Assignment 3

Yizhou Tang, 250888541

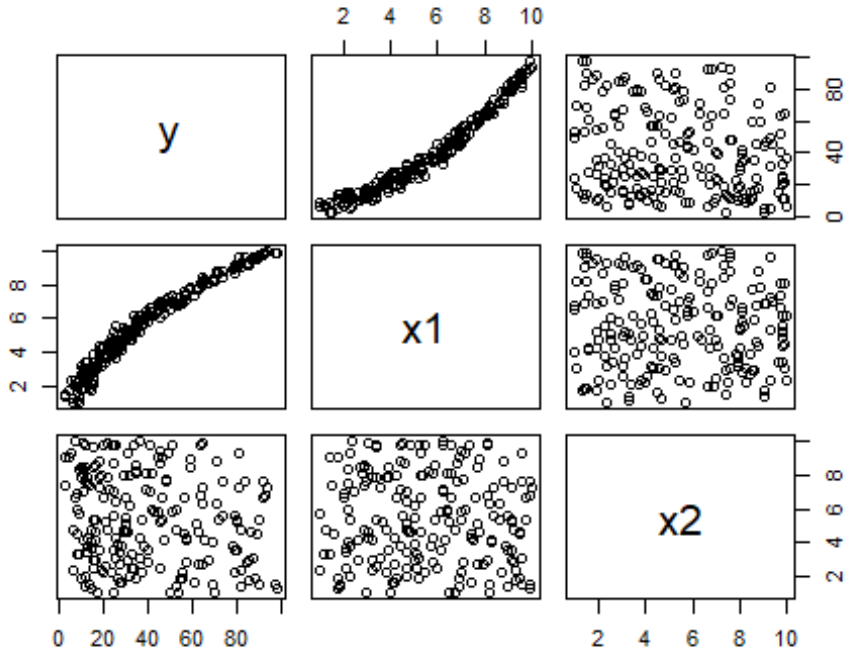
November 13, 2019

Question 1

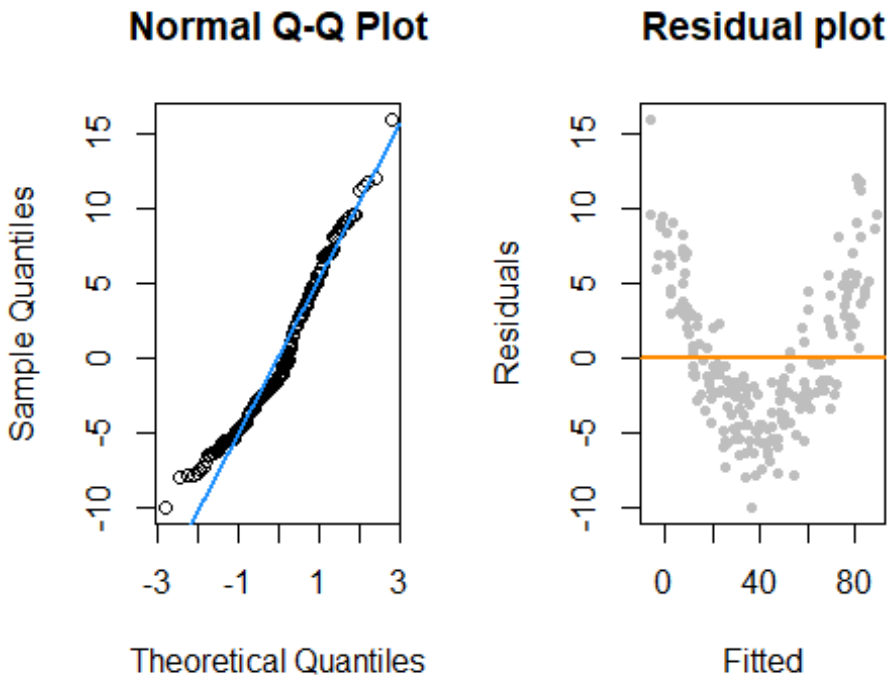
```
#1
#a)
#True, increasing the number of predictors will increase R^2.The model curve
fit more as it gets more complex.

#b)
#True, When multicollinearity exists, it increases the variance of the
estimated beta parameters, which will the increase in standard error of the
parameter estimates.

#c)
#False, VIF of beta estimates are depended on the R^2 of the regression
response of xj on the other predictors

#d)
#False, a high leverage point "could have a large influence", but not always.
A highly influential point needs to have both a high leverage and residual.

#e)
#No. For example, BIC penalizes the complexity more than AIC. Hence, the
result set of predictor variables would most likely be less complex.
```

Question 2

```
hw3data =
read.csv("https://raw.githubusercontent.com/hgweon2/ss3859/master/hw3-
data.txt")

#a)
pairs(hw3data)
```




```r
#The scatter plot shows a very significant linear relationship between y and
x1
#y & x2: No relationship reflected on the scatter plot
#x1 & x2: No relationship reflected on the scatter plot

#b)
model = lm(y ~ x1+x2, data = hw3data)
par(mfrow=c(1,2)) # Combining plots
qqnorm(resid(model))
qqline(resid(model), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(model), resid(model), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```



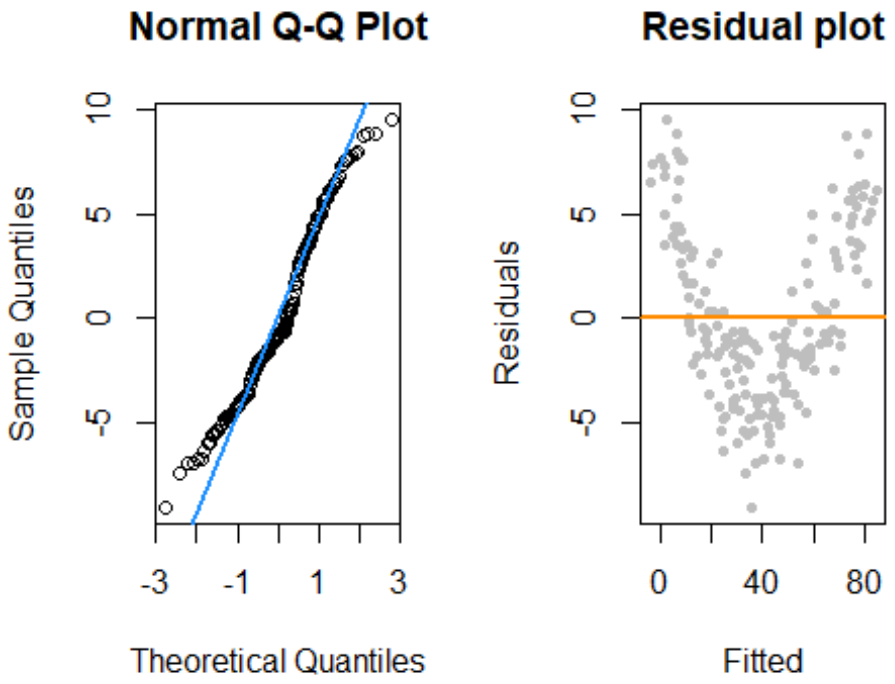

```
#Normality is vioated: the observations do not seem to follow a normal
distribution when comparing the tails.

#Linearity is violated, because residual plot shows mean of e varies
systematically

#Equal variance is not violated, because the spread of e does appear to be
constant

#c)
lev_fit = lm(y~.,data = hw3data)

# Cook's distance
temp = cooks.distance(lev_fit)
n = 200 #number of observations

#The influential points' indices are:
influPoints = temp[temp > 4 /n]
influPoints
```

```
##          6         18         24         31         35         51
## 0.03559460 0.02208370 0.04714715 0.04217429 0.03089461 0.02312918
##         74         87        111        126        128        139
## 0.03465779 0.03253310 0.02052246 0.04094386 0.02308269 0.04012721
##        143        193
## 0.07935835 0.04637037
```

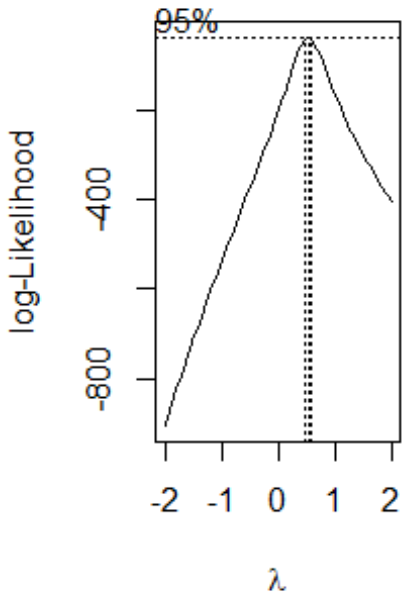
```r
#d)

# checking outliers
 #standardized residuals
rstandard(lev_fit)[temp>4/n] >2
```

```
##     6    18    24    31    35    51    74    87   111   126   128   139
## FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
##   143   193
##  TRUE  TRUE
```

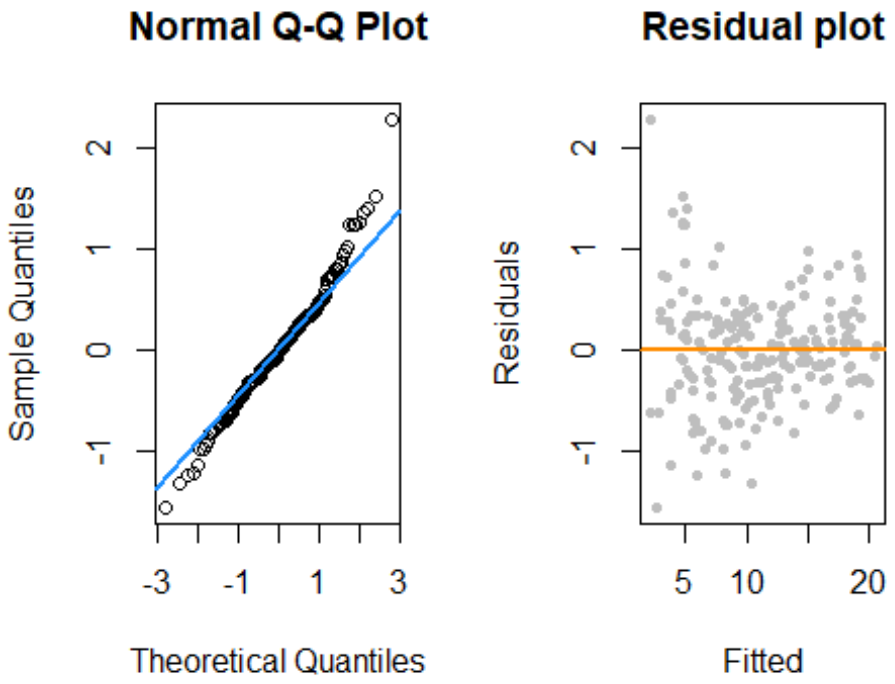
```r
#Outlier indices are 24,31,139,143,and 193

#e)
#Remove influential points found in c)
reducedData = hw3data
#Add a column that matches the cook's distance for each data point
reducedData$cooks = temp
#Remove the influential points
reducedData<-reducedData[!(reducedData$cooks>4/n),]

#Repeat b on the new data
model = lm(y ~ x1+x2, data = reducedData)
par(mfrow=c(1,2)) # Combining plots
qqnorm(resid(model))
qqline(resid(model), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(model), resid(model), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```



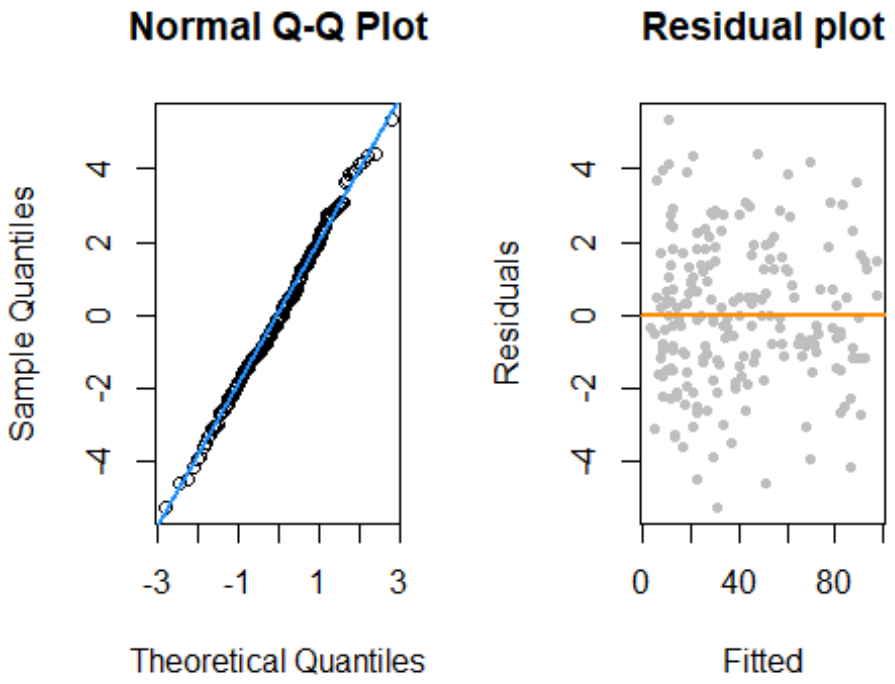

```r
#No, it was not useful, the previously violated assumptions were not fixed.

#f)
library(MASS)
model = lm(y ~ x1+x2, data = hw3data)
bc = boxcox(model)

#Find the appropriate lambda
lambda <- bc$x[which.max(bc$y)]
lambda

## [1] 0.5454545

#Transformation
lm_cox <- lm(((y^(lambda)-1)/(lambda)) ~ x1+x2, data = hw3data)

#Repeat b on the new data

par(mfrow=c(1,2)) # Combining plots
```



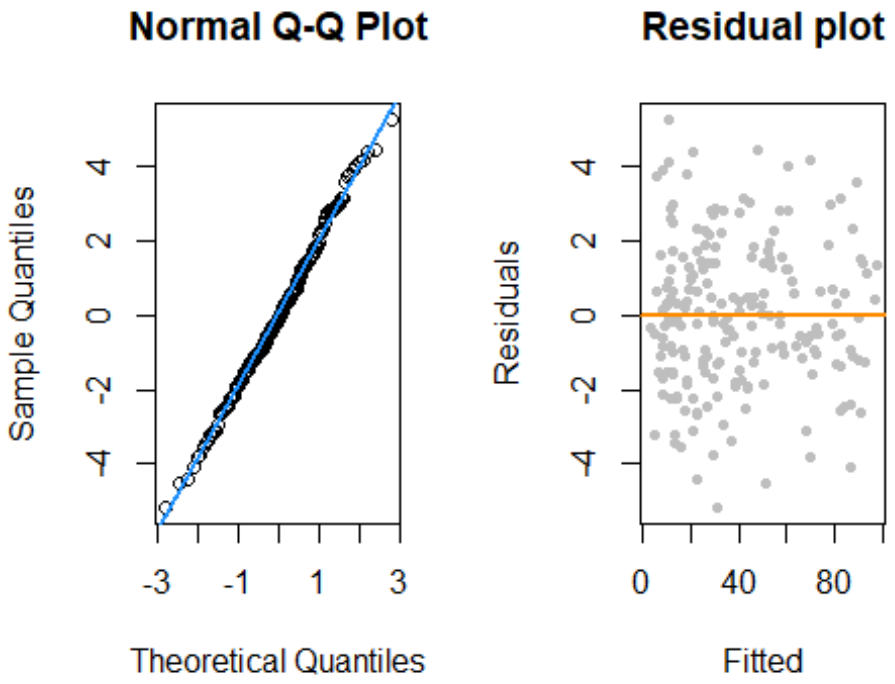

```r
qqnorm(resid(lm_cox))
qqline(resid(lm_cox), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(lm_cox), resid(lm_cox), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```




```
#Normality is stil vioated, however, it follows normal distribution a lot
better than before

#Linearity is still violated, however, the residual plot appears a lot more
random than before.

#Equal variance is violated, the spread of e does not appear to be constant
anymore


#g)

model1 = lm(y ~ x1+x2+I(x1^2)+I(x2^2), data = hw3data)
par(mfrow=c(1,2)) # Combining plots
qqnorm(resid(model1))
qqline(resid(model1), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(model1), resid(model1), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```




```
#This model is preferable to the previous resulting models in b and f.
#Because the model assumptions are met.
#Normality is not violated. The distribution tracks normal distribution very
closely.
#Linearity is not violated, residual plot shows mean of e does not
systematically
#Equal variance is not violated, because the spread of e does appear to be
overall constant

#h)

model2 = lm(y ~ x1+x2+I(x1^2)+I(x2^2)+I(x1^3)+I(x2^3), data = hw3data)
par(mfrow=c(1,2)) # Combining plots
qqnorm(resid(model2))
qqline(resid(model2), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(model2), resid(model2), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```




```
#Clearly, all three assumptions are met
#We can't simply compare model from h and g based on meeting assumptions
#Need to use model selection algorithms
#Choose AIC and BIC:
AIC(model1,model2) # AIC
```

```
##        df      AIC
## model1  6 850.8645
## model2  8 854.4720
```

```
BIC(model1,model2) # BIC
```

```
##        df      BIC
## model1  6 870.6544
## model2  8 880.8586
```

```
#The quadratic model has lower AIC and BIC scores than the cubic one. Hence,
the quadratic model from g is preferred to this cubic one.
```

Question 3

```
library(faraway)
```

```
## Warning: package 'faraway' was built under R version 3.5.3
```

```
#a)
model_a = lm(mpg ~ cyl+disp+hp+wt+drat, data = mtcars)
vif(model_a)
```

```
##       cyl      disp        hp        wt      drat
##  7.869010 10.463957  3.990380  5.168795  2.662298

#The vif values clearly reflected collinearity as most of the VIF values are
> 1, implying high Rj^2 values.

#In particular, disp has a VIF of 10.463957.

#Yes, collinearity exists. Collinearity affects regression analysis because a
high VIF on regresssion coefficients, which implies high variance estimates,
which implies high standard error of the particular parameter estimate.

#b)
#Local function for vif
myVIF <- function(model){
  r_squared = summary(model)$r.squared
  return(1/(1-r_squared))
}

cylLM = lm(cyl ~ hp+wt+drat, data = mtcars)
cyl_VIF = myVIF(cylLM)
cyl_VIF

## [1] 6.17356

hpLM = lm(hp~ cyl+wt+drat, data = mtcars)
hp_VIF = myVIF(hpLM)
hp_VIF

## [1] 3.78467

wtLM = lm(wt~cyl + hp+drat, data = mtcars)
wt_VIF = myVIF(wtLM)
wt_VIF

## [1] 3.076225

dratLM = lm(drat ~ cyl+ hp+wt, data = mtcars)
drat_VIF = myVIF(dratLM)
drat_VIF

## [1] 2.639229

#Collinearity stil exists, however,the model has improved, since there are no
more values over 10.

#c)
fit_null=lm(mpg~1,data=mtcars)
fit_step_aic = step(fit_null,
                    mpg~cyl+disp+hp+wt+drat,
                    direction = "forward")
```

```
## Start:  AIC=115.94
## mpg ~ 1
##
##        Df Sum of Sq     RSS     AIC
## + wt    1    847.73  278.32  73.217
## + cyl   1    817.71  308.33  76.494
## + disp  1    808.89  317.16  77.397
## + hp    1    678.37  447.67  88.427
## + drat  1    522.48  603.57  97.988
## <none>              1126.05 115.943
##
## Step:  AIC=73.22
## mpg ~ wt
##
##        Df Sum of Sq    RSS    AIC
## + cyl   1    87.150 191.17 63.198
## + hp    1    83.274 195.05 63.840
## + disp  1    31.639 246.68 71.356
## <none>              278.32 73.217
## + drat  1     9.081 269.24 74.156
##
## Step:  AIC=63.2
## mpg ~ wt + cyl
##
##        Df Sum of Sq    RSS    AIC
## + hp    1   14.5514 176.62 62.665
## <none>              191.17 63.198
## + disp  1    2.6796 188.49 64.746
## + drat  1    0.0010 191.17 65.198
##
## Step:  AIC=62.66
## mpg ~ wt + cyl + hp
##
##        Df Sum of Sq    RSS    AIC
## <none>              176.62 62.665
## + disp  1    6.1762 170.44 63.526
## + drat  1    2.2453 174.38 64.255
```

```
# Resulting model
fit_step_aic
```

```
##
## Call:
## lm(formula = mpg ~ wt + cyl + hp, data = mtcars)
##
## Coefficients:
## (Intercept)           wt          cyl           hp
##    38.75179     -3.16697     -0.94162     -0.01804
```

```r
#d)
n = nrow(mtcars)
fit_step_bic = step(model_a, direction = "backward",k = log(n))
```

```
## Start:  AIC=73.75
## mpg ~ cyl + disp + hp + wt + drat
##
##        Df Sum of Sq    RSS    AIC
## - drat  1     3.018 170.44 70.854
## - disp  1     6.949 174.38 71.584
## - cyl   1    15.411 182.84 73.100
## <none>              167.43 73.748
## - hp    1    21.066 188.49 74.075
## - wt    1    77.476 244.90 82.453
##
## Step:  AIC=70.85
## mpg ~ cyl + disp + hp + wt
##
##        Df Sum of Sq    RSS    AIC
## - disp  1     6.176 176.62 68.528
## - hp    1    18.048 188.49 70.609
## <none>              170.44 70.854
## - cyl   1    24.546 194.99 71.694
## - wt    1    90.925 261.37 81.069
##
## Step:  AIC=68.53
## mpg ~ cyl + hp + wt
##
##        Df Sum of Sq    RSS    AIC
## - hp    1    14.551 191.17 67.595
## - cyl   1    18.427 195.05 68.237
## <none>              176.62 68.528
## - wt    1   115.354 291.98 81.147
##
## Step:  AIC=67.6
## mpg ~ cyl + wt
##
##        Df Sum of Sq    RSS    AIC
## <none>              191.17 67.595
## - cyl   1     87.15 278.32 76.149
## - wt    1    117.16 308.33 79.426
```

```r
# Resulting model
fit_step_bic
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt, data = mtcars)
##
## Coefficients:
```

```
## (Intercept)          cyl           wt
##      39.686       -1.508       -3.191
```

```
anova(fit_step_bic, fit_step_aic)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ cyl + wt
## Model 2: mpg ~ wt + cyl + hp
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     29 191.17
## 2     28 176.62  1    14.551 2.3069   0.14
```

```
#Not significant because 0.14 < alpha, no evidence against null hypothesis.
The two models are not significantly different from each other.
```

Question 4

```
modelA = lm(lpsa~lcavol+lweight+svi,data= prostate)
modelB = lm(lpsa~lcavol+lweight+svi+lbph,data = prostate)
modelC = lm(lpsa~lcavol+lweight+svi+lbph+lcp+gleason,data = prostate)

#a)
AIC(modelA,modelB,modelC)
```

```
##        df      AIC
## modelA  5 216.5979
## modelB  6 215.9223
## modelC  8 218.9735
```

```
#Best Model: modelB

BIC(modelA,modelB,modelC)
```

```
##        df      BIC
## modelA  5 229.4714
## modelB  6 231.3705
## modelC  8 239.5712
```

```
#Best Model: modelA

#Adj.R squared:
#Model A:
summary(modelA)$adj.r.squared
```

```
## [1] 0.6143899
```

```
#Model B:
summary(modelB)$adj.r.squared
```

```
## [1] 0.6208036
```

```r
#Model C:
summary(modelC)$adj.r.squared
```

```
## [1] 0.6161501
```

```r
#Best model: modelB

#b)
sqrt(sum((resid(modelA)/(1-hatvalues(modelA)))^2)/n)
```

```
## [1] 1.285099
```

```r
sqrt(sum((resid(modelB)/(1-hatvalues(modelB)))^2)/n)
```

```
## [1] 1.280599
```

```r
sqrt(sum((resid(modelC)/(1-hatvalues(modelC)))^2)/n)
```

```
## [1] 1.298576
```

```r
#Best model: modelB

#c)
#R squared:
#Model A:
summary(modelA)$r.squared
```

```
## [1] 0.6264403
```

```r
#Model B:
summary(modelB)$r.squared
```

```
## [1] 0.6366035
```

```r
#Model C:
summary(modelC)$r.squared
```

```
## [1] 0.6401407
```

```r
#Best model: modelC

#R^2 is not an appropriate measure for model comparison because it will
always pick the most complex model. This could lead to overfitting, resulting
in incorrect models and poor out of sample predictability.
```