Homework #12

asm366

Things to complete:

a. All examples completed by the TAs
b. All of the homework problems assigned to you

Refer to **Homework & Discussion Submission Rules** on Canvas.

Refer to Gradescope for the Due Date and Due time.

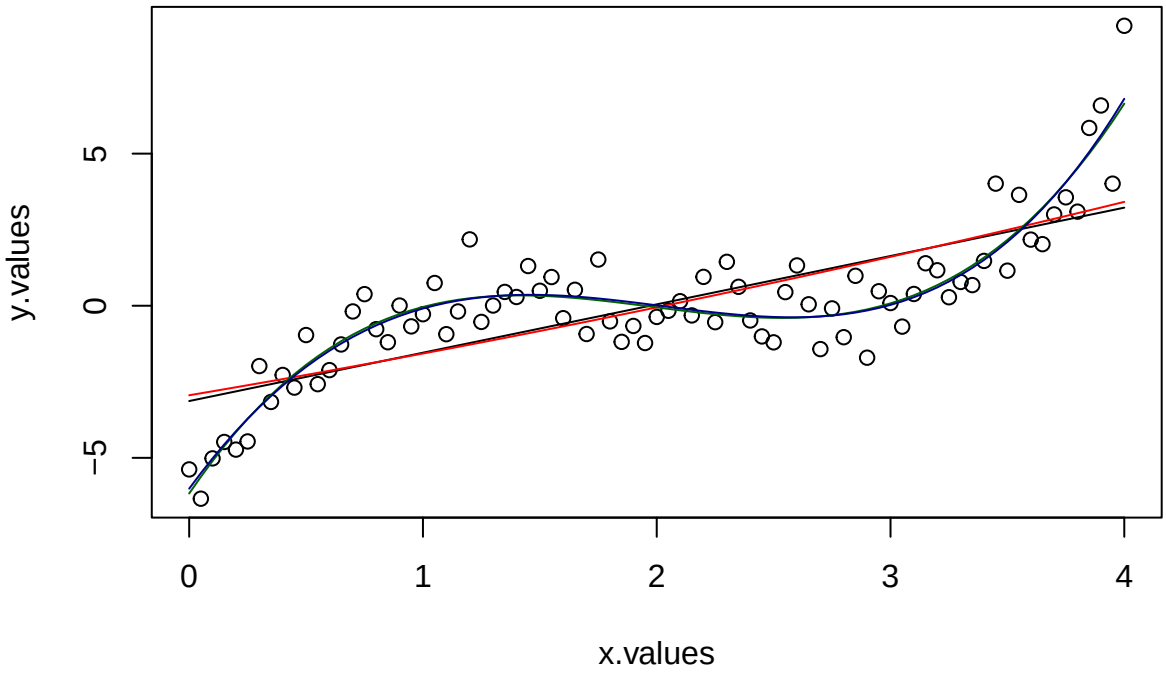
If you needed to download a *.RDATA file with this lab, make sure it was downloaded into the same folder as this file.

Additionally, run the first code chunk in this file before you start doing any work. By running the first code chunk, some variables will be loaded for you. You should use these variables in your work, when necessary.

Do not assign/select the cover page in GRADESCOPE.

Example 1:

```
# SHOW YOUR WORK HERE.
x.values <- seq(from = 0, to = 4, by = 0.05)
ln <- length(x.values)
errors <- rnorm(n = ln, sd = 1)

y.values <- (x.values - 1) * (x.values - 2) * (x.values - 3) + errors

simulated.dat <- data.frame(x.values, y.values)

plot(simulated.dat)

lm.fit <- lm(y.values ~ x.values, dat = simulated.dat)

quadratic.fit <- lm(y.values ~ x.values + I(x.values^2), dat = simulated.dat)

cubic.fit <- lm(y.values ~ x.values + I(x.values^2) + I(x.values^3), dat = simulated.dat)

quatric.fit <- lm(y.values ~ x.values + I(x.values^2) + I(x.values^3) +
    I(x.values^4), dat = simulated.dat)

points(x.values, predict(lm.fit), col = "black", type = "l")
points(x.values, predict(quadratic.fit), col = "red", type = "l")
points(x.values, predict(cubic.fit), col = "darkgreen", type = "l")
points(x.values, predict(quatric.fit), col = "darkblue", type = "l")
```

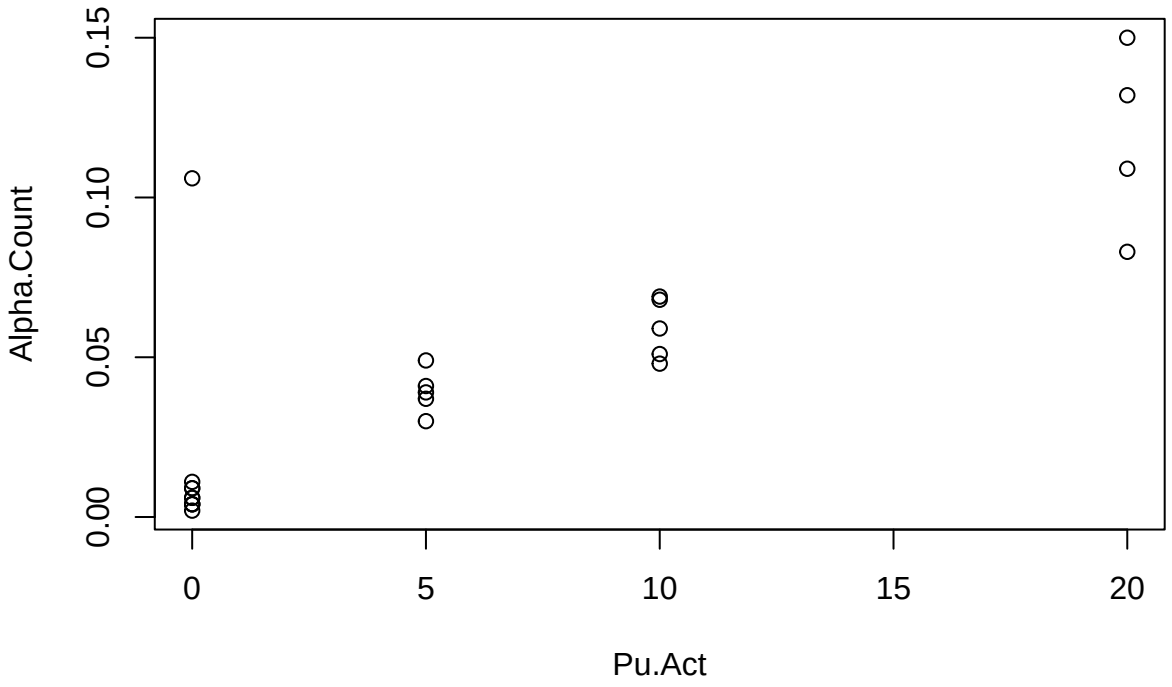



```
summary(quatric.fit)
##
## Call:
## lm(formula = y.values ~ x.values + I(x.values^2) + I(x.values^3) +
##     I(x.values^4), data = simulated.dat)
##
```

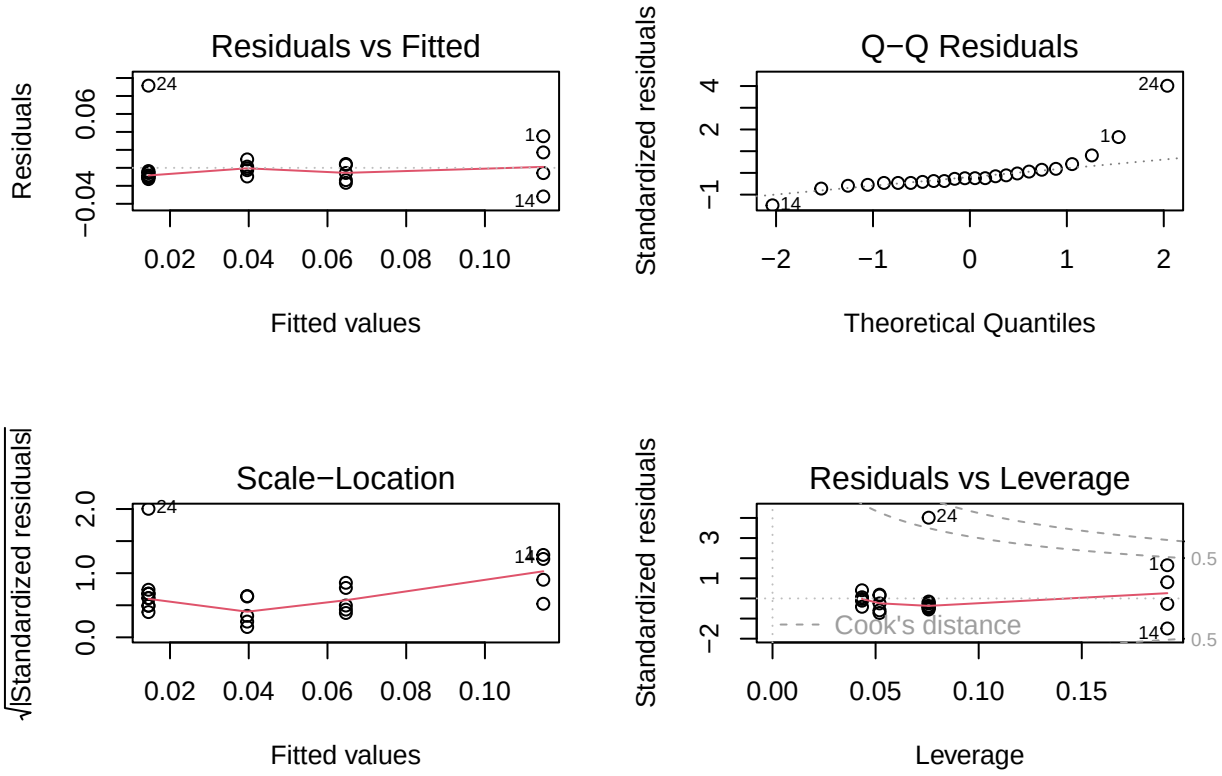
```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1705 -0.7139 -0.1297  0.6322  2.4007
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -6.01307    0.48857 -12.308  < 2e-16 ***
## x.values      10.45885    1.71478   6.099 4.13e-08 ***
## I(x.values^2) -5.25804    1.75905  -2.989  0.00377 **
## I(x.values^3)  0.67418    0.66271   1.017  0.31223
## I(x.values^4)  0.04672    0.08217   0.569  0.57136
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9453 on 76 degrees of freedom
## Multiple R-squared:  0.8609, Adjusted R-squared:  0.8536
## F-statistic: 117.6 on 4 and 76 DF,  p-value: < 2.2e-16

# FINISHED? KNIT YOUR WORK BEFORE GOING TO THE NEXT QUESTION.

## REVIEW Homework & Discussion Submission Rules ON CANVAS.

### THIS SHOULD BE SUBMITTED BEFORE 5:00PM ON THE DUE DATE LISTED ON
### GRADESCOPE.

#### EXTENSIONS ARE NOT GIVEN BECAUSE YOUR DOCUMENT WILL NOT KNIT.
```

Example 2:

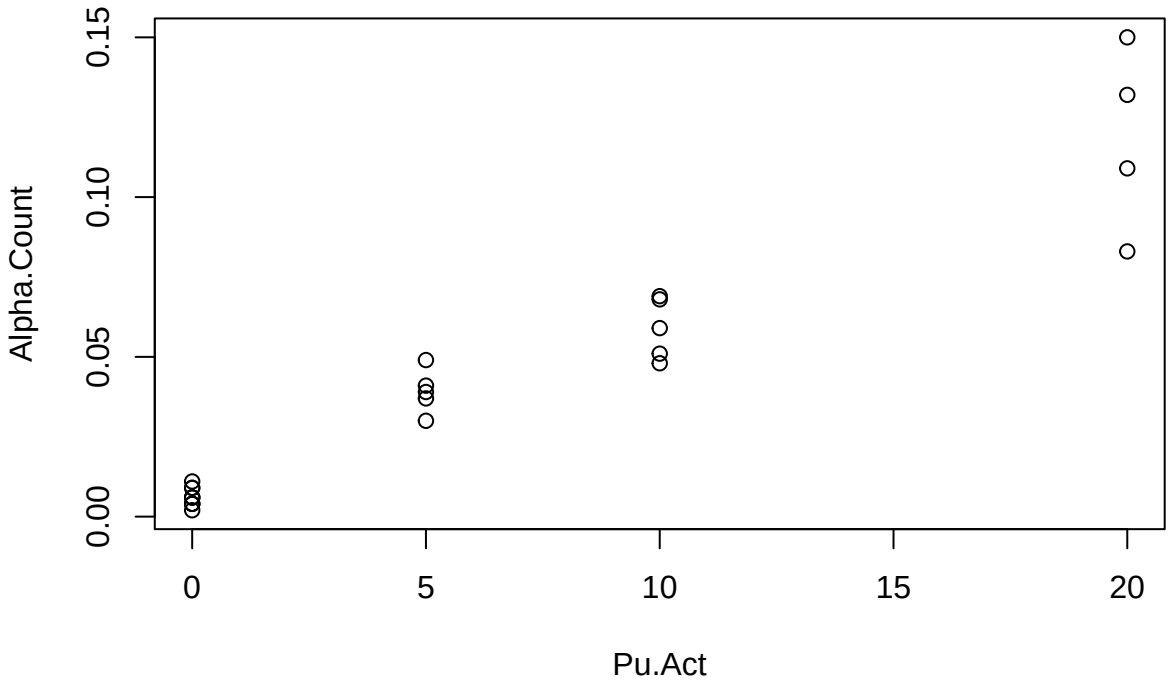
```
# SHOW YOUR WORK HERE.

pluto <- read.csv("Plutonium.csv")
pluto.fit <- lm(Alpha.Count ~ Pu.Act, dat = pluto)
plot(pluto[2:1])
```



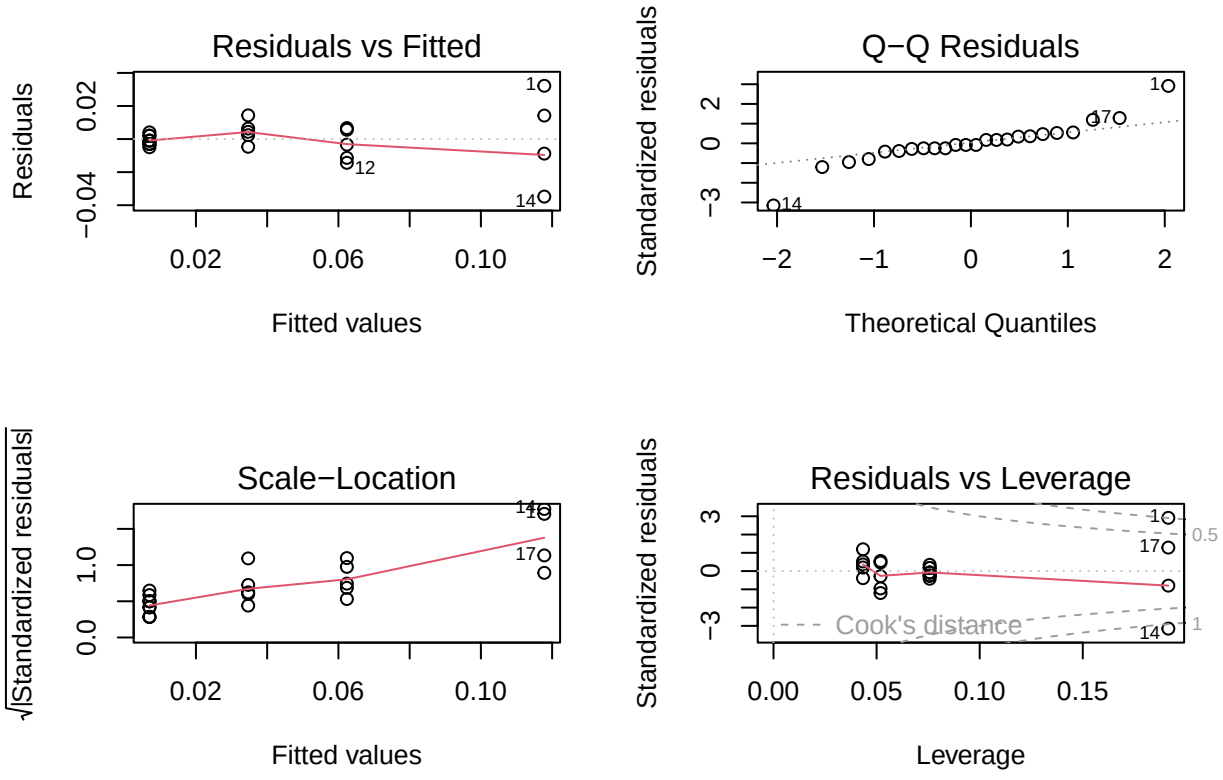

```
par(mfrow = c(2, 2))
plot(pluto.fit)
```




```
pluto[24, 1]
## [1] 0.106


pluto[24, 1] <- 0.006
pluto.fit <- lm(Alpha.Count ~ Pu.Act, dat = pluto)
par(mfrow = c(1, 1))
plot(pluto[2:1])
```

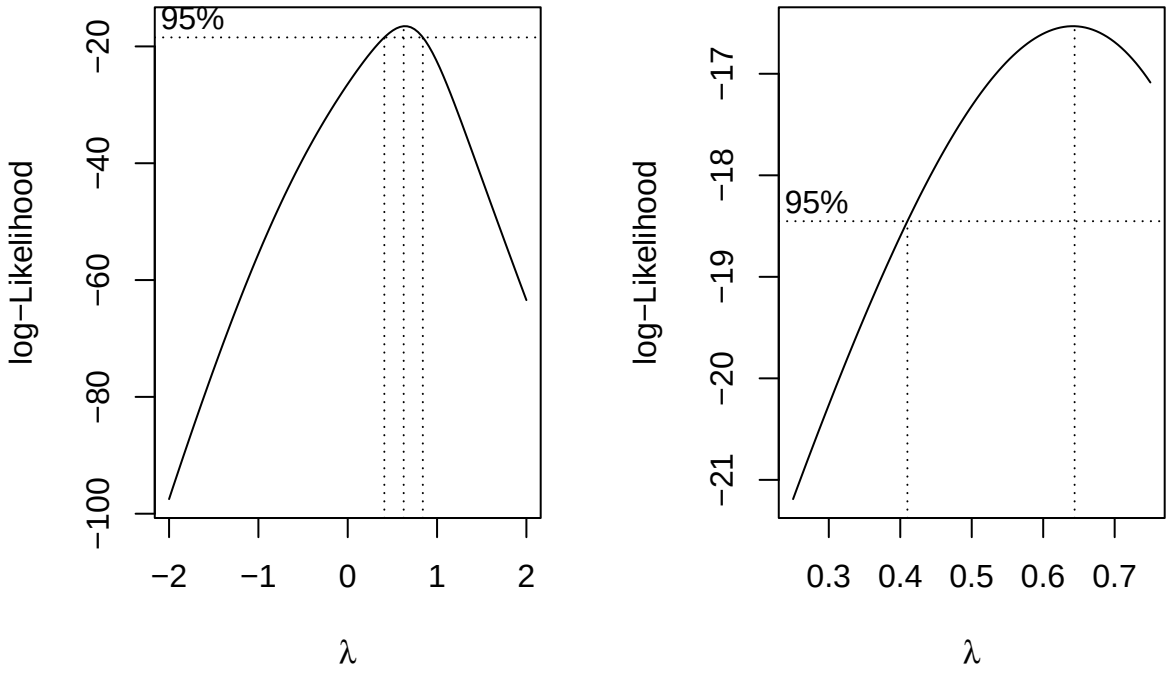



```
par(mfrow = c(2, 2))
plot(pluto.fit)
```

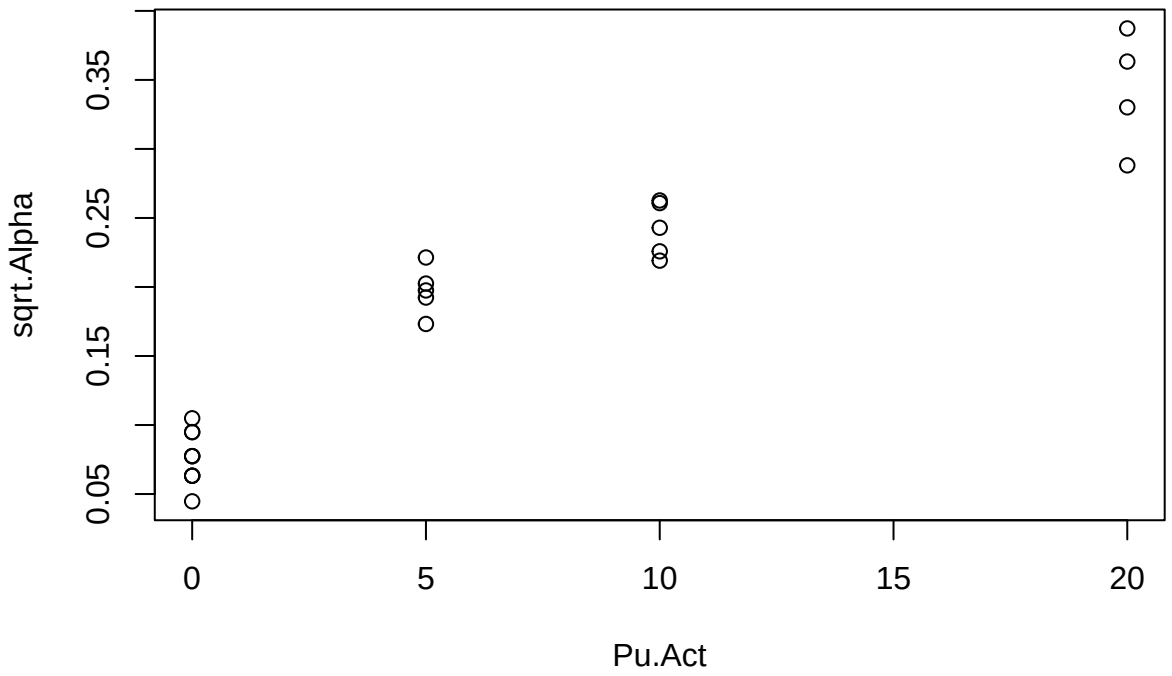



```
library(MASS)

par(mfrow = c(1, 2))
boxcox(pluto.fit, lambda = seq(-2, 2, 1/10))
```

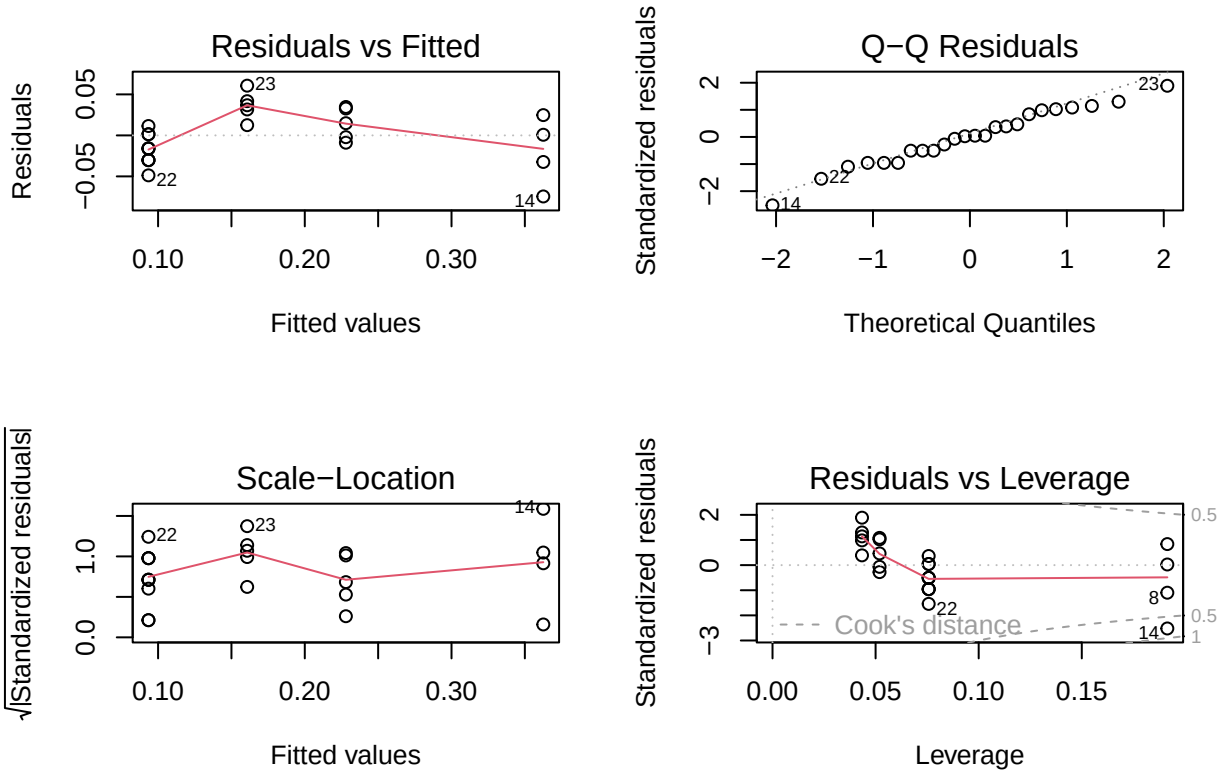
```
boxcox(pluto.fit, lambda = seq(0.25, 0.75, 1/20))
```




```
pluto$sqrt.Alpha <- sqrt(pluto$Alpha.Count)
pluto.fit.sqrt <- lm(sqrt.Alpha ~ Pu.Act, data = pluto)
par(mfrow = c(1, 1))
plot(pluto[2:3])
```




```
par(mfrow = c(2, 2))
plot(pluto.fit.sqrt)
```

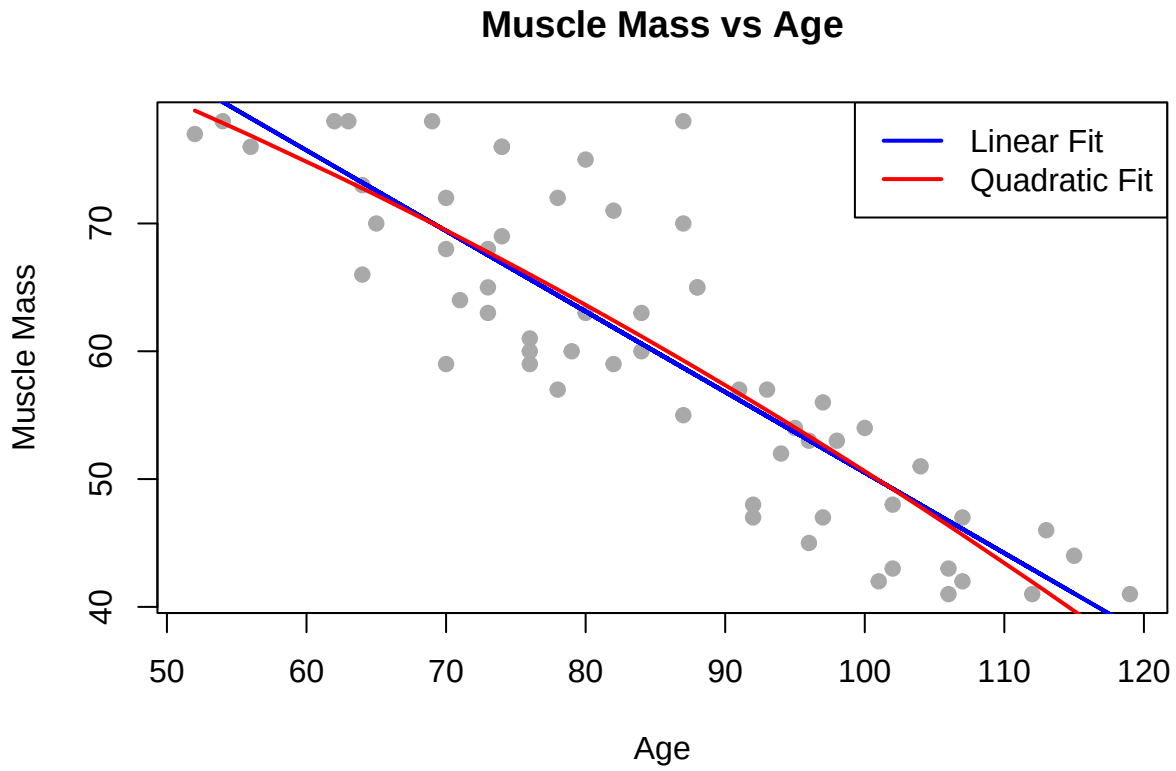



```
pluto$sqrt.Pu <- sqrt(pluto$Pu.Act)
pluto.fit.both <- lm(sqrt.Alpha ~ sqrt.Pu, data = pluto)
par(mfrow = c(1, 1))
pluto.fit.both
##
## Call:
## lm(formula = sqrt.Alpha ~ sqrt.Pu, data = pluto)
##
## Coefficients:
## (Intercept)      sqrt.Pu
##     0.07341      0.05719

summary(pluto.fit.both)
##
## Call:
## lm(formula = sqrt.Alpha ~ sqrt.Pu, data = pluto)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.041067 -0.010463  0.001092  0.011335  0.058134
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.073410   0.007300   10.06 1.09e-09 ***
## sqrt.Pu     0.057188   0.002872   19.91 1.47e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02422 on 22 degrees of freedom
```

```
## Multiple R-squared:  0.9474, Adjusted R-squared:  0.945
## F-statistic: 396.4 on 1 and 22 DF,  p-value: 1.465e-15



# FINISHED? KNIT YOUR WORK BEFORE GOING TO THE NEXT QUESTION.

## REVIEW Homework & Discussion Submission Rules ON CANVAS.

### THIS SHOULD BE SUBMITTED BEFORE 5:00PM ON THE DUE DATE LISTED ON
### GRADESCOPE.

#### EXTENSIONS ARE NOT GIVEN BECAUSE YOUR DOCUMENT WILL NOT KNIT.
```

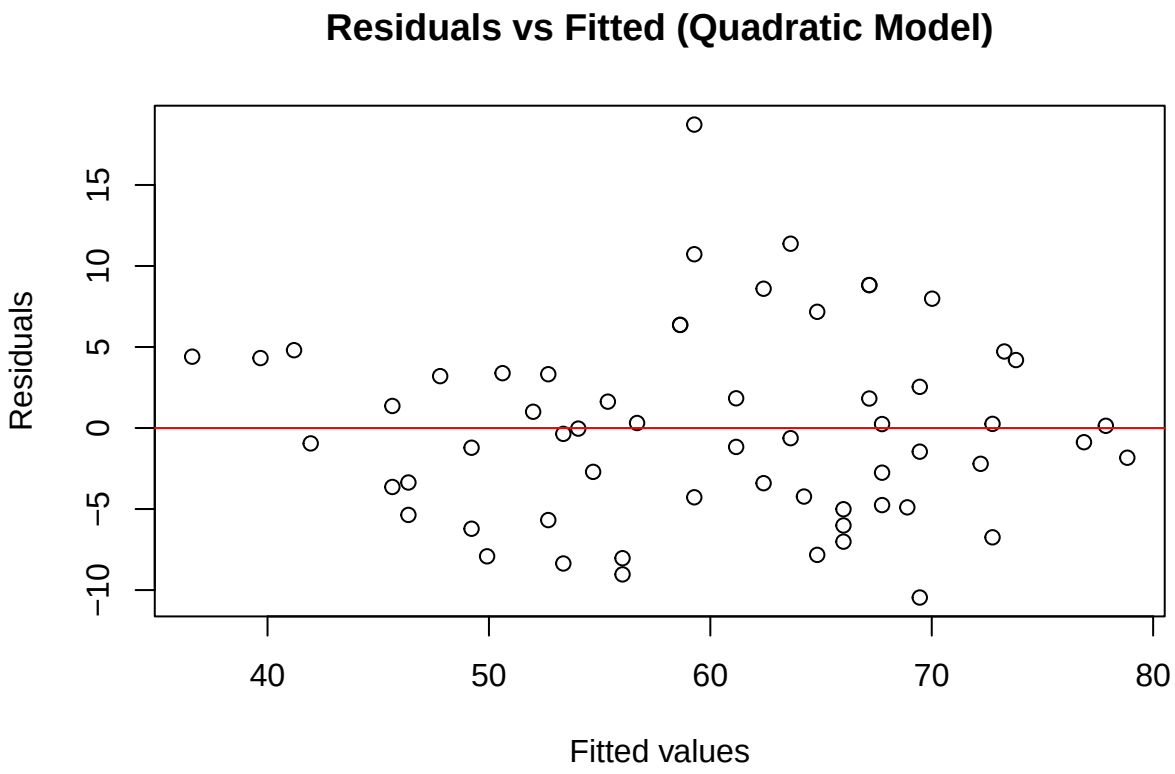
Problem 3:

```r
data <- read.csv("muscleMass.csv")
model1 <- lm(mass ~ age, data = data)

model2 <- lm(mass ~ age + I(age^2), data = data)

plot(data$age, data$mass, main = "Muscle Mass vs Age", xlab = "Age", ylab = "Muscle Mass",
    pch = 19, col = "darkgray")

lines(data$age, fitted(model1), col = "blue", lwd = 2)
lines(sort(data$age), fitted(model2)[order(data$age)], col = "red", lwd = 2)
legend("topright", legend = c("Linear Fit", "Quadratic Fit"), col = c("blue",
    "red"), lwd = 2)
```

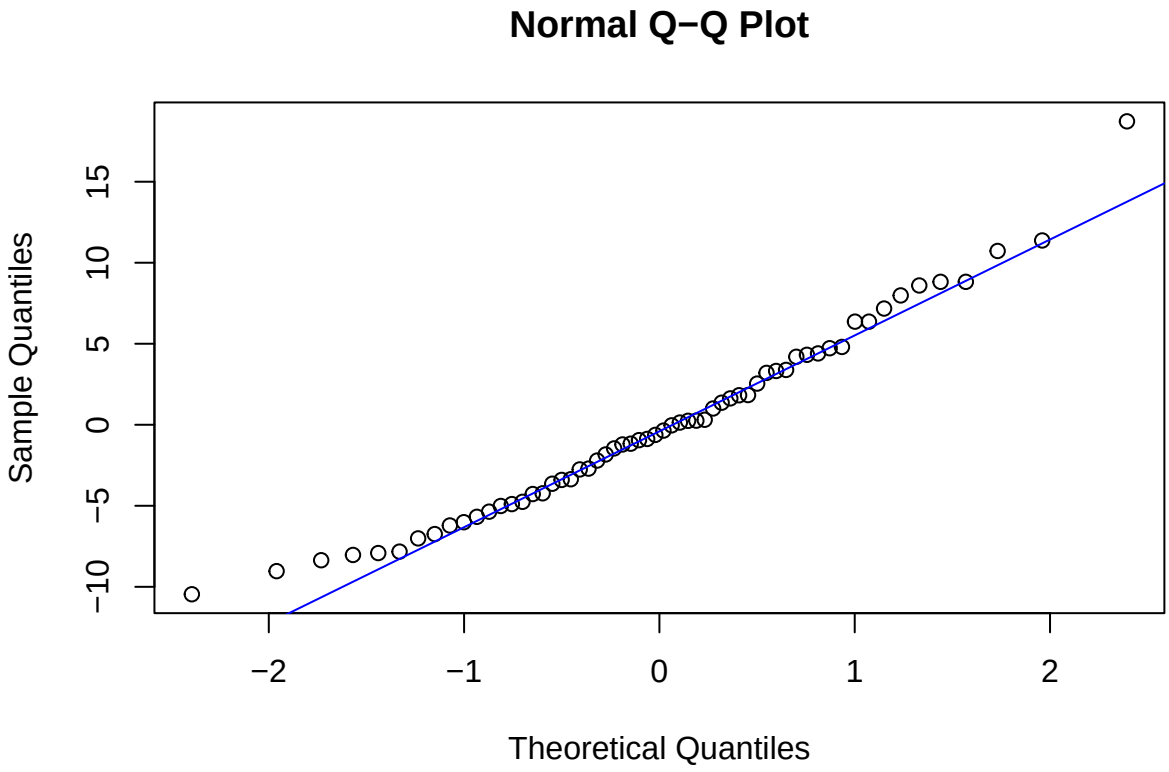



```r
summary(model2)
##
## Call:
## lm(formula = mass ~ age + I(age^2), data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4593  -4.3954  -0.4927   3.5878  18.7248
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 97.768763  19.364306   5.049 4.86e-06 ***
## age         -0.247811   0.461302  -0.537    0.593
## I(age^2)    -0.002237   0.002684  -0.834    0.408
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.964 on 57 degrees of freedom
## Multiple R-squared:  0.7531, Adjusted R-squared:  0.7444
## F-statistic: 86.92 on 2 and 57 DF,  p-value: < 2.2e-16

anova(model1, model2)
## Analysis of Variance Table
##
## Model 1: mass ~ age
## Model 2: mass ~ age + I(age^2)
##   Res.Df    RSS Df Sum of Sq     F Pr(>F)
## 1     58 2052.2
## 2     57 2027.5  1    24.722 0.695 0.4079

plot(model2$fitted.values, model2$residuals, xlab = "Fitted values", ylab = "Residuals",
    main = "Residuals vs Fitted (Quadratic Model)")
abline(h = 0, col = "red")
```




```
qqnorm(model2$residuals)
qqline(model2$residuals, col = "blue")
```




```
new_data <- data.frame(age = 48)

predict(model1, newdata = new_data, interval = "confidence", level = 0.99)
##        fit     lwr      upr
## 1 83.28382 78.1548 88.41283

predict(model2, newdata = new_data, interval = "confidence", level = 0.99)
##        fit      lwr      upr
## 1 80.71912 71.03998 90.39827
# FINISHED? KNIT YOUR WORK BEFORE GOING TO THE NEXT QUESTION.

## REVIEW Homework & Discussion Submission Rules ON CANVAS.

### THIS SHOULD BE SUBMITTED BEFORE 5:00PM ON THE DUE DATE LISTED ON
### GRADESCOPE.

#### EXTENSIONS ARE NOT GIVEN BECAUSE YOUR DOCUMENT WILL NOT KNIT.
```

Null Hypothesis: B1 = B2 = 0 (the terms are not significant, so the relationship is not isgnifcant) Alternate Hypothesis B1 != 0 or B2 != 0 ((at least one of) the terms are significant, so the relationship is signifcant) F-statistic p-value: 2.2e-16, because p < alpha at the 0.05 level, there is an significant regression relationship using the quadratic model.

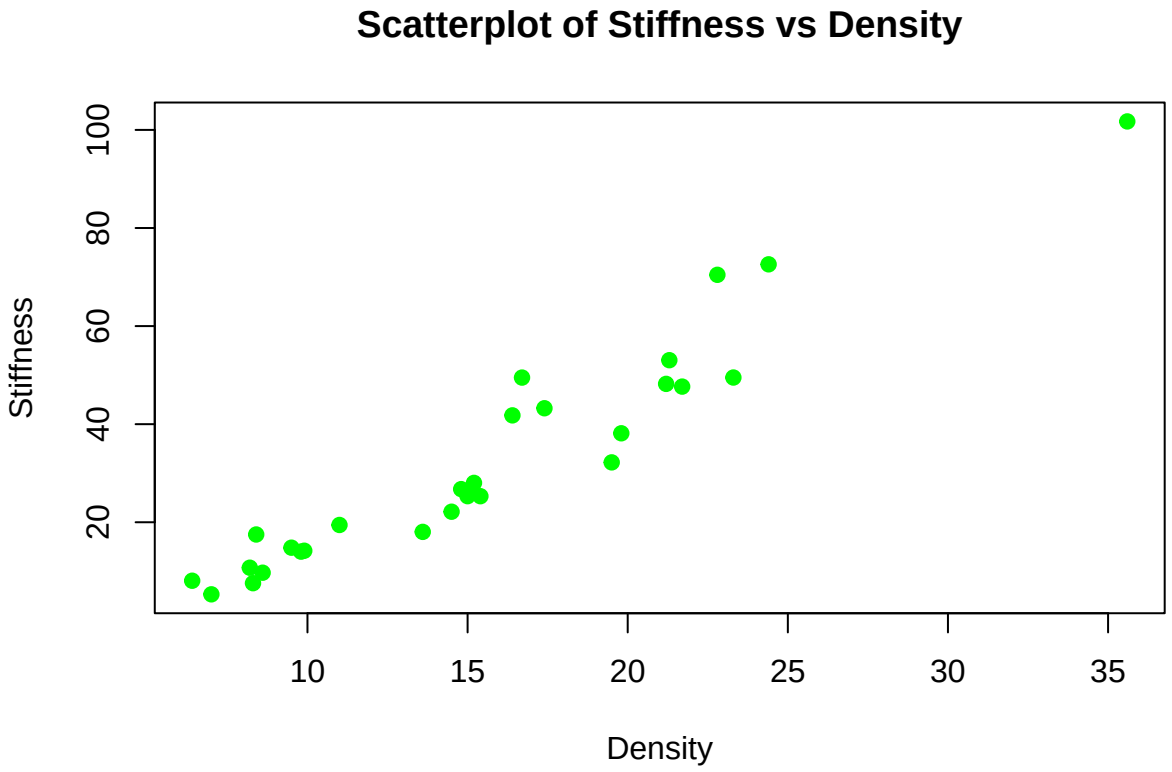
Null Hypothesis: B2 = 0 (the quadratic term is not significant) Alternate Hypothesis B2 != 0 (the quadratic term is significant) F-statistic p-value: 0.4079, because p > alpha at the 0.05 level, there is an insignificant difference between the two so the quadratic term is unnecessary.

Based on the residual vs. fitted values, I see a general random shape in the plot, satisfying the condition of linearity. The variances seem to be quite evenly spread out too, satisfying the condition of homoskedacsity. Finally, the plot of the residuals follows the line on the Q-Q plot really well, so the residuals should be

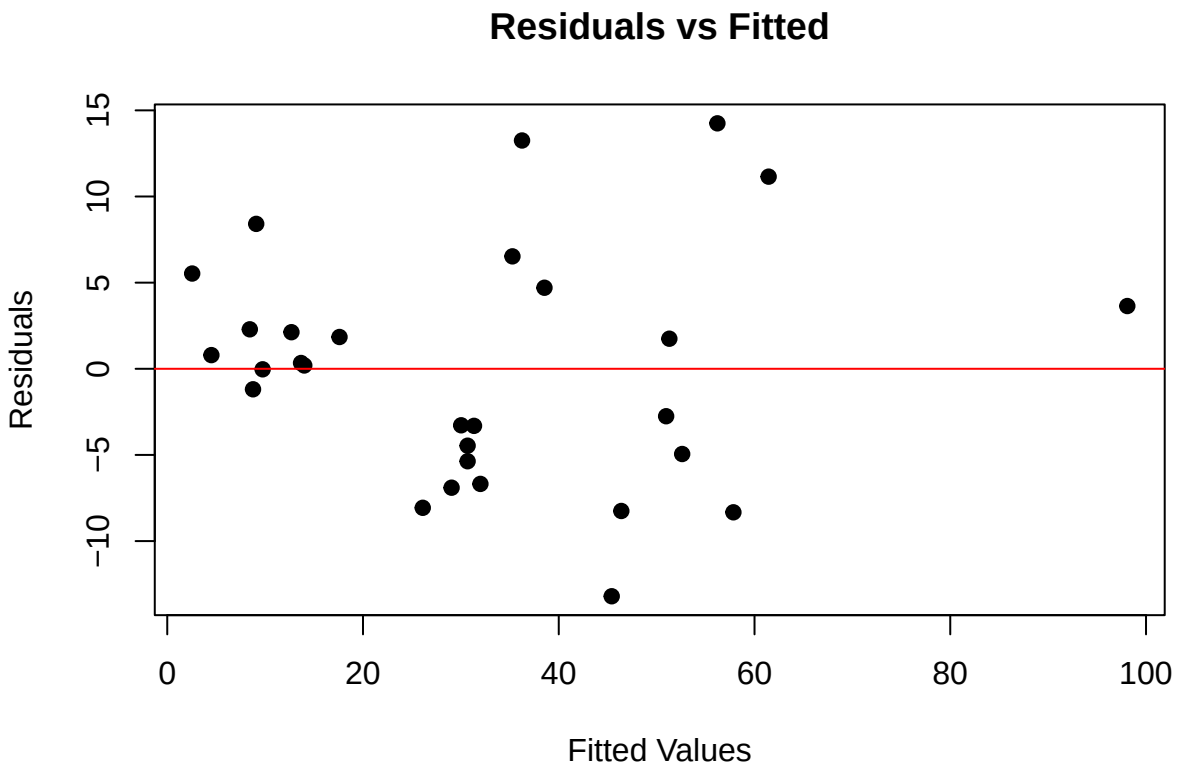
normally distributed.

Problem 4:

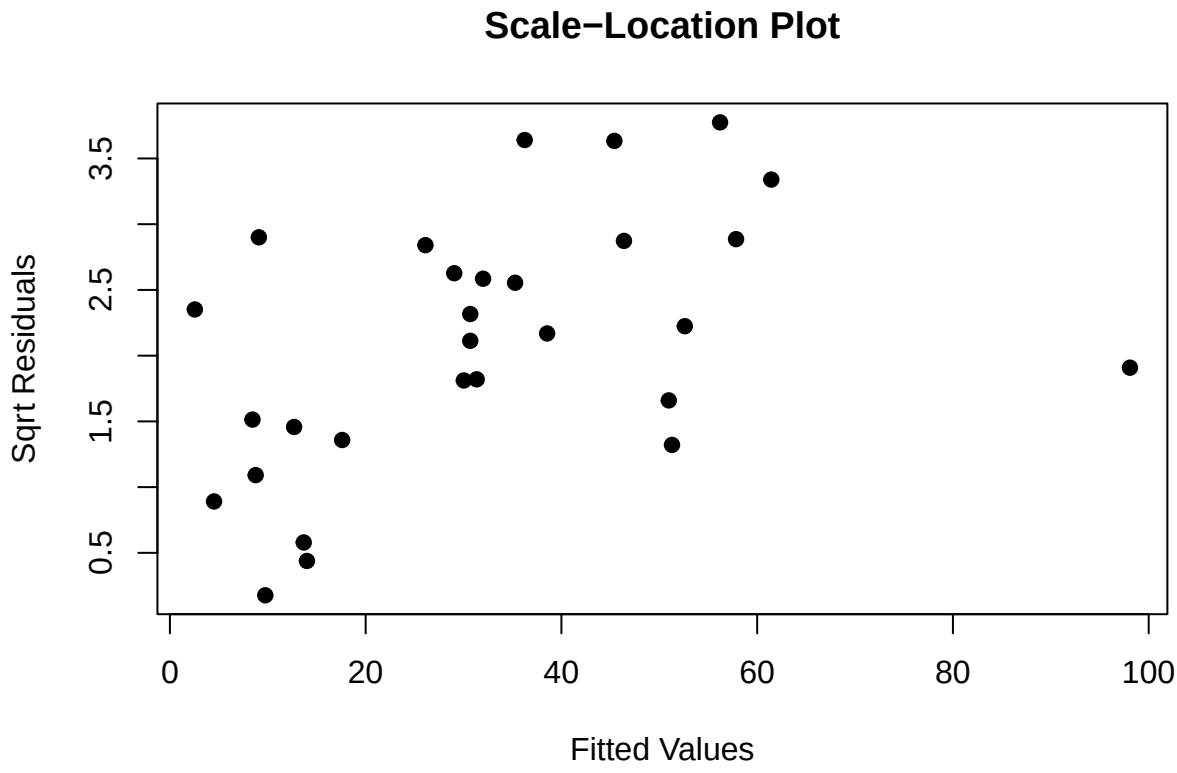
```
data <- read.csv("ParticleBoard.csv")
head(data)
##   Density Stiffness     Temp
## 1     9.5    14.814 70.61056
## 2     8.4    17.502 72.30000
## 3     9.8    14.007 66.15377
## 4    11.0    19.443 70.05781
## 5     8.3     7.573 69.33919
## 6     9.9    14.191 69.12882

plot(data$Density, data$Stiffness, main = "Scatterplot of Stiffness vs Density",
    xlab = "Density", ylab = "Stiffness", pch = 19, col = "green")
```



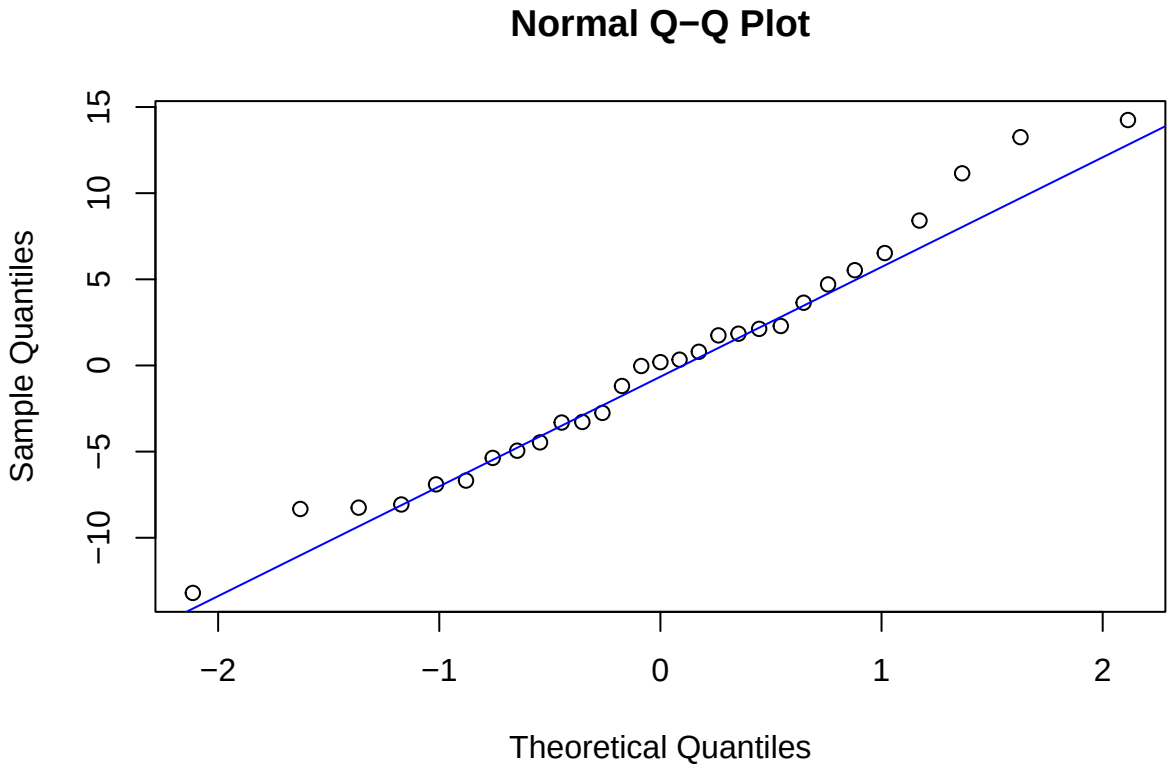

```
model <- lm(Stiffness ~ Density, data = data)
plot(model$fitted.values, model$residuals, main = "Residuals vs Fitted",
    xlab = "Fitted Values", ylab = "Residuals", pch = 19)
abline(h = 0, col = "red")
```




```
sqrt_abs_resid <- sqrt(abs(model$residuals))
plot(model$fitted.values, sqrt_abs_resid, main = "Scale-Location Plot",
    xlab = "Fitted Values", ylab = "Sqrt Residuals", pch = 19)
```



```r
qqnorm(model$residuals)
qqline(model$residuals, col = "blue")
```




```r
data$sqrt_Stiffness <- sqrt(data$Stiffness)

powers <- seq
```

```r
# FINISHED? KNIT YOUR WORK BEFORE GOING TO THE NEXT QUESTION.

## REVIEW Homework & Discussion Submission Rules ON CANVAS.

### THIS SHOULD BE SUBMITTED BEFORE 5:00PM ON THE DUE DATE LISTED ON
### GRADESCOPE.

#### EXTENSIONS ARE NOT GIVEN BECAUSE YOUR DOCUMENT WILL NOT KNIT.
```

They appear to have a positive linear relationship.

The residuals appear to have a pattern however. They start near zero then have a sort of downward dip and then go back up again. This means we might need a quadratic or non linear term.

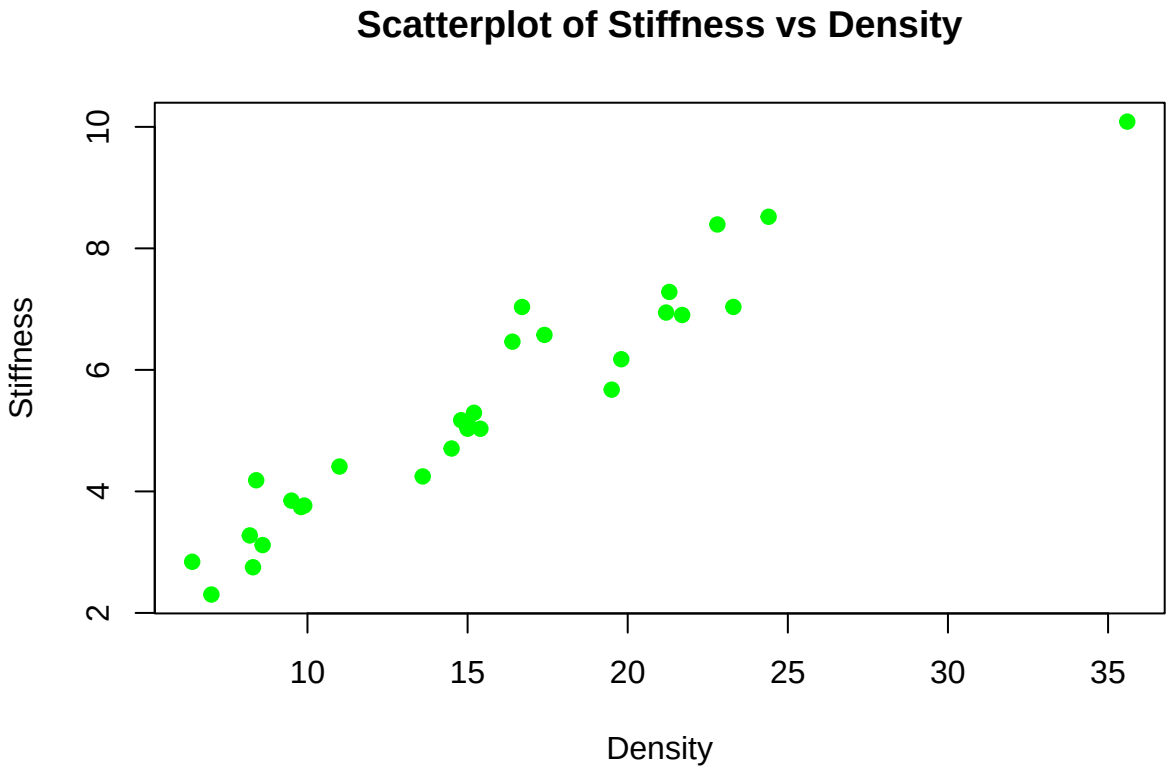
In the scale-location plot there seems to be a sort of pattern, kind of curved maybe square root curved. This is interesting and liekly means the data violates the principles of heteroskedacdisity.

The dots generally stick to the line but its definitely not a perfect fit to the normal distribution.

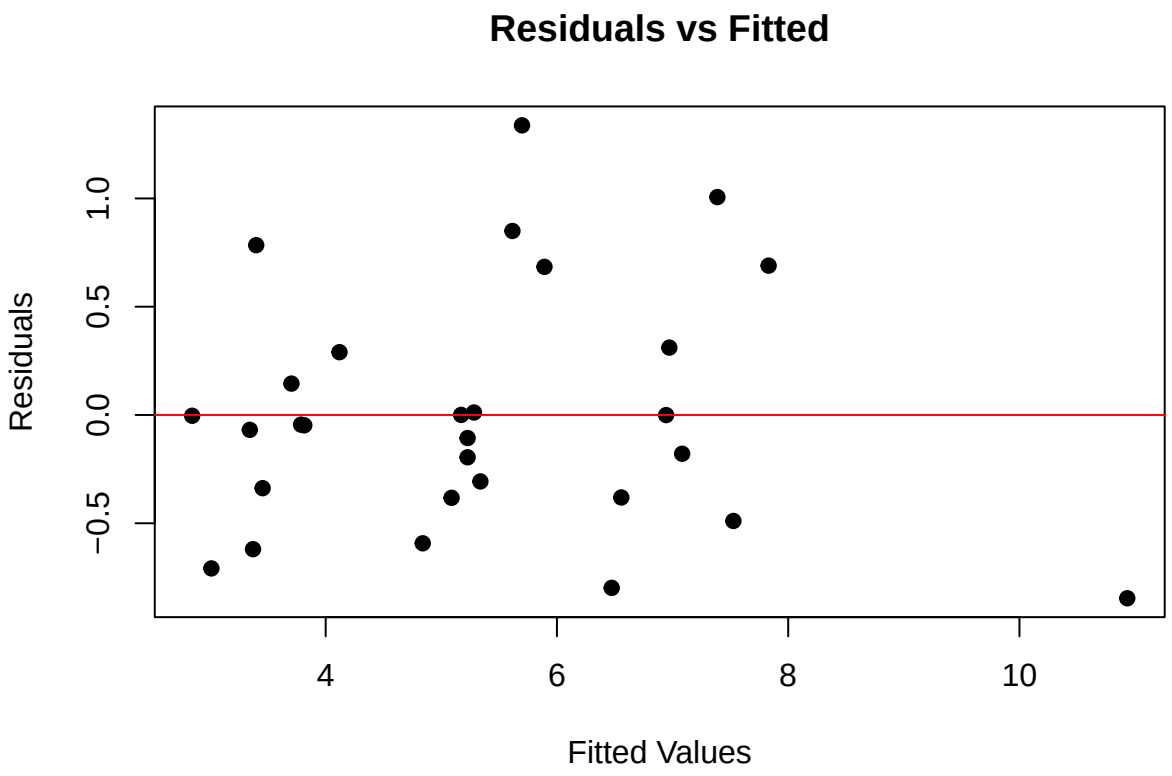
I think I would do Stiffness to the square root and then density based on the graphs. The residuals sort of fan out at higher values and Stiffness looks slightly right skewed.

I would transform Stiffness for sure because as we see in the residual plot there seems to be some sort of non-constant variance and transforming the response (Stiffness) and help sovle this.

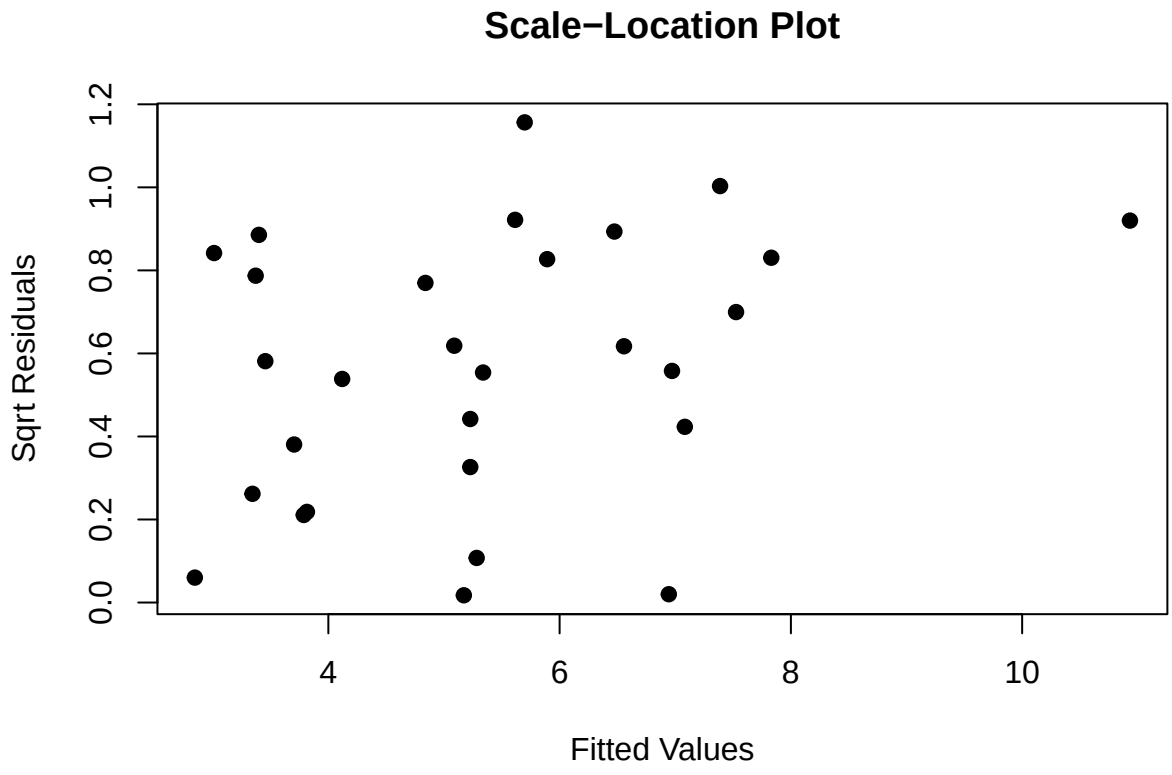
```r
data$Stiffness = sqrt(data$Stiffness)
plot(data$Density, data$Stiffness, main = "Scatterplot of Stiffness vs Density",
    xlab = "Density", ylab = "Stiffness", pch = 19, col = "green")
```



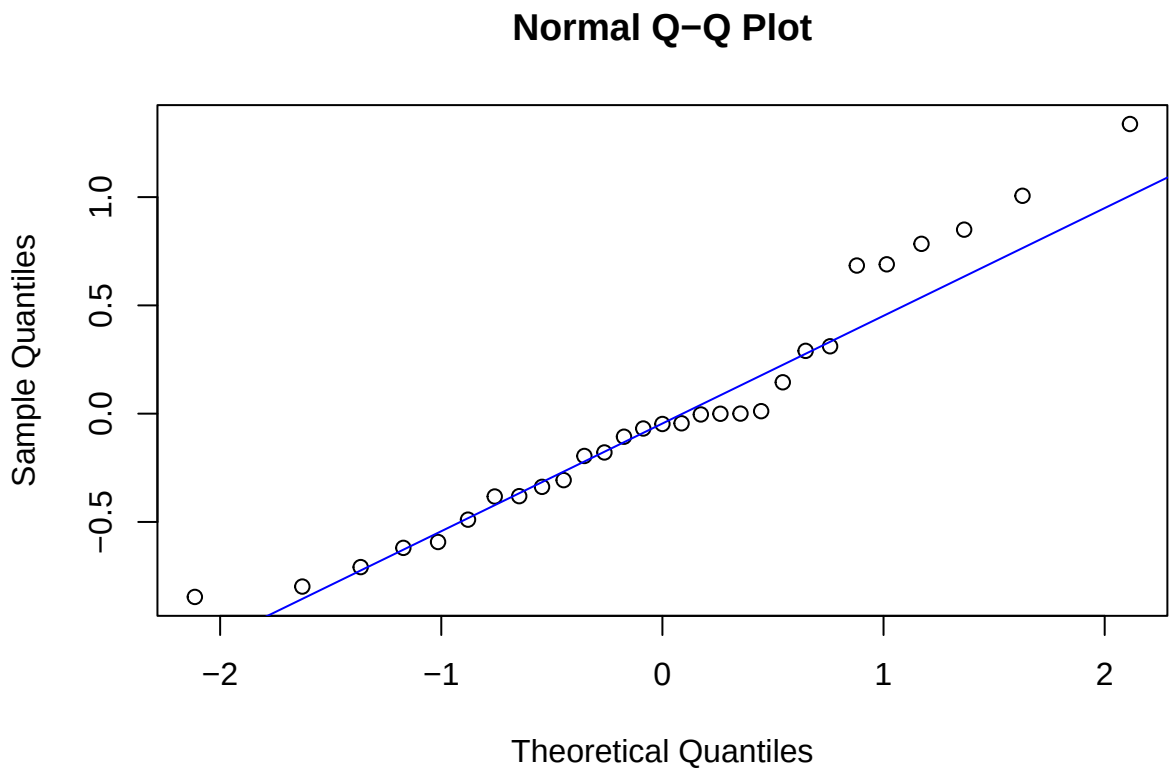

```r
model <- lm(Stiffness ~ Density, data = data)
plot(model$fitted.values, model$residuals, main = "Residuals vs Fitted",
    xlab = "Fitted Values", ylab = "Residuals", pch = 19)
abline(h = 0, col = "red")
```




```
sqrt_abs_resid <- sqrt(abs(model$residuals))
plot(model$fitted.values, sqrt_abs_resid, main = "Scale-Location Plot",
    xlab = "Fitted Values", ylab = "Sqrt Residuals", pch = 19)
```



```r
qqnorm(model$residuals)
qqline(model$residuals, col = "blue")
```




```r
data$sqrt_Stiffness <- sqrt(data$Stiffness)

powers <- seq
```