CS498 AML HOMEWORK 6

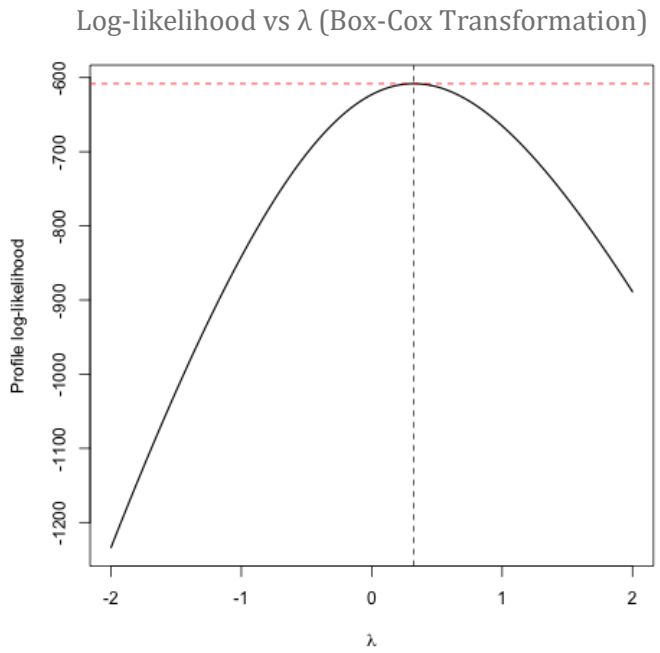
Zhenbang WU (zw12)
Chao XU (chaox2)

1. List all the points (row numbers) you removed (indexed on the original dataset) as outlier points

   365, 366, 368, 369, 370, 371, 372, 373

2. Box-Cox Transformation - Plot the Box-Cox transformation curve (Log-likelihood vs Parameter value). What is the best value of the parameter you got?

   i. the Box-Cox transformation curve



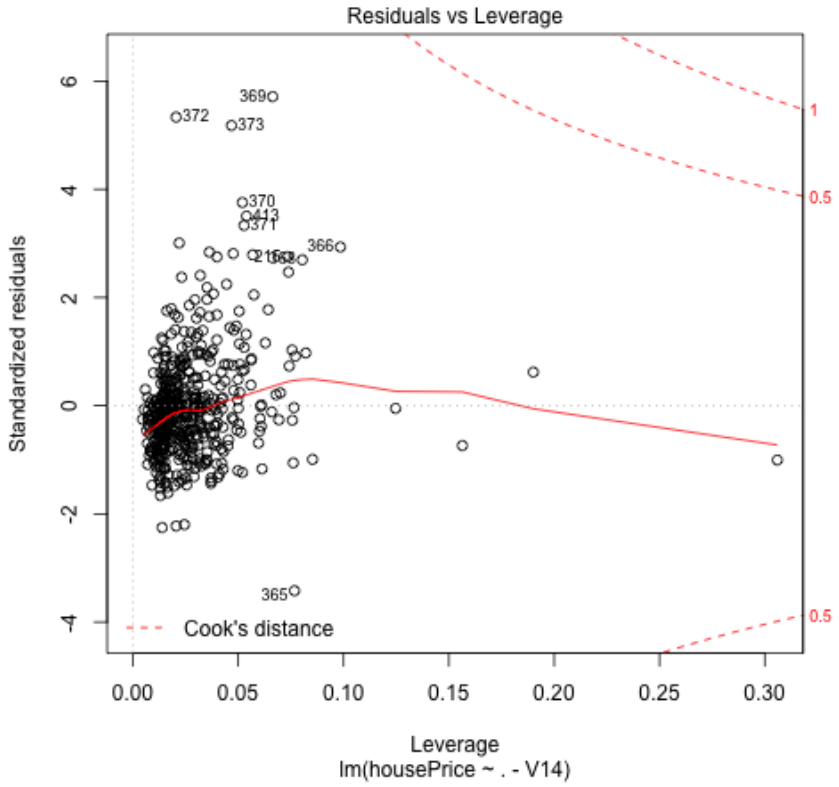
   ii. best value of the parameter: 0.3229465

CS498 AML HOMEWORK 6

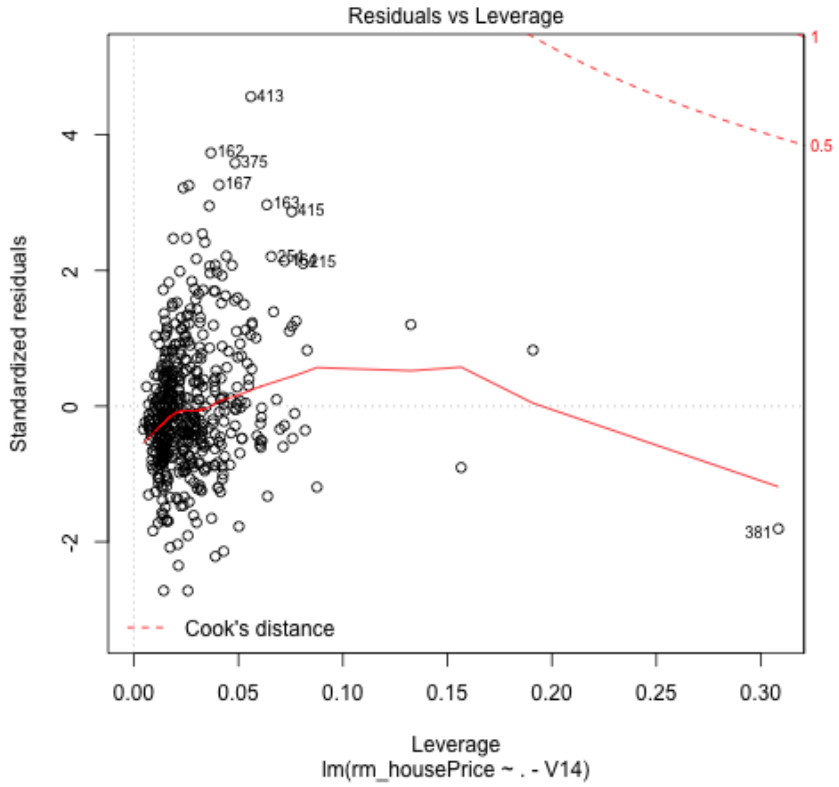
Zhenbang WU (zw12)
Chao XU (chaox2)

3. Diagnostic plots used for identification of outliers. Please only include the Standard residuals vs Leverage vs Cook's distance plots (do not put other 3 plots you obtain for R). The final diagnostic plot obtained after removing all outliers should also be included.

i. the Original Diagnostic Plot



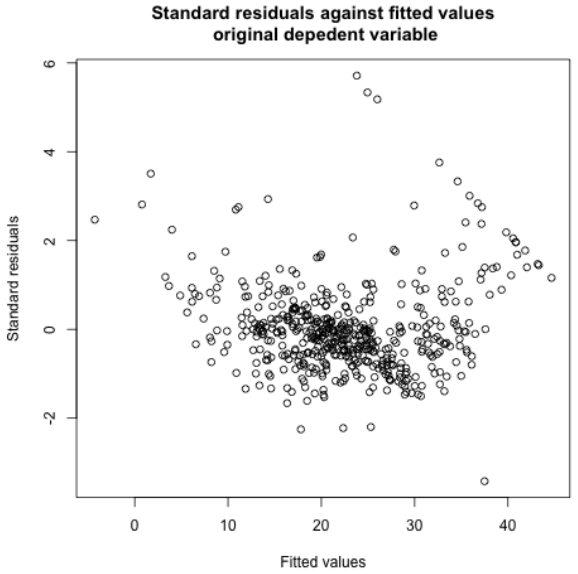
ii. the Final Diagnostic Plot

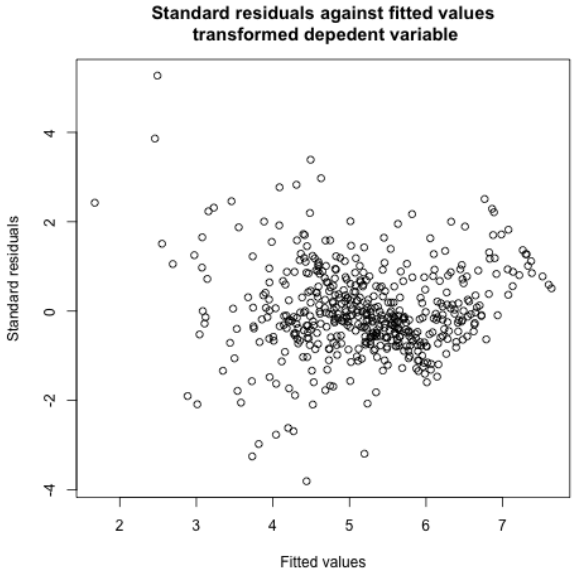


CS498 AML HOMEWORK 6

Zhenbang WU (zw12)
Chao XU (chaox2)

4. Plot of Standardized residuals vs Fitted values for the linear regression model obtained without any transforms (like removing outliers or transforming dependent variables) -**THE LEFT FIGURE BELOW**
5. Plot of Standardized residuals vs Fitted values for the final linear regression model obtained after removing all outliers and transforming the dependent variable -**THE RIGHT FIGURE BELOW**





6. Compare the two plots. What do you observe?

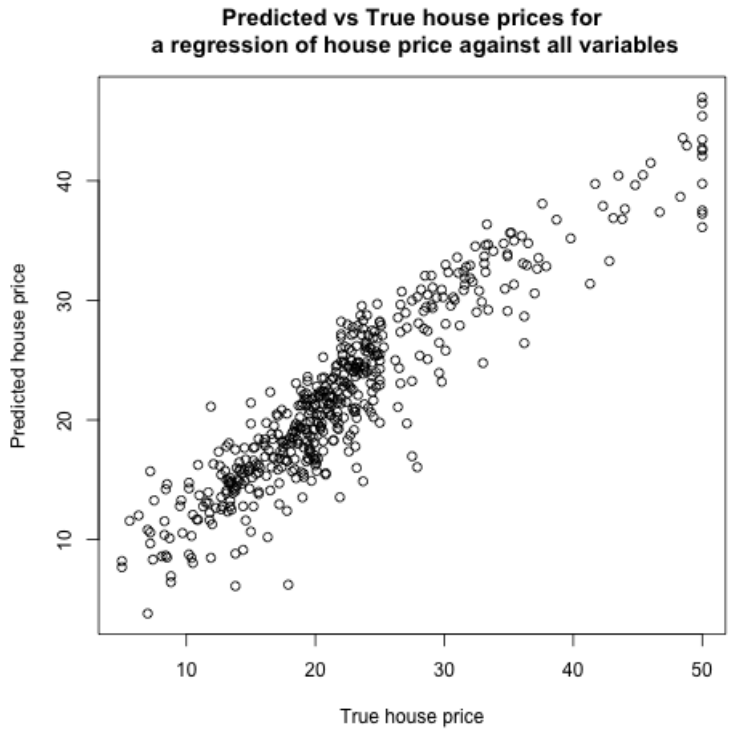
i. The original plot is more banana-shaped (the residuals are larger for small and for large predicted values), and its residual seems to depend more strongly on the predicted value (denser). The transformed plot's residual seems more random/independent.

ii. In the original plot, there are some points that seem to come from some other model (e.g. top three & bottom one). After removing outliers, those strange points disappear, and the standard residual improves. Notice that the axes are different (the transformed one is the residual of the power of λ). Surely the fitted values change too since the power of λ in the transformation.

CS498 AML HOMEWORK 6

Zhenbang WU (zw12)
Chao XU (chaox2)

7. Final plot of Fitted house price vs True house price. What do you observe?



Observations:

i. This regression looks fairly good at predicting housing price from other measurement (especially when the housing prices are in the middle range);

ii. This regression does not perform very well when the housing prices are too high or too low (which is reasonable because most data we used to train fall in the middle range, and also, from our life experience, extremely low-price houses and extremely high-price houses should be of various types).

CS498 AML HOMEWORK 6

Zhenbang WU (zw12)
Chao XU (chaox2)

8. One-page Code screenshot. It should include code for Linear regression, Box-Cox transformation and how you used the parameter value to transform the dependent variable.

   i. Linear regression

```
3  # regress house price against all others
4  relation<-lm(housePrice~.-V14,data=housingData)
5  # ------ some other code ------
6  # regress house price against all others after removing outliers
7  rm_relation<-lm(rm_housePrice~.-V14, data=rm_housingData)
8  # ------ some other code ------
9  # regress house price against all others after removing outliers
10 # and transforming depedeng variable
11 transf_rm_relation<-lm(transf_rm_housePrice~.-V14, data=rm_housingData)
```

   ii. Box-Cox transformation

```
13 # box-cox transform
14 boxcox_list<-boxcox(rm_relation)
15 # get the best value of parameter
16 best_lamdba<-boxcox_list$lambdahat
```

   iii. Transform the dependent variable

```
18 # transform depedent variable
19 transf_rm_housePrice<-lapply(rm_housePrice,
20                             function(x) (x^best_lamdba - 1) / best_lamdba)
21 # convert list to double in order to fit the input type for lm()
22 transf_rm_housePrice<-as.numeric(unlist(transf_rm_housePrice))
23 # ------ some other code ------
24 # transfrom our predicted values back to get the final prediction
25 final_prediction<-lapply(transf_rm_relation$fitted.values,
26                          function(x) (x*best_lamdba + 1)^(1/best_lamdba))
```

```
# CS498AML HW6
# written by Zhanbang Wu and Chao Xu on Oct 22 2018

#####################################################################
# Initialization
#####################################################################
# set the workspace here
setwd('/Users/Zachary/playground/CS498/HW6')
getwd()

# include library
# https://www.rdocumentation.org/packages/base/versions/3.5.1
library("base")
# https://www.rdocumentation.org/packages/trafo/versions/1.0.0
library("trafo")

# read the data and extract out measurements and housing price
housingData<-read.table('housing.data.txt')

#####################################################################
# Deal with original data
#####################################################################
# extract dependant variable
housePrice<-housingData[,14]
# regress house price against all others
relation<-lm(housePrice~.-V14,data=housingData)
# plot residuals vs leverage vs cook's distance
png(filename='output/orig_diagnostic_plot.png')
plot(relation, which=c(5), id.n=10)
dev.off()
# plot residuals vs fitted values
png(filename='output/res_fitted.png')
plot(relation$fitted.values, rstandard(relation),
     xlab="Fitted values", ylab="Standard residuals",
     main="Standard residuals against fitted values\n original depedent
variable")
dev.off()

#####################################################################
# Deal with removed-outliers data
#####################################################################
# run several test here
# to_removed<-c(-365, -369, -372, -373)
# to_removed<-c(-365, -369,  -372, -373, -366, -370)
# to_removed<-c(-365, -369,  -372, -373, -366, -370, -368, -371, -413)
# to_removed<-c(-365, -369,  -372, -373, -366, -370, -368, -371, -381)
```

```
# remove possible outliers
to_removed<-c(-365, -369,  -372, -373, -366, -370, -368, -371)
rm_housingData<-housingData[to_removed,]
# extract dependant variable
rm_housePrice<-rm_housingData[,14]
# regress house price against all others after removing outliers
rm_relation<-lm(rm_housePrice~.-V14, data=rm_housingData)
# plot residuals vs leverage vs cook's distance
png(filename='output/rm_diagnostic_plot.png')
plot(rm_relation, which=c(5), id.n=10)
dev.off()

################################################################
# Deal with transformed and  removed-outliers data
################################################################
# Box-Cox transformation
png(filename='output/orig_box_plot.png')
boxcox_list<-boxcox(rm_relation)
dev.off()
# get the best value of parameter
best_lamdba<-boxcox_list$lambdahat
# transform depedent variable
transf_rm_housePrice<-lapply(rm_housePrice, function(x) (x^best_lamdba - 1)
/ best_lamdba)
# convert list to double in order to fit the input type for lm()
transf_rm_housePrice<-as.numeric(unlist(transf_rm_housePrice))
# regress house price against all others after removing outliers
# and transforming depedeng variable
transf_rm_relation<-lm(transf_rm_housePrice~.-V14, data=rm_housingData)
# plot standard residuals vs leverage vs cook's distance
png(filename='output/transf_rm_diagnostic_plot.png')
plot(transf_rm_relation, which=c(5), id.n=10)
dev.off()
# plot standard residuals vs fitted values
png(filename='output/transf_rm_res_fitted.png')
plot(transf_rm_relation$fitted.values, rstandard(transf_rm_relation),
     xlab="Fitted values", ylab="Standard residuals",
     main="Standard residuals against fitted values\n transformed depedent
variable")
dev.off()

################################################################
# Get the final plot
################################################################
# transfrom our predicted values back to get the final prediction
final_prediction<-lapply(transf_rm_relation$fitted.values,
                         function(x) (x*best_lamdba + 1)^(1/best_lamdba))
# final plot of Fitted house price vs True house price
png(filename='output/final_plot.png')
```

```
plot(rm_housePrice, final_prediction,
     xlab="True house price", ylab="Predicted house price",
     main="Predicted vs True house prices for\n a regression of house price
against all variables")
dev.off()
```