Although our dataset contains movie characteristics such as “Name” “Director” “ID” etc, We only focus on the relationship between Year and Score this time.

```
movie <- read.csv("movie.csv")
summary(movie)
```

```
##        ID                Name              Released          Year
##  Min.   :     417   Length:10436       Min.   :0.000   Min.   :1902
##  1st Qu.:  111255   Class :character   1st Qu.:1.000   1st Qu.:1991
##  Median : 1045670   Mode  :character   Median :1.000   Median :2006
##  Mean   : 2718015                      Mean   :0.972   Mean   :2001
##  3rd Qu.: 4679114                      3rd Qu.:1.000   3rd Qu.:2016
##  Max.   :13399862                      Max.   :1.000   Max.   :2023
##                                                        NA's   :2902
##      Score         RatingCount        Director
##  Min.   : 1.300   Min.   :      5   Length:10436
##  1st Qu.: 5.700   1st Qu.:   1561   Class :character
##  Median : 6.500   Median :   5304   Mode  :character
##  Mean   : 6.413   Mean   :  26069
##  3rd Qu.: 7.300   3rd Qu.:  19091
##  Max.   :10.000   Max.   :1038909
##  NA's   :292      NA's   :292
```

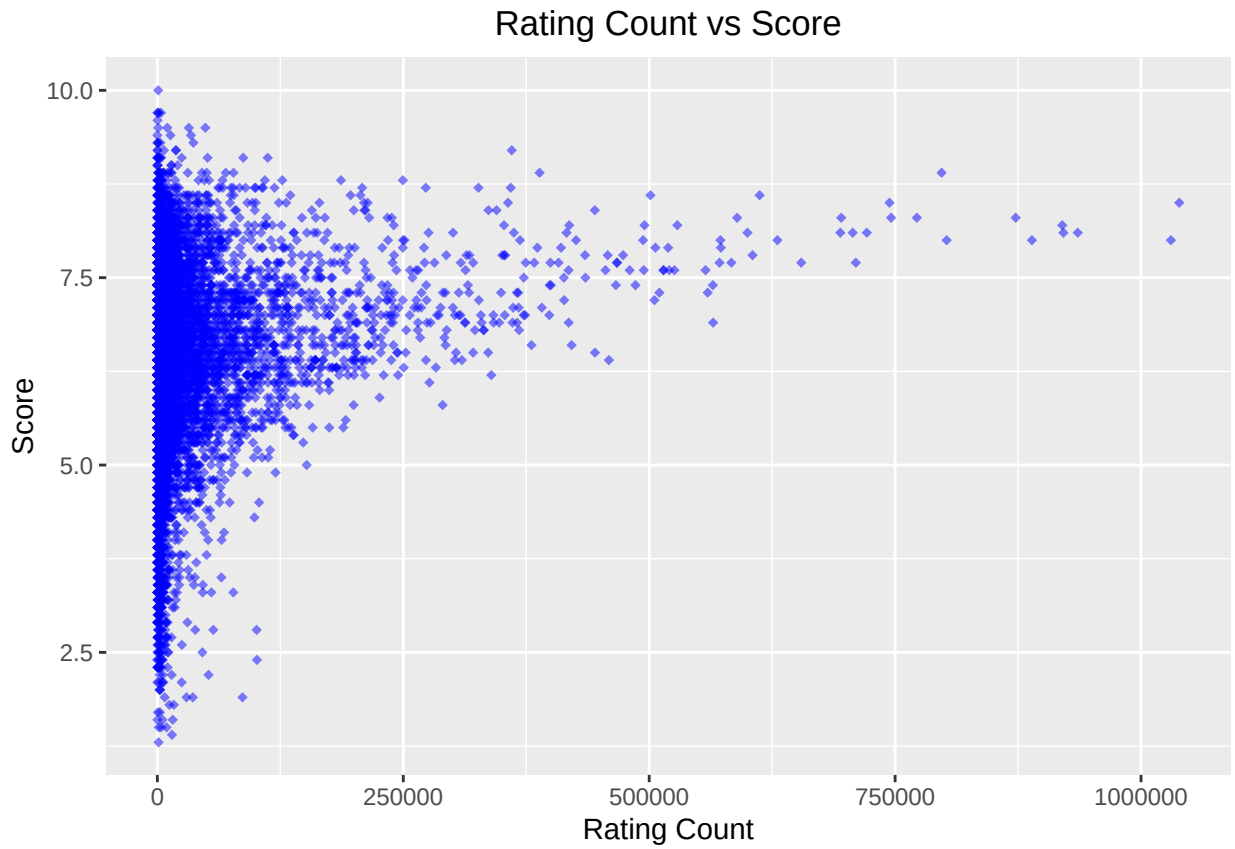
```
year <- movie$Year
score <- movie$Score
count <- movie$RatingCount
id <- movie$ID
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
```

```
ggplot(movie, aes(count, score)) +  geom_point(colour = "blue",shape = 18,alpha = 1/2) +
labs(title="Rating Count vs Score", x="Rating Count", y="Score")+
  theme(plot.title = element_text(hjust = 0.5))
```

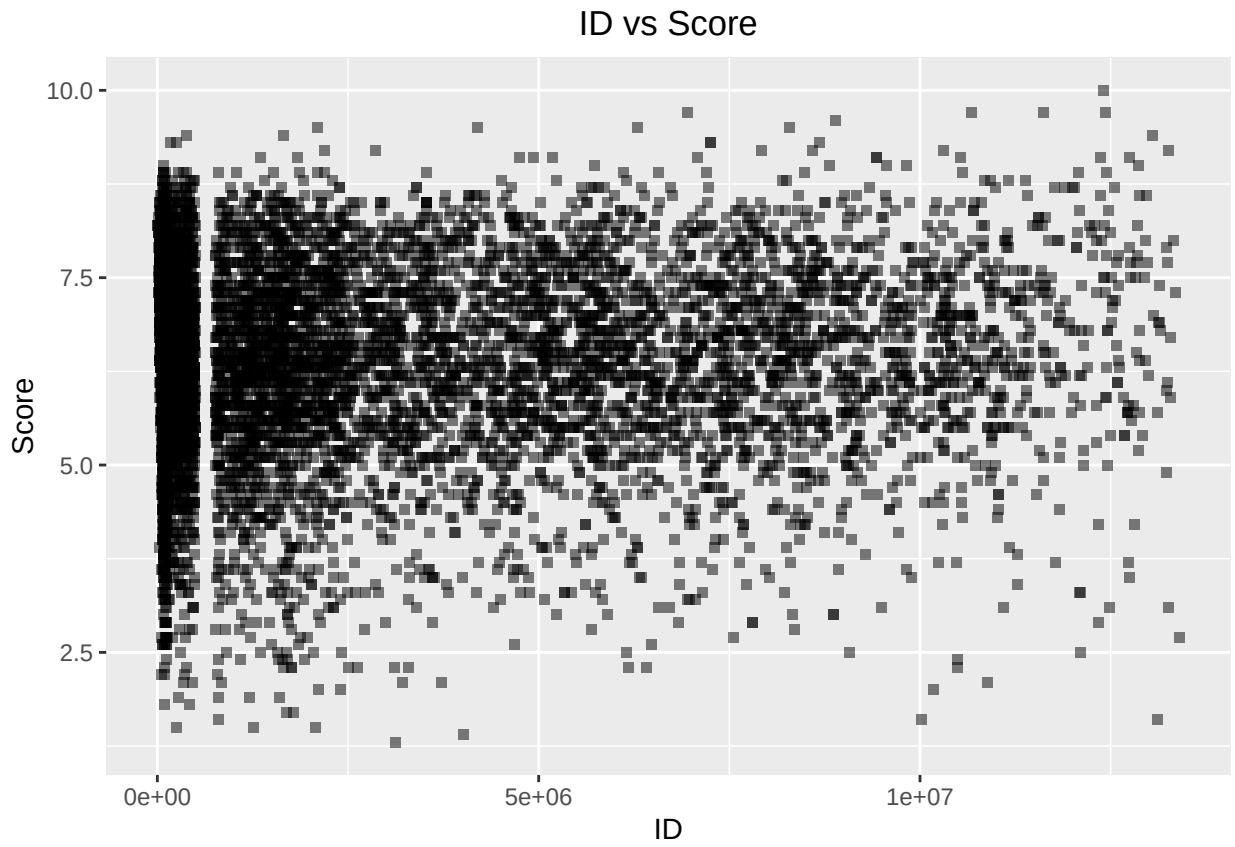
```
## Warning: Removed 292 rows containing missing values (geom_point).
```



For the scatter plot of Rating Count vs Score, we have that as the rating count gets higher and higher, the rating of movies tend to concentrate on a higher score than those movies that are not that frequently rated, this can be because people are more interested in rating a movie that shocks them or provides an amazing feeling,if the movie gets more rating, in other words, is more popular, that actually indicates that this movie is more attractive and should get a relatively higher rating. And if the movie is bad or is just an ordinary make, it will not attract anyone to comment on it or gives it a rating.

```
ggplot(movie, aes(ID, score)) +  geom_point(colour = "black",shape = 15,alpha = 1/2) +
labs(title="ID vs Score", x="ID", y="Score")+
  theme(plot.title = element_text(hjust = 0.5))
```

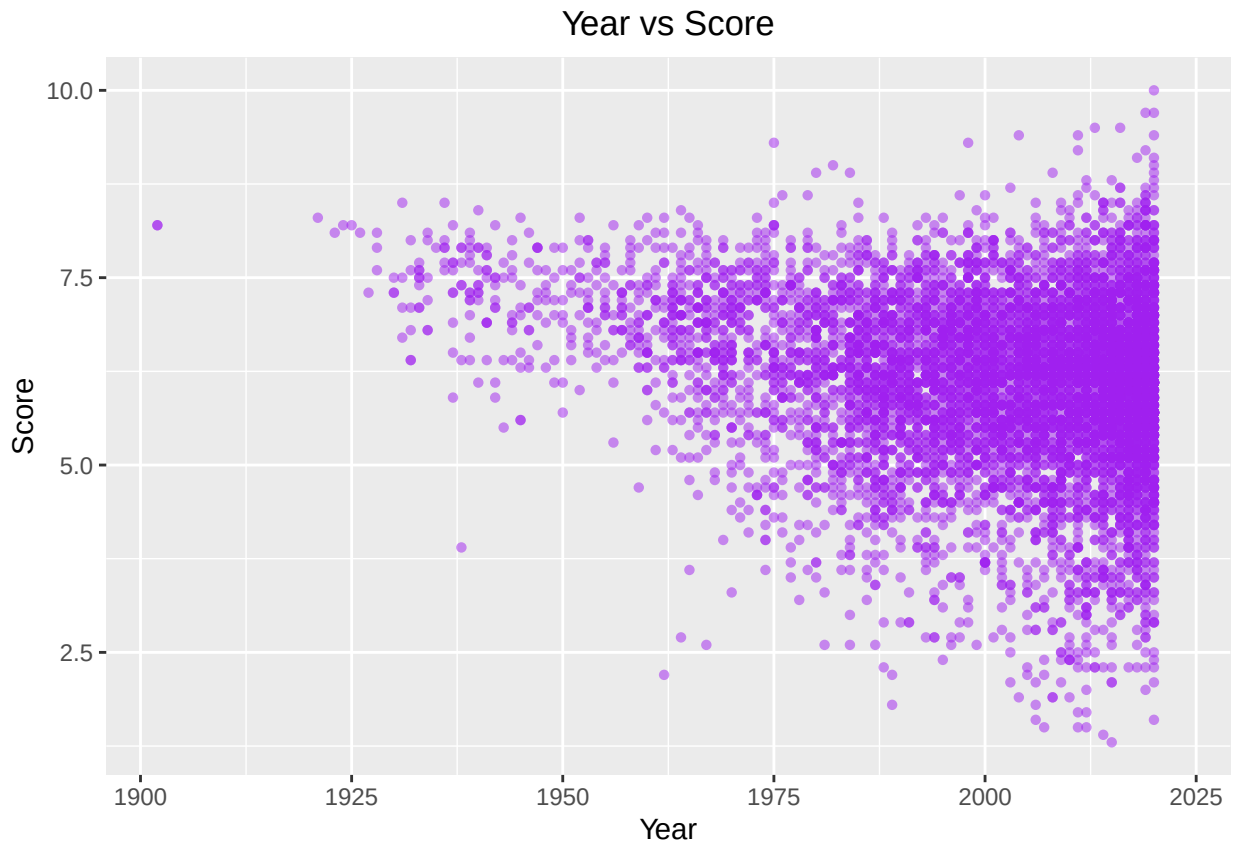
```
## Warning: Removed 292 rows containing missing values (geom_point).
```



For this one, ID vs Score, as we can see, it's quite randomly distributed, and it indeed make sense since the ID is just an index which has no meaning behind it, and it does not has any relationship with the score of movie.

```
ggplot(movie, aes(year, score)) +  geom_point(colour = "purple",shape = 16,alpha = 1/2) +
labs(title="Year vs Score", x="Year", y="Score")+
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 3028 rows containing missing values (geom_point).
```



From the scatter plot for Year vs Score, we get that as time goes by, number of movies becomes higher and higher, this may be because of the abrupt development of modern technology which makes it much easier to produce a movie.

And according to the plot, we can also see that movies from early times tend to have a higher quality than they are today,this maybe due to the fact that nowadays, people make movies because they want to get some money from it,not as before because they want to convey some beautiful thoughts, so they do not care about the quality that much.

Although our dataset contains movie characteristics such as “Name” “Director” “ID” etc, We only focus on the relationship between Year and Score this time.

We can fit a simple linear regression model

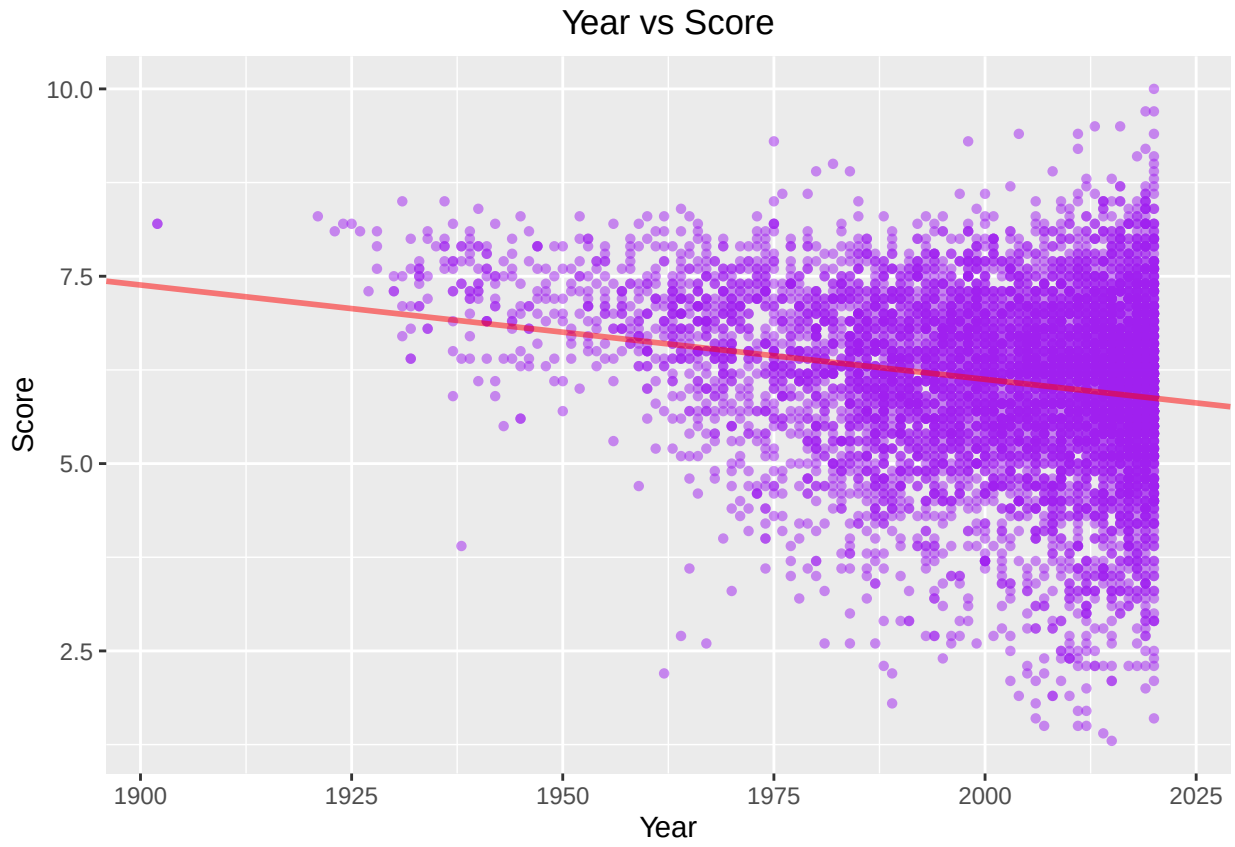
```
fit = lm(score~year)
summary(fit)
```

```
##
## Call:
## lm(formula = score ~ year)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6354 -0.6004  0.1251  0.7387  4.1275
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 31.2991946  1.4407928   21.72   <2e-16 ***
## year        -0.0125875  0.0007201  -17.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.125 on 7406 degrees of freedom
##   (3028 observations deleted due to missingness)
## Multiple R-squared:  0.03963,    Adjusted R-squared:  0.0395
## F-statistic: 305.6 on 1 and 7406 DF,  p-value: < 2.2e-16
```

```
ggplot(movie, aes(year, score)) +  geom_point(colour = "purple",shape = 16,alpha = 1/2) +
labs(title="Year vs Score", x="Year", y="Score")+
  theme(plot.title = element_text(hjust = 0.5))+
  geom_abline(slope = fit$coefficients[2], intercept = fit$coefficients[1],
              color = "red", size = 1, alpha = 0.5)
```

```
## Warning: Removed 3028 rows containing missing values (geom_point).
```



The equation of the fitted line is $\hat{y} = 31.2991946 - 0.0125875x$ since $\hat{\beta_0} = 31.2991946$ and $\hat{\beta_1} = -0.0125875$

Question: Does it appear to be a linear relationship between the two variables?

By observation, there's no obvious linear relationship, but we still need to test it.

We conduct a t-test at a 5% significance level to determine whether there's a linear relationship between the two variables.

The single linear regression model is defined as $y = \beta_0 + \beta_1 x$ as we know.

So we define $H_0 : \beta_1 = 0$ and $H_a : \beta_1 \neq 0$

Decision rule:If p-value is smaller than 5%(significance level), we reject the null hypothesis.

calculate t-value: as indicated in the summary, we have

```
t_value = (-0.0125875-0)/0.0007201
t_value
```

```
## [1] -17.48021
```

```
p_value = (2*pt(t_value,7406))
p_value
```

```
## [1] 4.445894e-67
```

As we can see, we obtain a extremely small p-value, also by looking at the summary,we get that p-value$<2.2*10^{-16}$, thus much smaller than 5%, so null hypothesis is rejected, thus there's a linear relationship between the two variables. The reason why it's not obvious is that number of observations for each year is quiet inconsistent.

We can also do some interesting stuff, such as estimating the general/average rating of the movies when we are in year 2040. So I'm gonna provide a confidence interval for that.

```
predict(fit, data.frame(year=2040), interval = "confidence", level = 0.90)
```

```
##        fit      lwr      upr
## 1 5.620712 5.569561 5.671862
```

```
predict(fit, data.frame(year=2040), interval = "confidence", level = 0.95)
```

```
##        fit      lwr      upr
## 1 5.620712 5.559759 5.681664
```

Meaning that the probability that the average rating of movies in 2040 lies between 5.569 and 5.671 is 90% and the probability that the average rating of movies in 2040 lies between 5.559 and 5.681 is 95%.

We can also use prediction interval to predict the range for a single movie that was produced in year 2040.

```
predict(fit, data.frame(year=2040), interval = "prediction", level = 0.90)
```

```
##        fit      lwr      upr
## 1 5.620712 3.768964 7.472459
```

```
predict(fit, data.frame(year=2040), interval = "prediction", level = 0.95)
```

```
##        fit      lwr     upr
## 1 5.620712 3.414134 7.82729
```

Meaning that the probability that the rating of a single movie produced in 2040 lies between 3.768 and 7.472 is 90% and the probability that the average rating of movies in 2040 lies between 3.414 and 7.827 is 95%.

We can notice that the prediction interval is much wider than the confidence interval, since a single observation always has more uncertainty to predict than the average of several observations.

That ends our presentation, thank you.