- The index of the outliers removed: 365, 369, 370, 372, 373, 381, 406, 411, 415, 419.
- Following is a box-cox plot of the lambda parameter and its log-likelihood. The best lambda is around **0.182**.

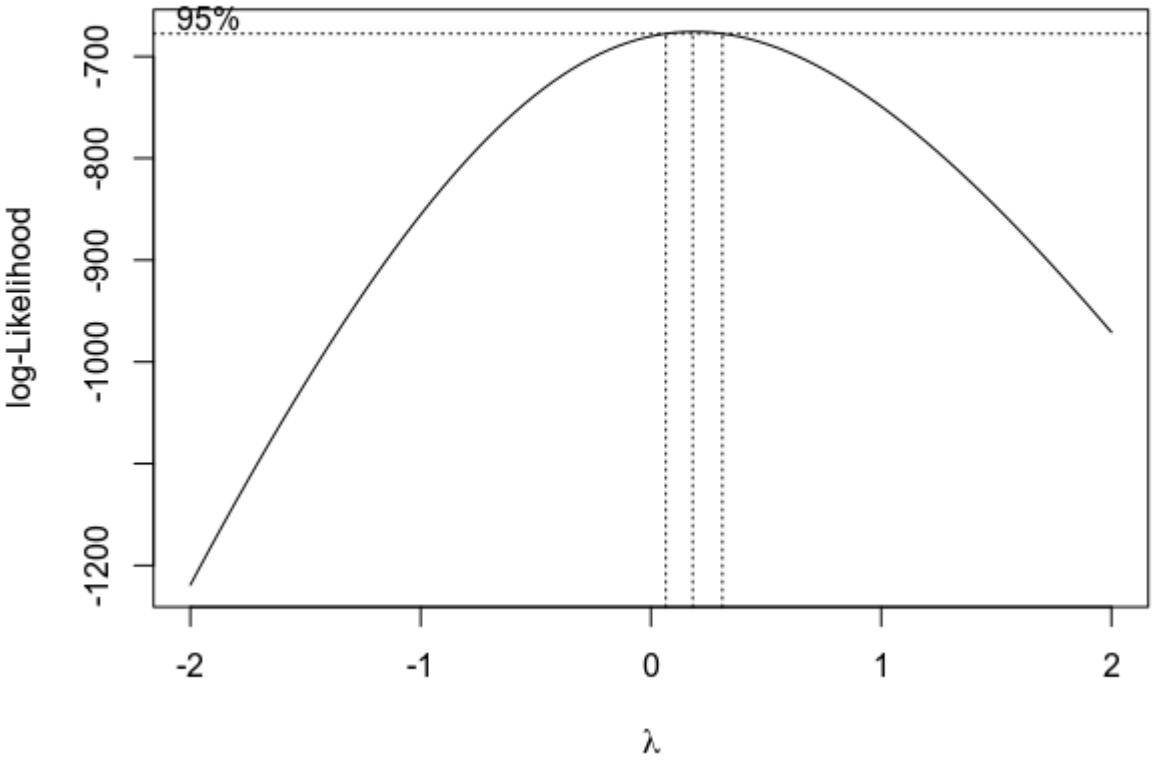

Figure 1: The log-likelihood vs. parameter of box-cox transformation.

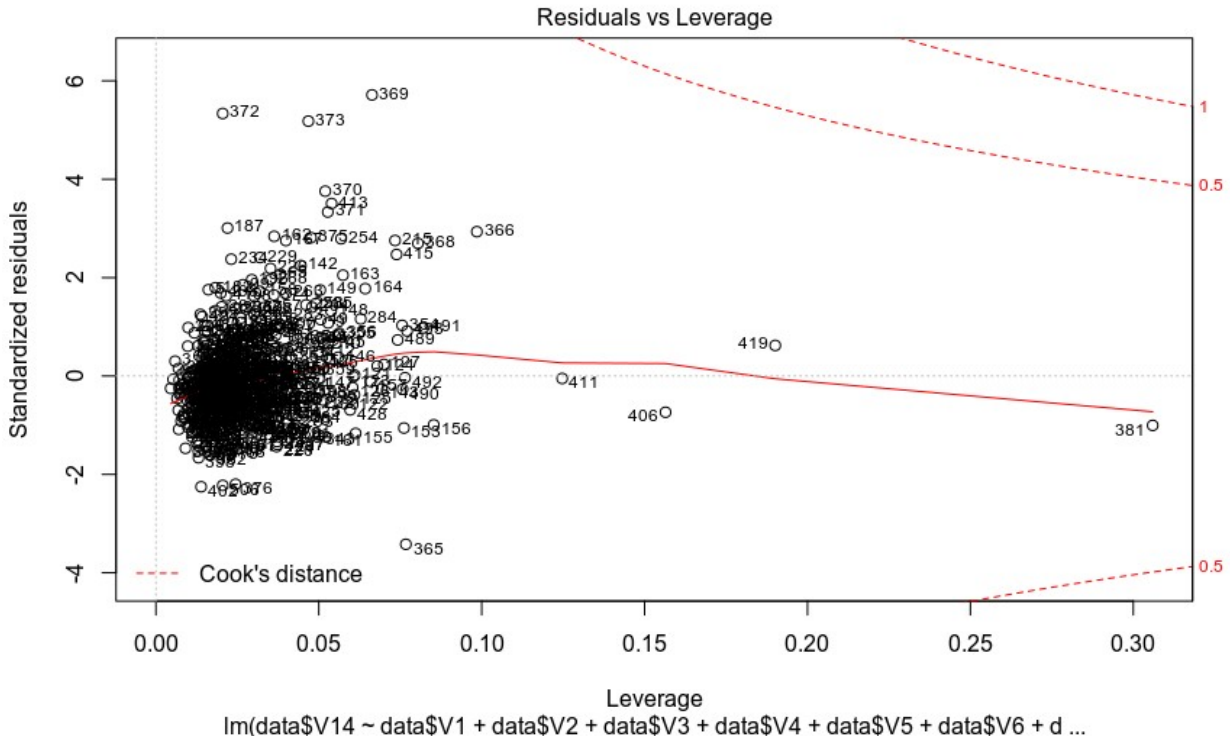

Figure 2: Standardized Residual vs. Leverage vs. Cook's Distance plot with outliers.

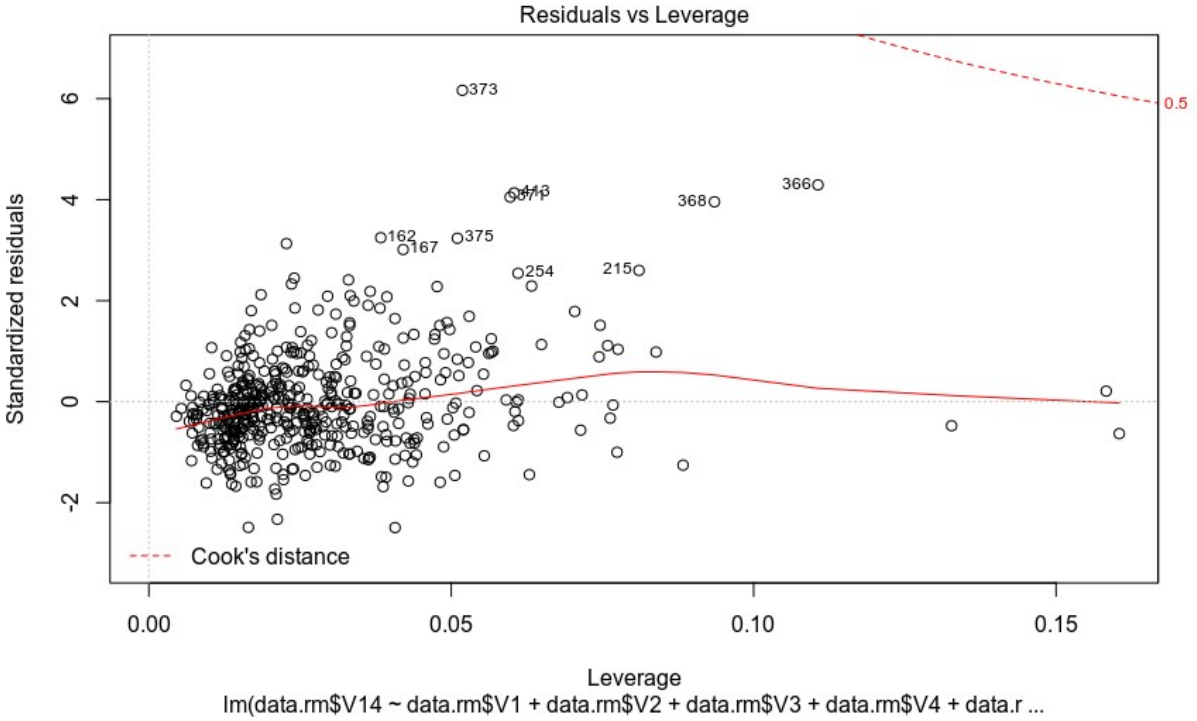

Figure 3: Standardized Residual vs. Leverage vs. Cook's Distance plot without outliers.

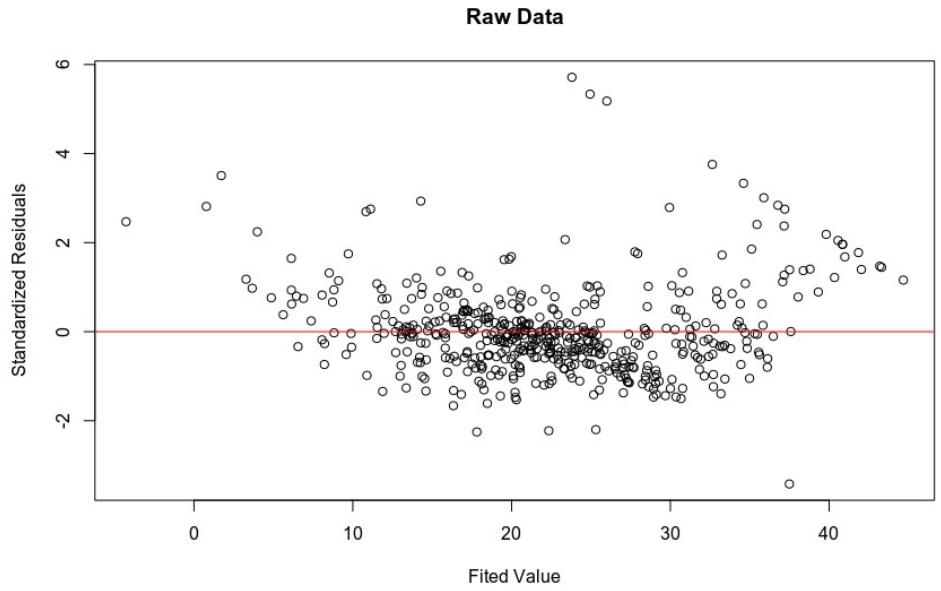

Figure 4: Standardized Residual vs. Fitted Values **without** transformation

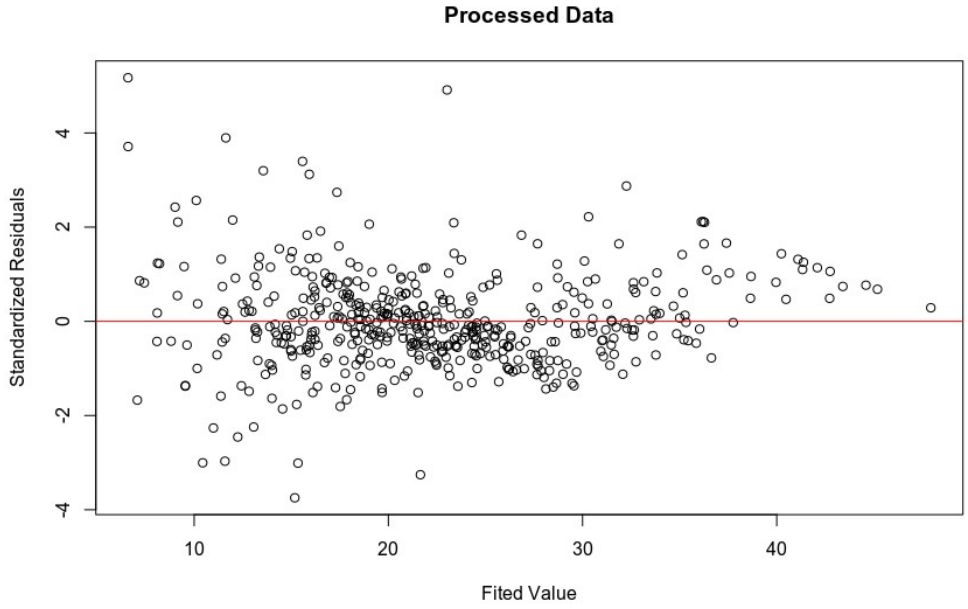

Figure 5: Std Residual vs. Fitted Values **with** 10 outliers removed & transforming the dependent variable

Comparing the above two graphs, we can observe that with transformation, the linear regression can yield results with standardized residuals that are closely center to 0 compared to the result generated by using original data.

Figure 6: The plot of fitted house price vs true house price

From the above graph, we can observe that the plot of fitted house price vs true house price follows the line of $y = x$ quite while. Although there are several points off, the trend is mostly linear. This indicates that our linear regression has yielded desirable results.

```r
setwd('~/Documents/Applied-Machine-Learning/HW-7/')
library(MASS)
data <- read.table('housing_data.txt', header = FALSE)

# Problem A, starting fiting using the data and plot the graphs
data.lm <-
lm(data$V14~data$V1+data$V2+data$V3+data$V4+data$V5+data$V6+data$V7+data$V8+data
$V9+data$V10+data$V11+data$V12+data$V13, data = data)
# par(mfrow=c(2,2), oma = c(0, 0, 2, 0))  # Used for formatting, not necessary
here
par(c(0, 0, 2, 0))
plot(data.lm, id.n = 506) # Use this line to show all the indices of all the
data
sum = summary(data.lm)

# Problem B, performing data removal
# Simply re-run many times of the reduction to finalized the following 9 points
data.rm <-data[-c(365, 369, 370, 372, 373, 381, 406, 411, 415, 419),]
data.rm.lm <-
lm(data.rm$V14~data.rm$V1+data.rm$V2+data.rm$V3+data.rm$V4+data.rm$V5+data.rm$V6
+data.rm$V7+data.rm$V8+data.rm$V9+data.rm$V10+data.rm$V11+data.rm$V12+data.rm$V1
3, data = data.rm)
# par(mfrow=c(2,2), oma = c(0, 0, 2, 0))  # Used for formatting, not necessary
here
par(c(0, 0, 2, 0))
plot(data.rm.lm, id.n = 10)


# Problem C
box_cox_result <- boxcox(data.rm.lm)
best_lam <- box_cox_result$x[which(box_cox_result$y == max(box_cox_result$y))]

# Problem D
data.box <- (data.rm$V14^best_lam-1)/best_lam
box.lm <-
lm(data.box~data.rm$V1+data.rm$V2+data.rm$V3+data.rm$V4+data.rm$V5+data.rm$V6+da
ta.rm$V7+data.rm$V8+data.rm$V9+data.rm$V10+data.rm$V11+data.rm$V12+data.rm$V13,
data = data.rm)
# Plot the original graph
data.stdres = rstandard(data.lm)
plot(data.lm$fitted.values, data.stdres, ylab="Standardized Residuals",
xlab="Fited Value", main="Raw Data")
abline(0, 0, col='red')                 # the horizon

# Plot the processed data
box.stdres = rstandard(box.lm)
box.lm.fitted_retrans = (box.lm$fitted.values*best_lam+1)^(1./best_lam)
plot(box.lm.fitted_retrans, box.stdres, ylab="Standardized Residuals",
xlab="Fited Value", main="Processed Data")
abline(0, 0, col='red')                 # the horizon

# Problem E
plot(box.lm.fitted_retrans, data.rm$V14, ylab="True House Price", xlab="Fitted
House Price", main="Fitted vs True House Price")
abline(1, 1, col='red')
```