C1M5_peer_reviewed

March 4, 2025

1 Module 5: Peer Reviewed Assignment

1.0.1 Outline:

The objectives for this assignment:

1. Understand what can cause violations in the linear regression assumptions.
2. Enhance your skills in identifying and diagnosing violated assumptions.
3. Learn some basic methods of addressing violated assumptions.

General tips:

1. Read the questions carefully to understand what is being asked.
2. This work will be reviewed by another human, so make sure that you are clear and concise in what your explanations and answers.

[1]:
```
# Load Required Packages
library(ggplot2)
```

1.1 Problem 1: Let's Violate Some Assumptions!

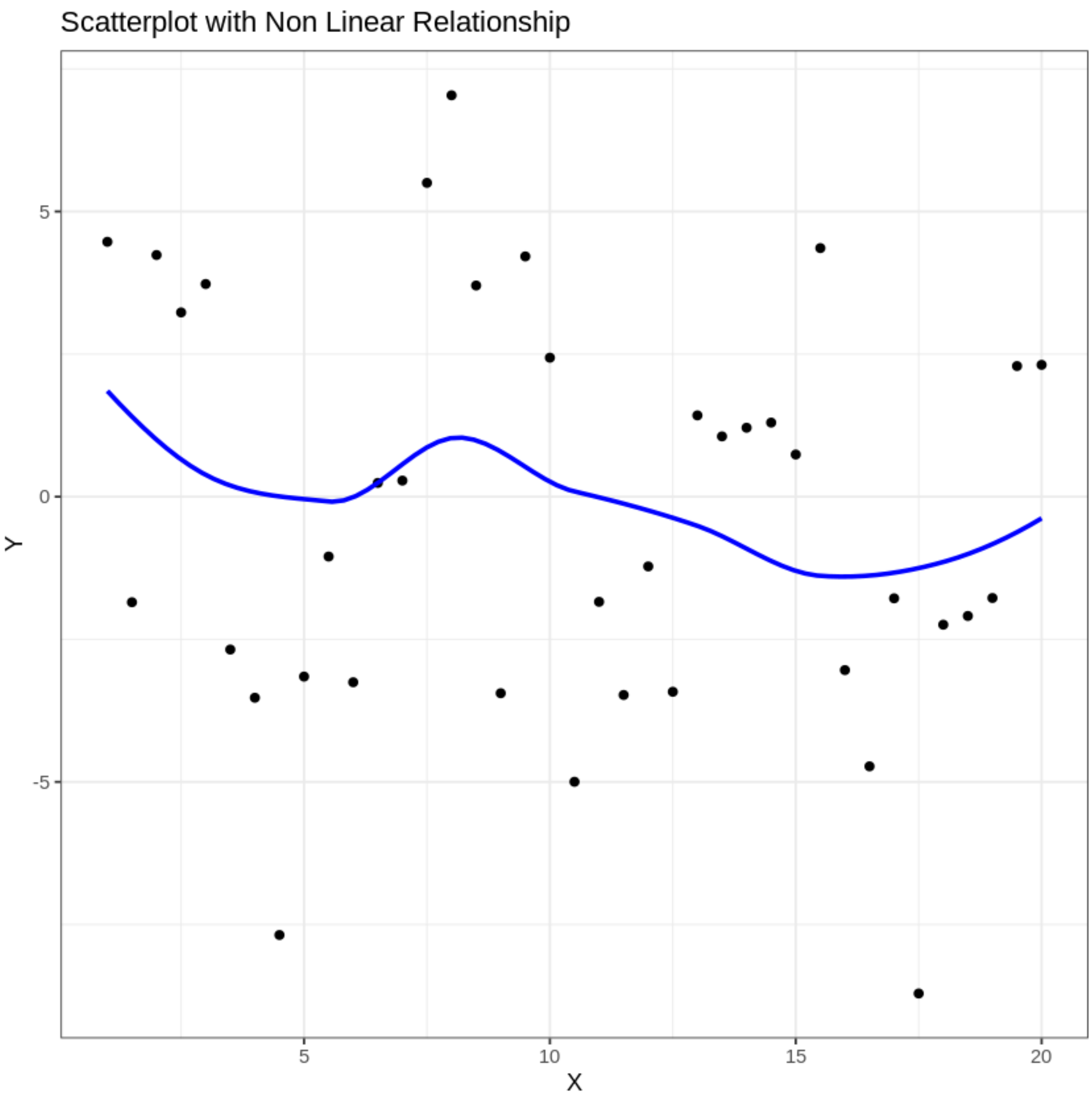
When looking at a single plot, it can be difficult to discern the different assumptions being violated. In the following problem, you will simulate data that purposefully violates each of the four linear regression assumptions. Then we can observe the different diagnostic plots for each of those assumptions.

1. (a) Linearity Generate SLR data that violates the linearity assumption, but maintains the other assumptions. Create a scatterplot for these data using ggplot.

Then fit a linear model to these data and comment on where you can diagnose nonlinearity in the diagnostic plots.

[2]:
```
x <- seq(1, 20, by = 0.5)
y <- 3*sin(x) + 2*rnorm(length(x))
df = data.frame(x1 = x, y = y)

ggplot(df, aes(x = x1, y = y)) +
geom_point() +
geom_smooth(se = F, col = "blue") +
labs(x = "X", y = "Y", title = "Scatterplot with Non Linear Relationship") +
theme_bw()

# Your Code Here
```

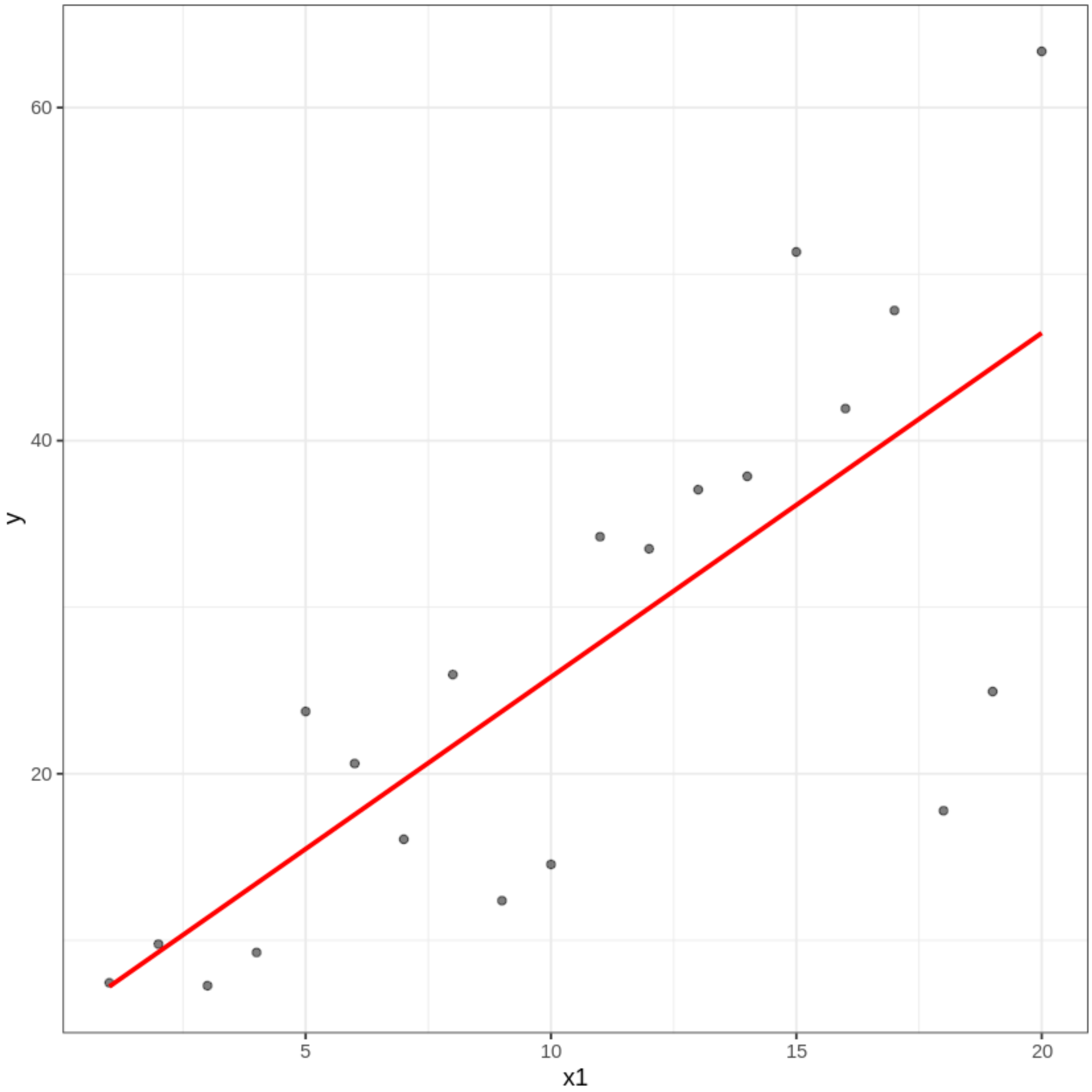
```
`geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



1. (b) Homoskedasticity Simulate another SLR dataset that violates the constant variance assumption, but maintains the other assumptions. Then fit a linear model to these data and comment on where you can diagnose non-constant variance in the diagnostic plots.

[3]:

```
set.seed(100000)
x <- seq(1, 20, by = 1)
y = 2*x + 5 + rnorm(length(x), mean = 0, sd = x)
df = data.frame(x1 = x, y = y)

ggplot(df, aes(x = x1, y = y)) +
geom_point(alpha = 0.5) +
geom_smooth(method = lm, formula = y ~ x, se = F, col = "red") +
theme_bw()

# Your Code Here
```

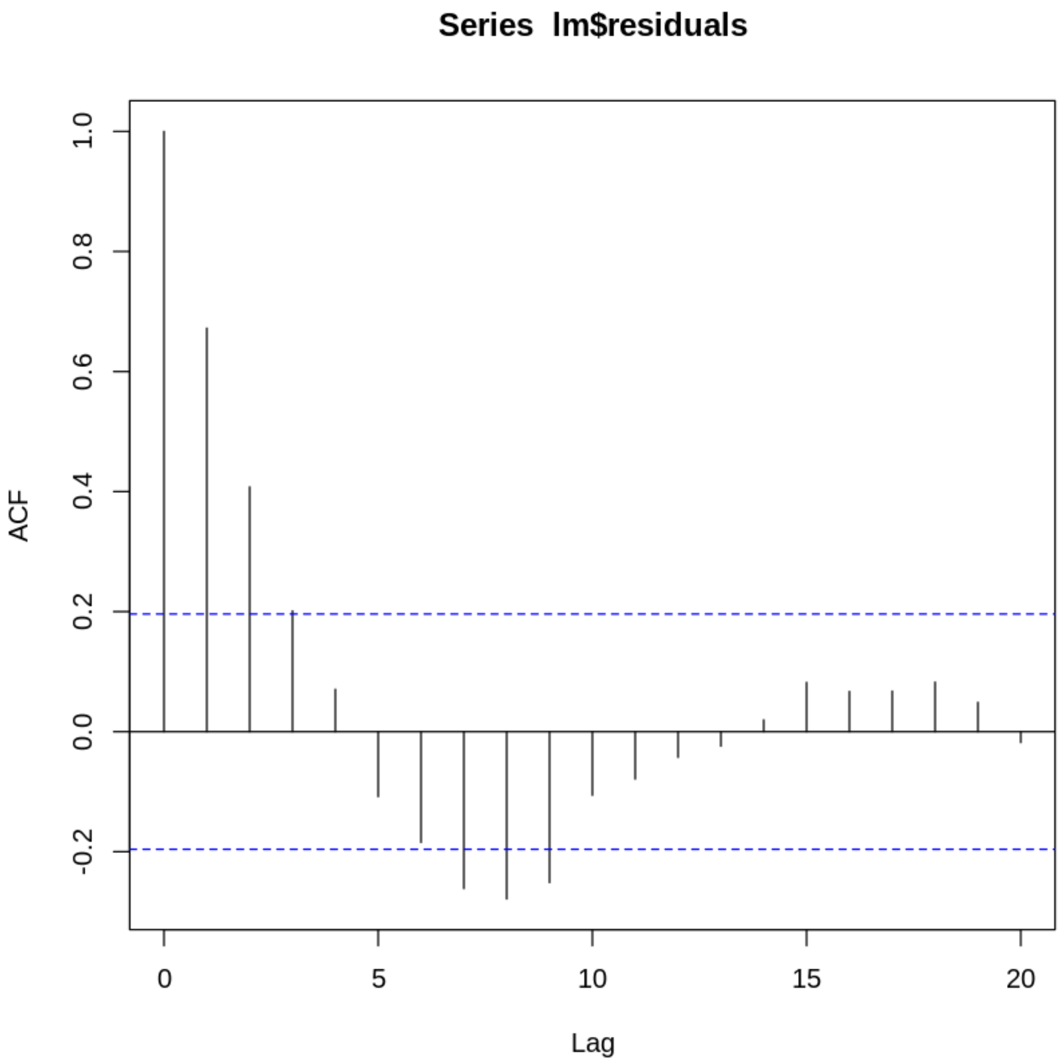


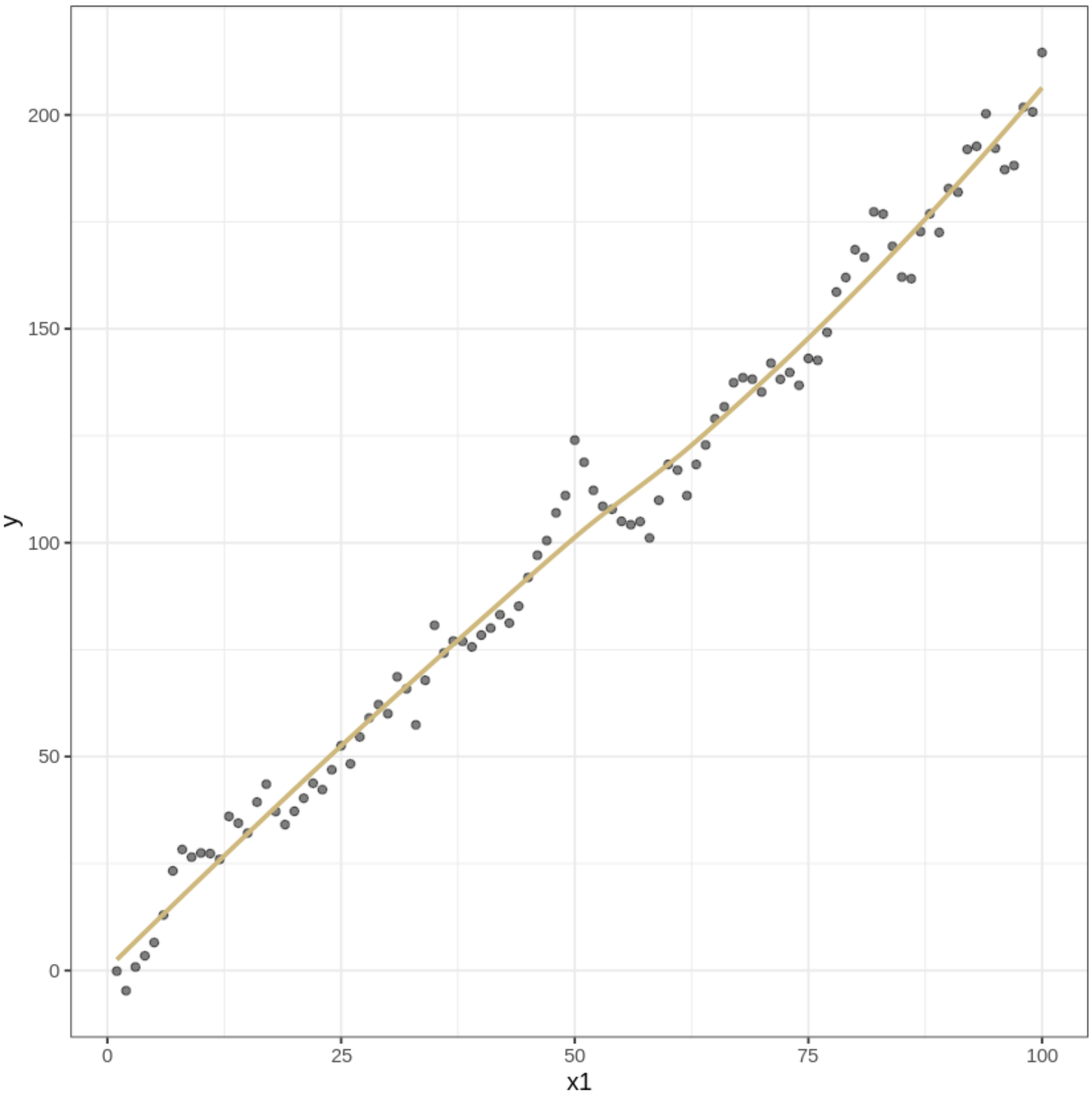
1. (c) Independent Errors Repeat the above process with simulated data that violates the independent errors assumption.

[4]:

```
set.seed(4567)
n <- 100
x <- 1:n
e <- arima.sim(model = list(ar = 0.7), n = n)
y <- 2*x + 5*e
df = data.frame(x1 = x, y = y)

lm <- lm(y~x, data = df)
acf(lm$residuals)

ggplot(df, aes(x = x1, y = y)) +
geom_point(alpha = 0.5) +
geom_smooth(se = F, col = "#CFB87C") +
theme_bw()# Your Code Here
```

```
Don't know how to automatically pick scale for object of type ts. Defaulting to
continuous.

`geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

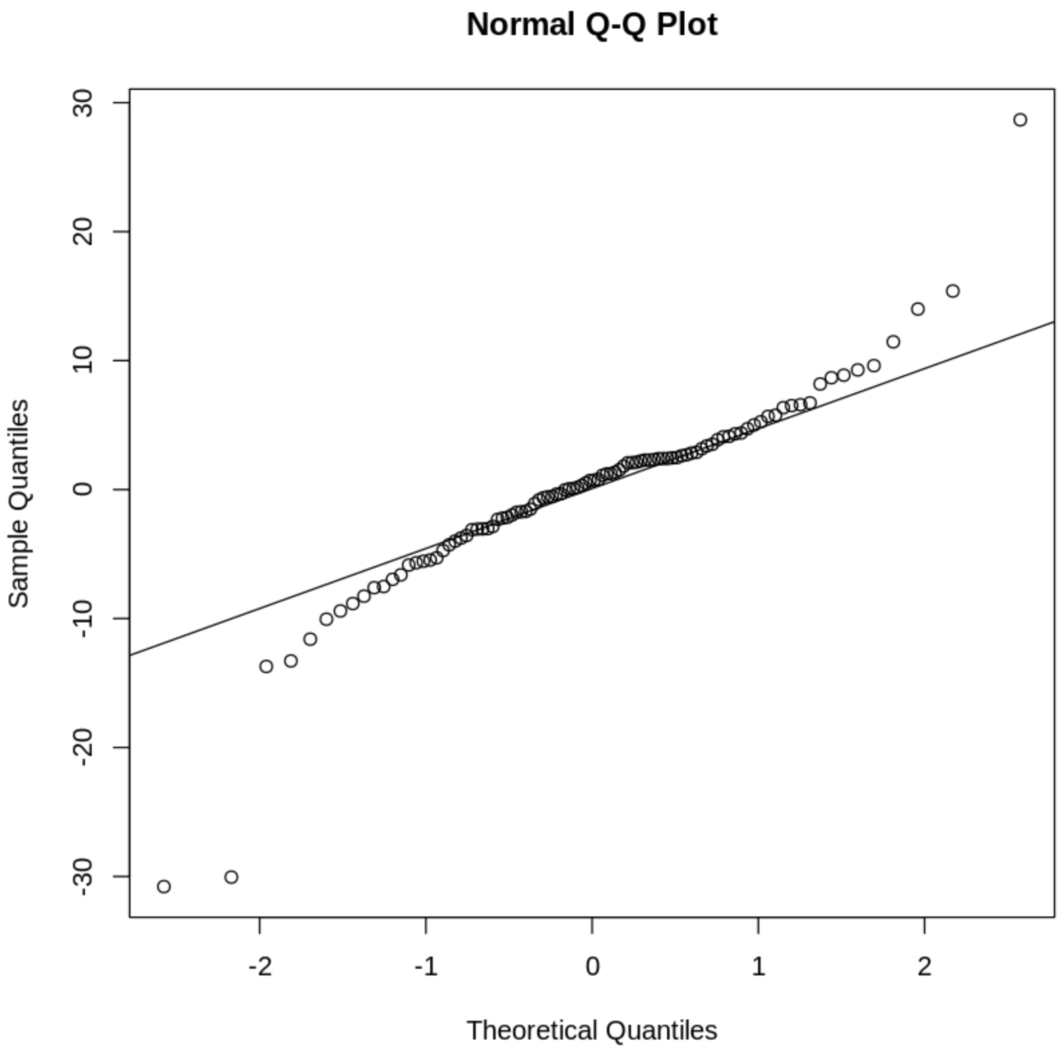

Series lm$residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0
5
10
15
20
Lag



1. (d) Normally Distributed Errors Only one more to go! Repeat the process again but simulate the data with non-normal errors.

[5]:
```
set.seed(927492)
n <- 100
x <- seq(1, n, by =1)
e <- rt(n, df = 3)
y <- 2*x + 5*e
df = data.frame(x1 = x, y = y)

lm2 <- lm(y~x, data = df)
qqnorm(lm2$residuals)
```

```
qqline(lm2$residuals)
```



2 Problem 2: Hats for Sale

Recall that the *hat* or *projection* matrix is defined as

$$H = X(X^T X)^{-1} X^T.$$

The goal of this question is to use the hat matrix to prove that the fitted values, $\widehat{\mathbf{Y}}$, and the residuals, $\widehat{\varepsilon}$, are uncorrelated. It's a bit of a process, so we will do it in steps.

2. (a) Show that $\widehat{Y} = HY$. That is, H “puts a hat on” Y.

[6]:
```
X <- matrix(c(1,1,1,1,2,3,4,5), ncol = 2)
H <- X %*% solve(t(X) %*% X) %*% t(X)

X
H
```

```
                              1    2
A matrix: 4 × 2 of type dbl   1    3
                              1    4
                              1    5

                              0.7    0.4    0.1    -0.2
A matrix: 4 × 4 of type dbl   0.4    0.3    0.2    0.1
                              0.1    0.2    0.3    0.4
                              -0.2   0.1    0.4    0.7
```

2. (b) Show that H is symmetric: $H = H^T$.

[7]:
```
H_inverse = solve(t(X) %*% X)
H_inverse
```

```
                              2.7    -0.7
A matrix: 2 × 2 of type dbl   -0.7   0.2
```

2. (c) Show that $H(I_n - H) = 0_n$, where 0_n is the zero matrix of size $n \times n$.**

[8]:
```
(X %*% H_inverse %*% t(X))^-1
```

```
                              1.428571    2.500000    10.000000   -5.000000
A matrix: 4 × 4 of type dbl   2.500000    3.333333    5.000000    10.000000
                              10.000000   5.000000    3.333333    2.500000
                              -5.000000   10.000000   2.500000    1.428571
```

2. (d) Stating that $\widehat{\mathbf{Y}}$ is uncorrelated with $\widehat{\varepsilon}$ is equivalent to showing that these vectors are orthogonal.* That is, we want their dot product to equal zero:

$$\widehat{\mathbf{Y}}^T \widehat{\varepsilon} = 0.$$

Prove this result. Also explain why being uncorrelated, in this case, is equivalent to the being orthogonal.

[9]:
```
print("Being uncorrelated is equivalent to being orthogonal because of the the␣
 ↪number of predictors is different.")
```

```
[1] "Being uncorrelated is equivalent to being orthogonal because of the the
number of predictors is different."
```

2.(e) Why is this result important in the practical use of linear regression?

[10]:
```
print("The line of best fit must show the average value of Y when all x's are␣
 ↪equal to zero.")
```

```
[1] "The line of best fit must show the average value of Y when all x's are
equal to zero."
```

2.1 Problem 3: Model Diagnosis

We here at the University of Colorado's Department of Applied Math love Bollywood movies. So, let's analyze some data related to them!

We want to determine if there is a linear relation between the amount of money spent on a movie (it's budget) and the amount of money the movie makes. Any venture capitalists among you will certianly hope that there is at least some relation. So let's get to modelling!

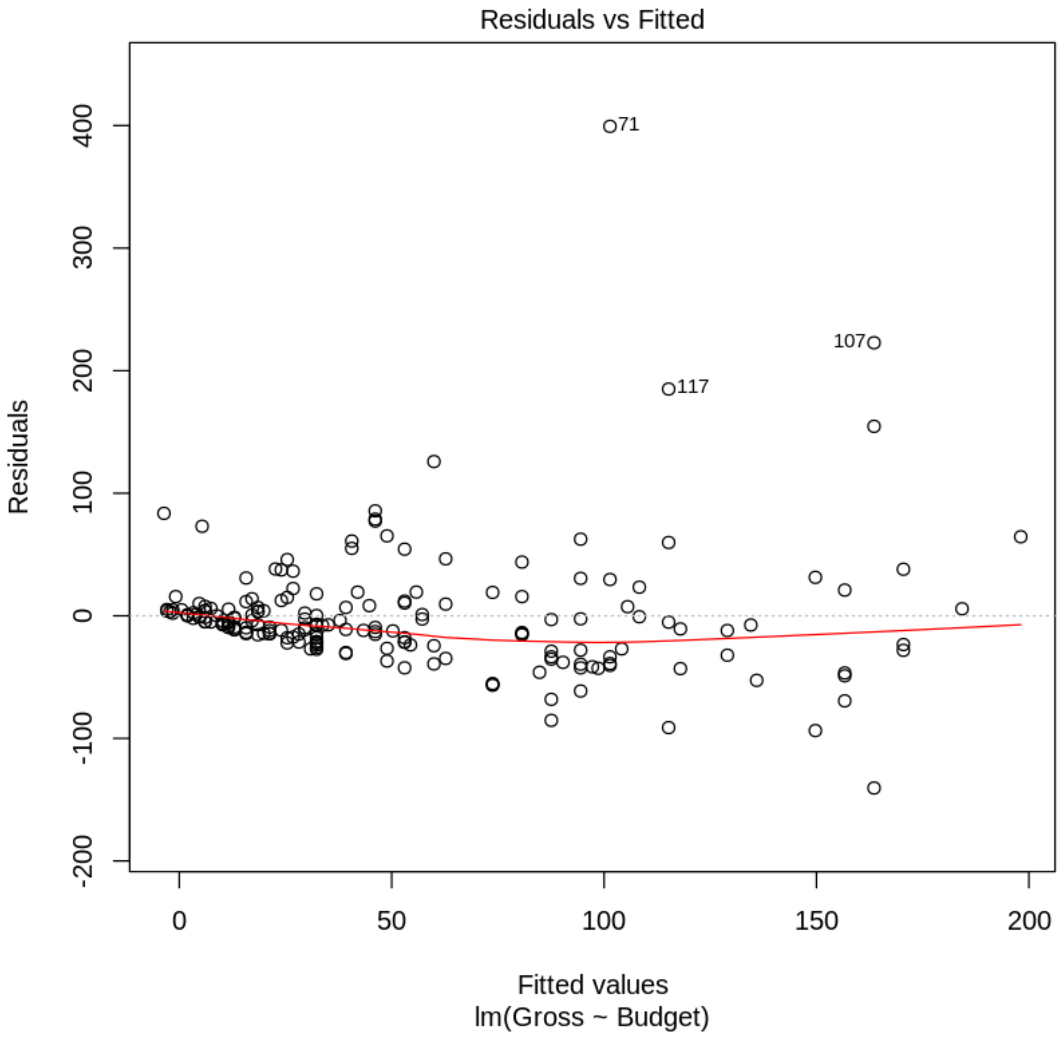
3. (a) Initial Inspection Load in the data from local directory and create a linear model with `Gross` as the response and `Budget` as the feature. The data is stored in the same local directory and is called `bollywood_boxoffice.csv`. Thank the University of Florida for this specific dataset.

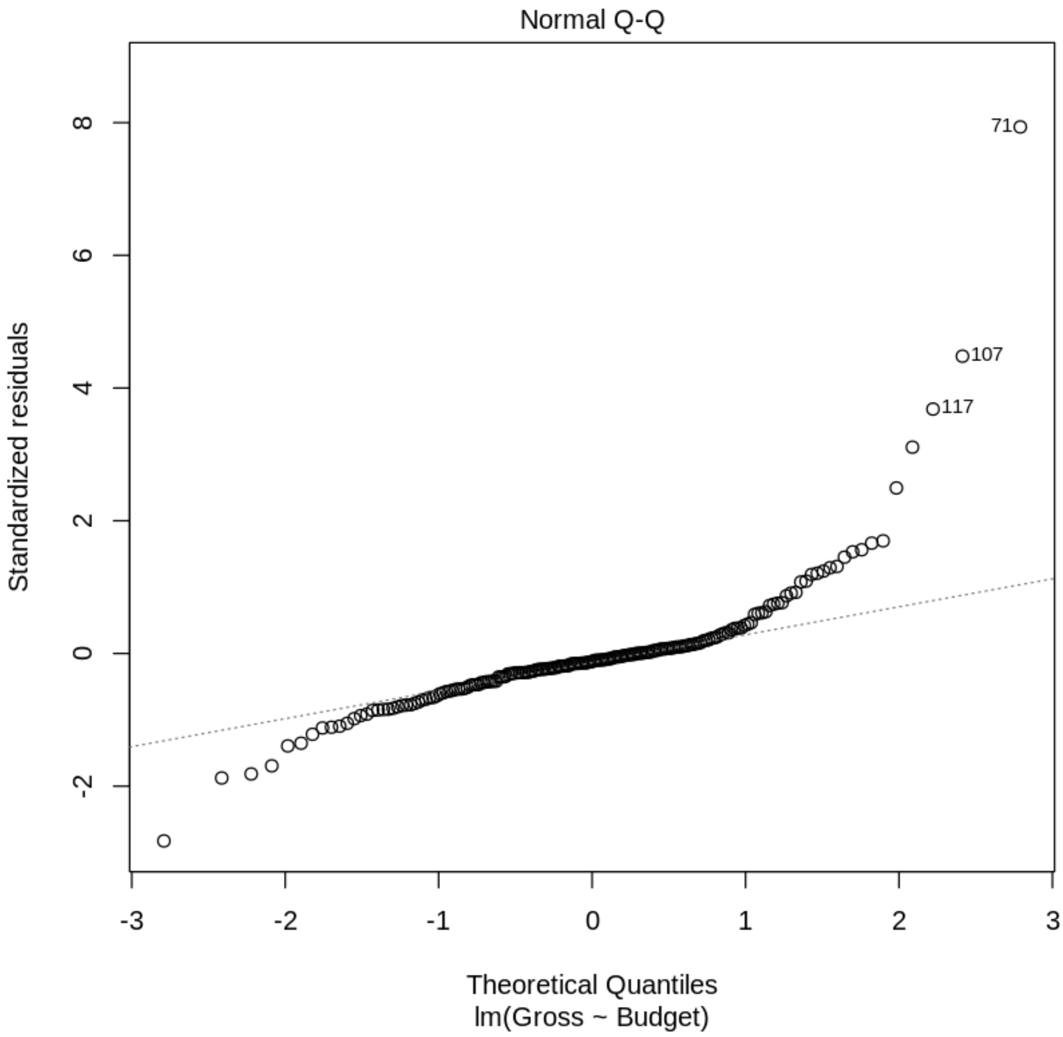
Specify whether each of the four regression model assumptions are being violated.

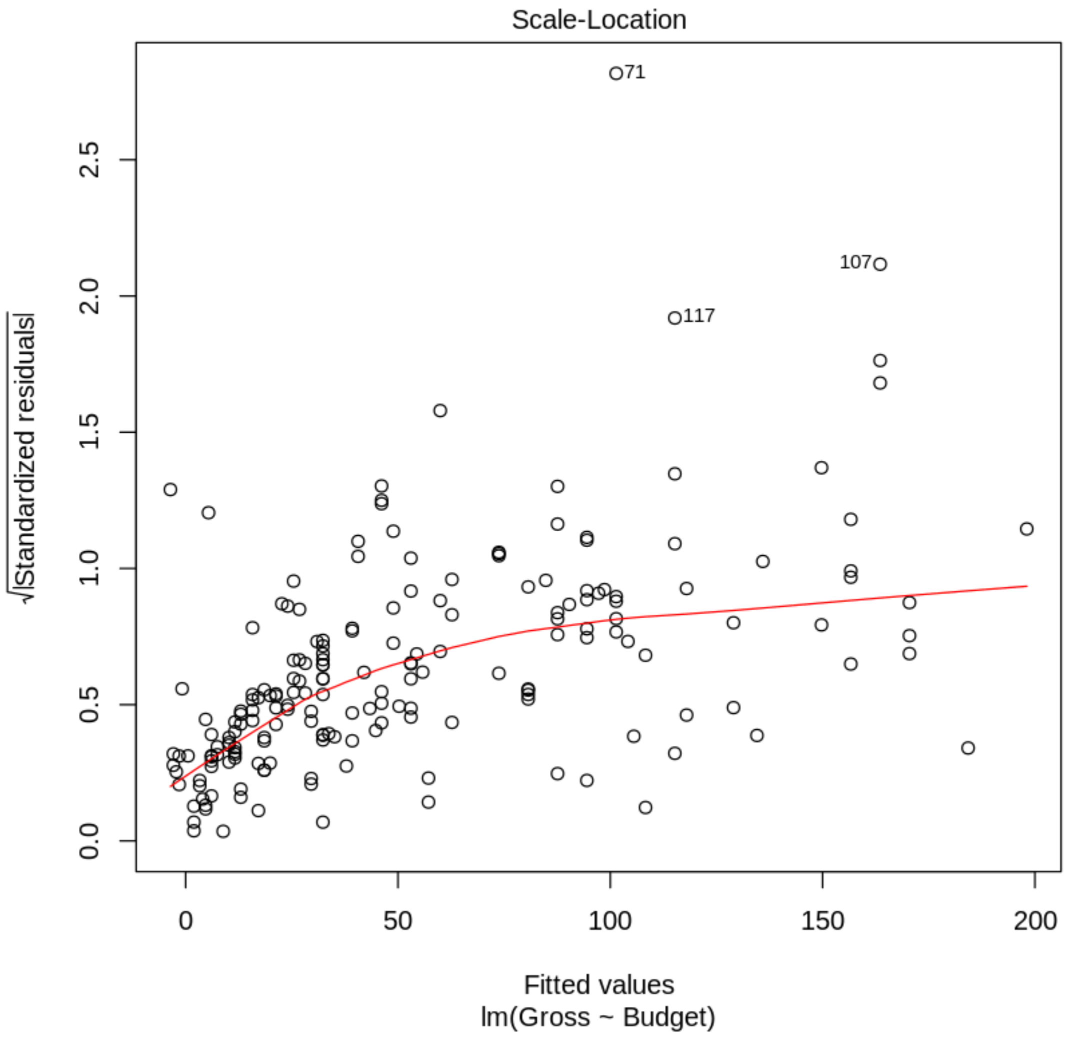
Data Source: http://www.bollymoviereviewz.com

[11]:
```
# Load the data
bollywood = read.csv("bollywood_boxoffice.csv")
summary(bollywood)

lm_bollywood <- lm(Gross ~ Budget, data = bollywood)
plot(lm_bollywood)
# Your Code Here


print("1. Linearity has been violated.
2. Independence has been violated.
3. Homoscedasticity has been violated.
4. Normality has been violated.")
```

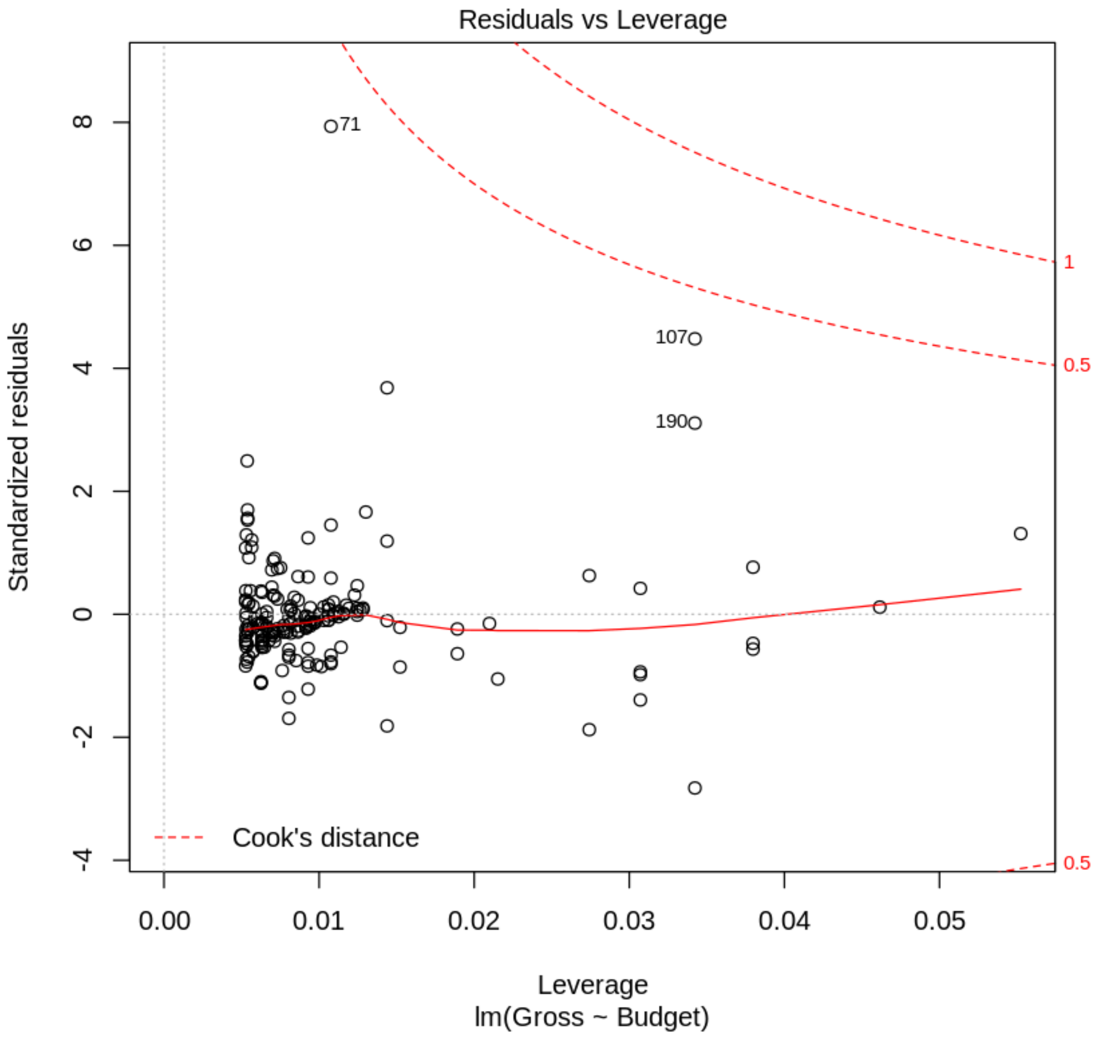
```
                  Movie          Gross            Budget
 1920London          :  1   Min.   :  0.63   Min.   :  4.00
 2 States\xa0        :  1   1st Qu.:  9.25   1st Qu.: 19.00
 24(Tamil,Telugu)    :  1   Median : 29.38   Median : 34.50
 Aashiqui 2          :  1   Mean   : 53.39   Mean   : 45.25
 AeDilHainMushkil\xa0:  1   3rd Qu.: 70.42   3rd Qu.: 70.00
 AGentleman          :  1   Max.   :500.75   Max.   :150.00
 (Other)             :184
```


Residuals vs Fitted
Residuals
400
300
200
100
0
-100
-200
0
50
100
150
200
71
107
117
Fitted values
lm(Gross ~ Budget)


Normal Q-Q
Standardized residuals
8
6
4
2
0
-2
-3
-2
-1
0
1
2
3
71
107
117
Theoretical Quantiles
lm(Gross ~ Budget)



```
[1] "1. Linearity has been violated.\n2. Independence has been violated.\n3.
Homoscedasticity has been violated.\n4. Normality has been violated."
```



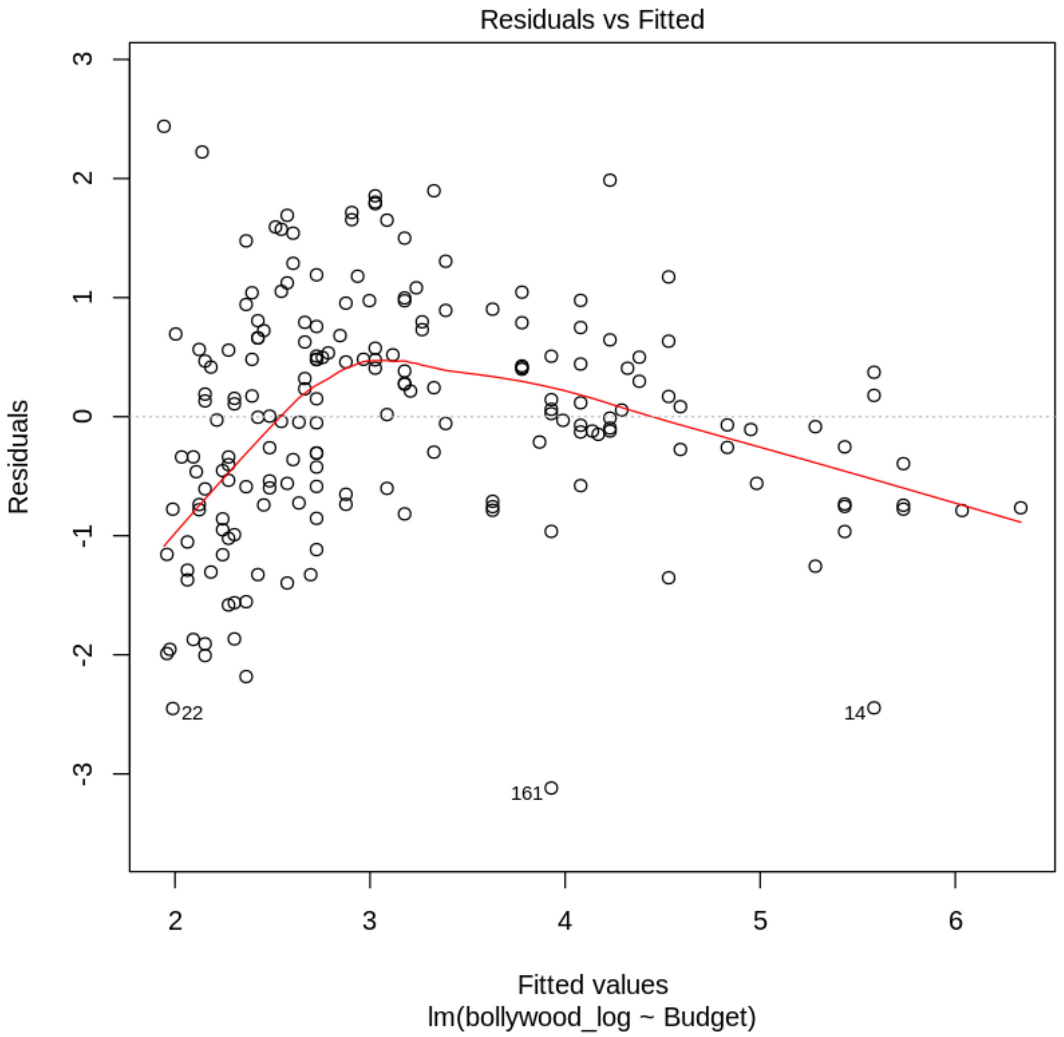
3. (b) Transformations Notice that the Residuals vs. Fitted Values plot has a 'trumpet" shape to it, the points have a greater spread as the Fitted value increases. This means that there is not a constant variance, which violates the homoskedasticity assumption.

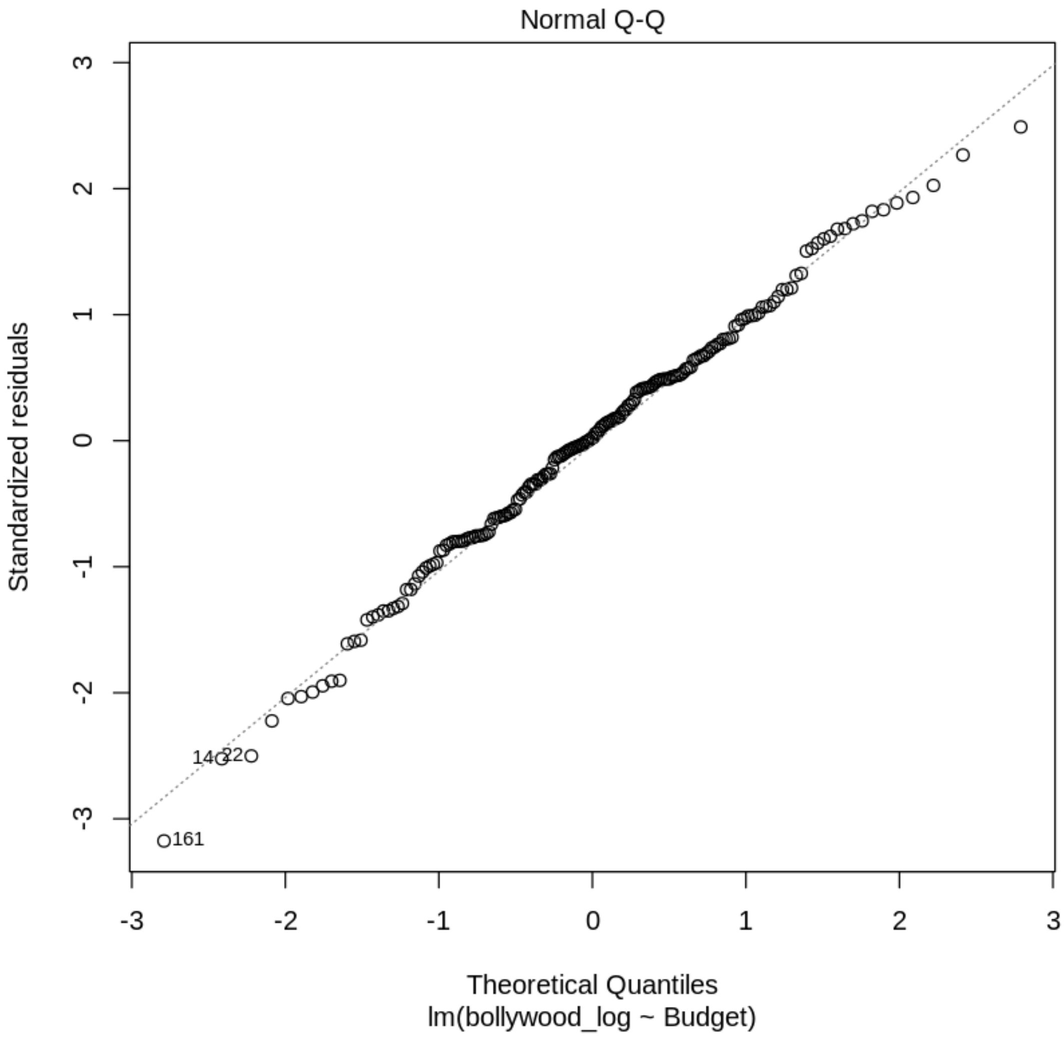
So how do we address this? Sometimes transforming the predictors or response can help stabilize the variance. Experiment with transfomrations on `Budget` and/or `Gross` so that, in the transformed scale, the relationship is approximately linear with a constant variance. Limit your transformations to square root, logarithms and exponentiation.

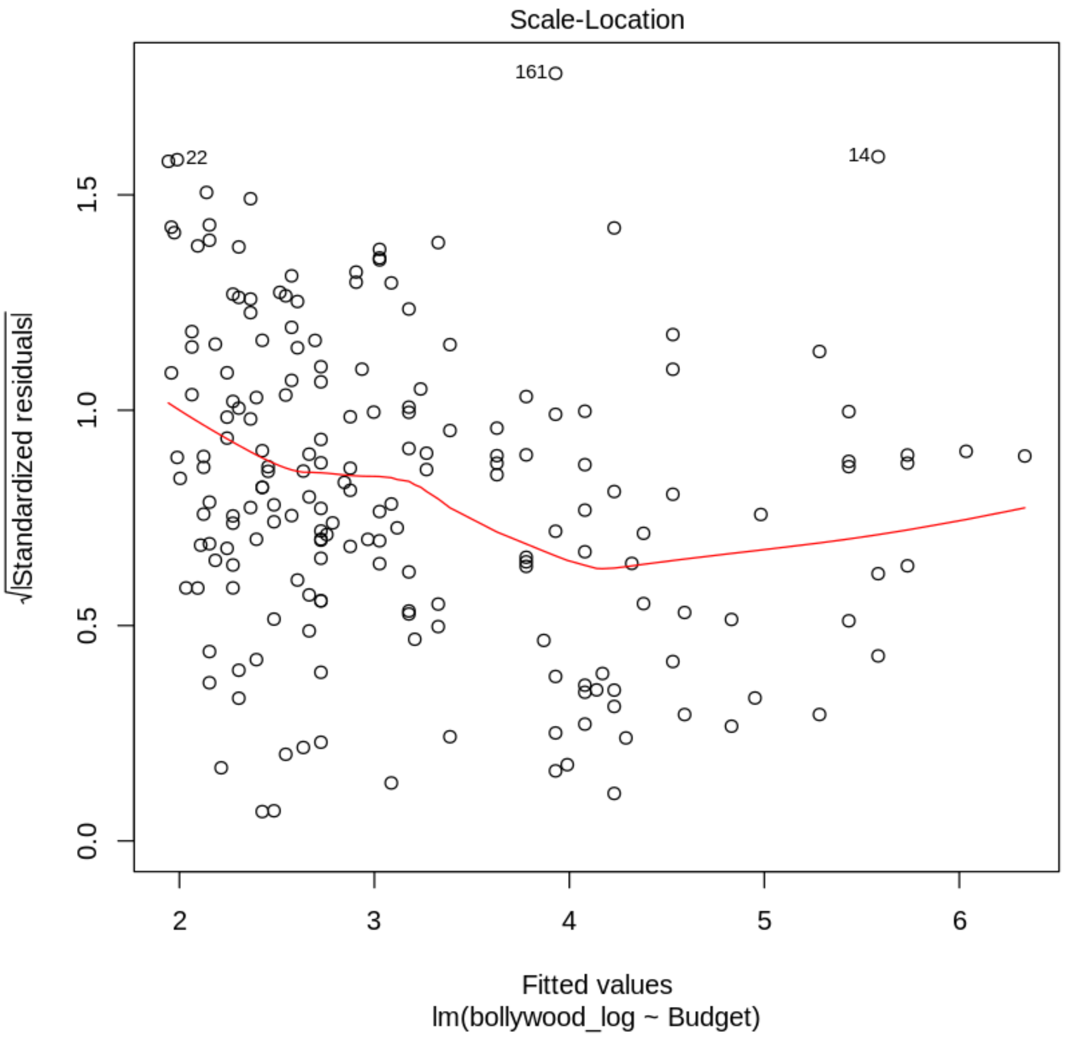
Note: There may be multiple transformations that fix this violation and give similar results. For the purposes of this problem, the transformed model doesn't have the be the "best" model, so long as it maintains both the linearity and homoskedasticity assumptions.

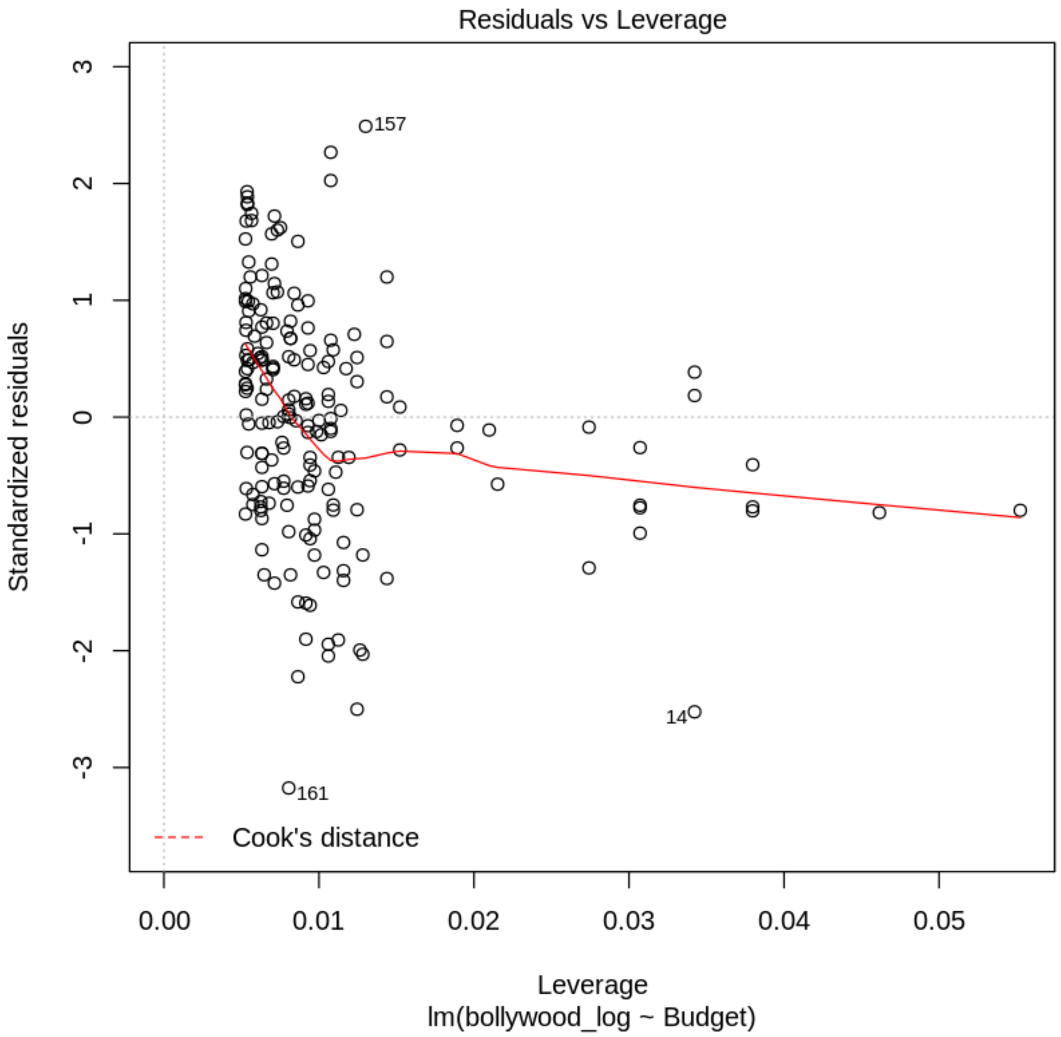
[12]:

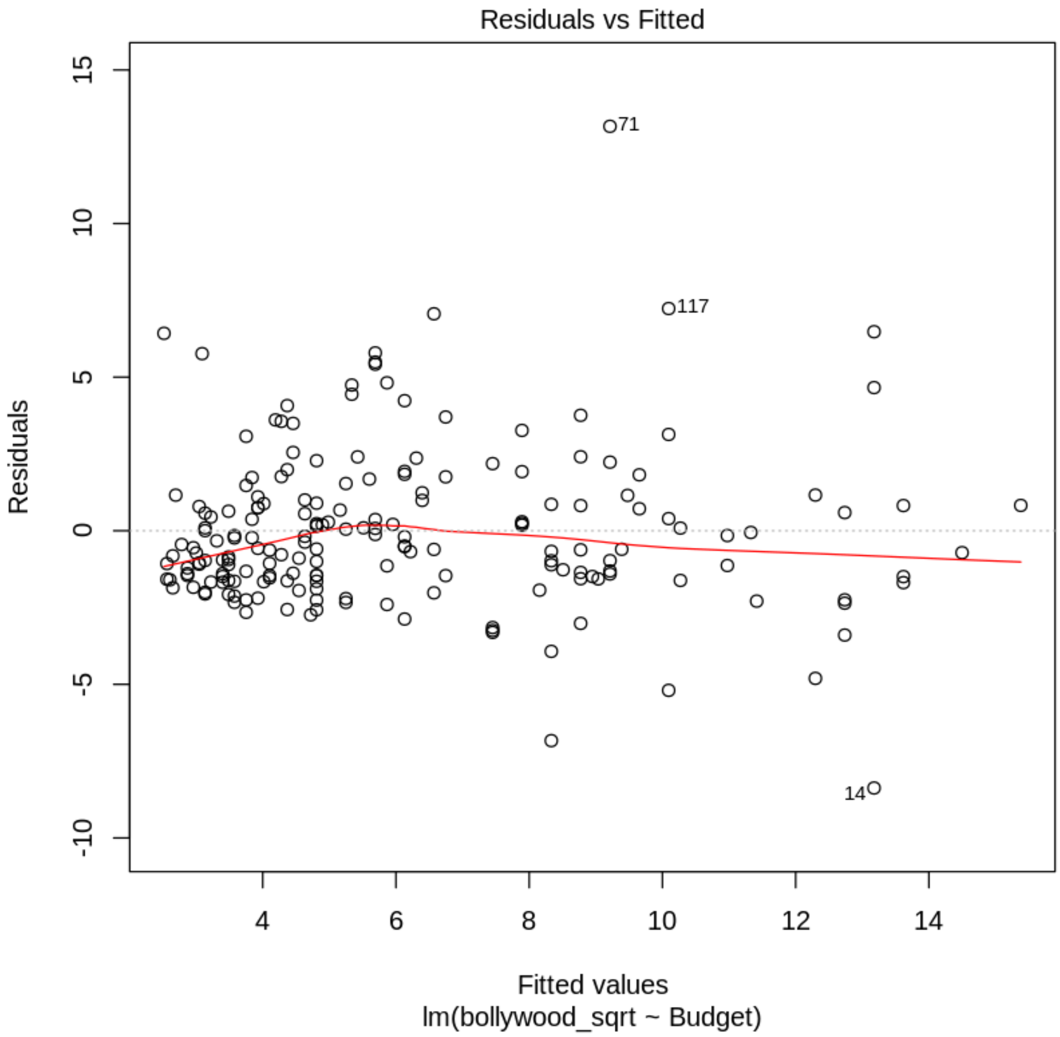
```r
bollywood_log <- log(bollywood$Gross)
log_model <- lm(bollywood_log ~ Budget, data = bollywood)
plot(log_model)

bollywood_sqrt <- sqrt(bollywood$Gross)
sqrt_model <- lm(bollywood_sqrt ~ Budget, data = bollywood)
plot(sqrt_model)

bollywood_exp <- exp(bollywood$Gross)
exp_model <- lm(bollywood_exp ~ Budget, data = bollywood)
plot(exp_model)

print("The log and square root transformations meet the linearity and␣
 ↪homoskedasticity assumptions effectively.")

# Your Code Here
```

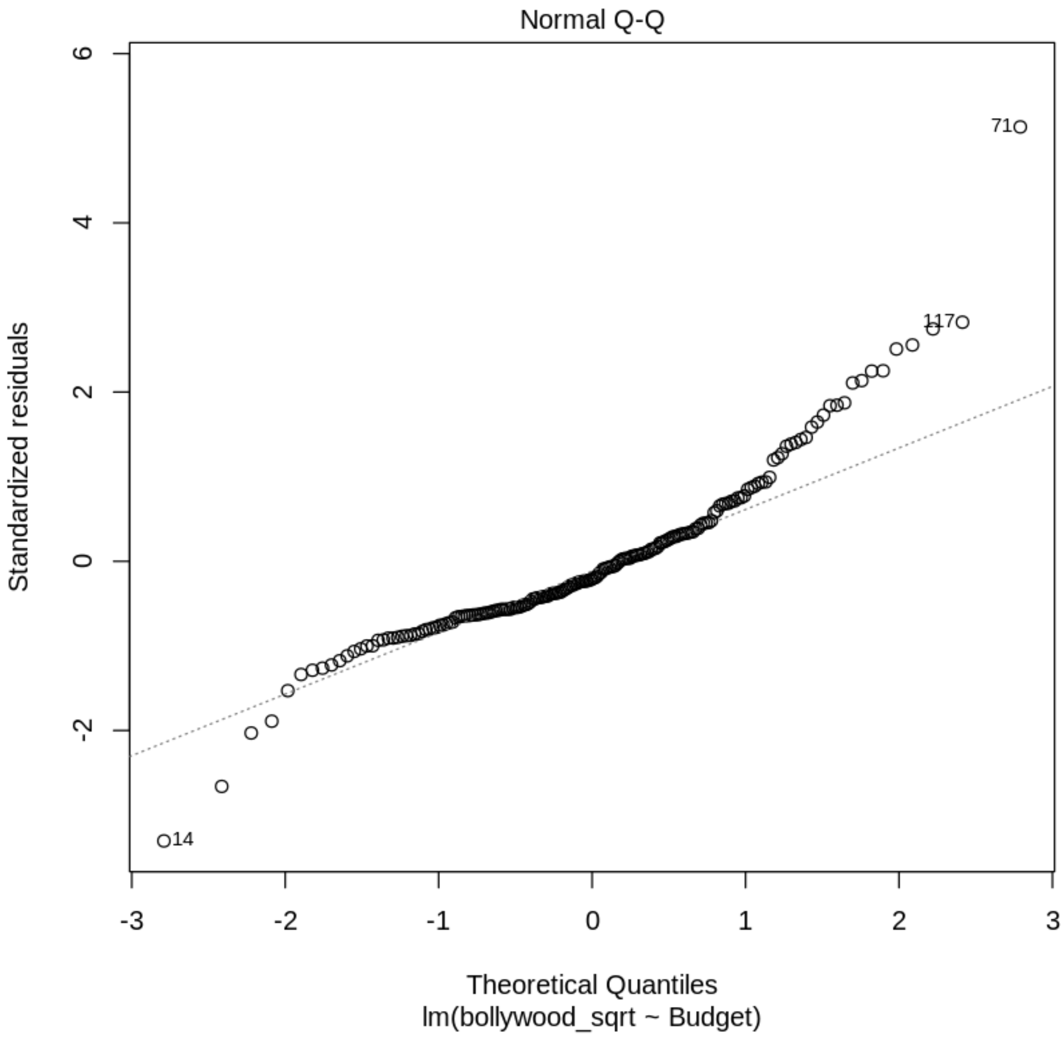

Residuals vs Fitted
Residuals
Fitted values
lm(bollywood_log ~ Budget)
3
2
1
0
-1
-2
-3
2
3
4
5
6
22
14
161

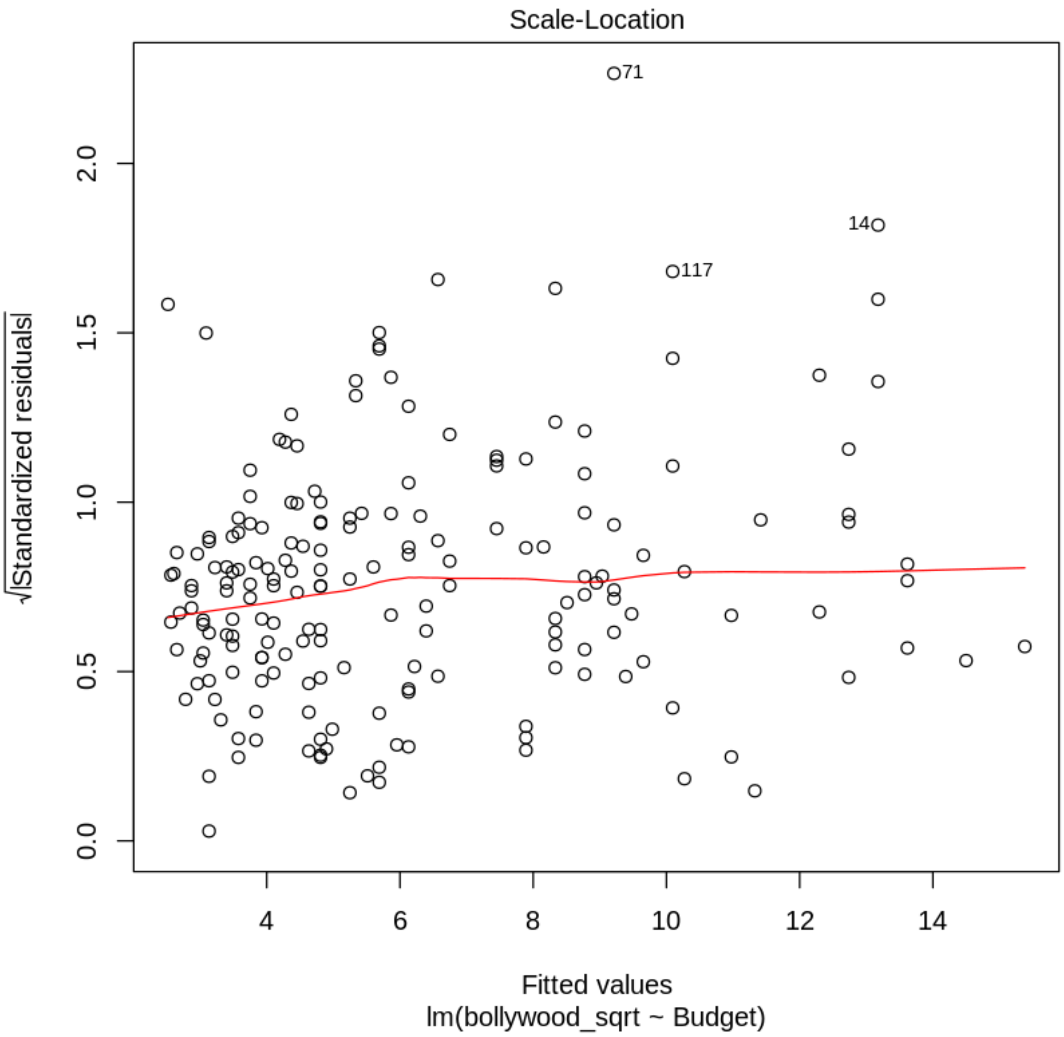

Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
161
14
22
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(bollywood_log ~ Budget)

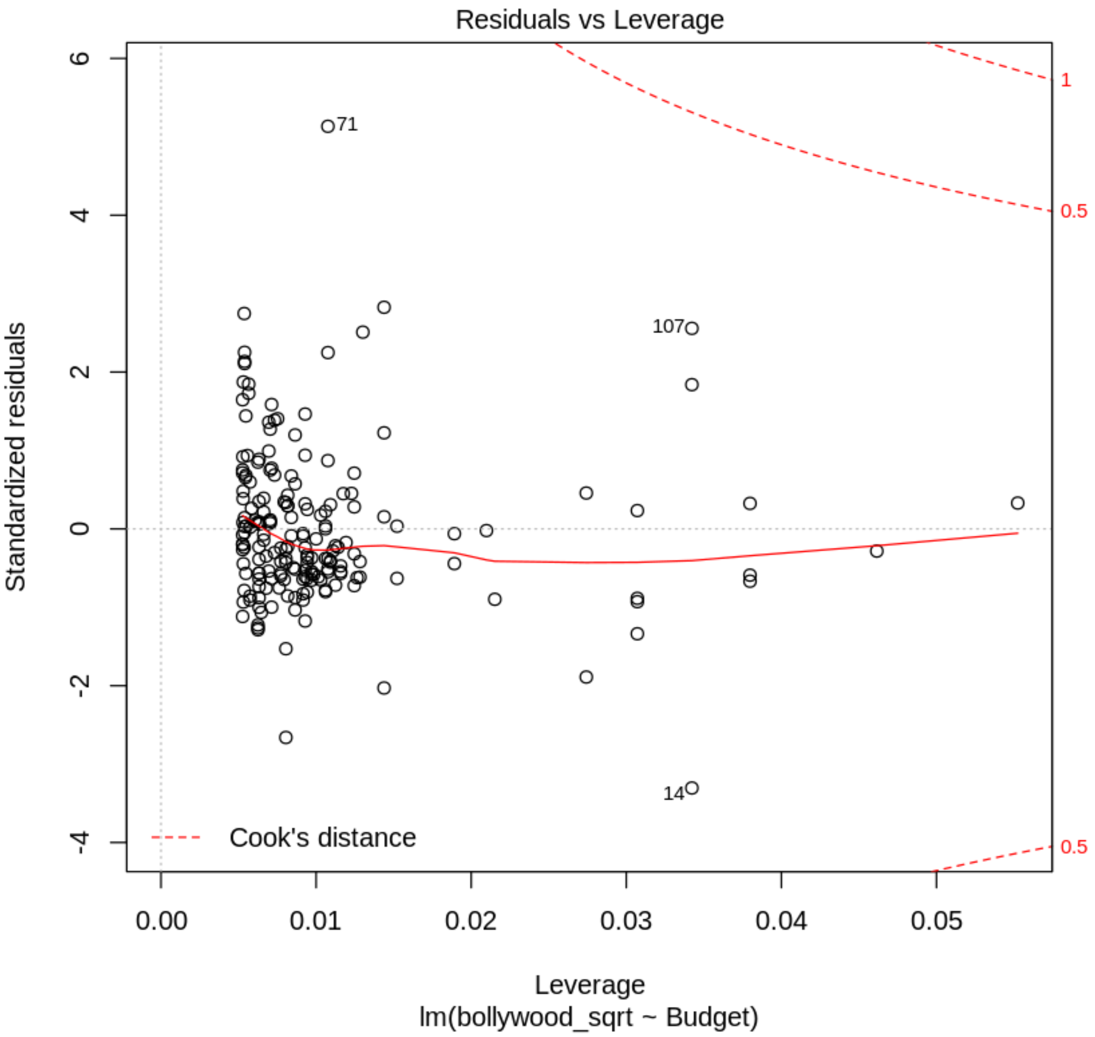

Scale-Location
161
22
14
1.5
1.0
0.5
0.0
√|Standardized residuals|
2
3
4
5
6
Fitted values
lm(bollywood_log ~ Budget)

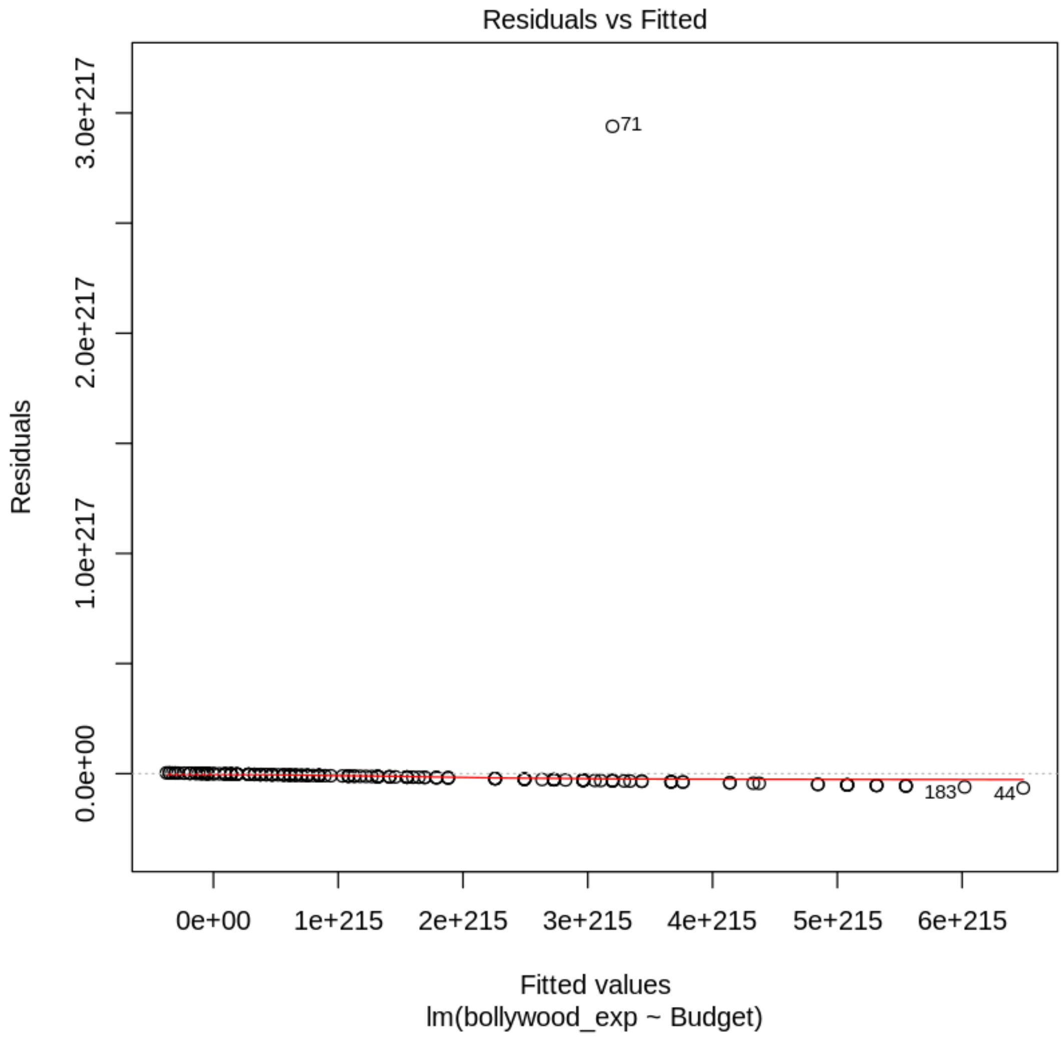

Residuals vs Leverage
Standardized residuals
3
2
1
0
-1
-2
-3
157
14
161
Cook's distance
0.00
0.01
0.02
0.03
0.04
0.05
Leverage
lm(bollywood_log ~ Budget)

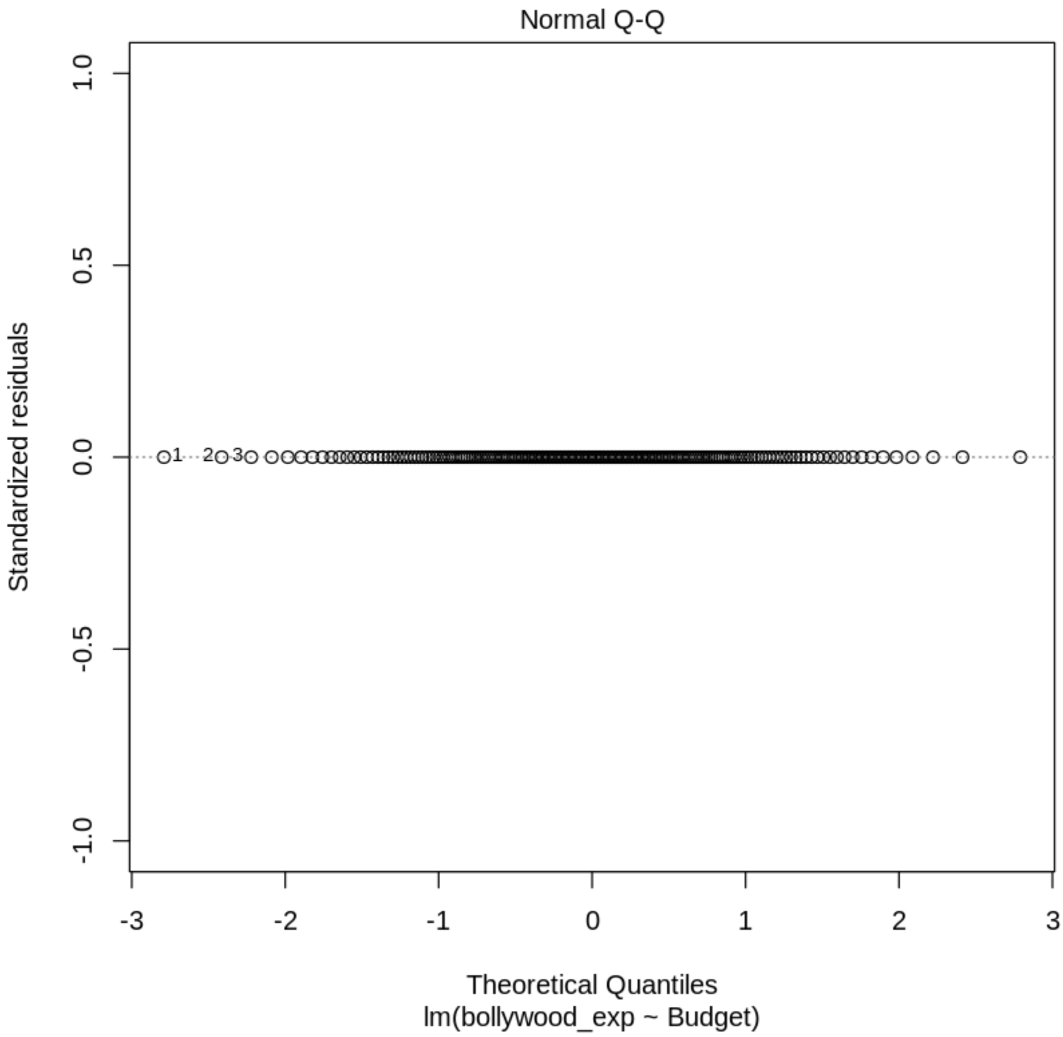

Residuals vs Fitted
Residuals
15
10
5
0
-5
-10
71
117
14
4
6
8
10
12
14
Fitted values
lm(bollywood_sqrt ~ Budget)

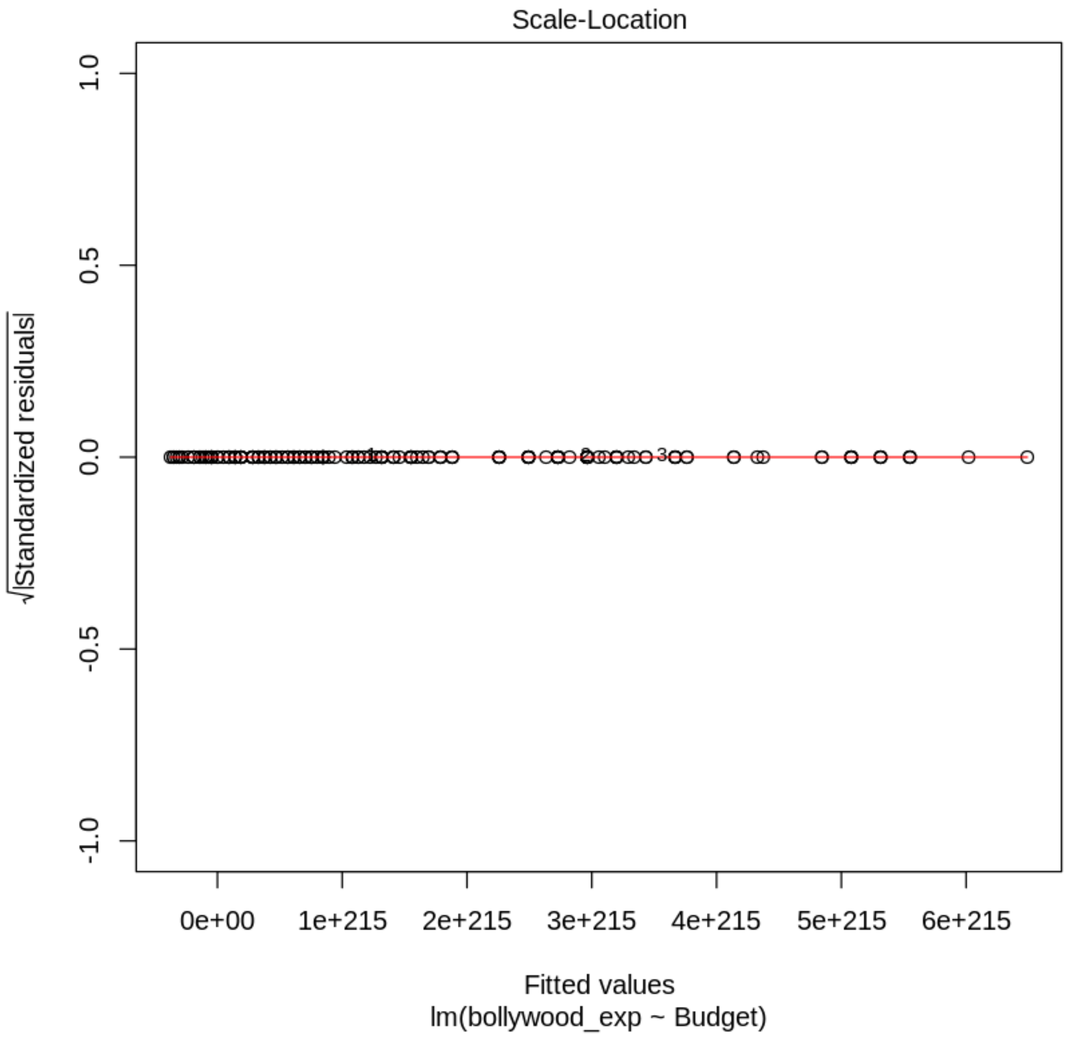

Normal Q-Q
Standardized residuals
6
4
2
0
-2
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(bollywood_sqrt ~ Budget)
71
117
14


Scale-Location
71
14
117
2.0
1.5
1.0
0.5
0.0
$\sqrt{|\text{Standardized residuals}|}$
4
6
8
10
12
14
Fitted values
lm(bollywood_sqrt ~ Budget)

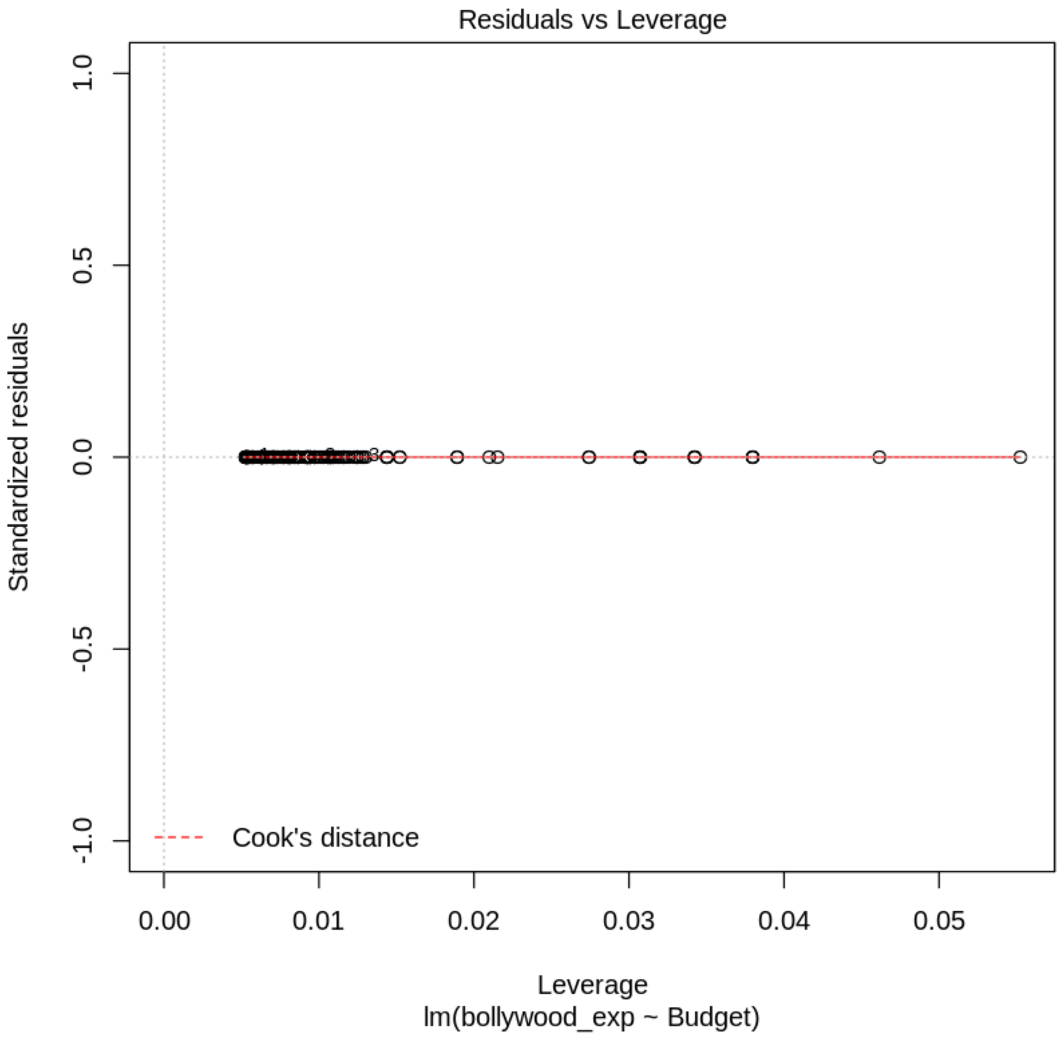

Residuals vs Leverage
Standardized residuals
6
4
2
0
-2
-4
0.00
0.01
0.02
0.03
0.04
0.05
Leverage
lm(bollywood_sqrt ~ Budget)
71
107
14
Cook's distance
1
0.5
0.5


Residuals vs Fitted
Residuals
3.0e+217
2.0e+217
1.0e+217
0.0e+00
71
183
44
0e+00
1e+215
2e+215
3e+215
4e+215
5e+215
6e+215
Fitted values
lm(bollywood_exp ~ Budget)


Normal Q-Q
Standardized residuals
1.0
0.5
0.0
-0.5
-1.0
-3
-2
-1
0
1
2
3
1
2
3
Theoretical Quantiles
lm(bollywood_exp ~ Budget)

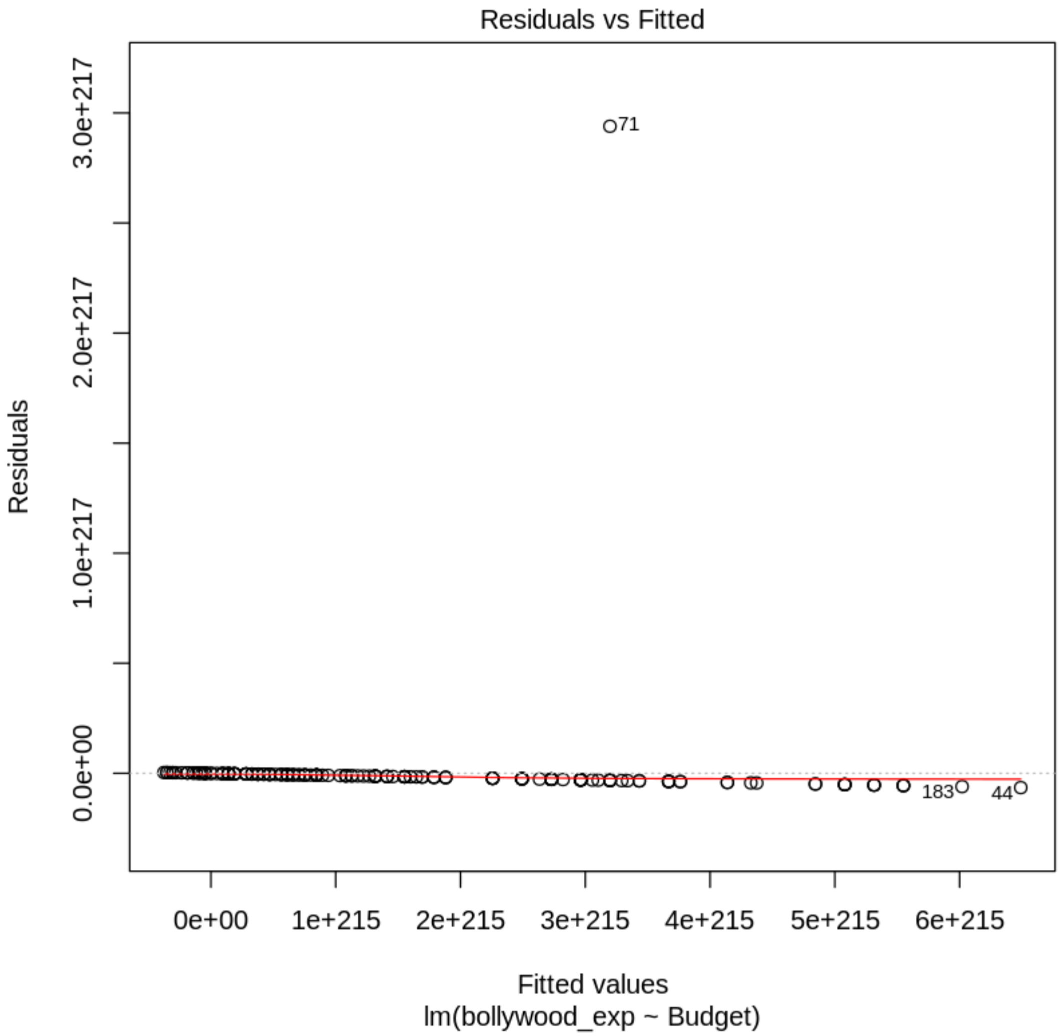


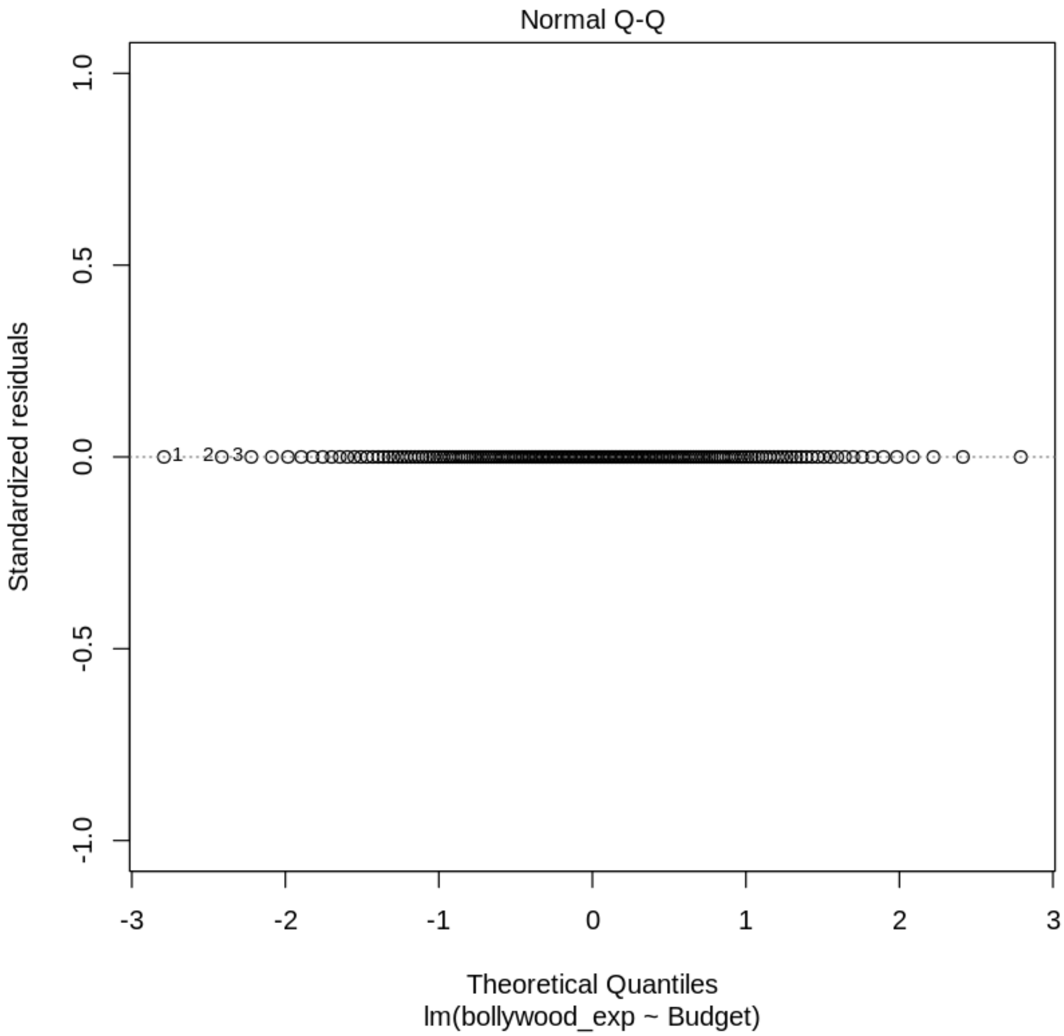
```
[1] "The log and square root transformations meet the linearity and
homoskedasticity assumptions effectively."
```

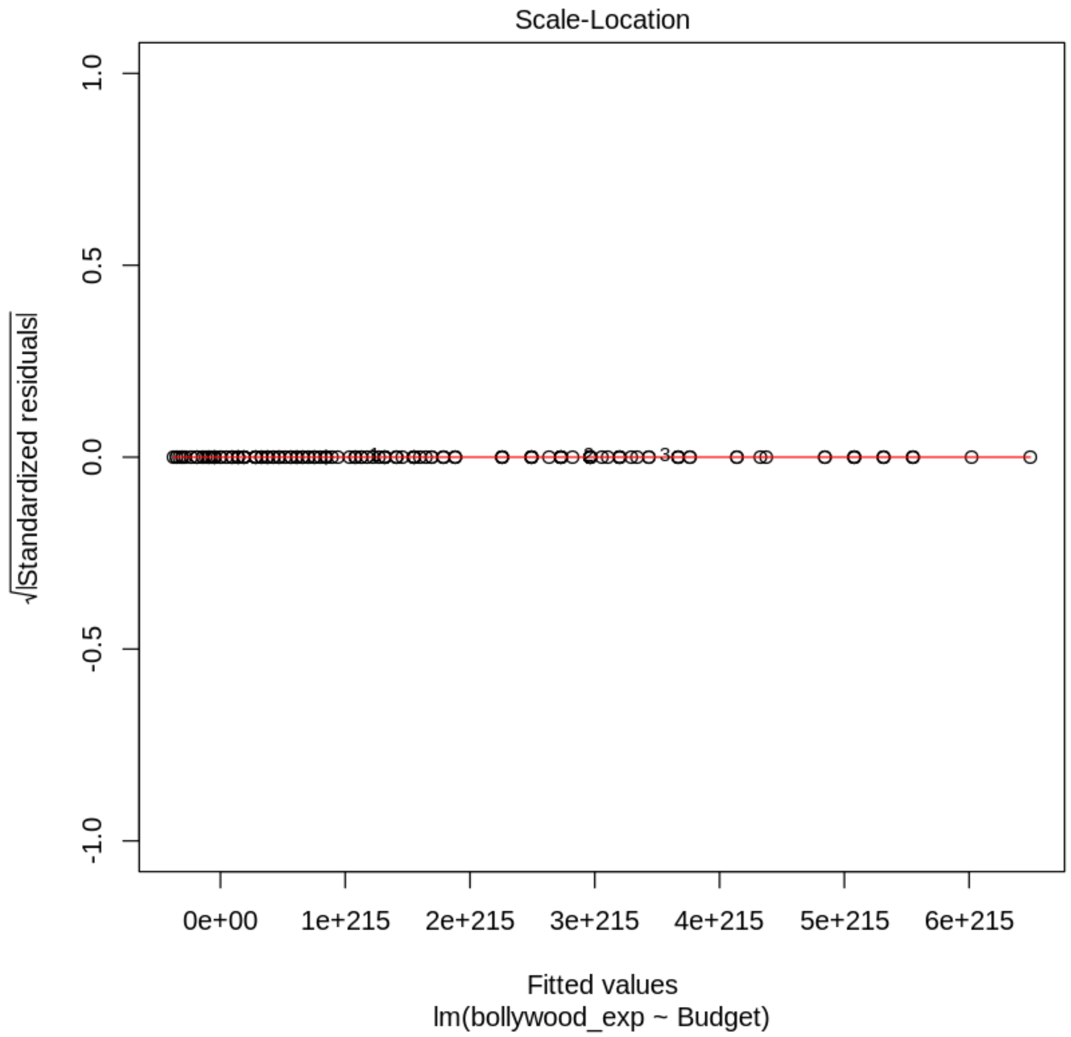


3. (c) Interpreting Your Transformation You've fixed the nonconstant variance problem! Hurray! But now we have a transformed model, and it will have a different interpretation than a normal linear regression model. Write out the equation for your transformed model. Does this model have an interpretation similar to a standard linear model?

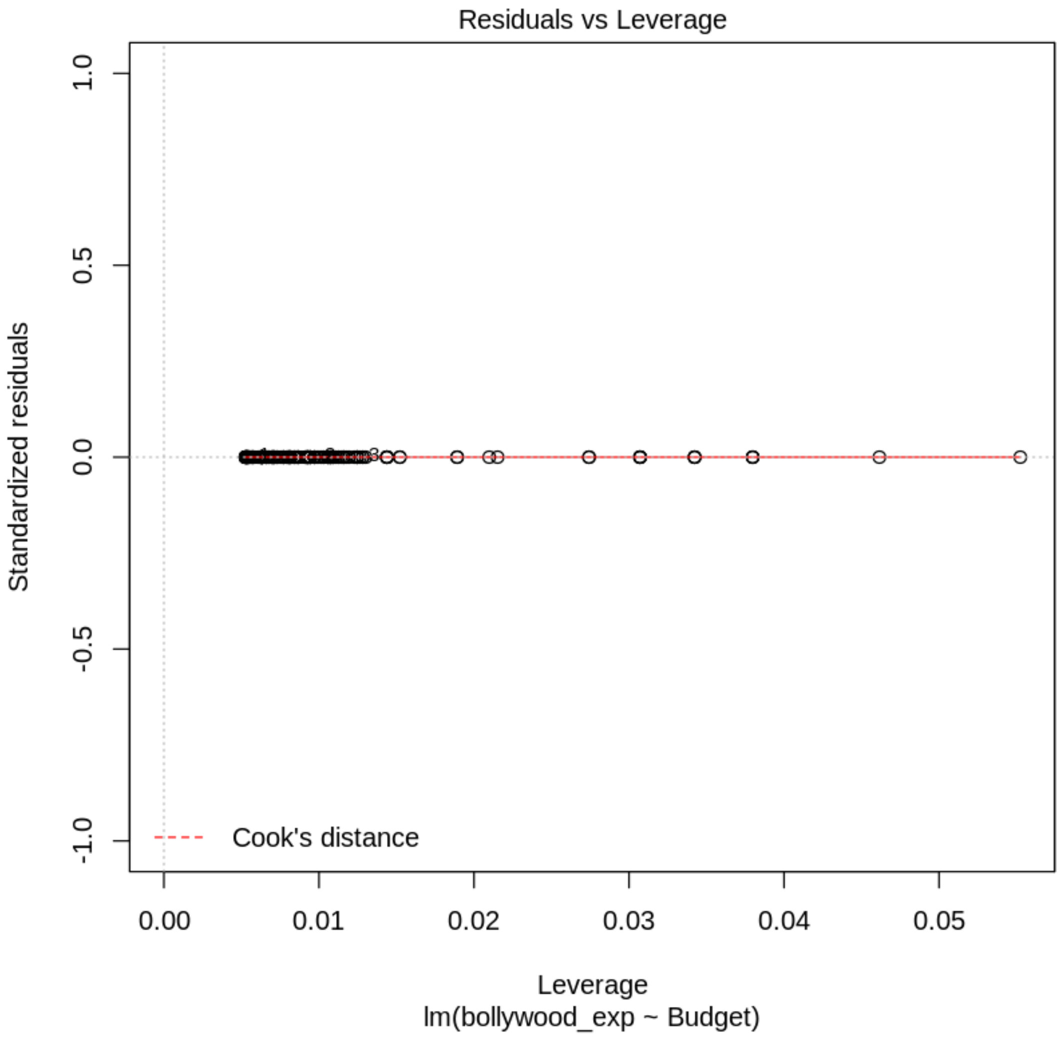
[13]:
```
bollywood_exp <- exp(bollywood$Gross)
exp_model <- lm(bollywood_exp ~ Budget, data = bollywood)
plot(exp_model)

print("The exp model is similar to a standard linear model.")
```


Residuals vs Fitted
Residuals
3.0e+217
2.0e+217
1.0e+217
0.0e+00
71
183
44
0e+00
1e+215
2e+215
3e+215
4e+215
5e+215
6e+215
Fitted values
lm(bollywood_exp ~ Budget)


Normal Q-Q
Standardized residuals
1.0
0.5
0.0
-0.5
-1.0
1
2
3
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(bollywood_exp ~ Budget)



```
[1] "The exp model is similar to a standard linear model."
```


Residuals vs Leverage
Standardized residuals
1.0
0.5
0.0
-0.5
-1.0
Cook's distance
0.00
0.01
0.02
0.03
0.04
0.05
Leverage
lm(bollywood_exp ~ Budget)

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```