Modeling and prediction for movies

Setup

Load packages

```
library(ggplot2)
library(dplyr)
library(statsr)
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.0.2
```

Load data

```
load("movies.Rdata")
```

Part 1: Data

According to this dataset's description, the data set consists of 651 randomly sampled movies produced and released before 2016. Besides, the sample size, 651 movies, is less than 10% of the population, and the sample size is greater than 30. With that being said, these samples are qualified to be independent. And since we aren't sure whether this dataset is randomly assigned, we could conclude that this data set is generalized but not causal.

Part 2: Research question

When it comes to the popularity of movies, undoubtedly, people are mostly obsessed with their content. But there are other factors worth exploring more about the famous film: Are the specific genres arousing people's interests, or the phenomenal cast meeting the public's expectations, or the box office giving audiences great confidence to take a watch at theatres? To find out the relationship between these possible factors and the popularity of movies, here we raised this research question:

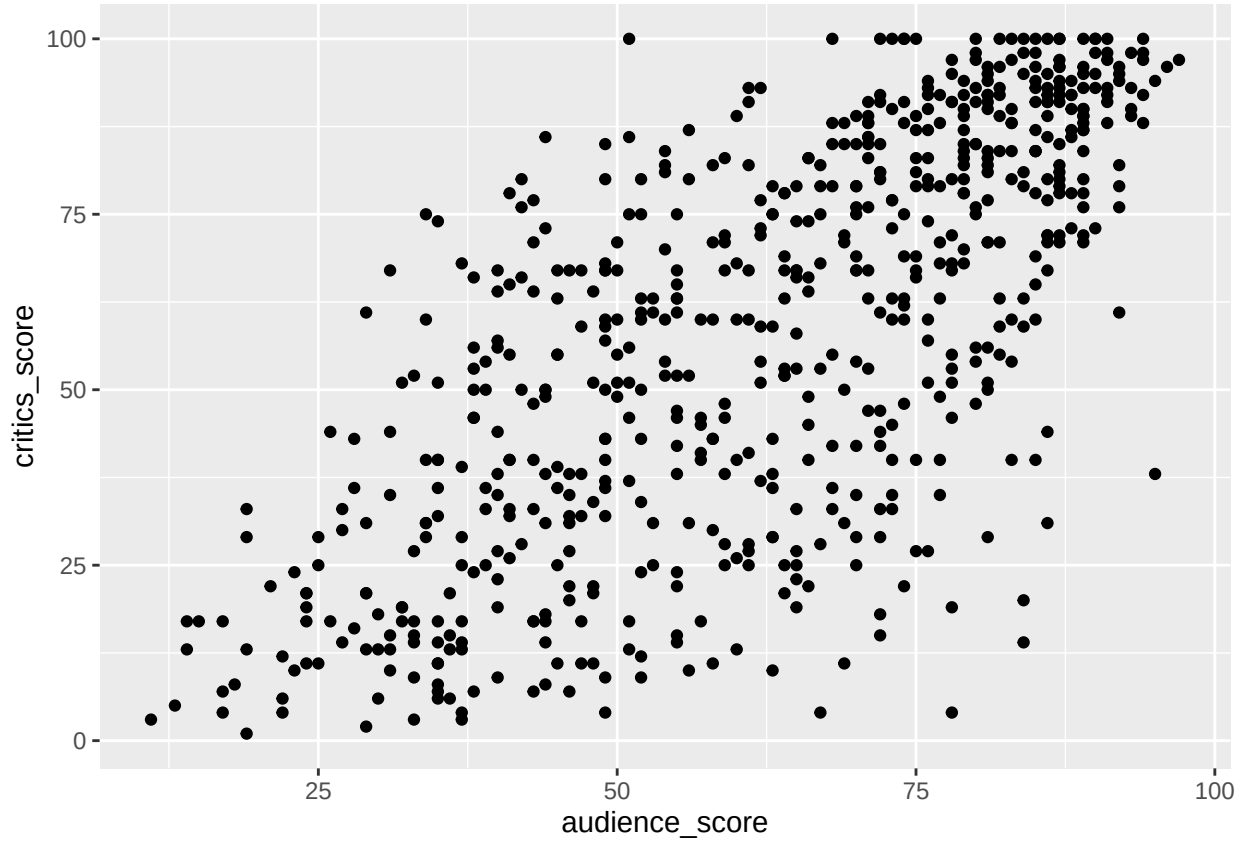
Research Question: The relationship between the critics scores of the movie on Rotten Tomatoes and its audience score and whether or not this movie won the best picture in Oscar.

Part 3: Exploratory data analysis

Step 1:

We create a scatterplot to see the relationship between the response variable (critics_score) and our numerical explanatory variable (audience_ score):

```
ggplot(data = movies, aes(x = audience_score, y = critics_score)) +
  geom_point()
```

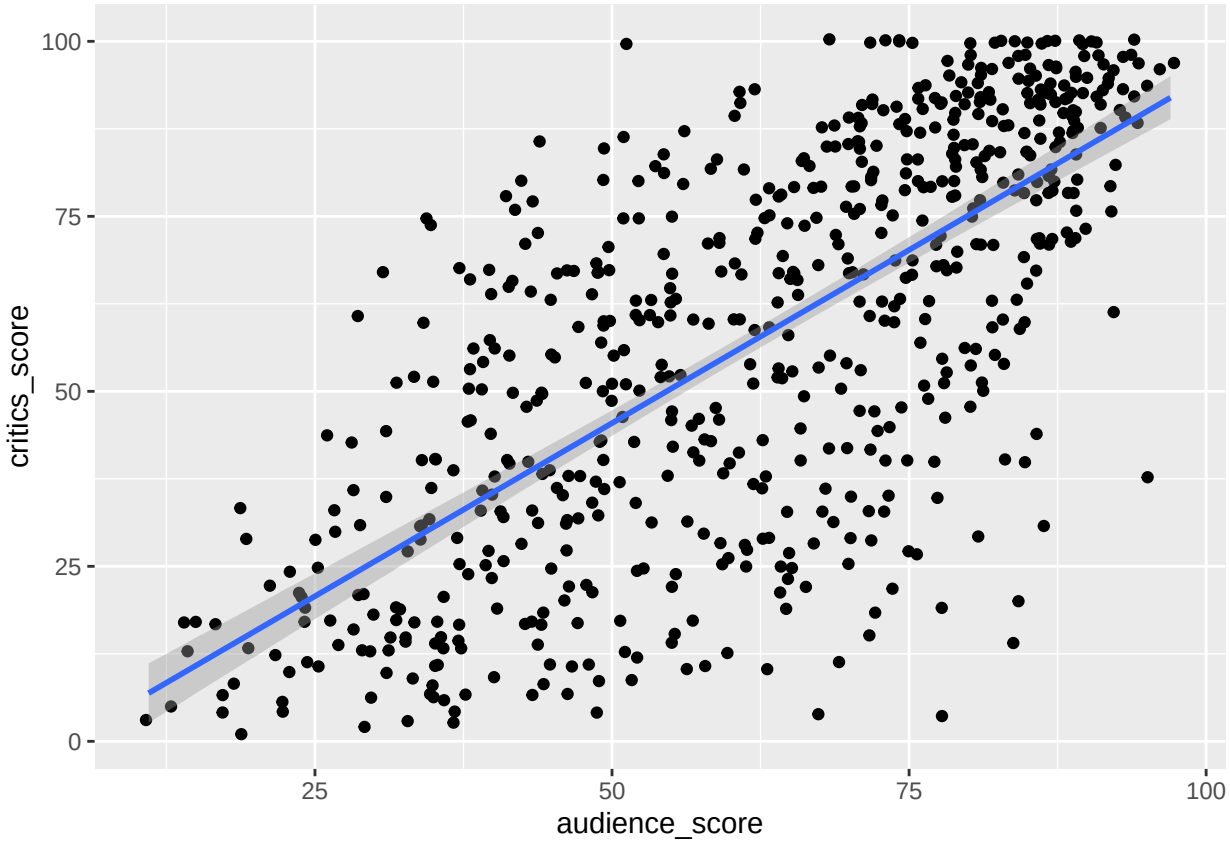


Apparently, we can tell there's a correlation between these two variables.

Then we want to see if the trend in the plot is something more than natural variation:

```
ggplot(data = movies, aes(x = audience_score, y = critics_score)) +
  geom_jitter() +
  geom_smooth(method = "lm")
```

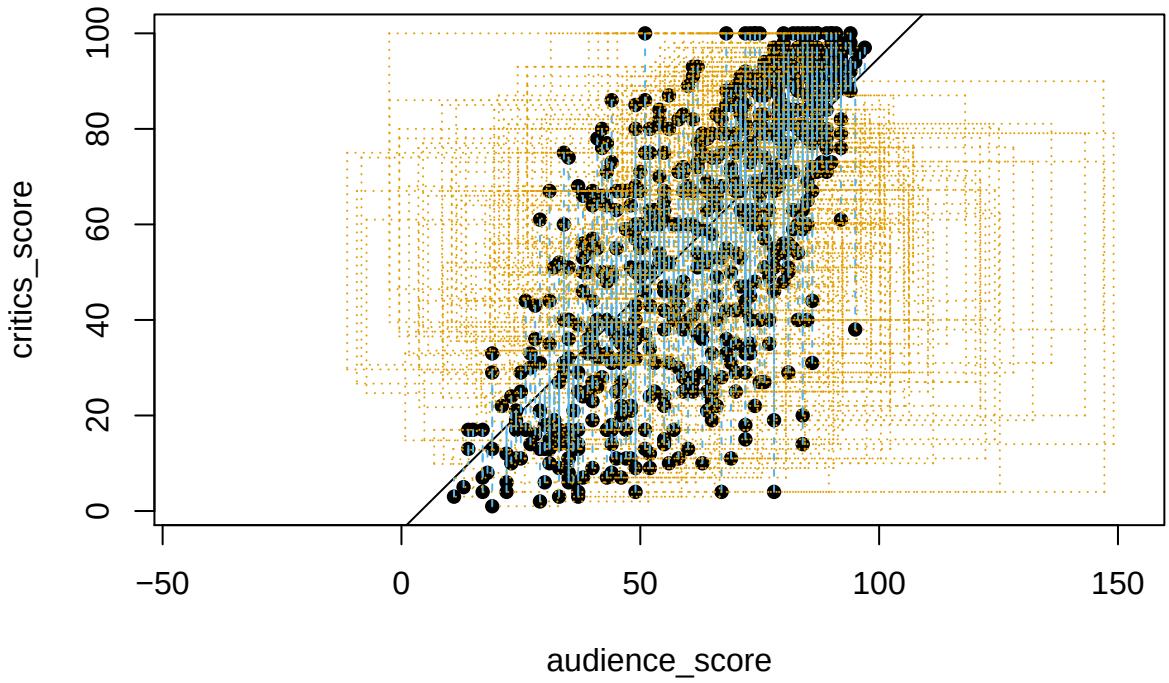
```
## `geom_smooth()` using formula 'y ~ x'
```



Since the shaded gray area around the line above tells us about the variability we might expect in our expectations. Now we know that the trend in the plot is within a natural variation.

Now we try to summarize the relationship between the two variables:

```
plot_ss(x = audience_score, y = critics_score, data = movies, showSquares = TRUE)
```



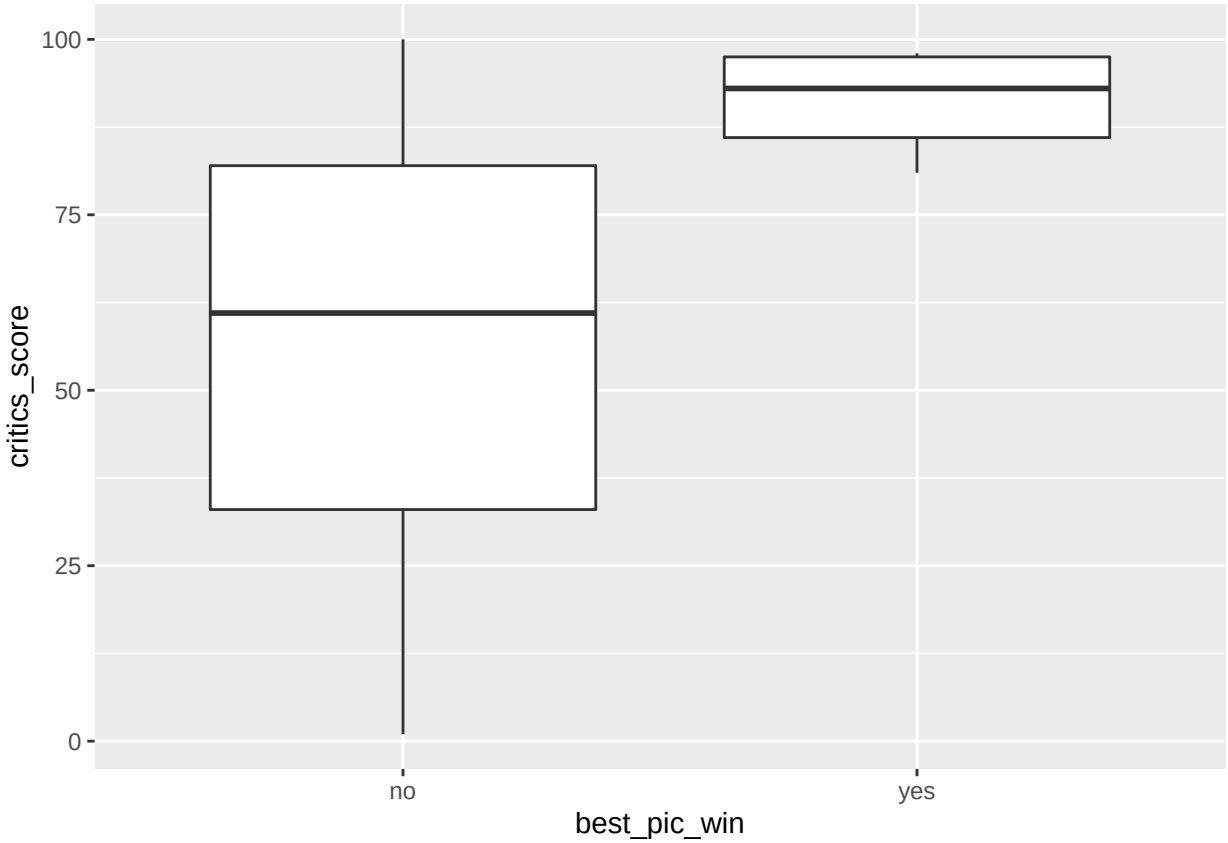

```
## Click two points to make a line.
## Call:
## lm(formula = y ~ x, data = pts)
##
## Coefficients:
## (Intercept)            x
##     -3.9987       0.9892
##
## Sum of Squares:  264281.8
```

Here we can see the correlation between the two variables is linear, which allows us to click the plot in two locations to draw the best fit line.

Step 2:

Now we might want to explore more about the two levels:(Yes or No) of the explanatory variable (best_pic_win) :

```
movies%>%
  ggplot(movies, mapping = aes(x = best_pic_win, y = critics_score)) +
  geom_boxplot()
```



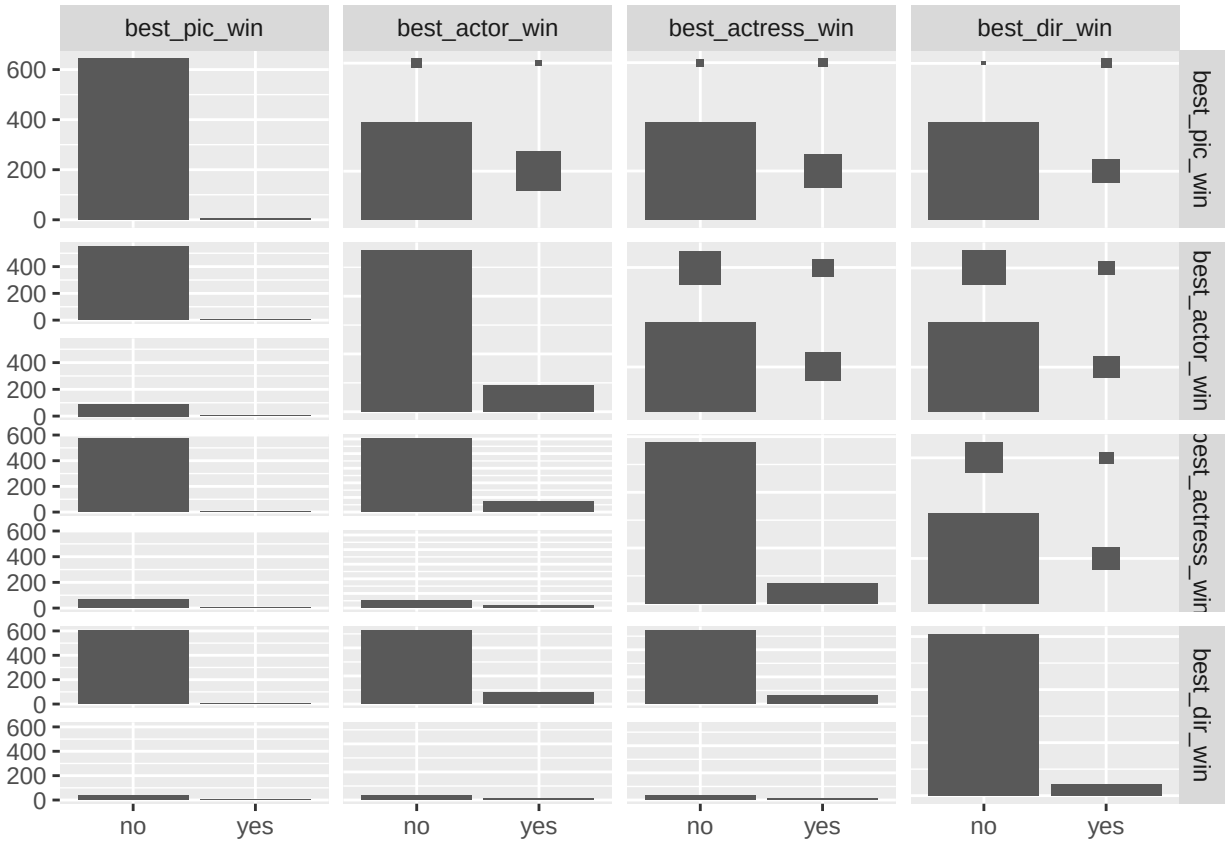
As shown above, none of these explanatory variables is distributed more constantly. Now we can start our multiple modeling process.

Part 4: Modeling

Step 1:

We want to know more about the relationship between different levels of our explanatory variables:(best_pic_win).

```
ggpairs(movies, columns = 20:23)
```



These variables are collinear (correlated), and adding more than one of these variables to the model would not add much value to the model. Therefore, in this application and with these highly-correlated predictors, it is reasonable to use the explanatory variable (best_pic_win) as the single representative of these variables.

Step 2:

According to our EDA, now we fit the linear model to see the multiple linear regression model between these three variables and access the model:

```
m1 <-lm( critics_score ~ audience_score + best_pic_win, data = movies)
summary(m1)
```

```
##
## Call:
## lm(formula = critics_score ~ audience_score + best_pic_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -68.921 -12.312   2.942  14.401  53.599
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -3.69326    2.57097  -1.437    0.151
## audience_score   0.98224    0.03936  24.954   <2e-16 ***
## best_pic_winyes 11.76932    7.71210   1.526    0.127
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.16 on 648 degrees of freedom
## Multiple R-squared:  0.4978, Adjusted R-squared:  0.4963
## F-statistic: 321.2 on 2 and 648 DF,  p-value: < 2.2e-16
```

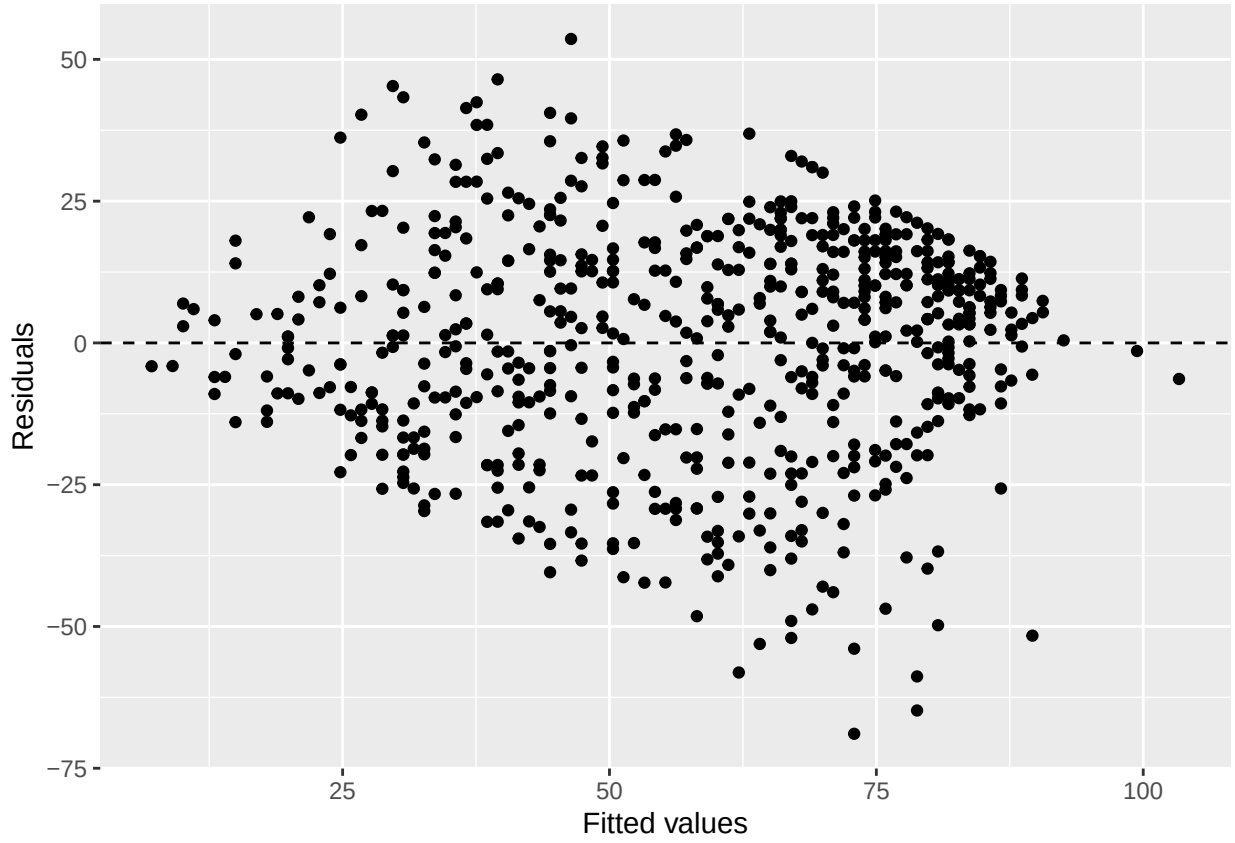
Interpreting the output: With all constant, if the movie ever won Oscar, its critic scores on Rotten Tomatoes are about 11.8 more than those that didn't win Oscar.

Here we don't interpret the intercept because it serves to adjust the height of the line.

Step 3:

Now we need to do the diagnostics for MLR: (1) linear relationships between (numerical) x and y, here we use (e vs.x):

```
ggplot(data = m1, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  xlab("Fitted values") +
  ylab("Residuals")
```



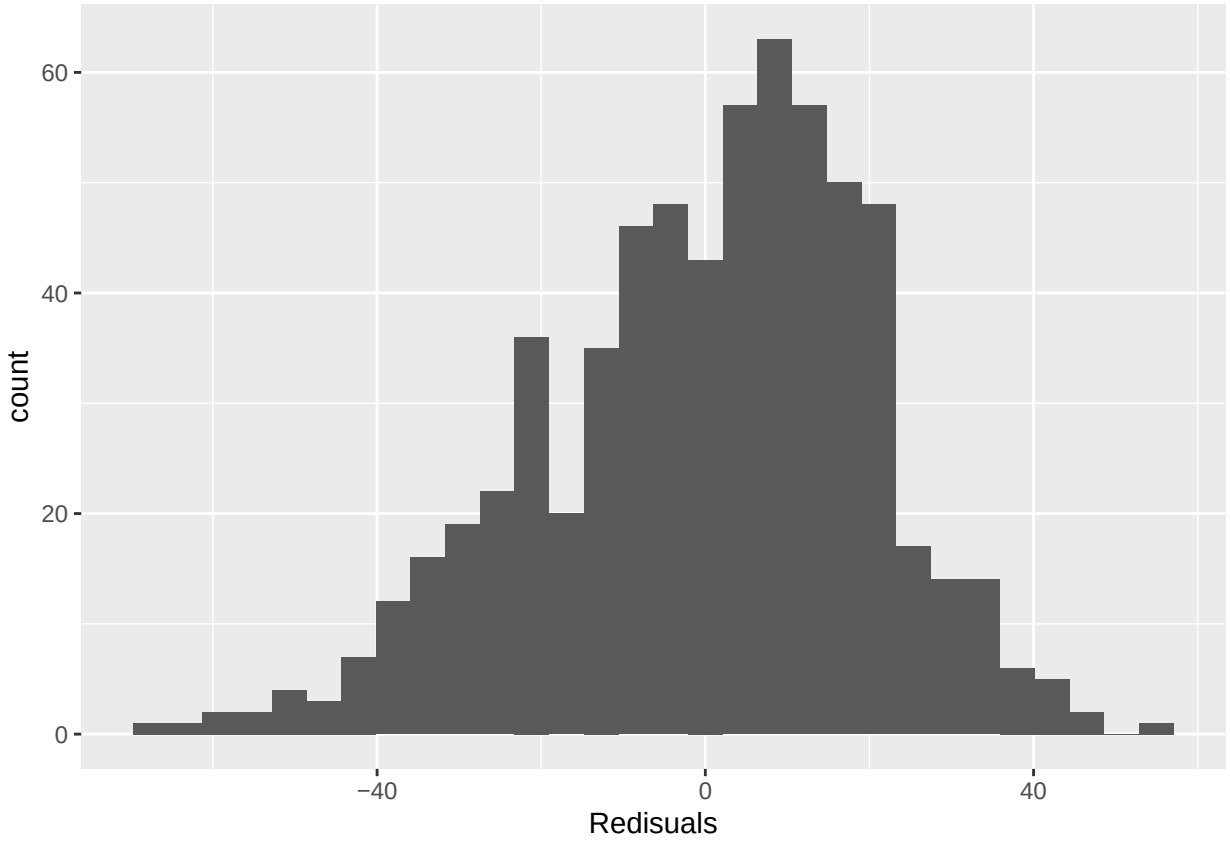
Here we can determine that residuals of this model are distributed around 0, therefore, it satisfies the conditions of linearity.

(2) nearly normal residuals:

Firstly we use a histogram to check this condition:

```
ggplot(data = m1, aes(x = .resid)) +
  geom_histogram() +
  xlab("Redisuals")
```

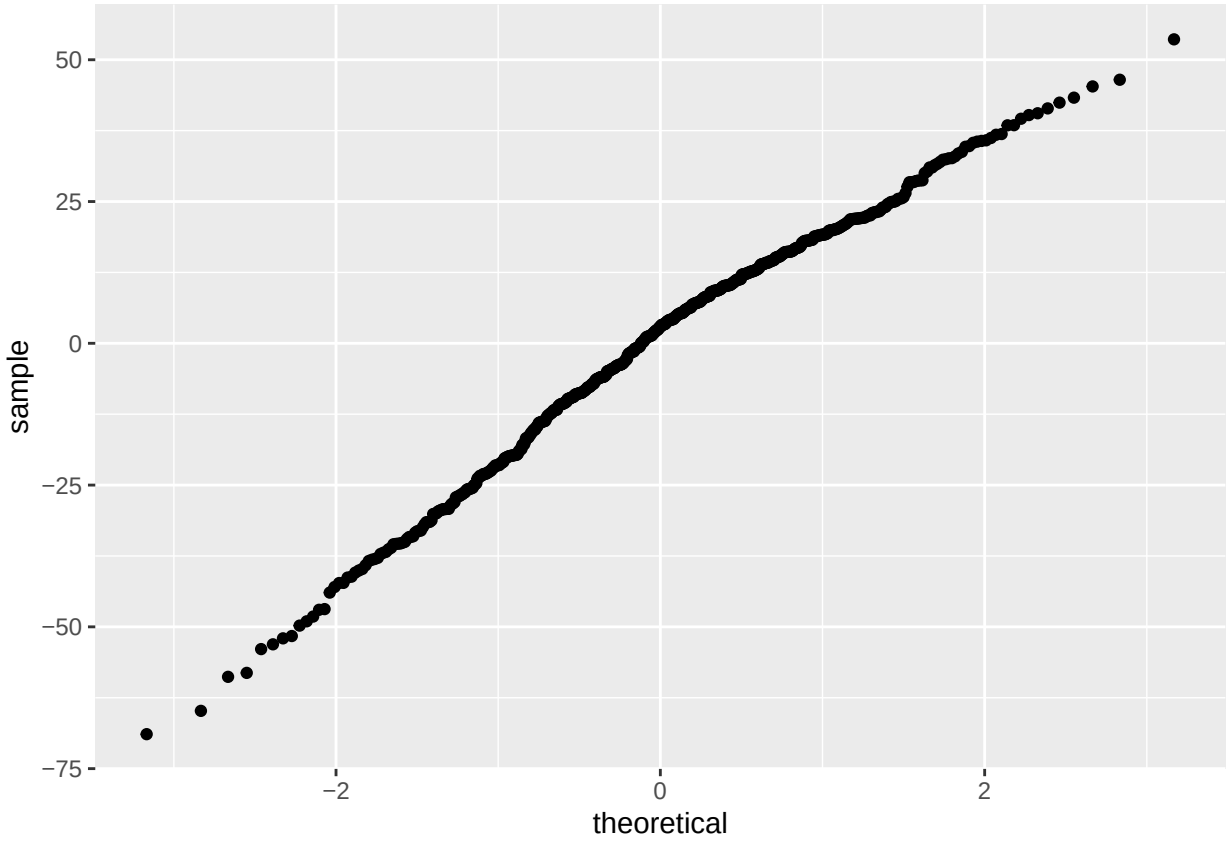
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The distribution of residuals is a little bit left-skewed but generally reasonably customarily distributed, centered at 0.

Besides, we can also use a normal probability plot of residuals to check if this condition is met:

```
ggplot(data = m1, aes(sample = .resid)) +
  stat_qq()
```

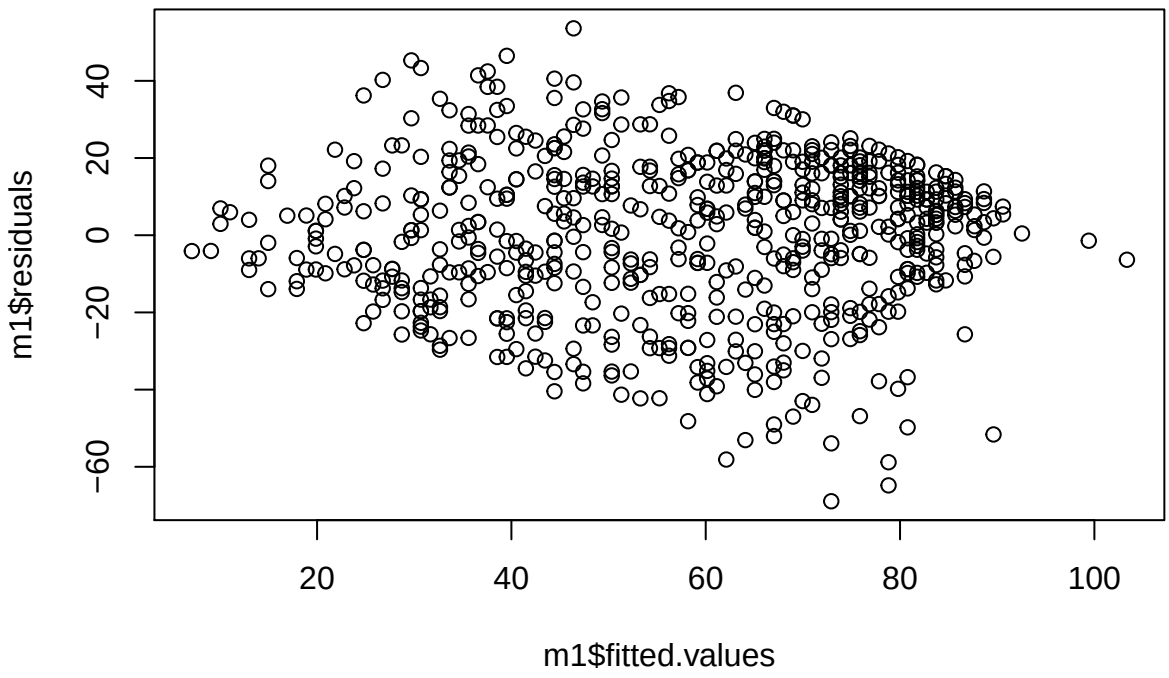


We can tell that this condition is met from the normal probability plot of residuals since they are roughly distributed around the line.

(3) constant variability:

Here we use a residuals plot to check residuals' constant variability:

```
plot(m1$residuals ~ m1$fitted.values)
```

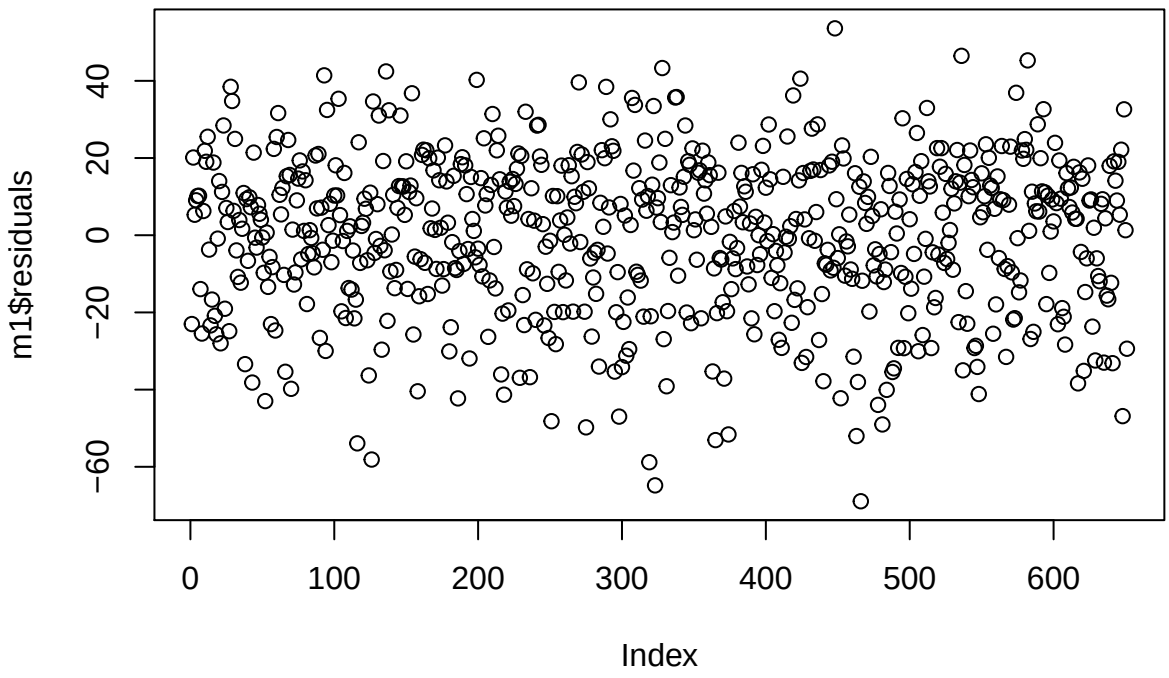


We can see that our model residuals randomly scattered in a band with a constant width around 0, and it's not a fan shape. Therefore, this condition is met, as well.

(4) independent residuals:

Now we check the independence of observations of residuals:

```
plot(m1$residuals)
```



With all discussed above, our MLR has met all four conditions. However, we didn't plot the absolute value of residuals, and since we are unaware of the details about how the data are randomly sampled, this model is not entirely perfect.

Part 5: Prediction

Suppose we want to use the model we created earlier to evaluate Rotten Tomatoes' critic score of a 2016 movie (not included in this sample), which didn't win an Oscar, and its audience score is 73.

First, we need to create a new data frame for this movie:

```
newmovie <- data.frame(best_pic_win = "no", audience_score = 73)
```

Next, we use the predict() function to do the prediction:

```
predict(m1, newmovie)
```

```
##        1
## 68.01017
```

According to the result, we know that the predicted critic score of this movie would be about 68.01.

Besides, we also want to construct a prediction interval around this prediction, which will measure uncertainty around the forecast.

```r
predict(m1, newmovie, interval = "prediction", level = 0.95)
```

```
##        fit      lwr      upr
## 1 68.01017 28.38584 107.6345
```

Hence, with 95% confidence, the model predicts that this movie, which didn't win Oscar and audience score is 73, expected to have an evaluation critic score on Rotten Tomatoes between 28.39 and 107.63.

Part 6: Conclusion

With all stated above, we can conclude that there's a linear relationship between the critic score of the movie in 2016 in Rotten Tomatoes and its audience score and whether or not it won an Oscar. And the more audience score it gets, it will be more likely to gain higher critic scores. Meanwhile, the same is true with if it won Oscar, and the effect imposed on critic score of wining Oscar is far greater than the one of high audience score.

However, there're some shortcomings in this regression model. To illustrate, we didn't consider the concrete details of sorting different levels of our explanatory variables and weighing other variables' correlation with our response variable. And we might need to use p-value and adjusted R squared to search for the best model instead.