HW week 12

w203: Statistics for Data Science

w203 teaching team

```
library(tidyverse)
library(ggplot2)


library(sandwich)
library(stargazer)
```

```
d <- load_and_clean(input = 'videos.txt')
```

```
## Rows: 9618 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (3): video_id, uploader, category
## dbl (6): age, length, views, rate, ratings, comments
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Regression analysis of YouTube dataset

You want to explain how much the quality of a video affects the number of views it receives on social media. In a world where people can now buy followers and likes, would such an investment increase the number of views that their content receives? **This is a causal question.**

You will use a dataset created by Cheng, Dale and Liu at Simon Fraser University. It includes observations about 9618 videos shared on YouTube. Please see this link for details about how the data was collected.

You will use the following variables:

- `views`: the number of views by YouTube users.
- `average_rating`: This is the average of the ratings that the video received, it is a renamed feature from `rate` that is provided in the original dataset. (Notice that this is different from `cout_of_ratings` which is a count of the total number of ratings that a video has received.
- `length`: the duration of the video in seconds.

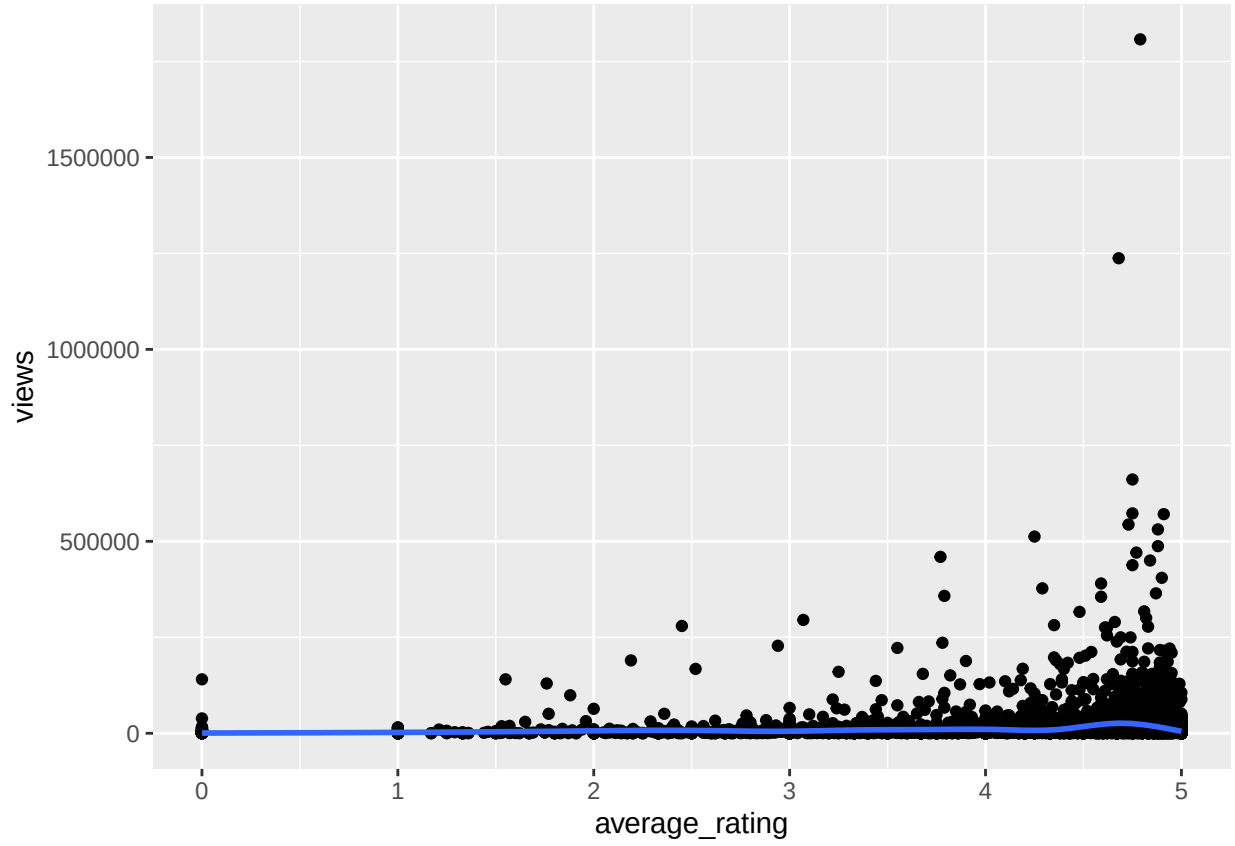
a. Perform a brief exploratory data analysis on the data to discover patterns, outliers, or wrong data entries and summarize your findings.

```
ggplot(d, aes(y=views, x=average_rating)) + geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 9 rows containing non-finite values (stat_smooth).
```

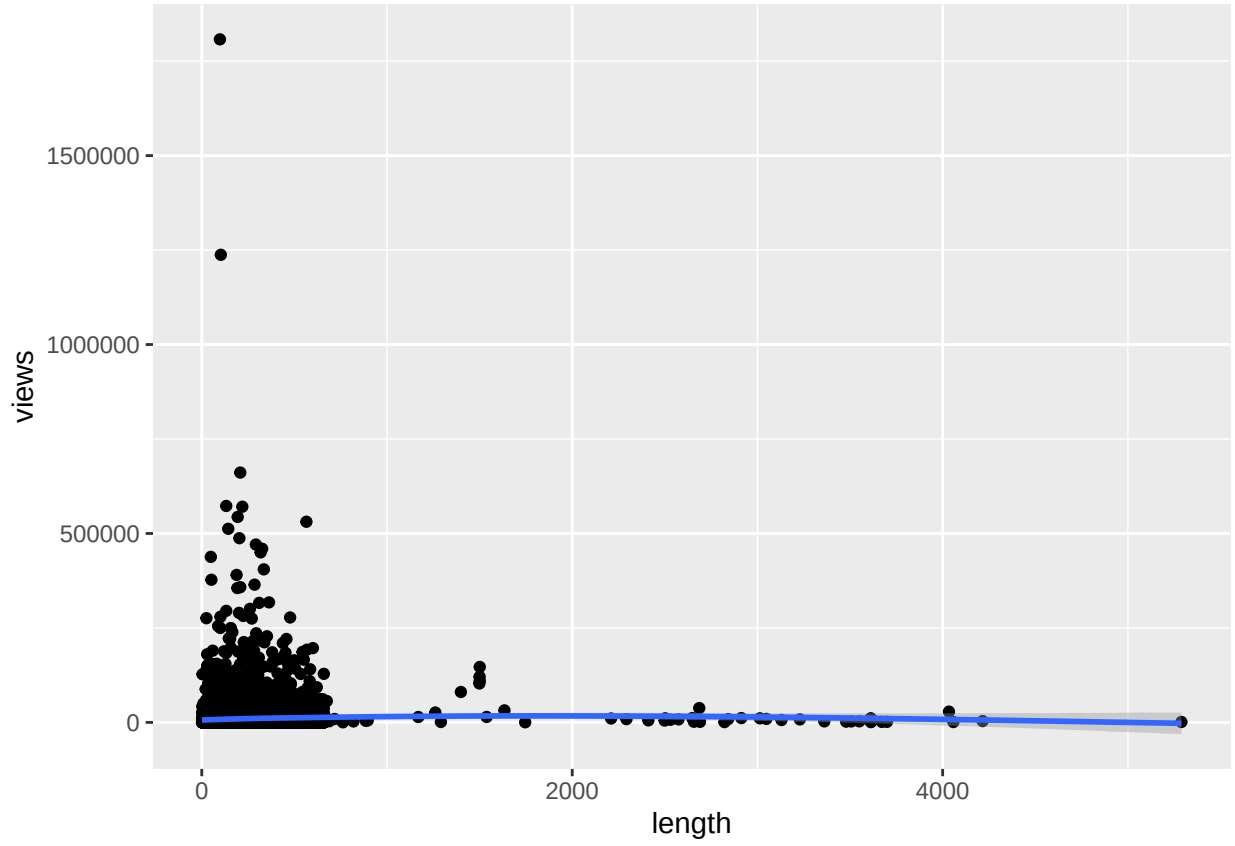
```
## Warning: Removed 9 rows containing missing values (geom_point).
```



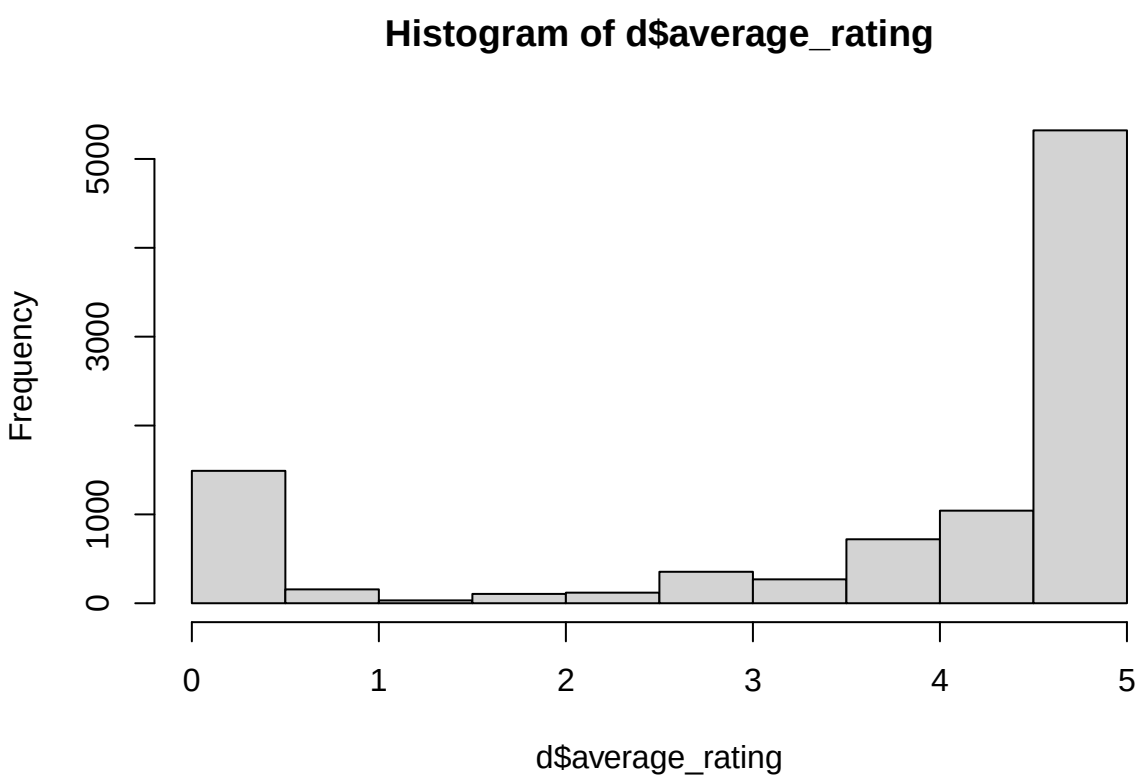

```
ggplot(d, aes(y=views, x=length)) + geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

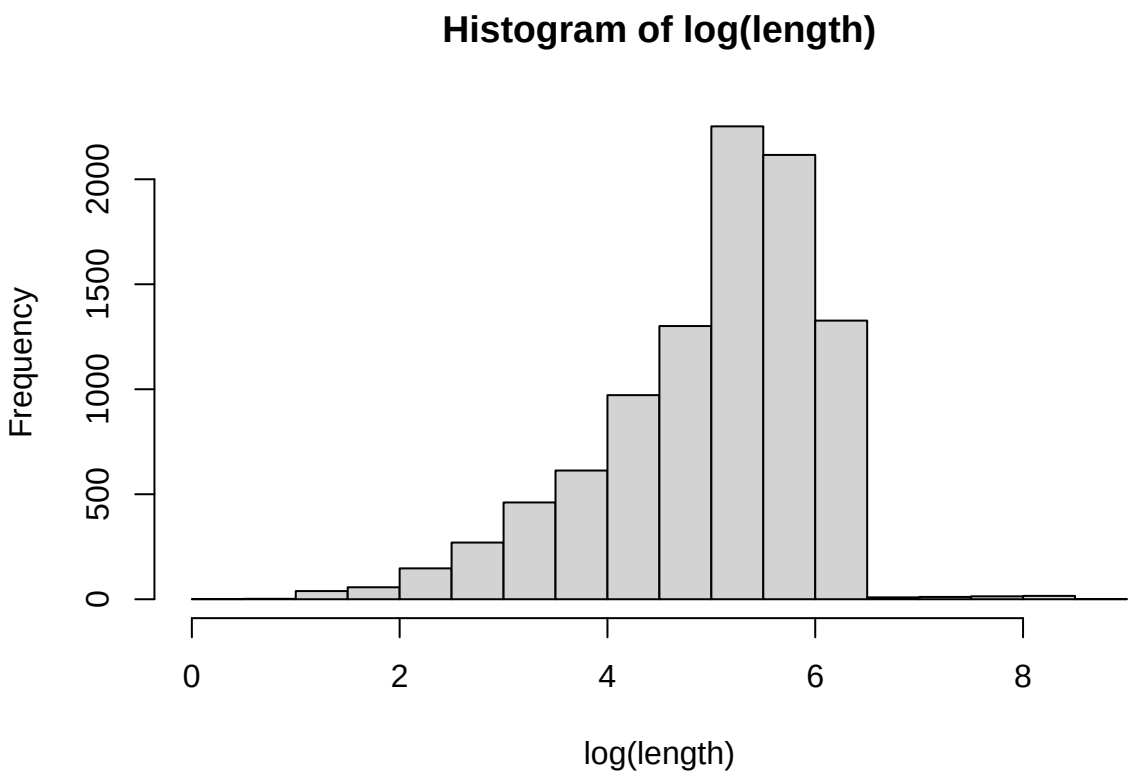
```
## Warning: Removed 9 rows containing non-finite values (stat_smooth).
## Removed 9 rows containing missing values (geom_point).
```




```
hist(d$average_rating)
```




```
attach(d)
hist(log(length))
```



- remove NAs as we do see some entries (only 9).

```
d2 <- na.omit(d)
#is.na(d)
nrow(d)
```

```
## [1] 9618
```

```
nrow(d2)
```

```
## [1] 9609
```

- Scatterplot of views and average_rating, we can observe visually that not many low-rating videos had high view counts, while more high-rated videos had higher view counts. However, there were also many videos with low view counts across all rating scores, which makes sense as there can be some videos with only a few ratings-
- Most videos got closer to zero views, so using the log(length) transformation which yields a more normal distribution will be more desirable for our model purposes.

b. Based on your EDA, select an appropriate variable transformation (if any) to apply to each of your three variables. You will fit a model of the type,

$$f(\text{views}) = \beta_0 + \beta_1 g(\text{rate}) + \beta_3 h(\text{length})$$

Where f, g and h are sensible transformations, which might include making *no* transformation.

```
model <- lm(log(views) ~ average_rating + log(length), data = d2)

 stargazer(
```

```
  model,
  type = 'text',
  se = list(get_robust_se(model))
  )
```

```
##
## ================================================
##                          Dependent variable:
##                      ---------------------------
##                                log(views)
## ------------------------------------------------
## average_rating                 0.467***
##                                 (0.010)
##
## log(length)                    0.105***
##                                 (0.018)
##
## Constant                       5.010***
##                                 (0.088)
##
## ------------------------------------------------
## Observations                     9,609
## R2                               0.189
## Adjusted R2                      0.189
## Residual Std. Error       1.799 (df = 9606)
## F Statistic          1,121.899*** (df = 2; 9606)
## ================================================
## Note:                *p<0.1; **p<0.05; ***p<0.01
```

c. Using diagnostic plots, background knowledge, and statistical tests, assess all five assumptions of the CLM. When an assumption is violated, state what response you will take. As part of this process, you should decide what transformation (if any) to apply to each variable. Iterate against your model until your satisfied that at least four of the five assumption have been reasonably addressed.

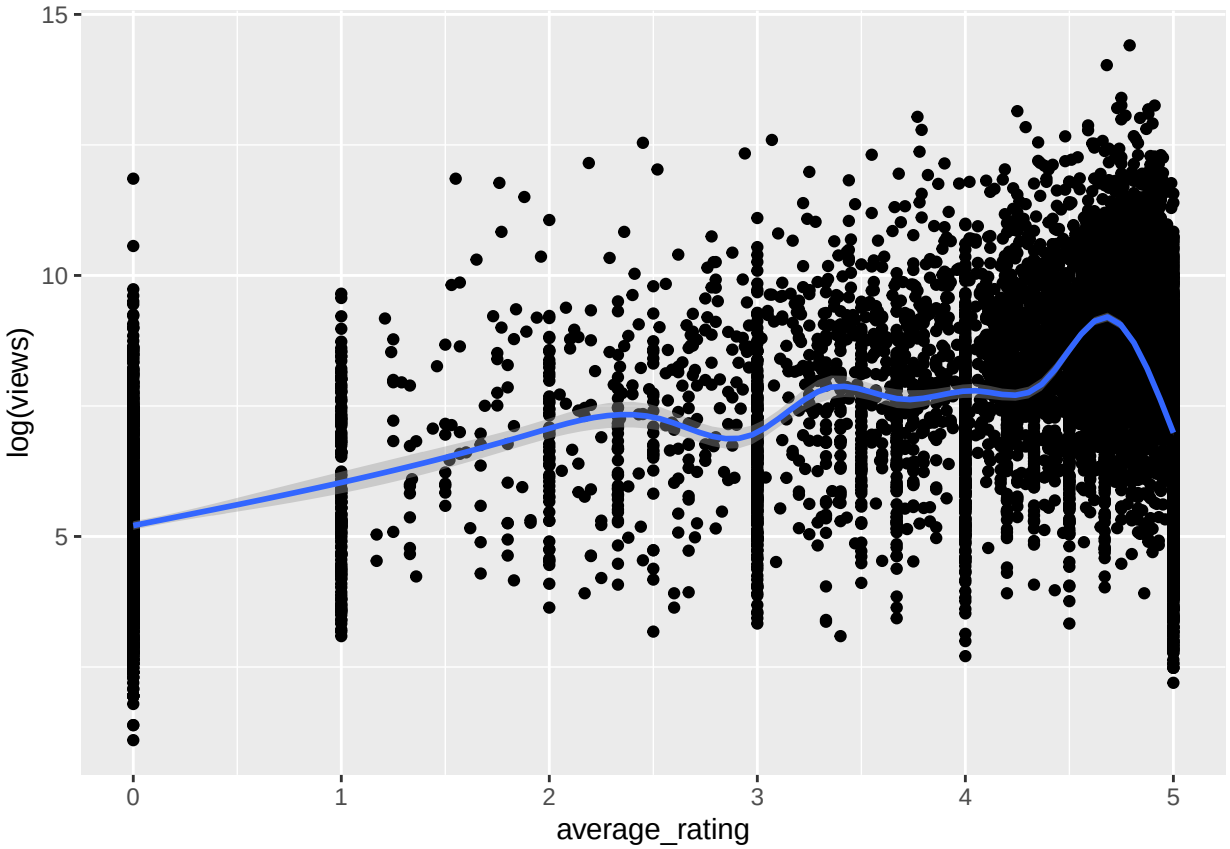
1. **IID Data:**

- Not IID as there are many entries that are by the same uploader, so it's likely that some of these videos may have similar quality (affecting ratings, length, etc.) as well as similar view counts. New iteration of model –> can we group-by the uploader and calculate the average ratings, views, length, etc. of all of the uploader's videos?
- Besides that, we can assume some clustering between different categories of video which are hard to fully isolate from one another.

2. **No Perfect Colinearity:** Very low correlation between the input variables (average_rating and log(length)).
3. **Linear Conditional Expectation:** Residuals should be evenly distributed across the model prediction range - Indicating a linear conditional expectation for all variables. Observing the scatterplot between our y-variable (log(views)) and our x-variables (average_rating, log(length)) we can see that the residuals distribution appears to be linear.

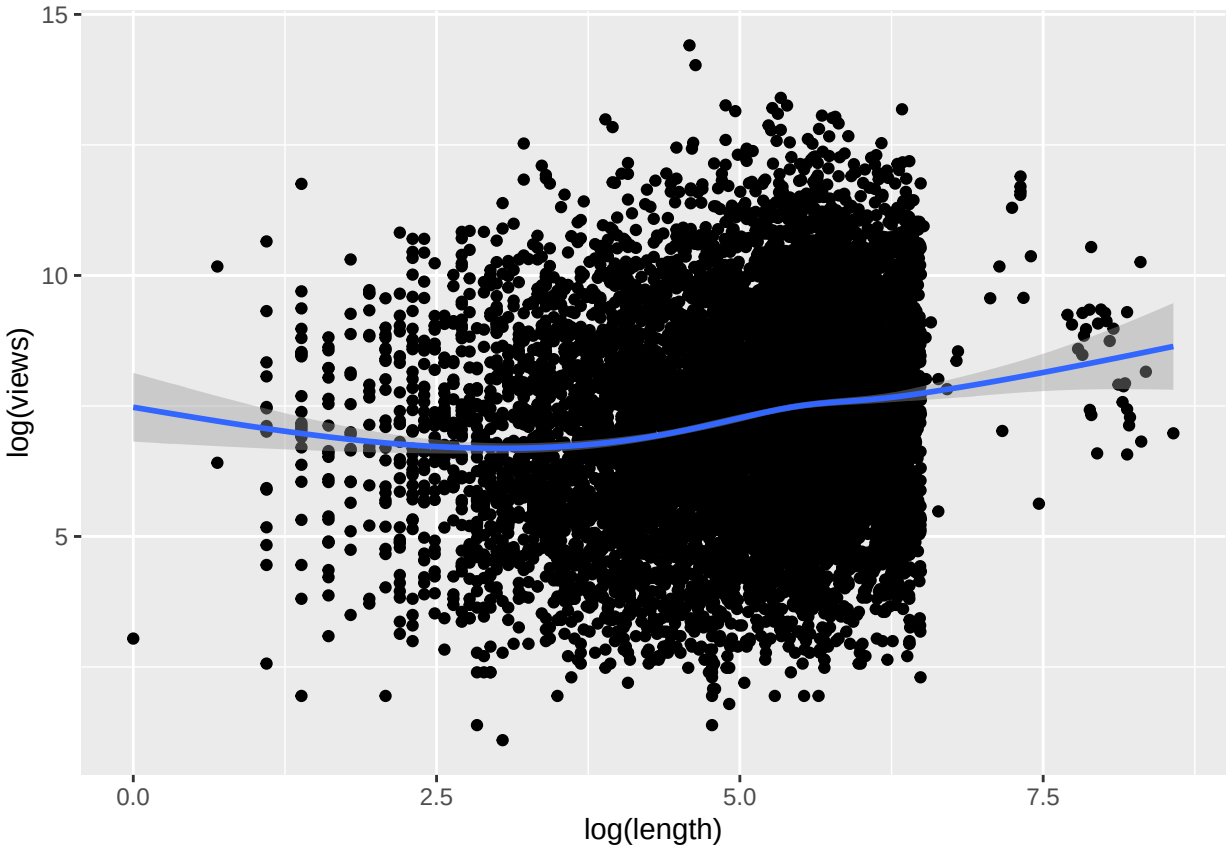
```
## remove this commented block and write code that can help you assess whether
## your model satisfied the requirement of a linear conditional expectation.
ggplot(d2, aes(y=log(views), x=average_rating)) + geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
ggplot(d2, aes(y=log(views), x=log(length))) + geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

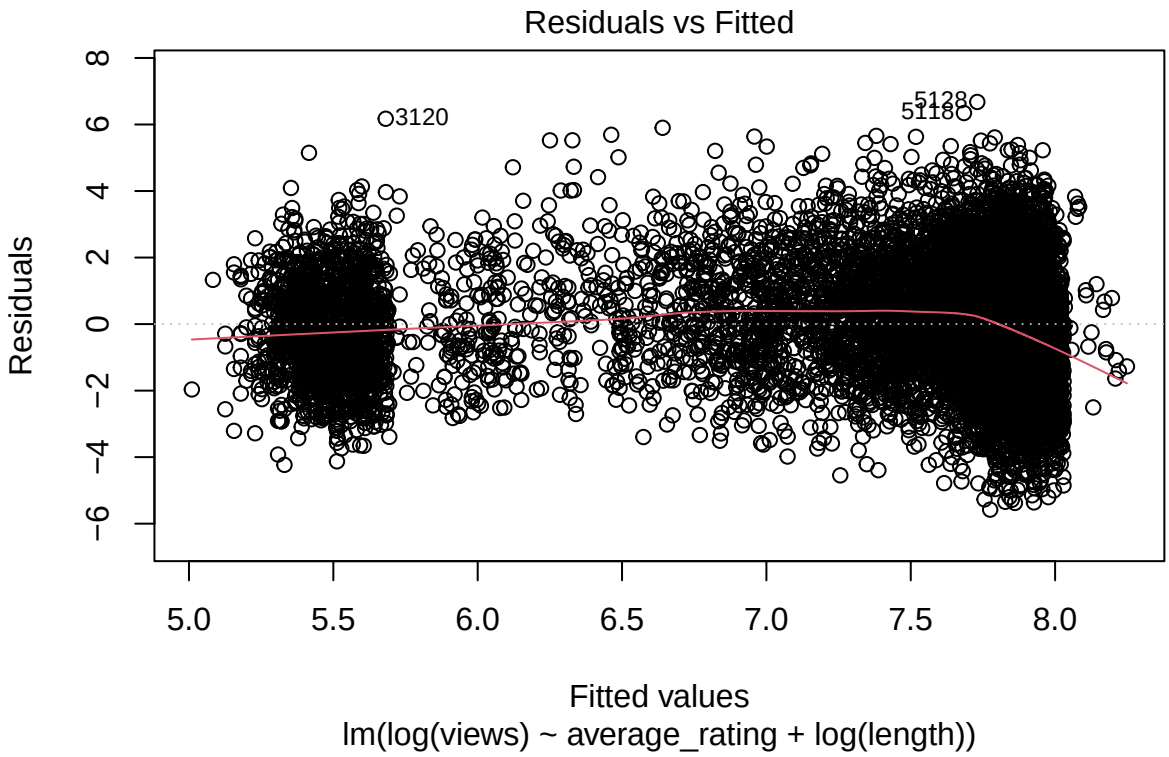


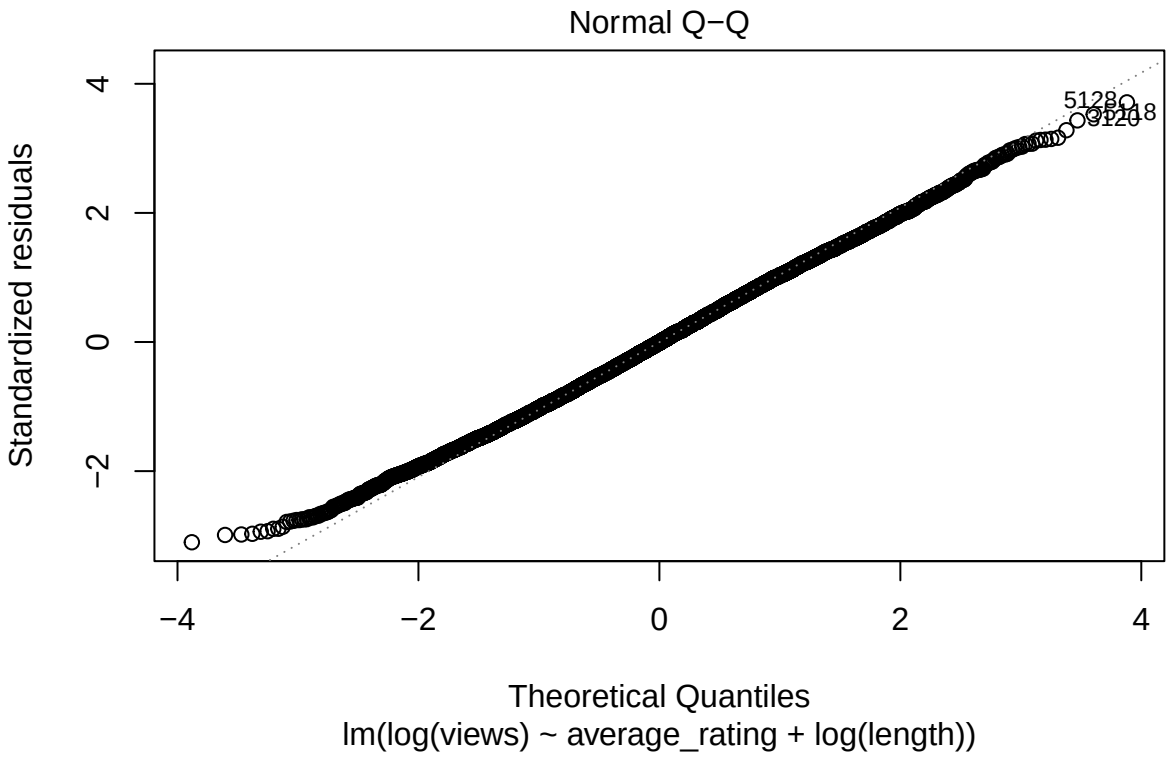
4. **Homoskedastic Errors:** 'Replace this quote with what you are looking for when you are assessing homoskedastic errors. Also include what you observe in your plot. In the model that you have chosen to report, do you satisfy the assumption of homoskedastic errors?'

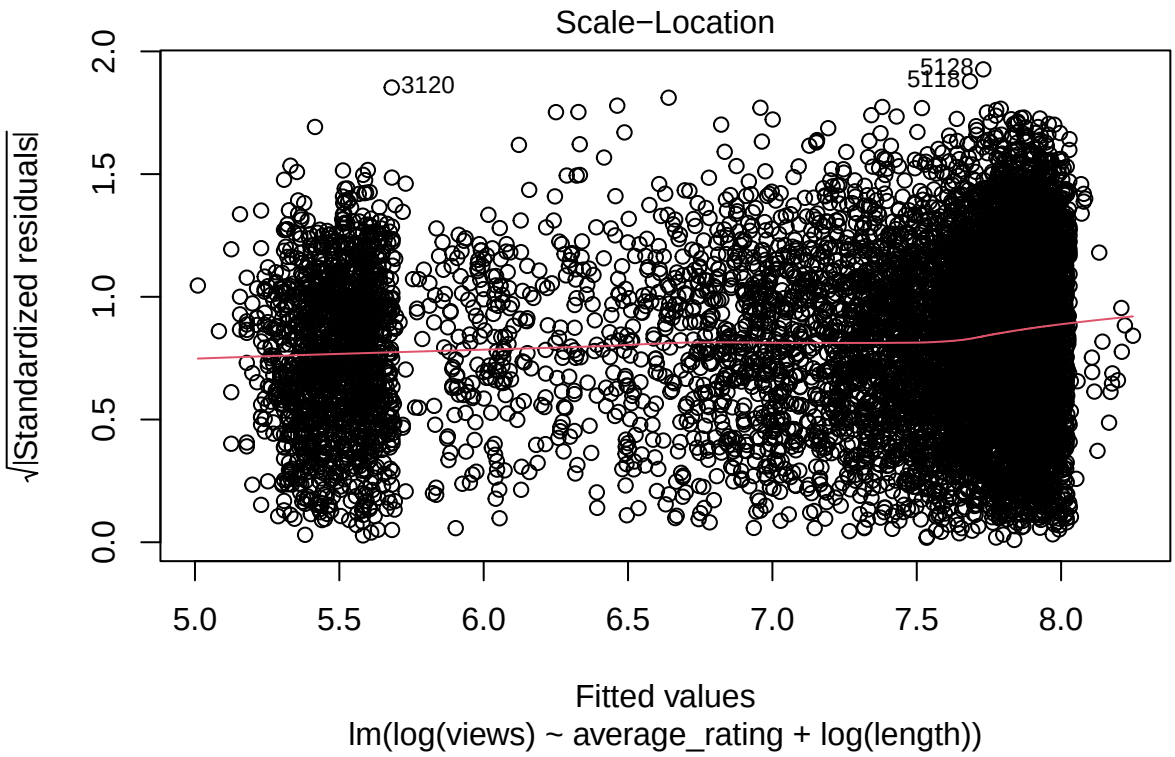
```
## remove this commented block and write code that can help you assess whether
## your model satisfied the requirement of homoskedastic errors
```

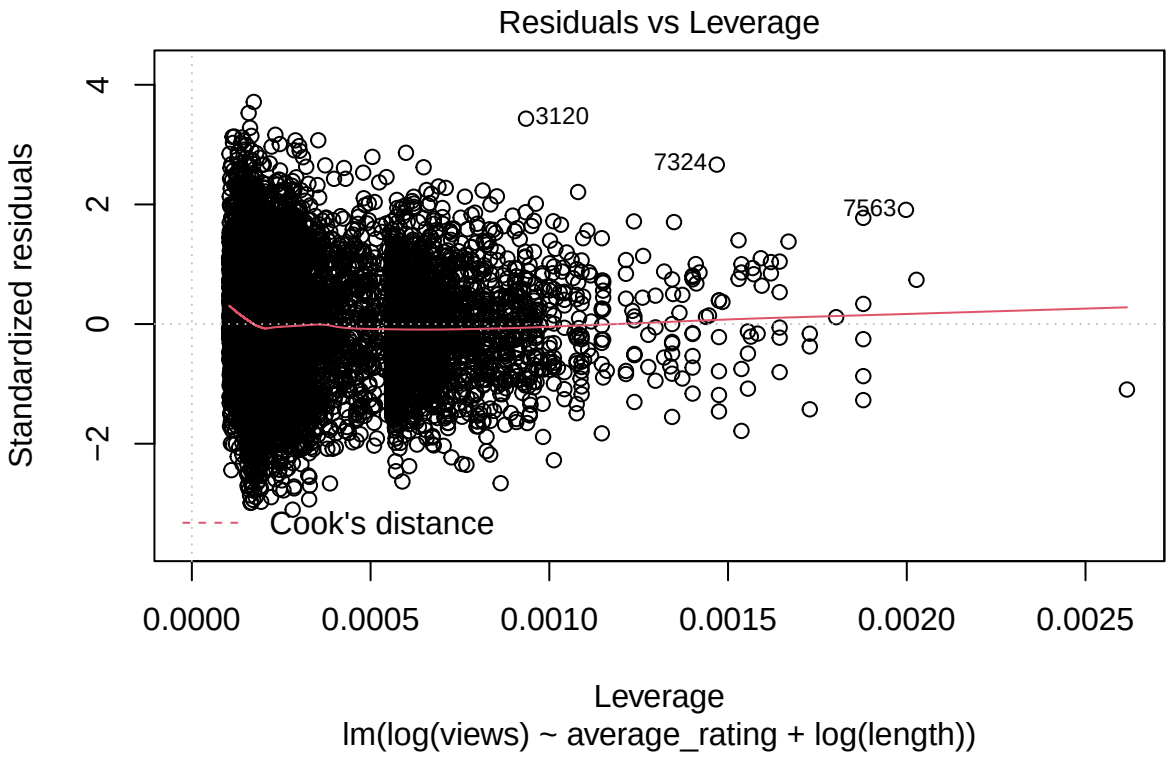
5. **Normally Distributed Errors:** 'Replace this quote with what you are looking for when you are assessing the normality of your error distribution. Also include what you observe in your plot. In the model that you have chosen to report, do you satisfy the assumption of normally distributed errors?'

```
## remove this commented block and write code that can help you assess whether
## your model satisfied the requirement of normally distributed errors
plot(model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(views) ~ average_rating + log(length))
3120
5128
5118
8
6
4
2
0
−2
−4
−6
5.0
5.5
6.0
6.5
7.0
7.5
8.0


Normal Q−Q
Standardized residuals
4
2
0
−2
−4
−2
0
2
4
5128
Theoretical Quantiles
lm(log(views) ~ average_rating + log(length))


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
3120
5128
5118
5.0
5.5
6.0
6.5
7.0
7.5
8.0
Fitted values
lm(log(views) ~ average_rating + log(length))



Given the QQ plot, doesn't appear to have much deviation except at the extreme ends with minor bimodal tendency at the extremes (which makes sense given the large distribution of zero and 5 ratings.) What could be done to improve this assumption is to remove all zero-rating rows in our data, as these are likely low-volume videos that may be outliers.