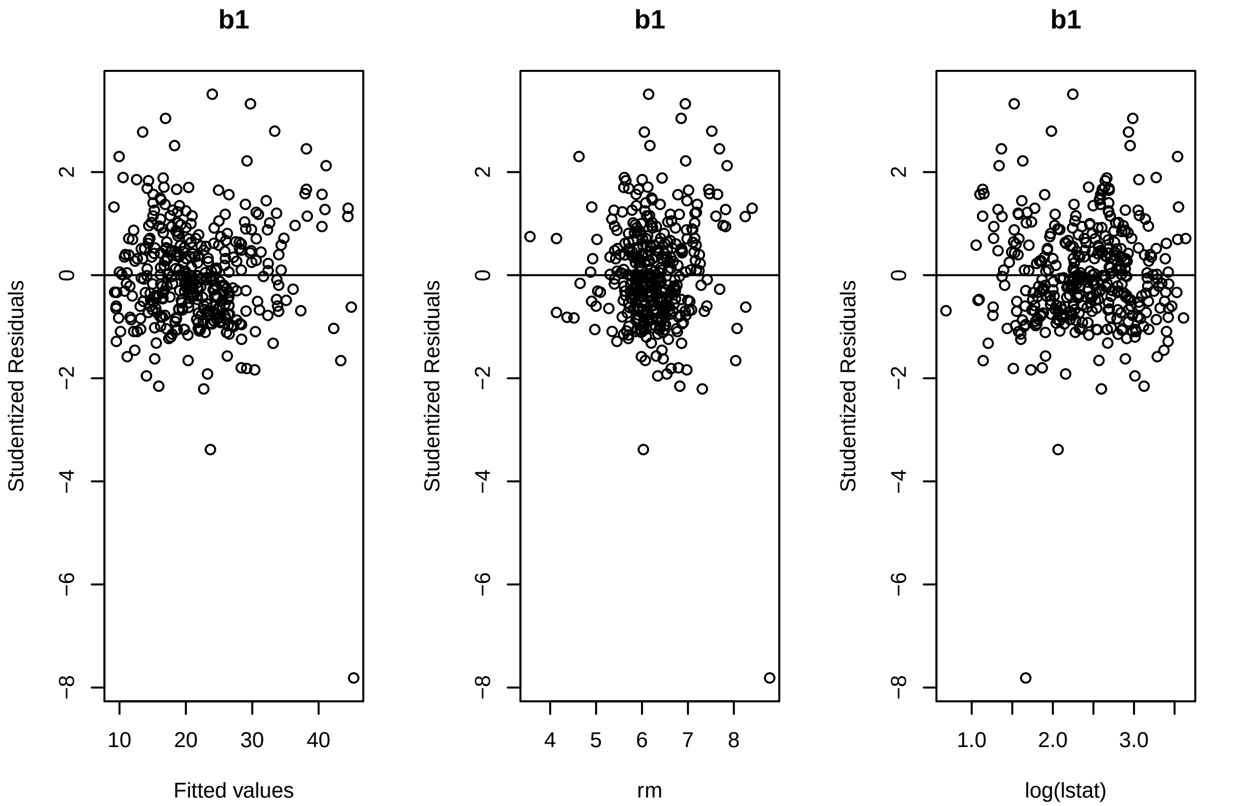
```
b1 = lm(medv ~ rm + I(rm^2) + tax + log(lstat), bost)
par(mfrow=c(1,3))
rp(b1,identify=TRUE)
```

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

```
rp(b1,"rm",identify=TRUE)
```

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

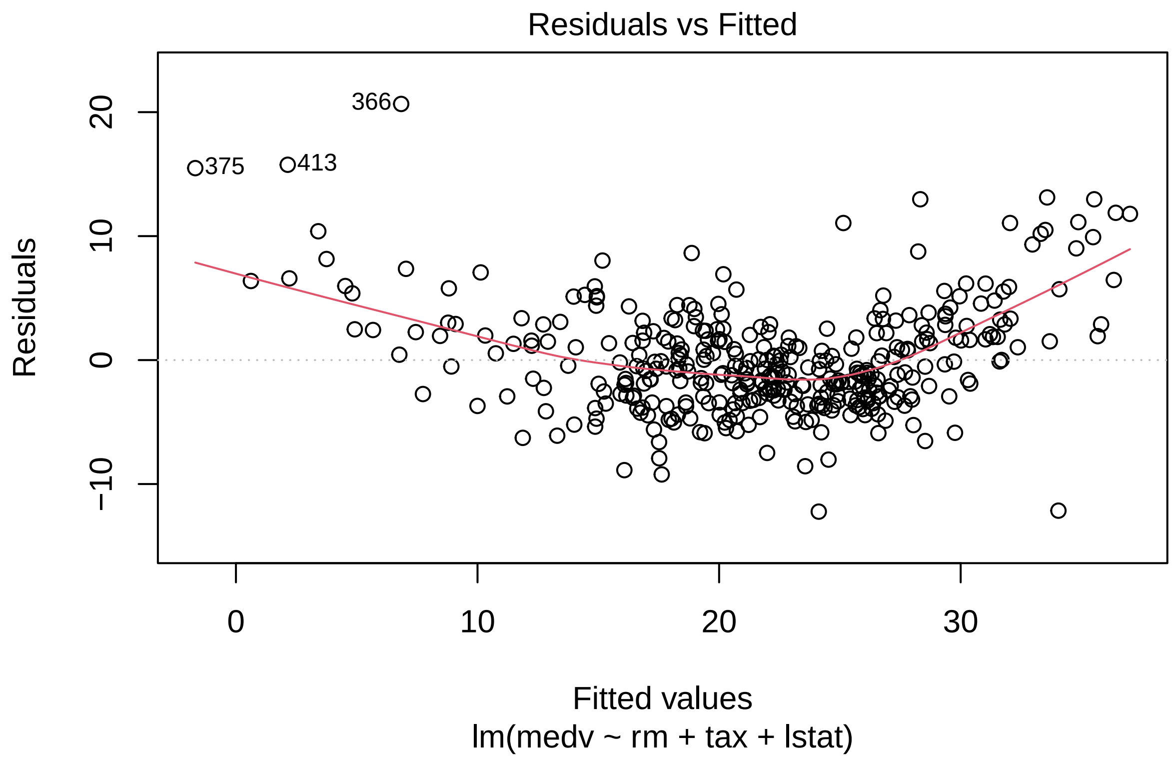
```
rp(b1,"log(lstat)",identify=TRUE)
```



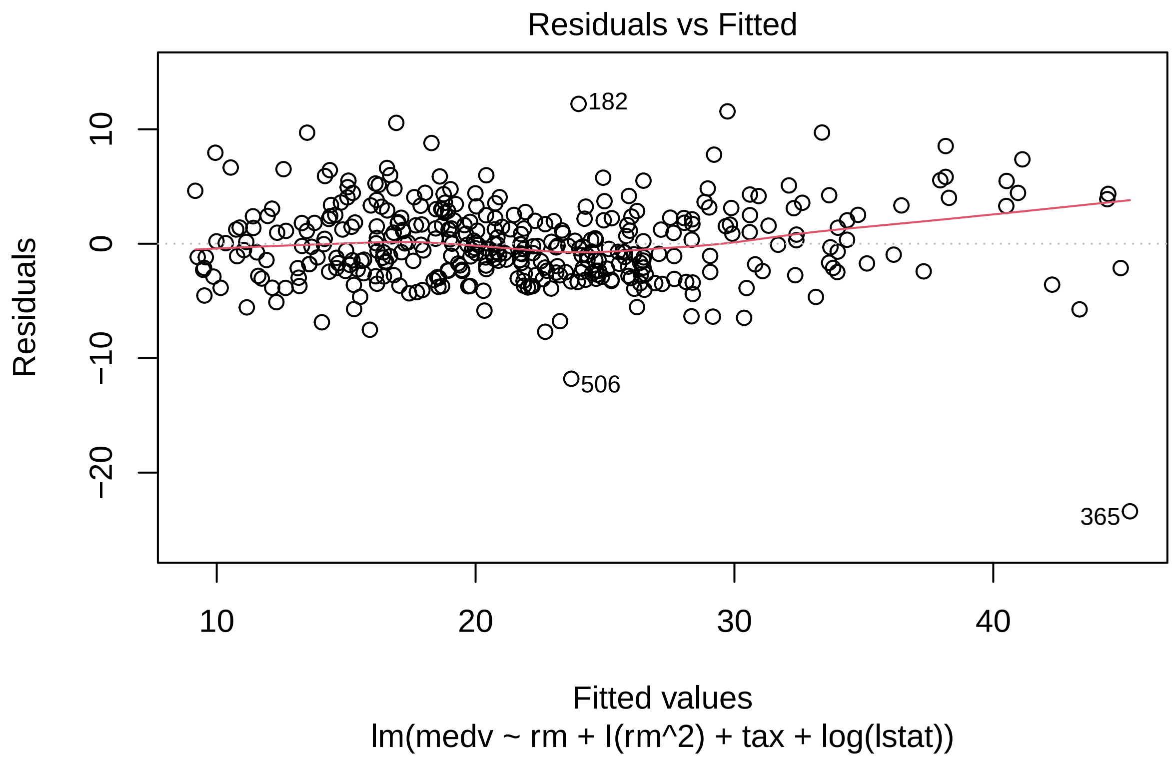

```
## integer(0)
```

For some reason the outliers I could not present using the rm function in the plot. Instead of that I am using fitted vs residual plot to indicate the outliers here.

```
plot(b0, which=c(1,1))
```




```
plot(b1, which=c(1,1))
```



The plot of the model after taking log transformation, shows that the row names of the worst outliers are 182, 506 and 365.

d

```
i1=influence.measures(b1)
# Count how many observations are "flagged" for a
 # particular influence measure:
sum(i1$is.inf[,"cook.d"])
```

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

```
# Show all influence measures for those observations
 # that were "flagged" for a particular influence measure:
 i1$infmat[i1$is.inf[,"cook.d"],]
```

```
##     dfb.1_     dfb.rm   dfb.I(^2    dfb.tax   dfb.lg()      dffit      cov.r
## -2.1254158  2.5812916 -2.8389275 -0.4273385 -0.6170212 -3.5726076  0.5677823
##     cook.d        hat
##  2.1945643  0.1729028
```

```
 # Show all data for those observations that were "flagged"
 # for a particular influence measure:
#bost[i1$is.inf[,"cook.d"],]

which(i1$is.inf[,"cook.d"])
```

```
## 365
```

```
## 279
```

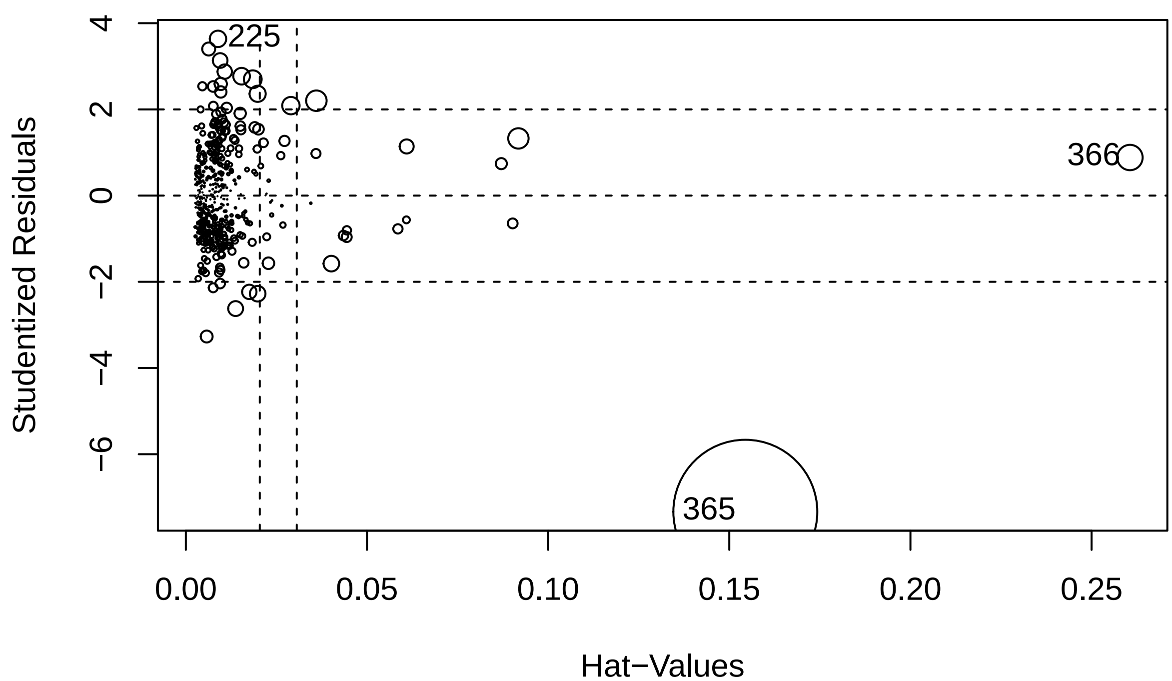
According to Cook's distance we found two suspected observation. But using the previous knowledge and this finding we could tell observation 365 should be investigated. ## e

```r
#install.packages("mice")
library(mice)
bost.mice = mice(bost,10)
```

```
##
##  iter imp variable
##   1   1  rm  lstat
##   1   2  rm  lstat
##   1   3  rm  lstat
##   1   4  rm  lstat
##   1   5  rm  lstat
##   1   6  rm  lstat
##   1   7  rm  lstat
##   1   8  rm  lstat
##   1   9  rm  lstat
##   1   10  rm  lstat
##   2   1  rm  lstat
##   2   2  rm  lstat
##   2   3  rm  lstat
##   2   4  rm  lstat
##   2   5  rm  lstat
##   2   6  rm  lstat
##   2   7  rm  lstat
##   2   8  rm  lstat
##   2   9  rm  lstat
##   2   10  rm  lstat
##   3   1  rm  lstat
##   3   2  rm  lstat
##   3   3  rm  lstat
##   3   4  rm  lstat
##   3   5  rm  lstat
##   3   6  rm  lstat
##   3   7  rm  lstat
##   3   8  rm  lstat
##   3   9  rm  lstat
##   3   10  rm  lstat
##   4   1  rm  lstat
##   4   2  rm  lstat
##   4   3  rm  lstat
##   4   4  rm  lstat
##   4   5  rm  lstat
##   4   6  rm  lstat
##   4   7  rm  lstat
##   4   8  rm  lstat
##   4   9  rm  lstat
##   4   10  rm  lstat
##   5   1  rm  lstat
##   5   2  rm  lstat
##   5   3  rm  lstat
##   5   4  rm  lstat
```

```
##   5   5  rm  lstat
##   5   6  rm  lstat
##   5   7  rm  lstat
##   5   8  rm  lstat
##   5   9  rm  lstat
##   5   10  rm  lstat
```

```
i=2
influencePlot(lm(medv ~ rm + I(rm^2) + tax + log(lstat),
data = complete(bost.mice, i)))
```




```
##        StudRes         Hat      CookD
## 225  3.6361175 0.008872418 0.02308927
## 365 -7.3353569 0.154429957 1.77243116
## 366  0.8837176 0.260600779 0.05507442
```

Most of the data have a similar influence except the suspected observations like observation 365 according to the Cook's distance or any other measure.

f

```
bost.lms = with(bost.mice, lm(medv ~ rm + I(rm^2) + tax + log(lstat)))
summary(pool(bost.lms))
```

```
##          term     estimate    std.error  statistic       df      p.value
## 1 (Intercept)  95.68114451 10.647691179   8.986093 50.38291 4.826806e-12
## 2          rm -21.77796865  3.484919363  -6.249203 38.43232 2.476396e-07
```

```
## 3     I(rm^2)   2.06160542  0.284868055   7.237054  34.07066 2.215143e-08
## 4         tax  -0.01165264  0.001304822  -8.930439 278.93414 0.000000e+00
## 5  log(lstat)  -6.01745084  0.531014183 -11.331996 116.38564 0.000000e+00
```

From the model summary we found all of the regressors are highly significant. rm, tax and log(lstat) have a negative coefficient value. We did not consider any interaction between variables here.

g

```r
bostX=bost[-365,]
bostX.mice= mice(bostX,10)
```

```
##
##  iter imp variable
##   1   1  rm  lstat
##   1   2  rm  lstat
##   1   3  rm  lstat
##   1   4  rm  lstat
##   1   5  rm  lstat
##   1   6  rm  lstat
##   1   7  rm  lstat
##   1   8  rm  lstat
##   1   9  rm  lstat
##   1   10  rm  lstat
##   2   1  rm  lstat
##   2   2  rm  lstat
##   2   3  rm  lstat
##   2   4  rm  lstat
##   2   5  rm  lstat
##   2   6  rm  lstat
##   2   7  rm  lstat
##   2   8  rm  lstat
##   2   9  rm  lstat
##   2   10  rm  lstat
##   3   1  rm  lstat
##   3   2  rm  lstat
##   3   3  rm  lstat
##   3   4  rm  lstat
##   3   5  rm  lstat
##   3   6  rm  lstat
##   3   7  rm  lstat
##   3   8  rm  lstat
##   3   9  rm  lstat
##   3   10  rm  lstat
##   4   1  rm  lstat
##   4   2  rm  lstat
##   4   3  rm  lstat
##   4   4  rm  lstat
##   4   5  rm  lstat
##   4   6  rm  lstat
##   4   7  rm  lstat
##   4   8  rm  lstat
##   4   9  rm  lstat
##   4   10  rm  lstat
```

```
##   5   1  rm  lstat
##   5   2  rm  lstat
##   5   3  rm  lstat
##   5   4  rm  lstat
##   5   5  rm  lstat
##   5   6  rm  lstat
##   5   7  rm  lstat
##   5   8  rm  lstat
##   5   9  rm  lstat
##   5   10  rm  lstat
```

```
bostX.lms = with(bostX.mice, lm(medv ~ rm + I(rm^2) + tax + log(lstat)))
summary(pool(bostX.lms))
```

```
##          term     estimate    std.error  statistic        df      p.value
## 1 (Intercept)  95.27131095 11.092577624   8.588744  34.85280 4.012730e-10
## 2          rm -21.88405079  3.466370460  -6.313246  34.24754 3.282389e-07
## 3     I(rm^2)   2.07964078  0.276953679   7.508984  33.85190 1.051889e-08
## 4         tax  -0.01174219  0.001291738  -9.090230 277.12664 0.000000e+00
## 5  log(lstat)  -5.85235864  0.543603762 -10.765854  81.57558 0.000000e+00
```

After removing obervation 365 and imputation we have found little change in the beta coefficient estimate for the intercept and rm and also their standard error.