Applied Machine Learning, Fall 2018, Homework 6

Name: Emerson Sie (sie2), Junhao Pan (jpan22)

1. I removed 6 points. Their indices are $\{413, 373, 372, 371, 370, 369\}$.
2. The best value of λ is 0.263.

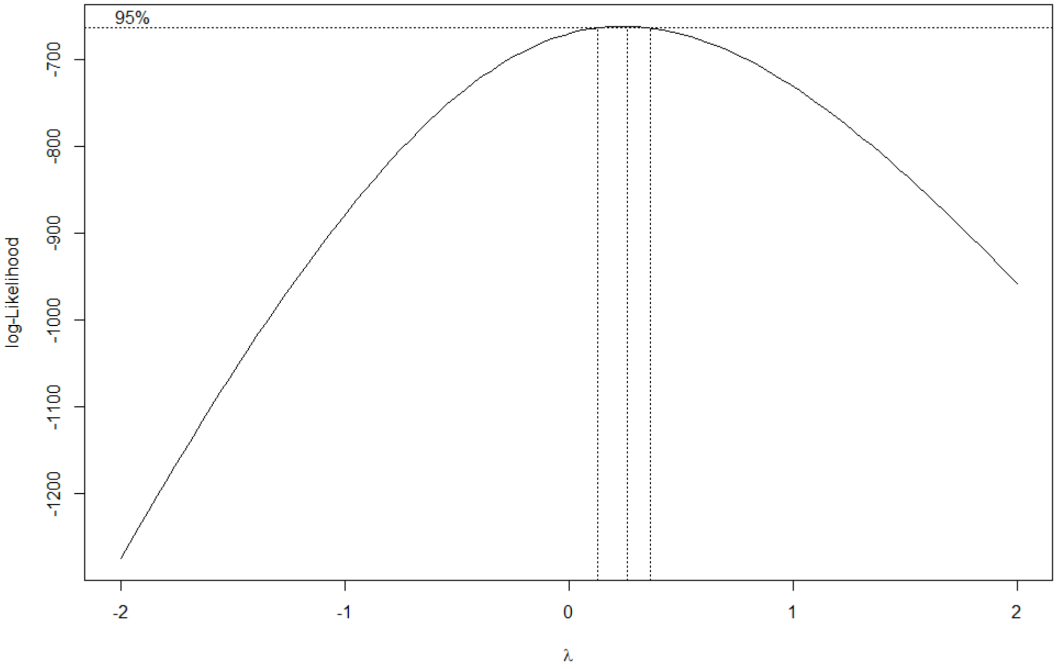


Figure 1: Box-Cox transformation curve.

3. To remove the outliers, we picked the points which have both high Cook's distance as well as leverage. We choose to remove six because removing too many outliers can skew the result.

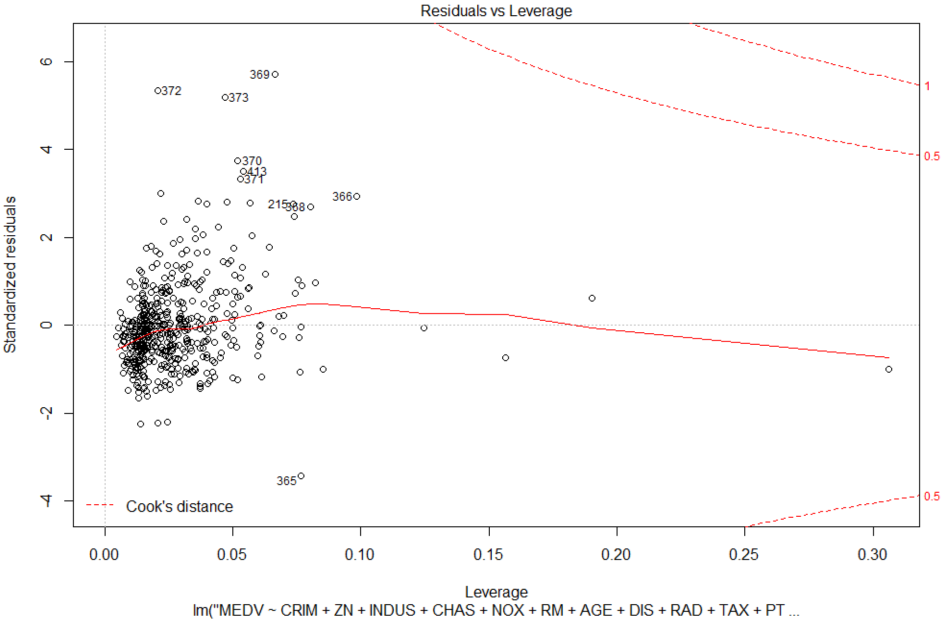


Figure 2: Diagnostic plot for part (a).

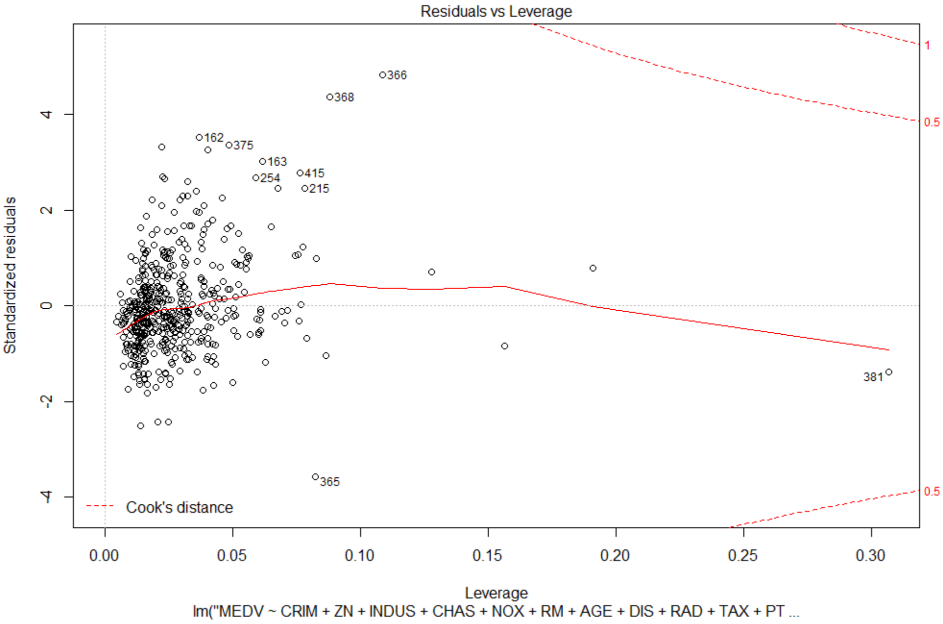


Figure 3: Final diagnostic plot for part (b), after removing outliers.

4.

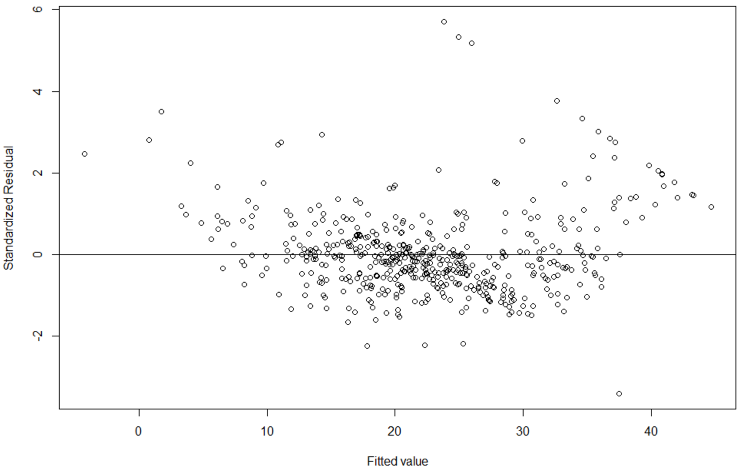


Figure 4: Standardized residuals vs fitted values for initial model.

5.

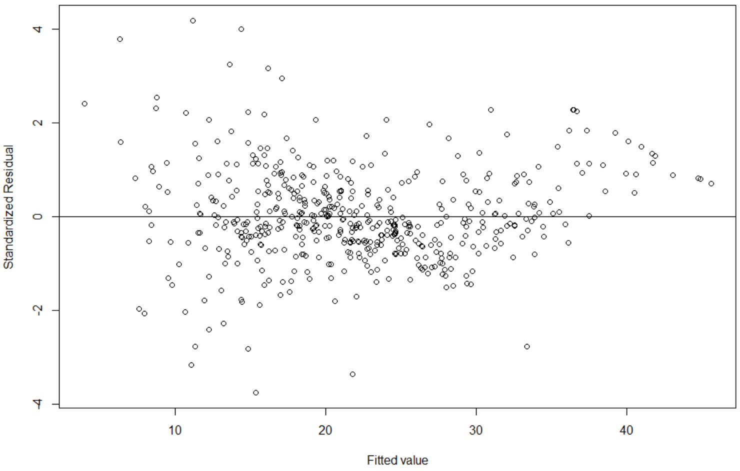


Figure 5: Standardized residuals vs fitted values after removing outliers and Box-Cox transform.

6. The points significantly more centered around the horizontal line after the Box-Cox transform and removing outliers. Also, the curvature of the set of points has decreased significantly. Both of these features suggest a better fit.

7.

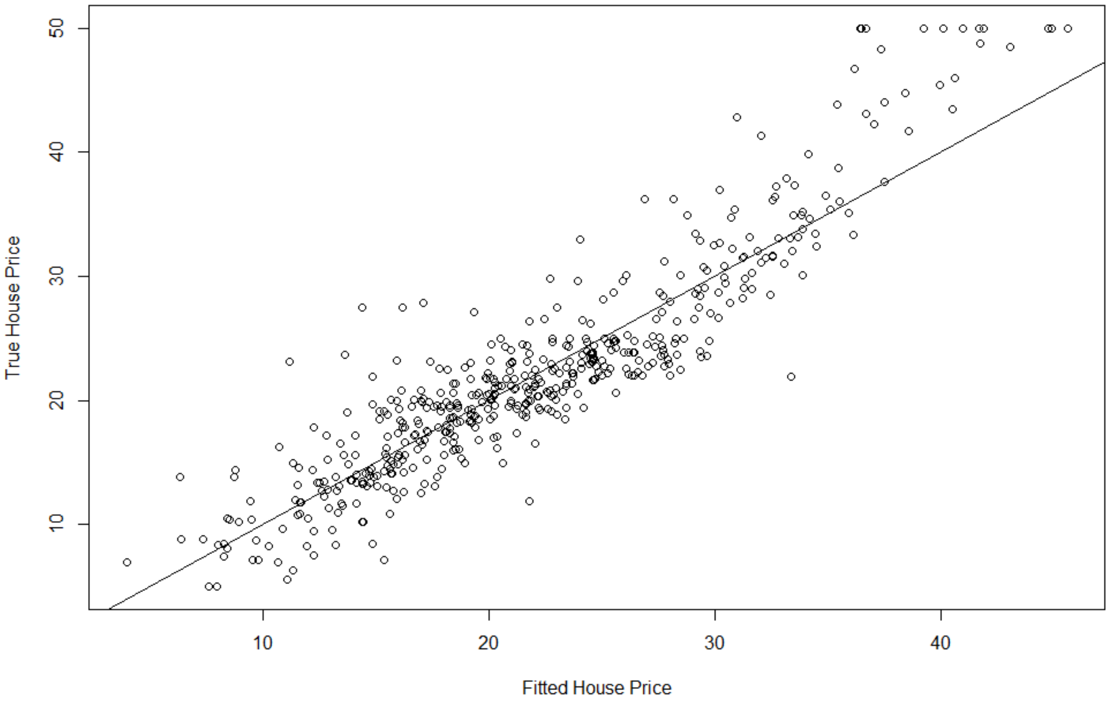
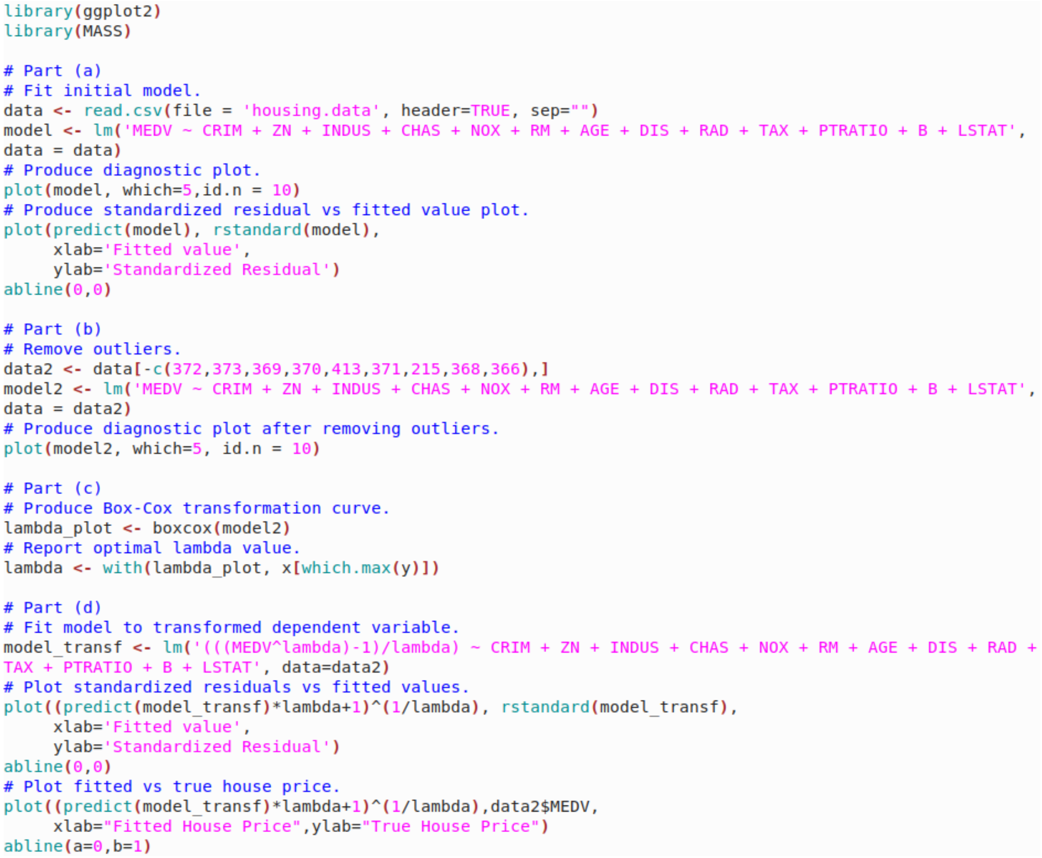


Figure 6: True vs fitted house price of final model.

The model produces a good fit of the data for low to intermediate values, but slightly under-predicts the house values at the higher end.

8.



```
library(ggplot2)
library(MASS)

# Part (a)
# Fit initial model.
data <- read.csv(file = 'housing.data', header=TRUE, sep="")
model <- lm('MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM + AGE + DIS + RAD + TAX + PTRATIO + B + LSTAT',
data = data)
# Produce diagnostic plot.
plot(model, which=5,id.n = 10)
# Produce standardized residual vs fitted value plot.
plot(predict(model), rstandard(model),
     xlab='Fitted value',
     ylab='Standardized Residual')
abline(0,0)

# Part (b)
# Remove outliers.
data2 <- data[-c(372,373,369,370,413,371,215,368,366),]
model2 <- lm('MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM + AGE + DIS + RAD + TAX + PTRATIO + B + LSTAT',
data = data2)
# Produce diagnostic plot after removing outliers.
plot(model2, which=5, id.n = 10)

# Part (c)
# Produce Box-Cox transformation curve.
lambda_plot <- boxcox(model2)
# Report optimal lambda value.
lambda <- with(lambda_plot, x[which.max(y)])

# Part (d)
# Fit model to transformed dependent variable.
model_transf <- lm('(((MEDV^lambda)-1)/lambda) ~ CRIM + ZN + INDUS + CHAS + NOX + RM + AGE + DIS + RAD +
TAX + PTRATIO + B + LSTAT', data=data2)
# Plot standardized residuals vs fitted values.
plot((predict(model_transf)*lambda+1)^(1/lambda), rstandard(model_transf),
     xlab='Fitted value',
     ylab='Standardized Residual')
abline(0,0)
# Plot fitted vs true house price.
plot((predict(model_transf)*lambda+1)^(1/lambda),data2$MEDV,
     xlab="Fitted House Price",ylab="True House Price")
abline(a=0,b=1)
```

Figure 7: Code screenshot.