Data Transformation- Lab 11_New

Anamika Khanal

1. Read the Dataset "class11_LAB_dataFrame_20190930T2158.csv" into a dataframe.

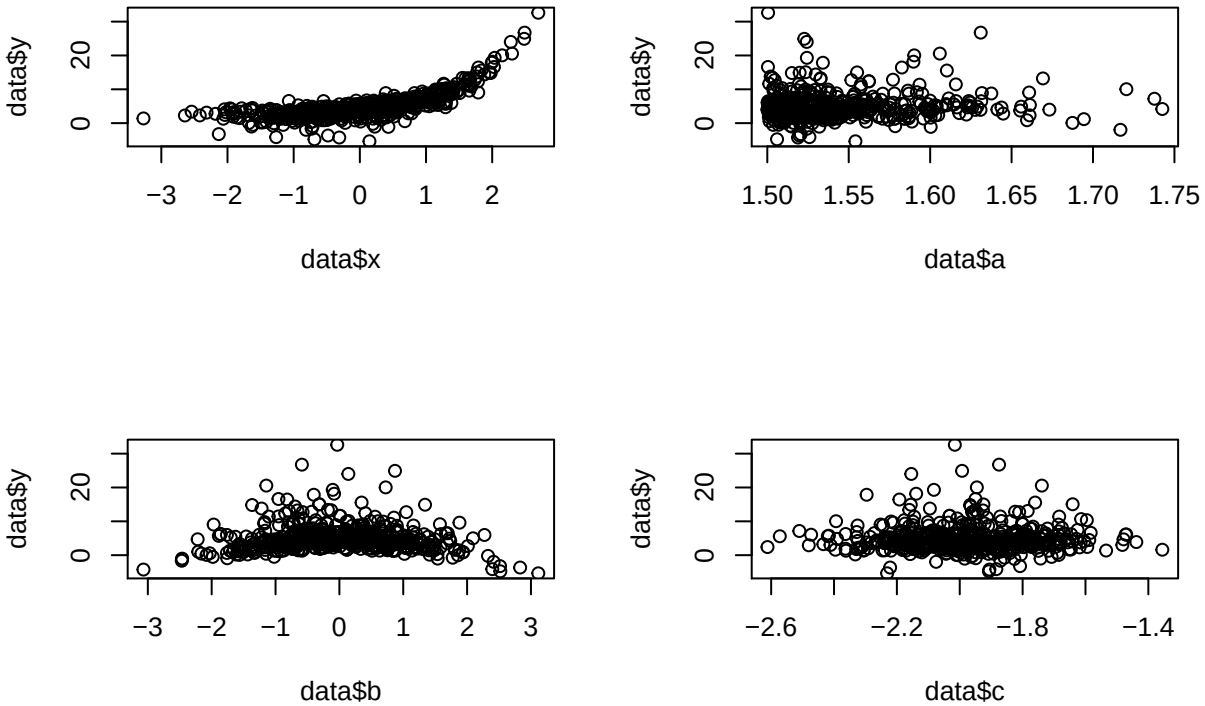
```
data = read.csv("class11_LAB_dataFrame_20190930T2158.csv", header= TRUE)
```

2. The target variable is **y**

Exploratory Data Analysis

1. Plot y vs.
2. x
3. a
4. b
5. c

```
par(mfrow=c(2,2))
plot(data$x,data$y)
plot(data$a,data$y)
plot(data$b,data$y)
plot(data$c,data$y)
```



2. Summarize the relationships between y and the set of explanatory variables (x,a,b,c). Summary: y is positively related to x, however the relationship doesn't appear to be linear. The relationship between y and a is not very apparent from the above plot.
The relationship between y and b seems slightly negative after a certain point. The relationship does not appear to be linear. The relationship between y and c is not very apparent either from the above plot.

Regression

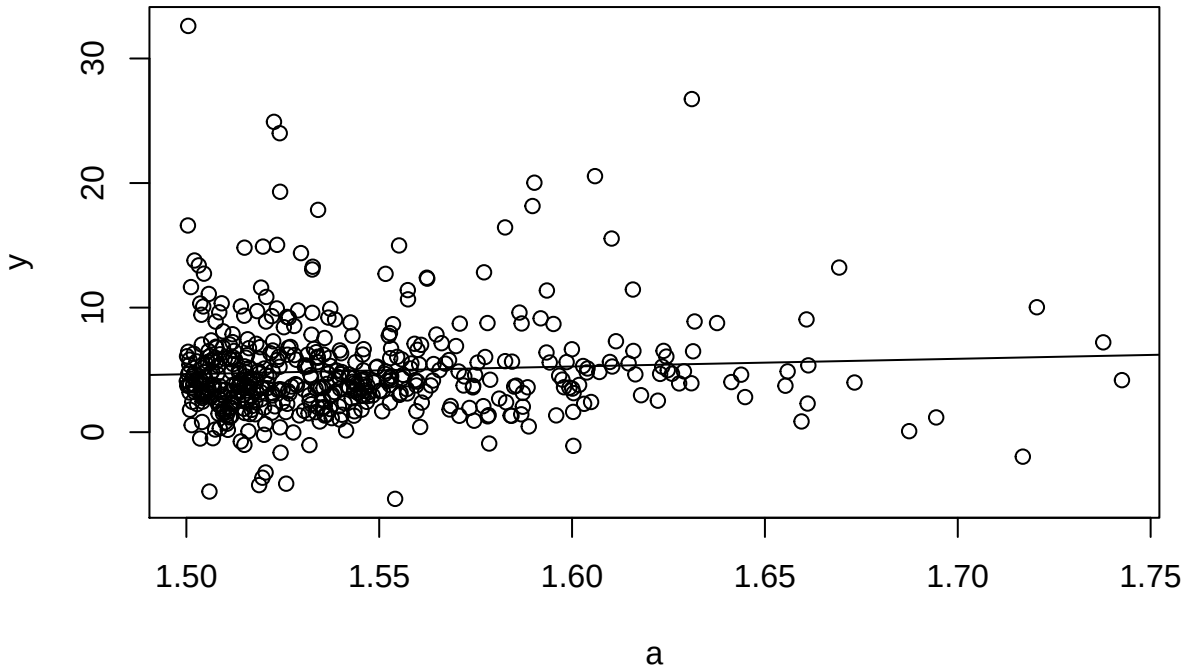
1. Please fit a simple linear regression model between **y** and **a**

```
model1 <- lm(y~a, data=data)
summary(model1)
```

```
##
## Call:
## lm(formula = y ~ a, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.341  -2.039  -0.807   1.172  27.957
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.641      6.712  -0.691    0.490
## a              6.195      4.348   1.425    0.155
##
## Residual standard error: 4.021 on 498 degrees of freedom
## Multiple R-squared:  0.00406,    Adjusted R-squared:  0.00206
## F-statistic:  2.03 on 1 and 498 DF,  p-value: 0.1548
```

2. Make a scatter plot of **y** and **a**
3. Overlay the regression line

```
plot(y ~ a, data = data)
abline(model1)
```



4. Comment on the predicted model versus ground truth Comment: The predicted model seems well approximate the true function. The relationship between y and a is linear and weakly positive.

Transforms

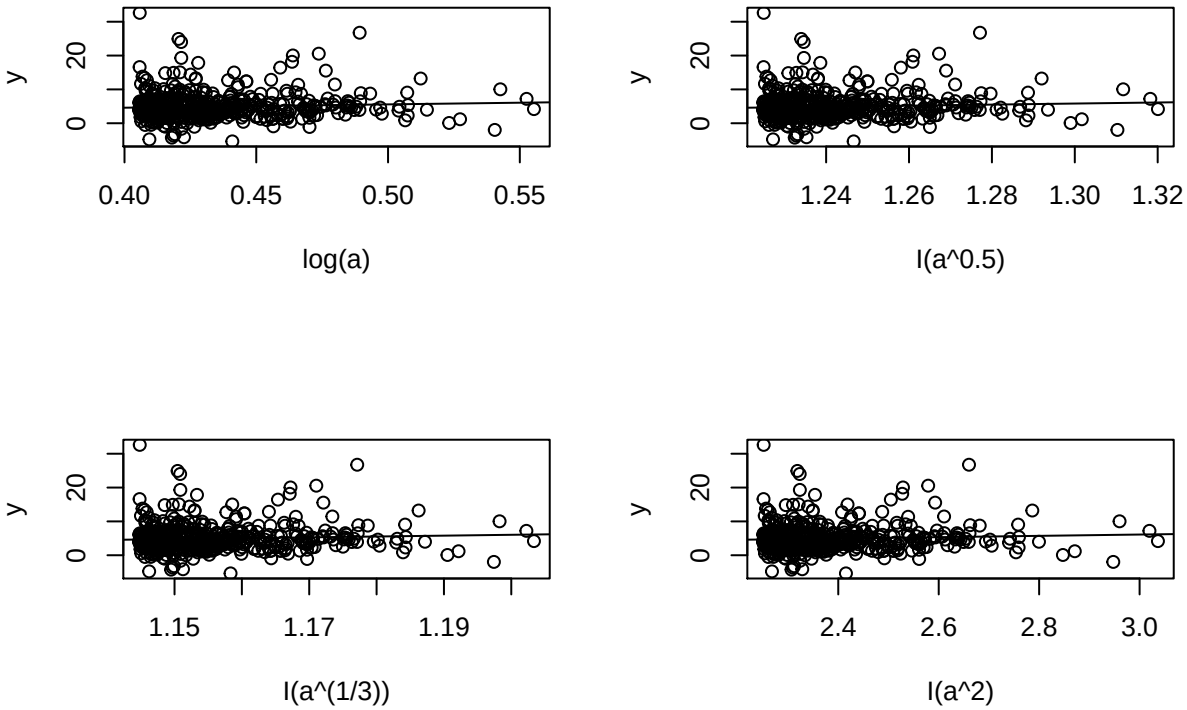
1. Transform **a** 4 different ways Answer: I will be using log(a), sqrt(a), a^1/3 and a^2 transformations.

2. Fit a linear regression

```
model_log <- lm(y~log(a), data=data)
model_sqrt <- lm(y~ I(a^0.5), data=data)
model_powerthird <- lm(y~I(a^(1/3)), data=data)
model_sq <- lm(y~I(a^2), data=data)

#summary(model_log)
#summary(model_sqrt)
#summary(model_powerthird)
#summary(model_sq)
```

3. Overlay the regression line

```
par(mfrow=c(2,2))
#for log(a)
plot(y ~ log(a), data = data)
abline(model_log)
#for sqrt(a)
plot(y ~ I(a^0.5), data = data)
abline(model_sqrt)
#for powerthird(a)
plot(y ~ I(a^(1/3)), data = data)
abline(model_powerthird)
#for sq(a)
plot(y ~ I(a^2), data = data)
abline(model_sq)
```



4. Comment on the predicted model versus ground truth Comment: All four transformations seem to well approximate the true function since most of the data points closely align with the direction and position of the predicted function.

Residuals

1. Plot a histogram of the above 4 model residuals

```
par(mfrow=c(2,2))
#model_log
data$residuals_log <- residuals(model_log)
typeof(data$residuals_log)
```

```
## [1] "double"
```

```
h= hist(data$residuals_log)
#title('log(a)')

#model_sqrt
data$residuals_sqrt <- residuals(model_sqrt)
typeof(data$residuals_sqrt)
```

```
## [1] "double"
```

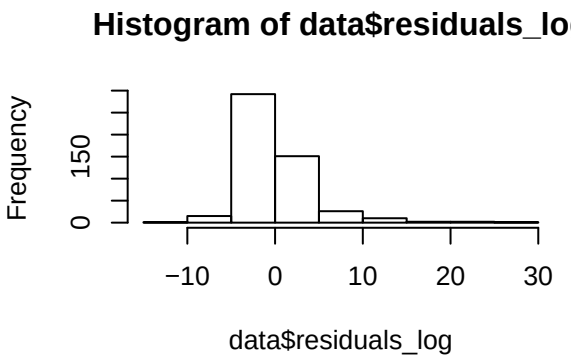
```
h1= hist(data$residuals_sqrt)
#title('sqrt(a)')

#model_powerthird
data$residuals_powerthird <- residuals(model_powerthird)
typeof(data$residuals_powerthird)
```

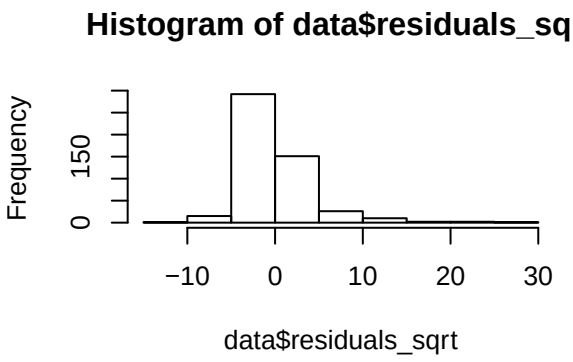
```
## [1] "double"
```

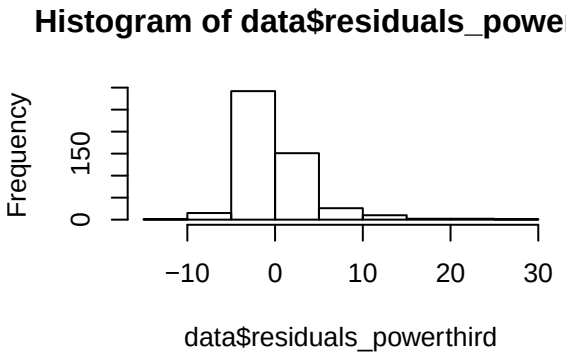
```
h2= hist(data$residuals_powerthird)
#title('a^1/3')

#model_sq
data$residuals_sq <- residuals(model_sq)
typeof(data$residuals_sq)
```

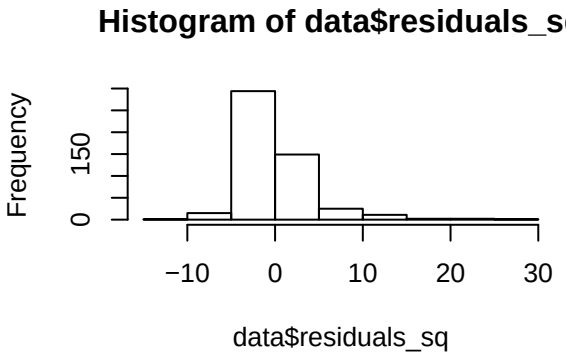
```
## [1] "double"
```

```
h3= hist(data$residuals_sq)
```









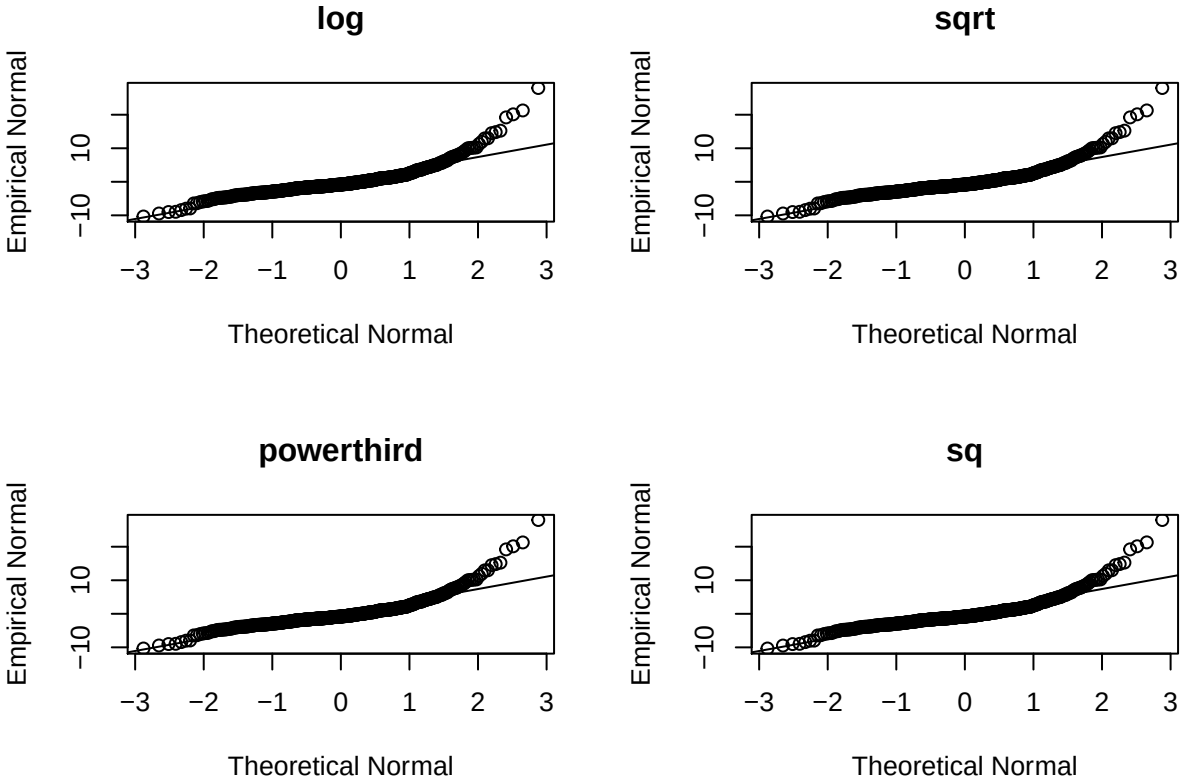

```
#title('x^2')
```

QQ

1. Plot a qqplot of the above 4 model residuals

```
par(mfrow=c(2,2))

# plot 1
epsilons_log = sort(data$residuals_log)
N = length(epsilons_log)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_log
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_log~theoreticalNormal))
title('log')

# plot 2
epsilons_sqrt = sort(data$residuals_sqrt)
N = length(epsilons_sqrt)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_sqrt
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
```

```
abline(lm(epsilons_sqrt~theoreticalNormal))
title('sqrt')

# plot 3
epsilons_powerthird = sort(data$residuals_powerthird)
N = length(epsilons_powerthird)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_powerthird
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_powerthird~theoreticalNormal))
title('powerthird')

# plot 4
epsilons_sq = sort(data$residuals_sq)
N = length(epsilons_sq)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_sq
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_sq~theoreticalNormal))
title('sq')
```



How do the residuals and qq plot compare between your 4 different transforms? Comment: From the qqplot, we can see that the distribution of error closely approximtes the normal distribution for all transformations

since the data points closely aligns with the line representing normal distribution. For the residual histogram, we can see that the errors are bell shaped and has mean around 0 for all 4 transformations.

MLR

1. Please fit a multiple linear regression model between **y** and (**x**, **a**, **b**, **c**)

```
model_multiple <- lm(y~x+a+b+c, data=data)
summary(model_multiple)
```

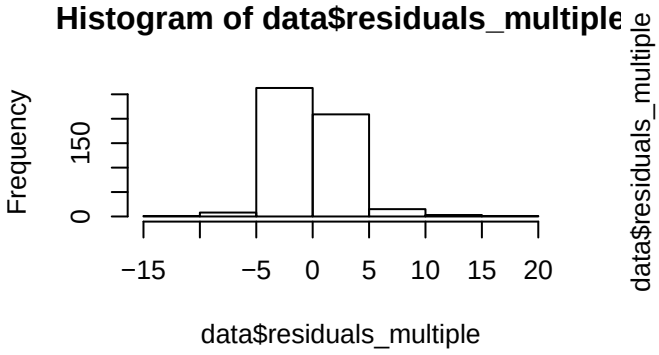
```
##
## Call:
## lm(formula = y ~ x + a + b + c, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.0914  -1.3495  -0.1577   1.0762  19.5012
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.5123     4.7373   1.375   0.1698
## x             3.0016     0.1226  24.487   <2e-16 ***
## a            -1.2917     2.9466  -0.438   0.6613
## b            -0.2199     0.1249  -1.760   0.0789 .
## c            -0.2159     0.6192  -0.349   0.7275
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.709 on 495 degrees of freedom
## Multiple R-squared:  0.5507, Adjusted R-squared:  0.547
## F-statistic: 151.7 on 4 and 495 DF,  p-value: < 2.2e-16
```

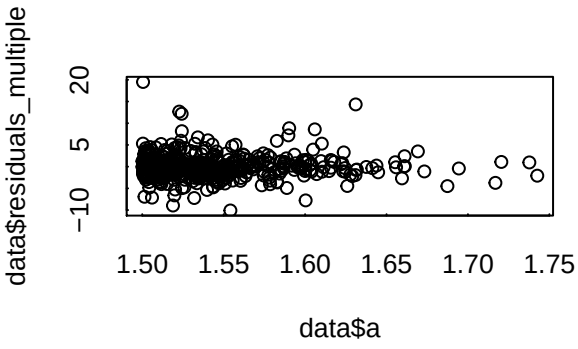
```
par(mfrow=c(2,2))
#histogram for residuals
data$residuals_multiple <- residuals(model_multiple)
typeof(data$residuals_multiple)
```

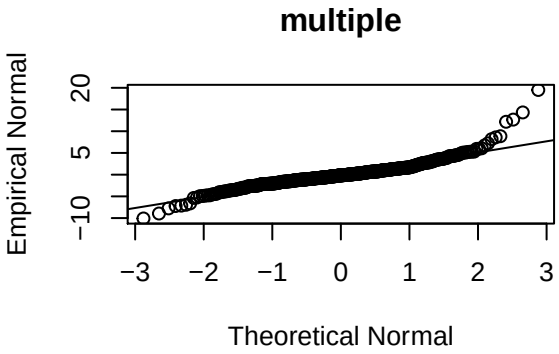
```
## [1] "double"
```

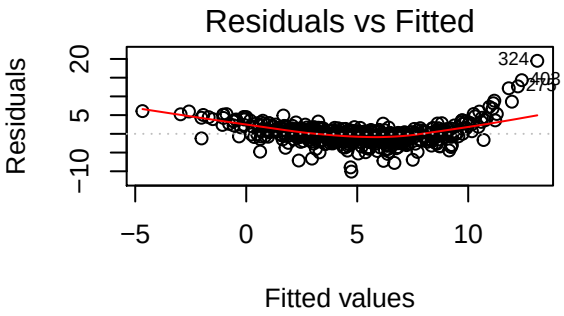
```
h= hist(data$residuals_multiple)
#title('Multiple')

# for checking equality of variance
plot(data$a,data$residuals_multiple, tck=0.01)

# qqplot
epsilons_multiple = sort(data$residuals_multiple)
N = length(epsilons_multiple)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_multiple
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_multiple~theoreticalNormal))
title('multiple')
```

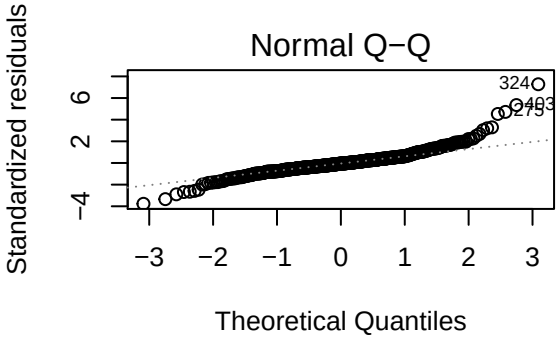
```
#for checking linearity
plot(model_multiple)
```

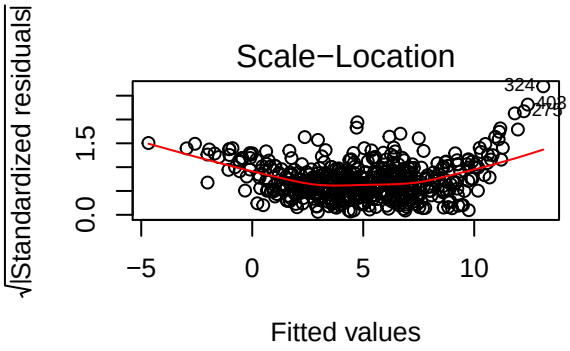


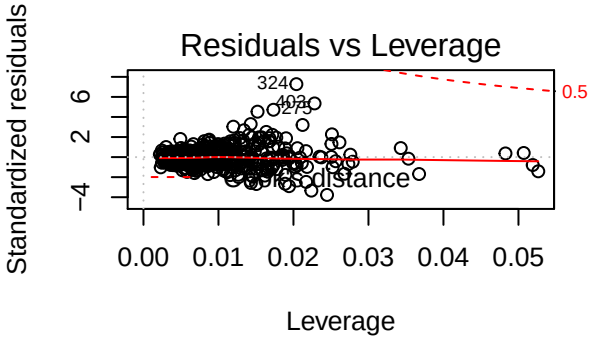












2. Does your MLR satisfy the LINE assumptions? Answer: We can assume that the choice of datapoints used

in MLR are independent to each others. From the qqplot, we can see that the datapoints do not closely align with the line for normal distribution. Thus, errors are not normally distributed. From the residual scatterplot, we can see that the residuals are not indifferent to a. Thus, equality of variance assumption is violated. Since the relationship between residuals and fitted is not random, linearity assumption is violated as well. Thus, MLR does not satisfy the LINE assumption.

3. If not, what transforms do you recommend be applied to each explanatory variable and why? Answer: We can start off using transforming a in various ways (like we did above) since we saw that there is a relationship between a and residuals from the above plots. If this does not fix, we can transform remaining explanatory variable as well and choose the model which fulfills the LINE assumptions and has the highest R-squared.

4. Transform the variable you feel most violates the LINE assumptions. Answer: I am transforming a to log(a), sqrt(a), a^(1/3), sq(a)

5. Plot a histogram of the above 4 model residuals

```
#log
model_multiplelog <- lm(y~log(a)+b+x+c, data=data)
model_multiplesqrt <- lm(y~sqrt(a)+b+x+c, data=data)
model_mutiplepowerthird <- lm(y~I(a^(1/3))+b+x+c, data=data)
model_multiplesq <- lm(y~I(a^2)+b+x+c, data=data)


par(mfrow=c(2,2))
#model_log
data$residuals_logM <- residuals(model_multiplelog)
typeof(data$residuals_logM)
```

```
## [1] "double"
```

```
h= hist(data$residuals_logM)
#title('log(a)')

#model_sqrt
data$residuals_sqrtM <- residuals(model_multiplesqrt)
typeof(data$residuals_sqrtM)
```

```
## [1] "double"
```

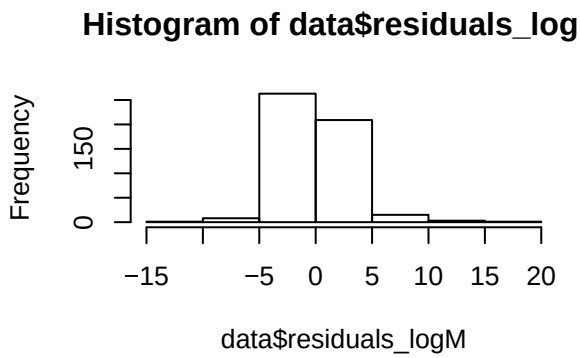
```
h1= hist(data$residuals_sqrtM)
#title('sqrt(a)')

#model_powerthird
data$residuals_powerthirdM <- residuals(model_mutiplepowerthird)
typeof(data$residuals_powerthirdM)
```

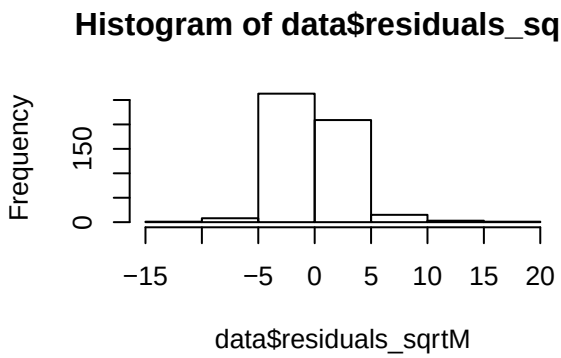
```
## [1] "double"
```

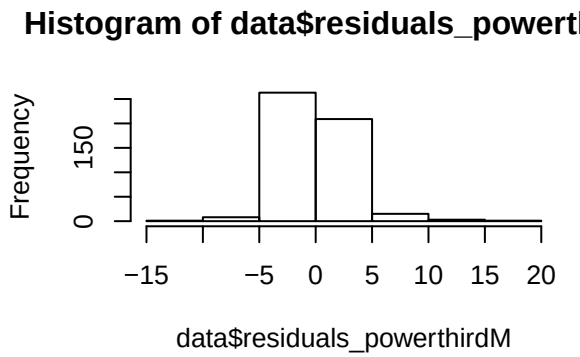
```
h2= hist(data$residuals_powerthirdM)
#title('a^1/3')

#model_sq
data$residuals_sqM <- residuals(model_multiplesq)
typeof(data$residuals_sqM)
```

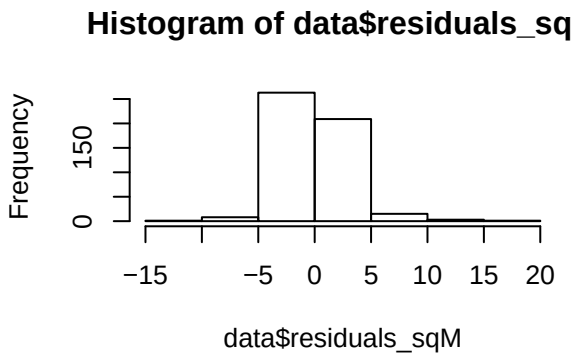
```
## [1] "double"
```

```
h3= hist(data$residuals_sqM)
```







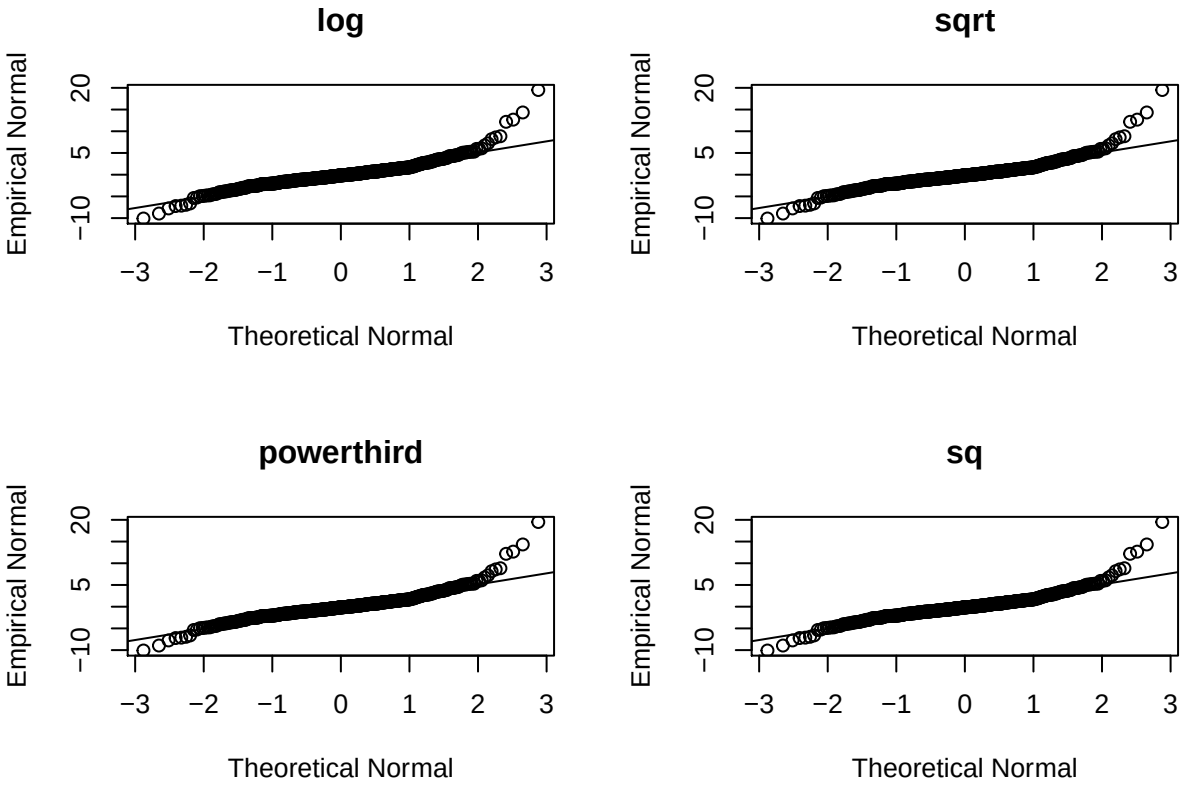



```
#title('x^2')
```

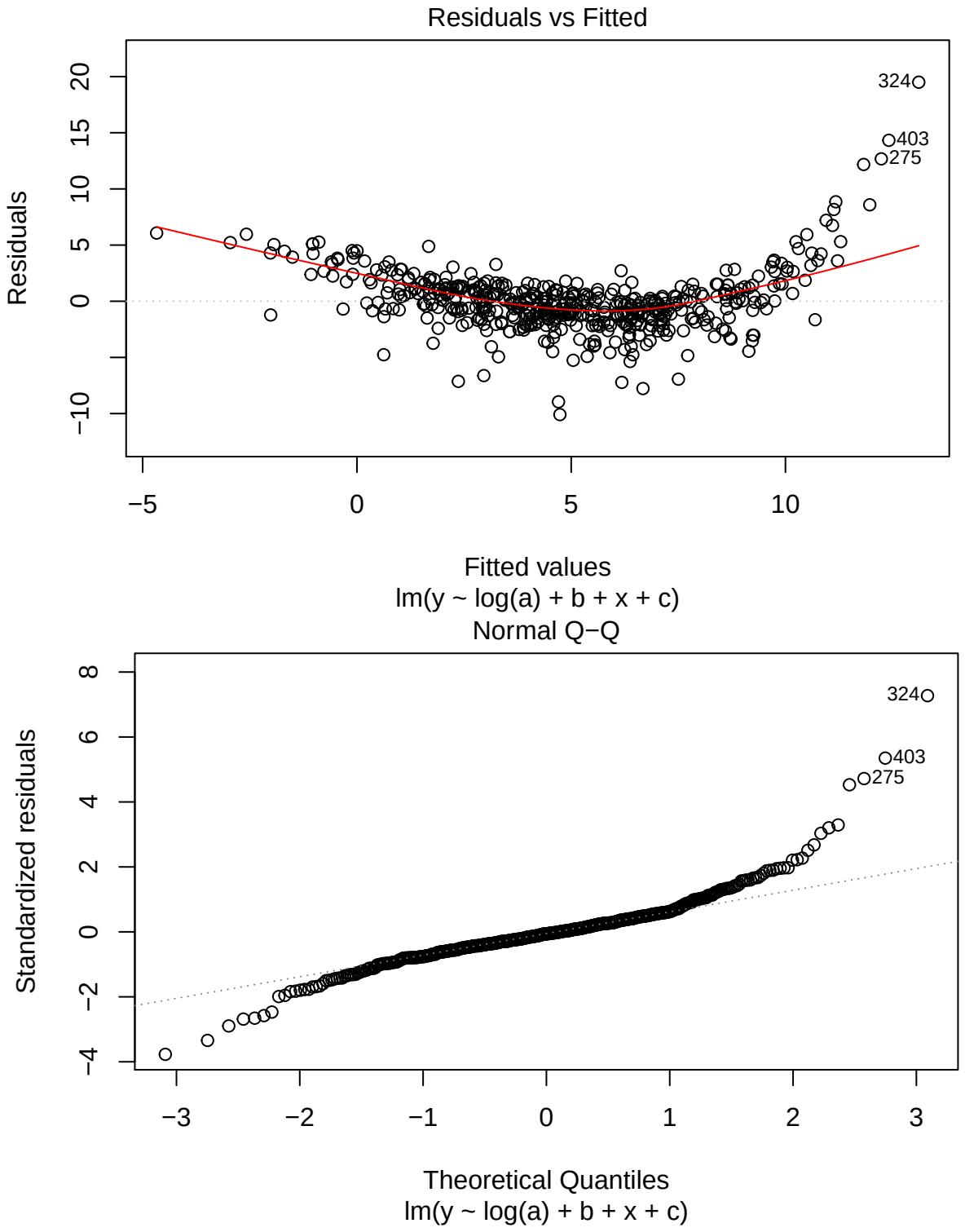
6. Plot a qqplot of the above 4 model residuals

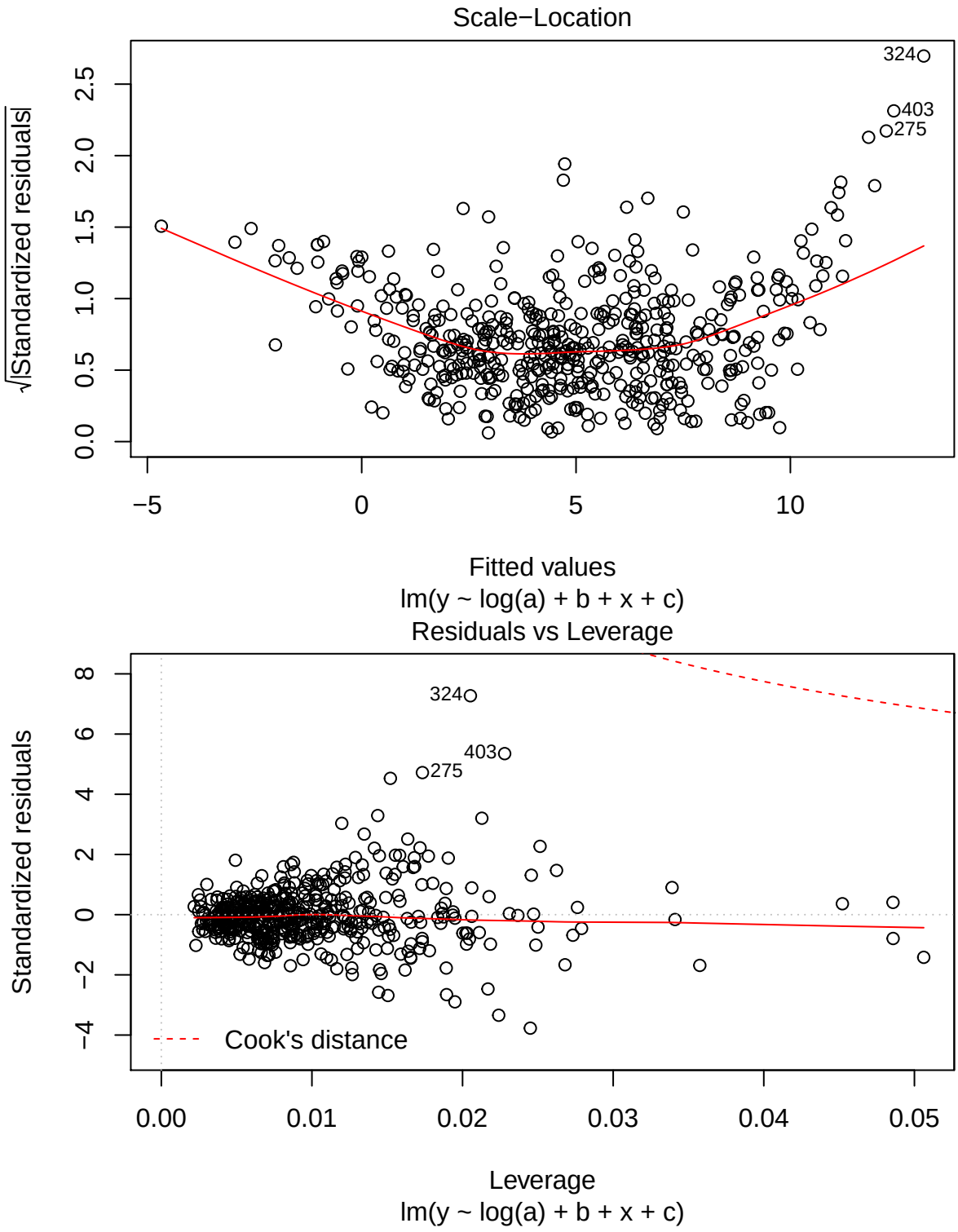
```
par(mfrow=c(2,2))

# plot 1
epsilons_logM = sort(data$residuals_logM)
N = length(epsilons_logM)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_logM
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_logM~theoreticalNormal))
title('log')

# plot 2
epsilons_sqrtM = sort(data$residuals_sqrtM)
N = length(epsilons_sqrtM)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_sqrtM
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
```

```
abline(lm(epsilons_sqrtM~theoreticalNormal))
title('sqrt')

# plot 3
epsilons_powerthirdM = sort(data$residuals_powerthirdM)
N = length(epsilons_powerthirdM)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_powerthirdM
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_powerthirdM~theoreticalNormal))
title('powerthird')

# plot 4
epsilons_sqM = sort(data$residuals_sqM)
N = length(epsilons_sqM)
probs = seq(1,N)/(N+1)
theoreticalNormal = qnorm(probs)

plot(theoreticalNormal,epsilons_sqM
     ,xlab='Theoretical Normal'
     ,ylab='Empirical Normal')
abline(lm(epsilons_sqM~theoreticalNormal))
title('sq')
```

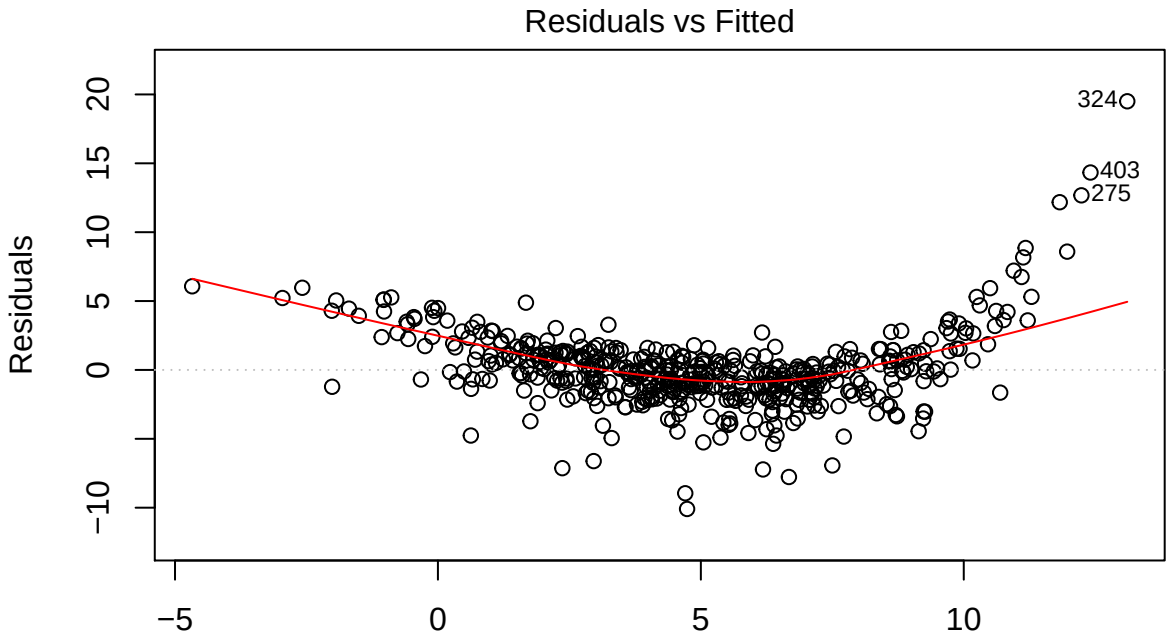


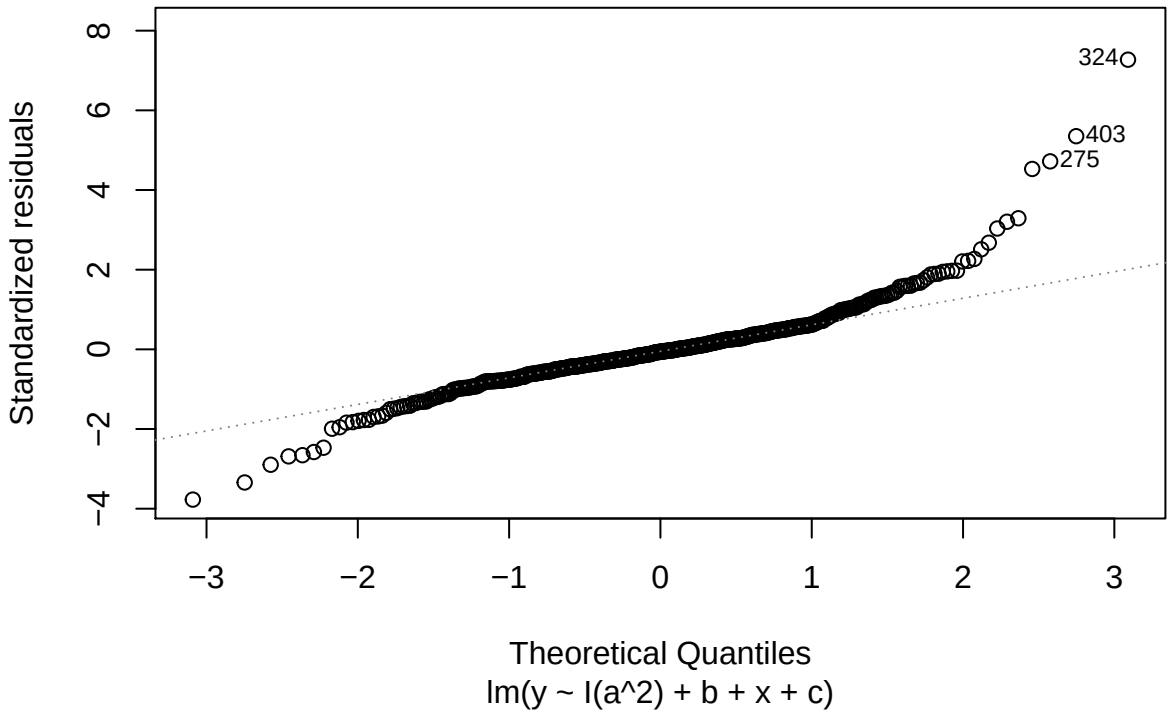
```
plot(model_multiplelog)
```

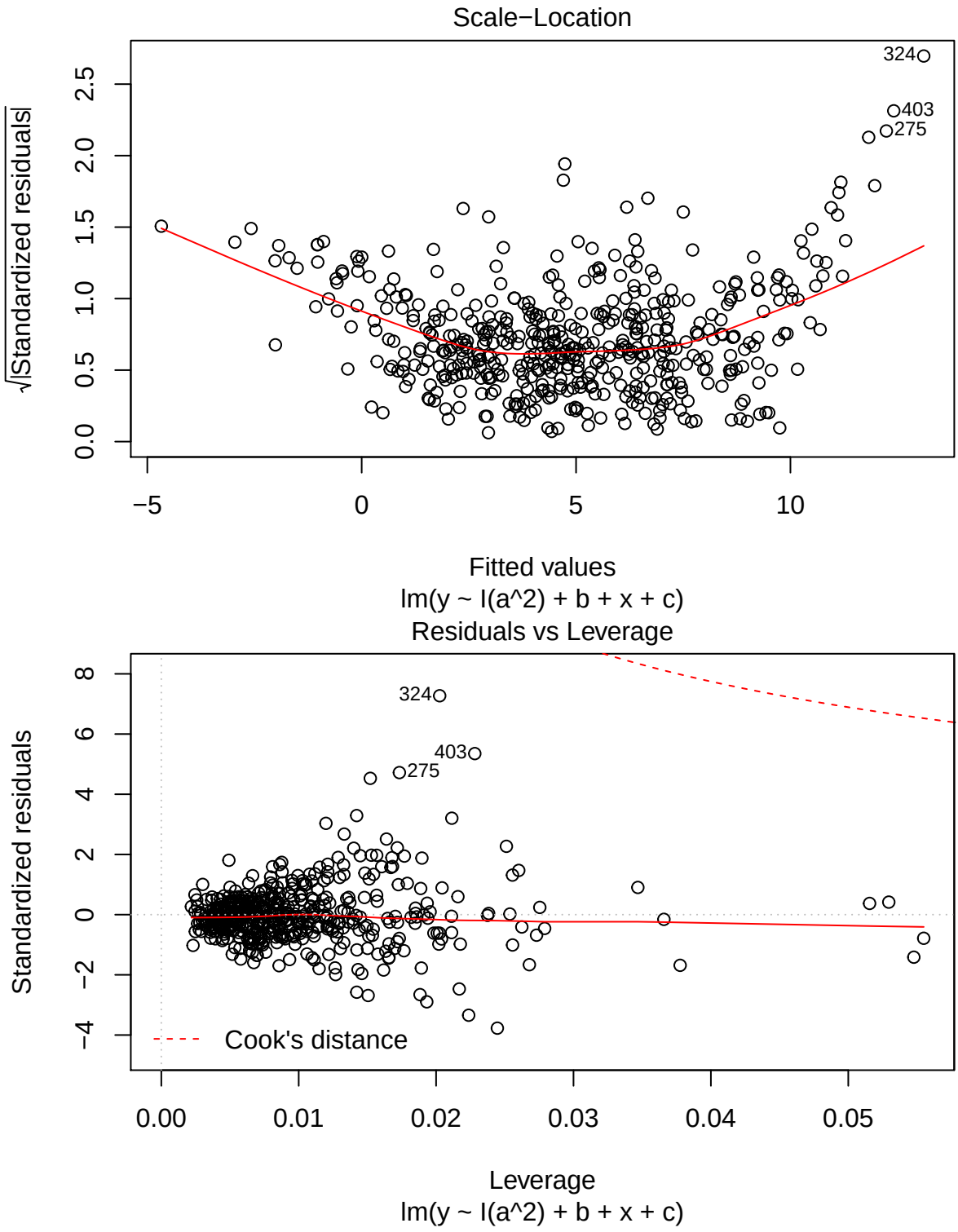




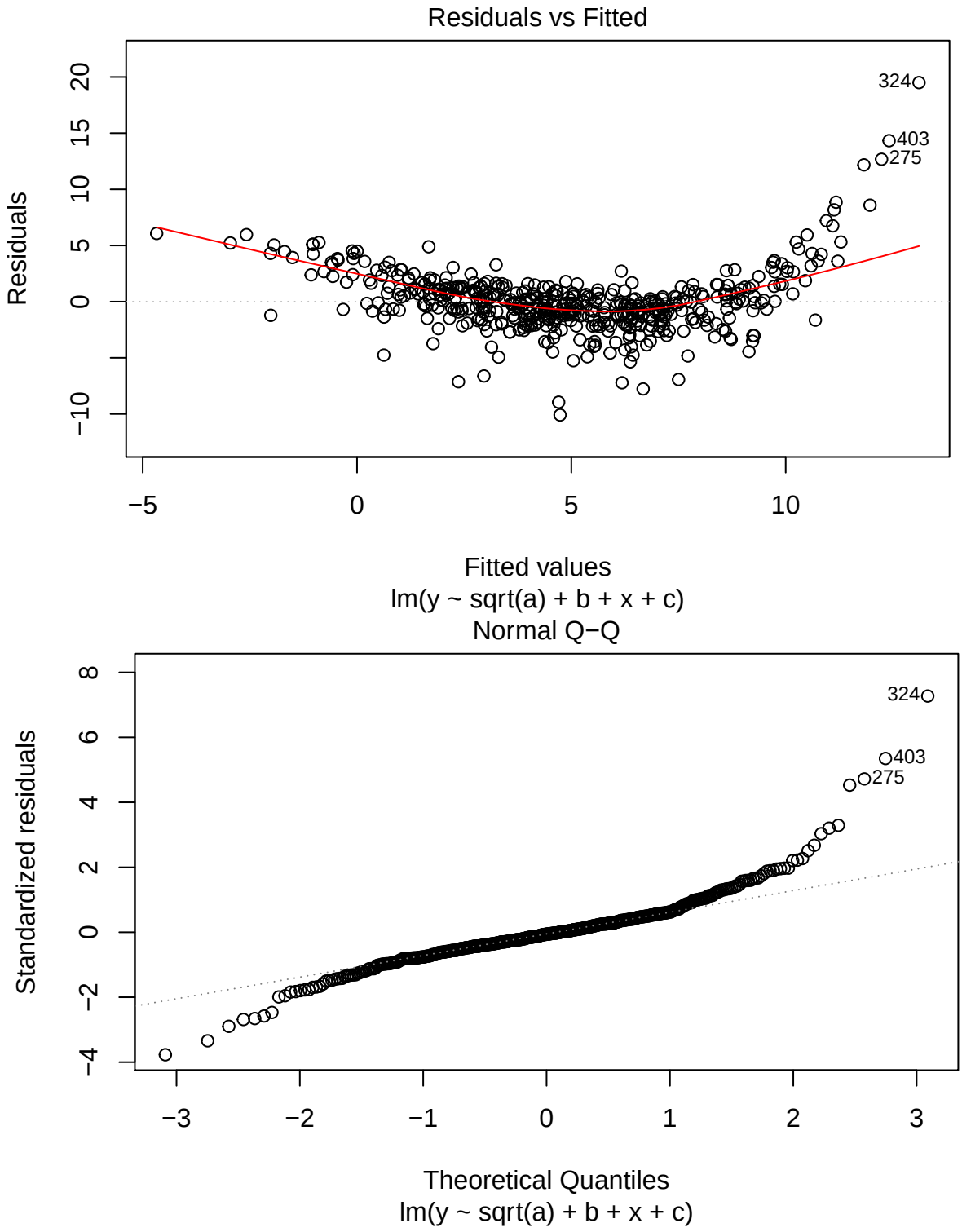

```
plot(model_multiplesq)
```

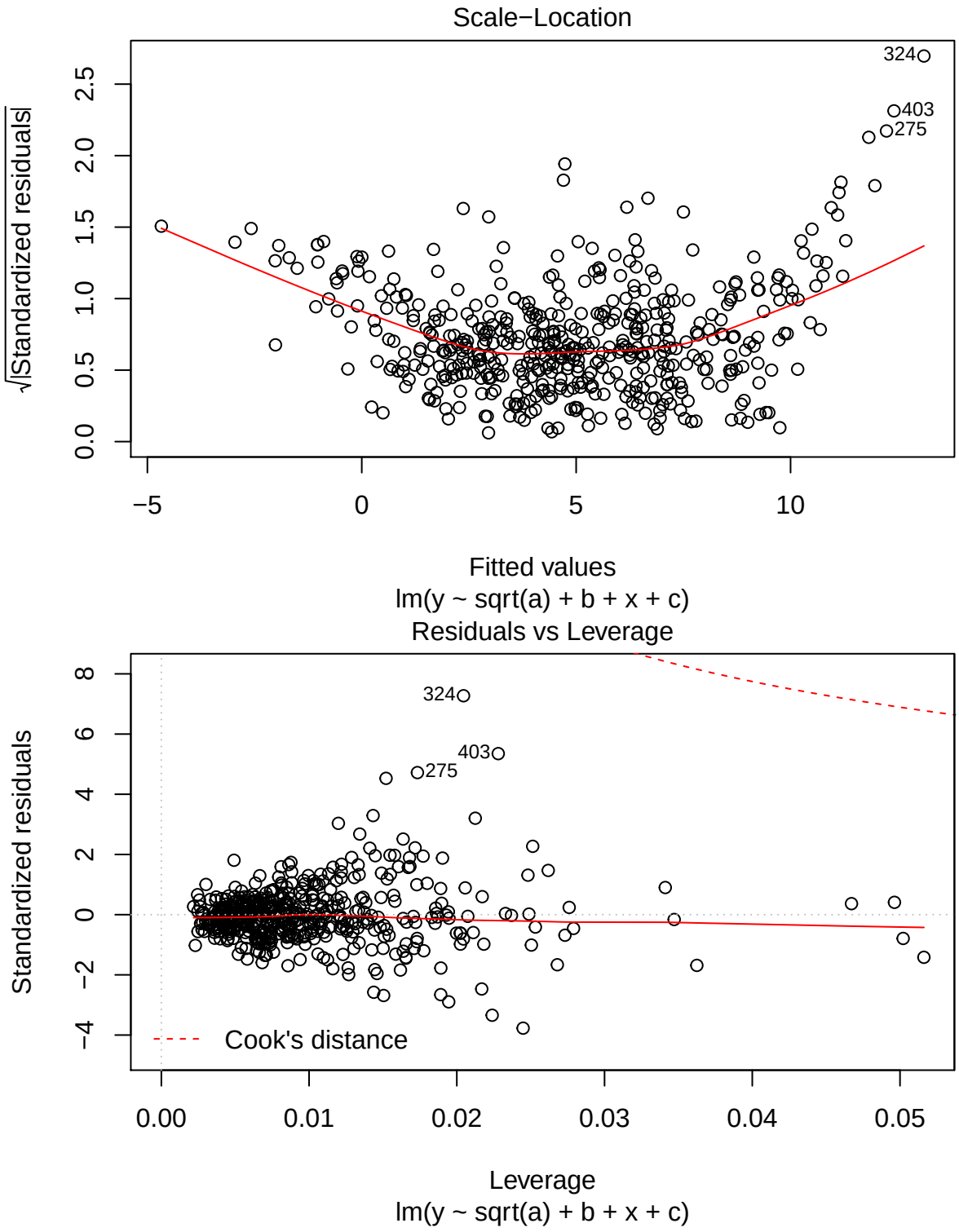

Residuals vs Fitted
Residuals
Fitted values
lm(y ~ I(a^2) + b + x + c)
324
403
275
20
15
10
5
0
−10
−5
0
5
10


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(y ~ I(a^2) + b + x + c)
324
403
275
8
6
4
2
0
−2
−4
−3
−2
−1
0
1
2
3

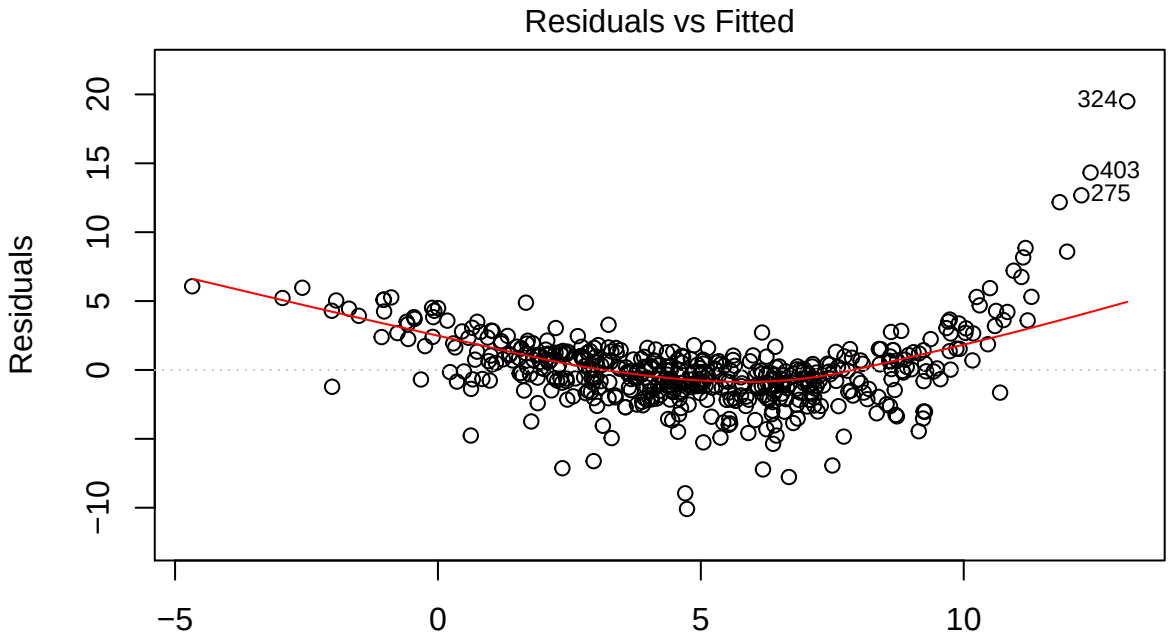


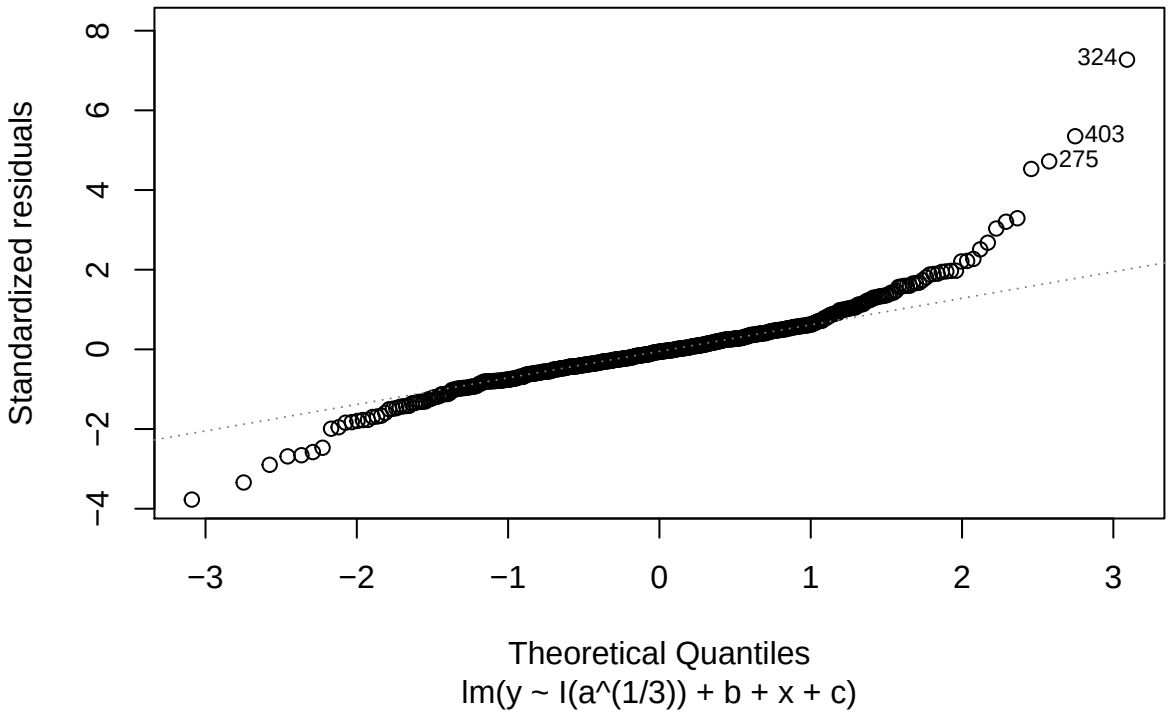

```
plot(model_multiplesqrt)
```

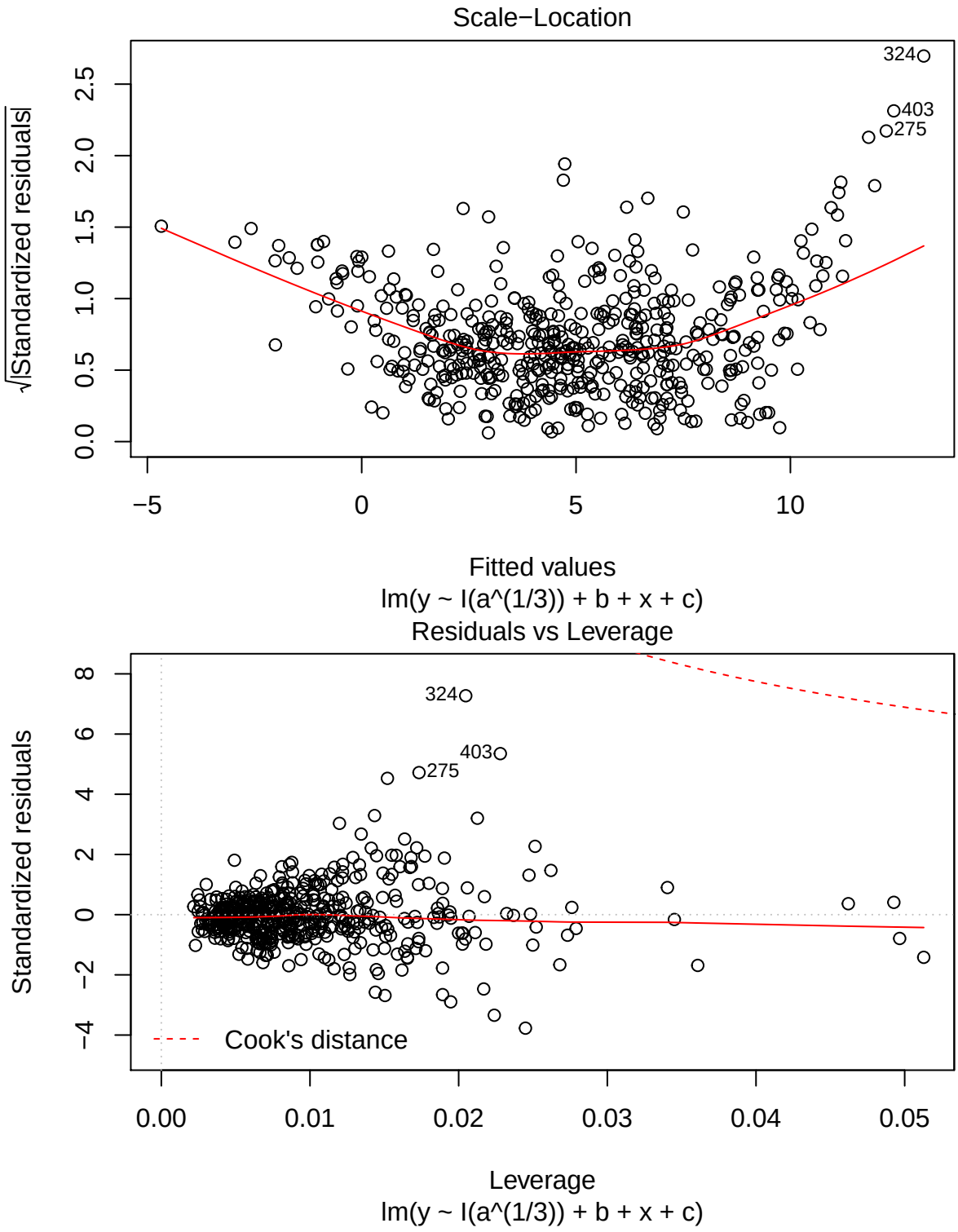

Residuals vs Fitted
Residuals
Fitted values
lm(y ~ sqrt(a) + b + x + c)
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(y ~ sqrt(a) + b + x + c)
324
403
275




```
plot(model_mutiplepowerthird)
```


Residuals vs Fitted
Residuals
Fitted values
lm(y ~ I(a^(1/3)) + b + x + c)
324
403
275


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(y ~ I(a^(1/3)) + b + x + c)
324
403
275



7. Does transforming this variable help? why or why not? Answer: No, none of the transformation in a works because: 1) From the qqplot, we can see that the the datapoints still does not align with the normal distribution– violation of normality. 2) From the residuals vs fitted plots, we can see that non-random pattern between residuals and predictions for all transformations– violation of linearity.