CS498 AML,AMO HW6

Shrashti Singhal, Ankush Singhal

TOTAL POINTS

95 / 100

QUESTION 1

1 code for regression and resulting model. **0 / 0**

✓ **- 0 pts Correct**

QUESTION 2

2 a screenshot of your diagnostic plot and a few sentences of your explanation. **45 / 50**

+ **10 pts** Correct

+ **45 Point adjustment**

remove too many outliers

QUESTION 3

3 a screenshot of your new diagnostic plot. **20 / 20**

+ **0 pts** Correct

+ **20 Point adjustment**

QUESTION 4

4 a screenshot of your code for subproblem 2. **10 / 10**

+ **0 pts** Correct

+ **10 Point adjustment**

QUESTION 5

5 a screenshot of Box-Cox transformation plot and the best value you chose. **10 / 10**

✓ **- 0 pts Correct**

- **10 pts** Click here to replace this description.

QUESTION 6

6 result of the standardized residuals of the regression after Box-Cox transformation and a plot of fitted house price against true house price. **10 / 10**

✓ **- 0 pts Correct**

QUESTION 7

7 code for subproblems 3 and 4. **0 / 0**

✓ **- 0 pts Correct**

- **5 pts** Click here to replace this description.

QUESTION 8

8 late penalty **0 / 0**

✓ **- 0 pts Correct**

- **5 pts** 1 day
- **10 pts** 2 days
- **15 pts** 3 days
- **20 pts** 4 days
- **30 pts** max

1) Code for regression and resulting model

Load Data

getwd()

setwd("C:/Users/asinghal/Downloads/aml_hw6-master/aml_hw6-master")

housing_data = read.table("housing.data.txt", header=FALSE, col.names = c("CRIM", "ZN", "INDUS", "CHAS", "NOX", "RM", "AGE", "DIS", "RAD", "TAX", "PTRATIO", "B", "LSTAT", "MEDV"))

Generate regression model

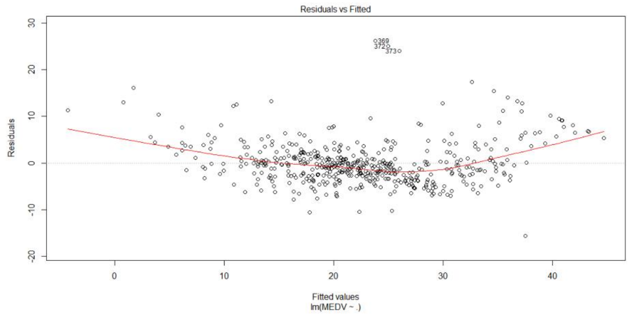
model = lm(MEDV ~ ., data = housing_data)

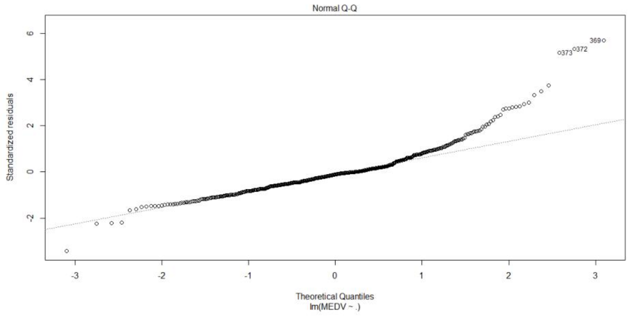
summary(model)

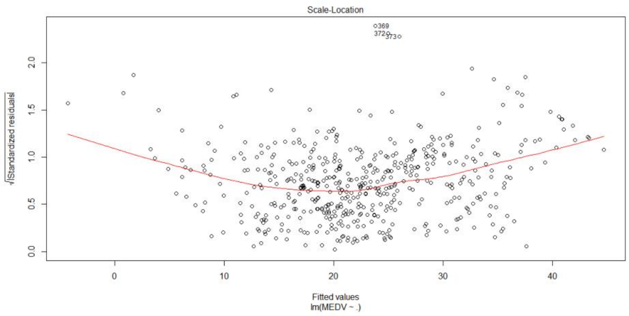
plot(model)

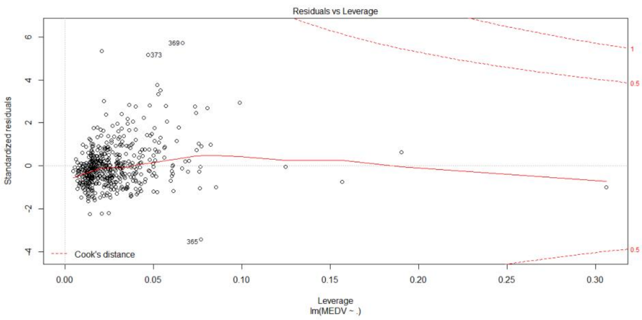
```
Call:
lm(formula = MEDV ~ ., data = housing_data)

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 ***
B            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
```

2) Screenshot of your diagnostic plot and a few sentences of your explanation:





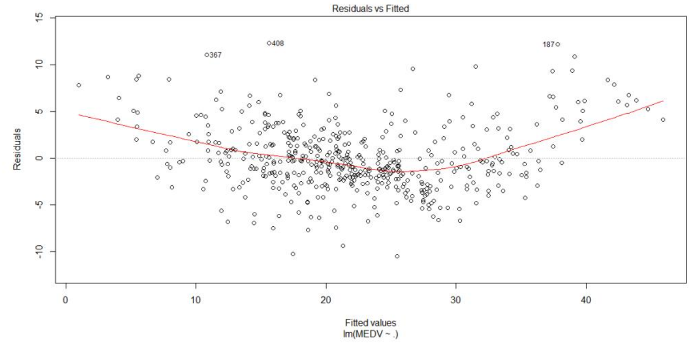


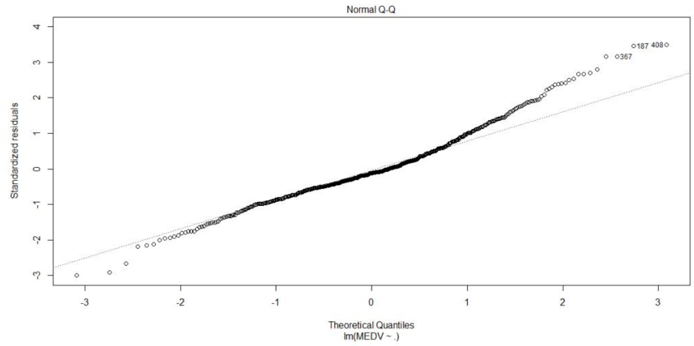


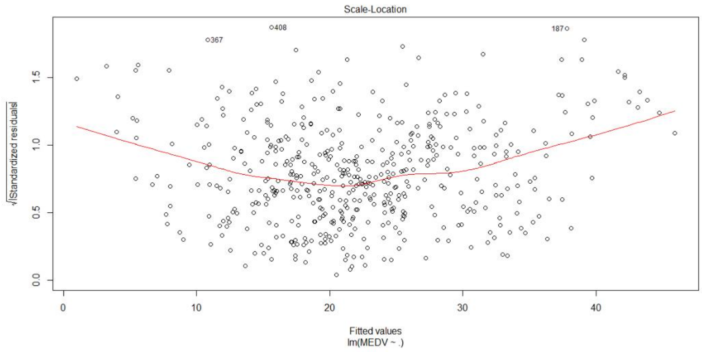
Following are the outliers were removed as we choose removed all points with a cook distance 4.

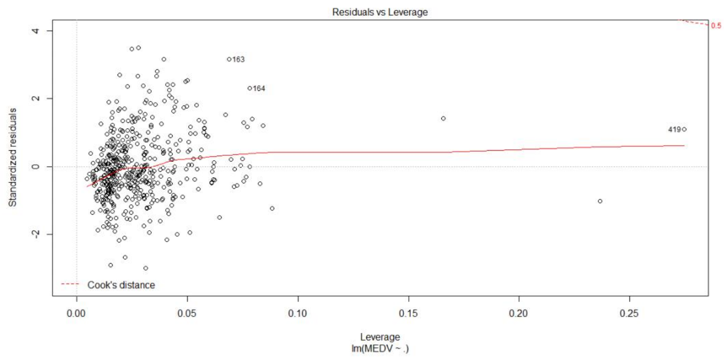
65,142,149,162,163,164,167,187,196,205,215,226,229,234,254,263,268,365,366,368,369,370,371,372,373,375,376,381,413,415.

3) Screenshot of your new diagnostic plot:







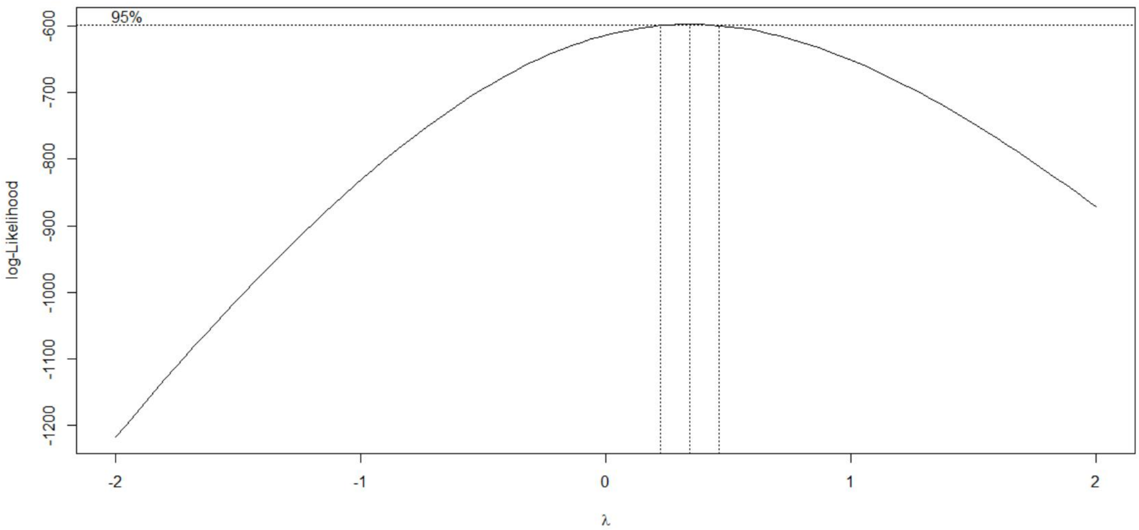


4) Screenshot of your code for subproblem 2.

```
fitted_resid = function(model, pointcol = "blue", linecol = "black") {
  plot(fitted(model), rstandard(model),
      col = pointcol, pch = 20, cex = 1.5,
      xlab = "Fitted", ylab = "Residuals")
  abline(h = 0, col = linecol, lwd = 2)
}
par(mfrow=c(1,1))
fitted_resid (model)

possible_outliers= as.numeric(names(resid(model)[cooks.distance(model) > 10 /
length(cooks.distance(model))]))

cat("Removing", length(possible_outliers), " outliers")
cleaned_data = housing_data[-possible_outliers, ]
new_model = lm(MEDV ~ ., data = cleaned_data)
summary(model)
plot(new_model)

par(mfrow=c(1,1))
fitted_resid (new_model)
```

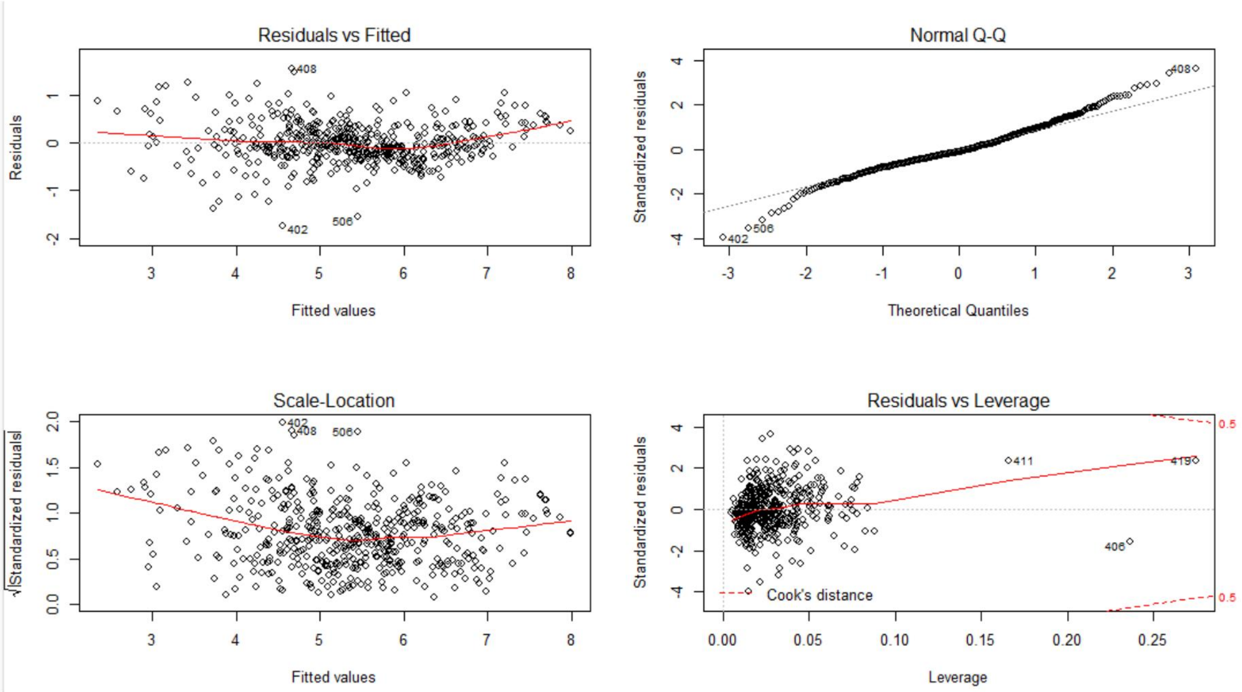
By: Shrashti Singhal & Ankush Singhal

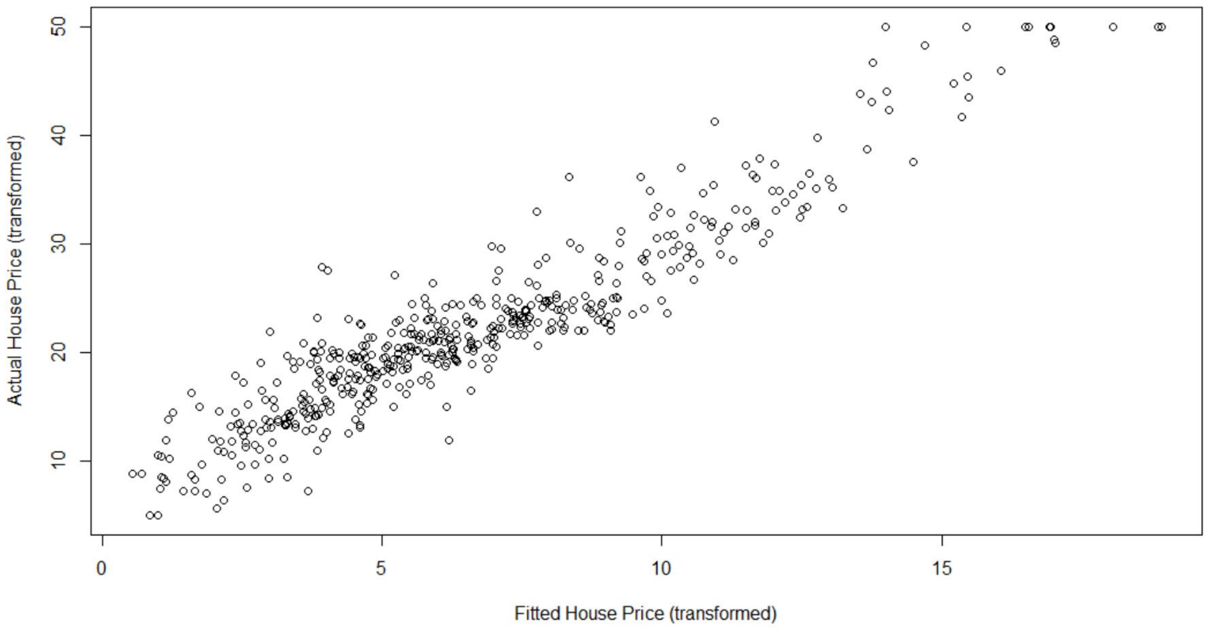
5) Screenshot of Box-Cox transformation plot and the best value you chose.



Best Value: 0.3434343

6) **Result of the standardized residuals of the regression after Box-Cox transformation and a plot of fitted house price against true house price.**





 By: Shrashti Singhal & Ankush Singhal

7) Code for subproblems 3 and 4.

```
# boxcox

library(MASS)

bc = boxcox(new_model)

lambda = bc$x[which.max(bc$y)]

lambda


transformed_model = lm(((MEDV ^ lambda - 1)/lambda) ~ ., data =
cleaned_data)

summary(transformed_model)

plot(transformed_model)

plot(((transformed_model$fitted.values)*lambda)^(1/lambda),
cleaned_housing_data$MEDV, xlab="Fitted House Price (transformed)",
ylab="Actual House Price (transformed)")
```