THE FINAL ASSIGNMENT

assignment_8- Edoardo Frigerio

THE GOAL

The goal of this research is to understand which variable have the most effect on movies revenue. In order to do this I have use a database where there are the movies from 2006 stored in the IMDb. I have select only the movies of the genre adventure,comedy and drama that are the most common one.

```
head(movies)
```

```
##   revenue  year rating runtime votes metascore genre
## 1    0.04  2007    5.7      88  8914        35 Drama
## 2    0.04  2014    6.9      95 31370        31 Comedy
## 3    0.05  2013    6.2     111  1356        51 Drama
## 4    0.08  2015    6.4     133 17565        72 Drama
## 5    0.11  2009    7.3      94 50946        73 Drama
## 6    0.15  2016    6.1      80  2417        69 Adventure
```

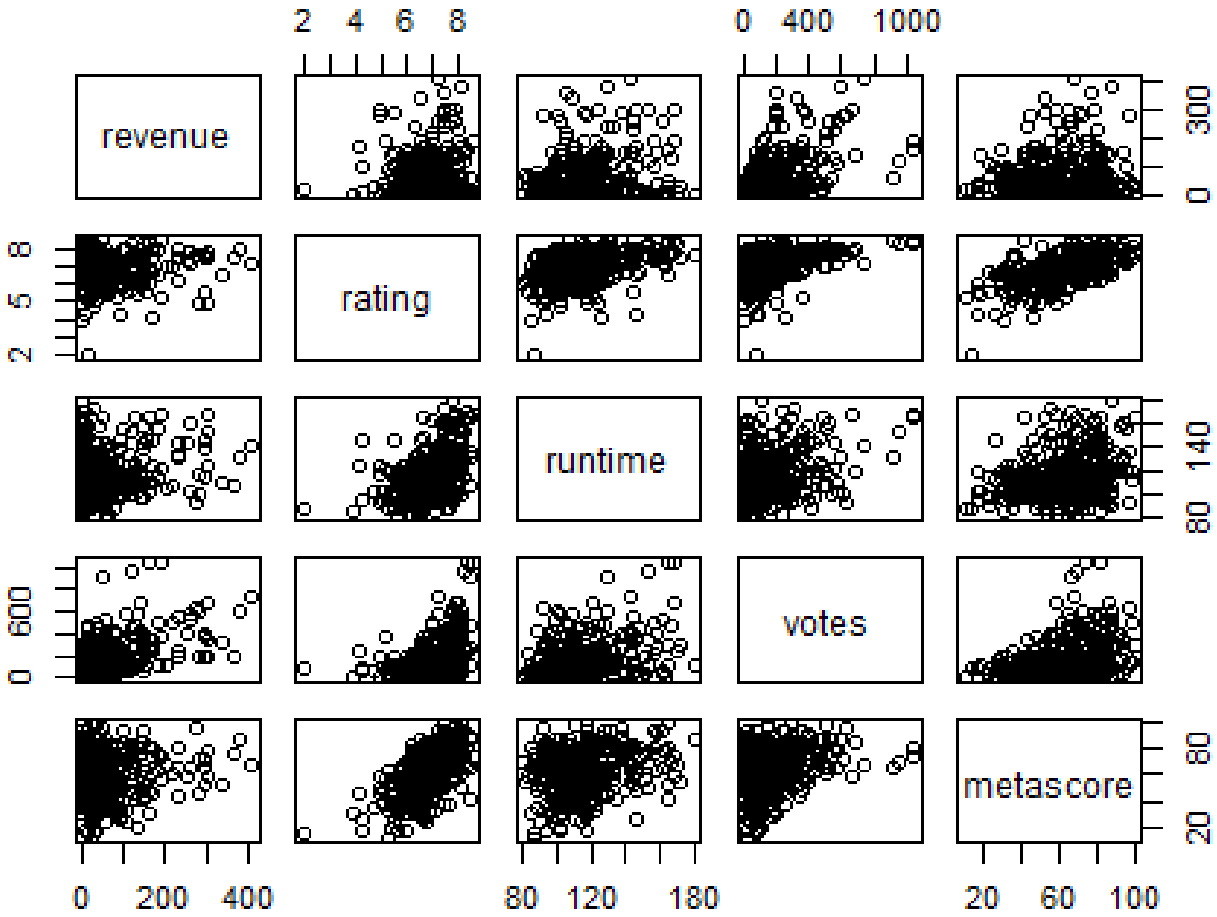
The variables, that I have chosen to explain the revenue(in millions US dollars) of the movies, are:

- **runtime** of the movies in minutes,
- **votes** that are the number of the people that on IMDb website have rate the movie or rewieving so it could be a sort of interaction (between people and a particular movie) measure,
- **year** of production of the movie,
- **score** of the movie that is a weighted average of the people rate on the IMDb website,
- **metascore** is the movie's rating from the film criticism
- **genre** the type of movie as I explain before only three: adventure,drama,and comedy.

THE DATA

As you can see from the scatter plot below it doesn't seem that exist a collinearity issue among the covariates but I refer this analysis to later in this assignment.

```
plot(movies[-c(2,7)])
```



The graph the correlation exclude that could exist a correlation among the covariates and the response variable. A light correlation could be between "rating" and "votes" and a medium-strong between "metascore" and "rating", in the last case it could be because the two variable express the same thing but one, the rating, is the people score while metascore is the score of movies pundits. If the two are high correlated (>0.80) it means one of this two things: the people are good as the pundit to rate movies or the pundit opinion isn't different from the normal people so isn't pundit of movies.

```
round(cor(x),4)
```

```
##            revenue    year  rating runtime   votes metascore
## revenue     1.0000 -0.1449  0.0312  0.2211  0.5659    0.0091
## year       -0.1449  1.0000 -0.1268 -0.0625 -0.3801   -0.0325
## rating      0.0312 -0.1268  1.0000  0.3999  0.4751    0.6425
## runtime     0.2211 -0.0625  0.3999  1.0000  0.3250    0.2563
## votes       0.5659 -0.3801  0.4751  0.3250  1.0000    0.3105
## metascore   0.0091 -0.0325  0.6425  0.2563  0.3105    1.0000
```

I have decided, due to this correlation among this two predictors, to remove them from the database and added a new variable called "avg_score" that is a mean, in range 1 to 10, of the variables that I have removed.

```
movies <- cbind(movies[,-c(3,6)],avg_score= ((movies$rating+movies$metascore/10)/2))
```

So now, the database is:

```
head(movies,4)
```

```
##   revenue year runtime  votes  genre avg_score
## 1    0.04    2      88  8.914  Drama      4.60
## 2    0.04    9      95 31.370 Comedy      5.00
## 3    0.05    8     111  1.356  Drama      5.65
## 4    0.08   10     133 17.565  Drama      6.80
```

THE LINEAR REGRESSION MODEL

The multiple linear regression model with all the predictors is:

$$\text{revenue} = \beta 0+\beta 1 \textit{year}+\beta 2 \text{rating}+\beta 3 \textit{runtime}+\beta 4 \text{votes}+\beta 5 \textit{metascore}+\beta 6 \text{genre}$$

The summary output is:

```
summary(ols1)
```

```
##
## Call:
## lm(formula = revenue ~ ., data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -180.225  -28.109   -7.754   16.950  269.731
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  84.69782   24.83770   3.410 0.000726 ***
## year          1.95403    0.99806   1.958 0.051047 .
## runtime       0.36887    0.17015   2.168 0.030845 *
## votes         0.26658    0.02144  12.435  < 2e-16 ***
## genreComedy -37.51758    8.37122  -4.482 1.01e-05 ***
## genreDrama  -55.17827    8.34993  -6.608 1.46e-10 ***
## avg_score   -13.66711    2.73390  -4.999 9.13e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.64 on 349 degrees of freedom
## Multiple R-squared:  0.4568, Adjusted R-squared:  0.4475
## F-statistic: 48.92 on 6 and 349 DF,  p-value: < 2.2e-16
```

As you can see from the summary out put there is the year variable the is not so significant in order to explain the response. So it is better to perform a best subset selection to explore all the possible models and find out the best to explain the variability of the response.

```
ols2<- regsubsets(revenue ~ .,data =movies,nvmax=11)
```

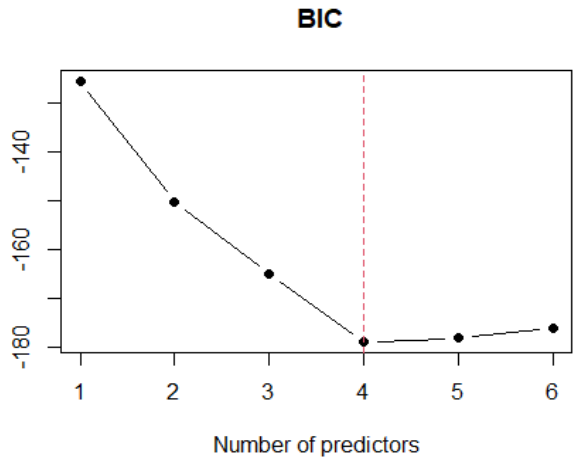
The summary output is:

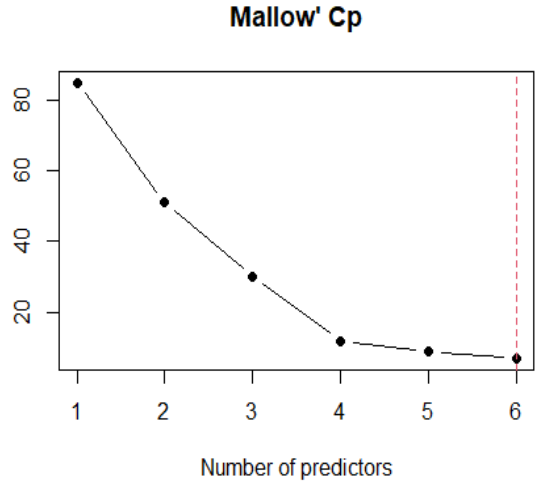
```
 summary(ols2)
```

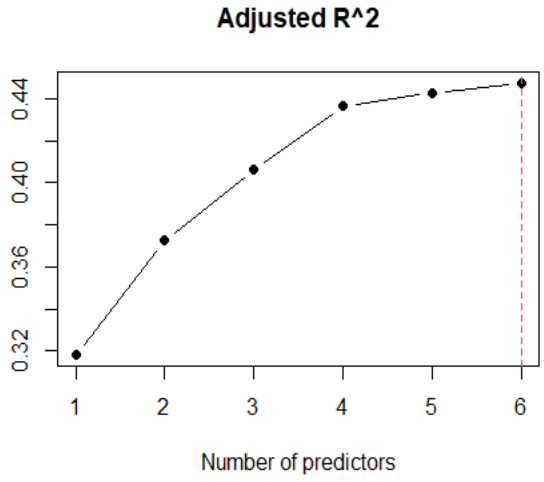
```
## Subset selection object
## Call: regsubsets.formula(revenue ~ ., data = movies, nvmax = 11)
## 6 Variables  (and intercept)
##             Forced in Forced out
```

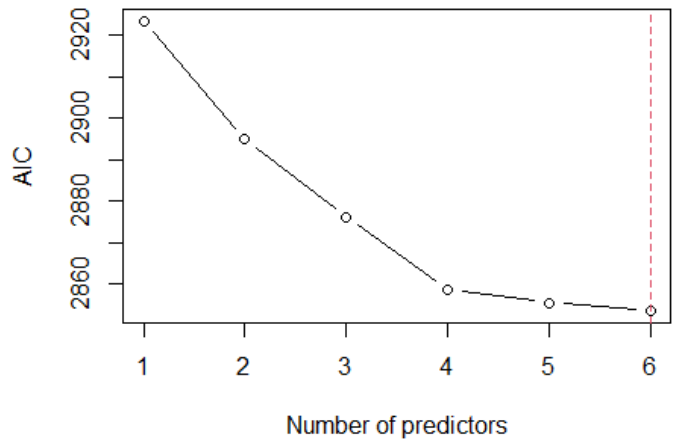
```
## year            FALSE      FALSE
## runtime         FALSE      FALSE
## votes           FALSE      FALSE
## genreComedy     FALSE      FALSE
## genreDrama      FALSE      FALSE
## avg_score       FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: exhaustive
##          year runtime votes genreComedy genreDrama avg_score
## 1  ( 1 ) " "  " "     "*"   " "         " "         " "
## 2  ( 1 ) " "  " "     "*"   " "         "*"         " "
## 3  ( 1 ) " "  " "     "*"   "*"         "*"         " "
## 4  ( 1 ) " "  " "     "*"   "*"         "*"         "*"
## 5  ( 1 ) " "  "*"     "*"   "*"         "*"         "*"
## 6  ( 1 ) "*"  "*"     "*"   "*"         "*"         "*"
```

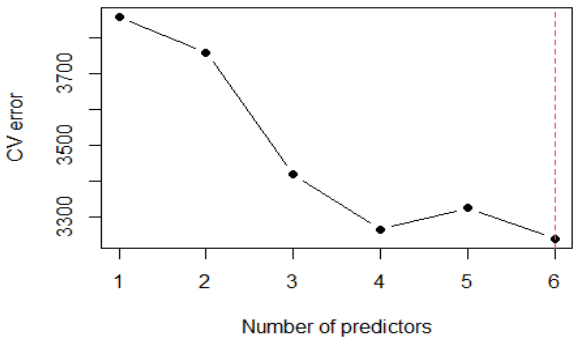
As you can see from the out put of the best subset selection the best one predictor model is the regression with only the votes variable, but to find out which is the best model overall there are 4 indices: BIC, adjusted R2, Mallow's Cp, Cross-validation error(LOOCV)











As you can see from the graphs only the BIC criteria considers the model with four predictors the best one and the other four indices consider the model with six covariates the best one. So by majority I choose the model with six covariates (the full model) that is according to the best subset selection:

$$revenue = \beta 0+\beta 1 year+\beta 2 runtime+\beta 3 votes+\beta 4 genre+\beta 5 avg\_score$$

So no variable is excluded, otherwise the BIC criteria indicates as best model the one with only four covariates, so without the variables "runtime" and "year":

$$revenue = \beta 0+\beta 1 votes+\beta 2 genre+\beta 3 avg\_score$$

and this should be reasonable because the people go to watch the movie nevertheless its duration or the year of production. If a movie is a good one is independently from this two variable. So I will continue to use the full model.

COLLINEARITY

The best way to check the collinerity issue among the predictors of the multiple linear regression model is use the VIF index:

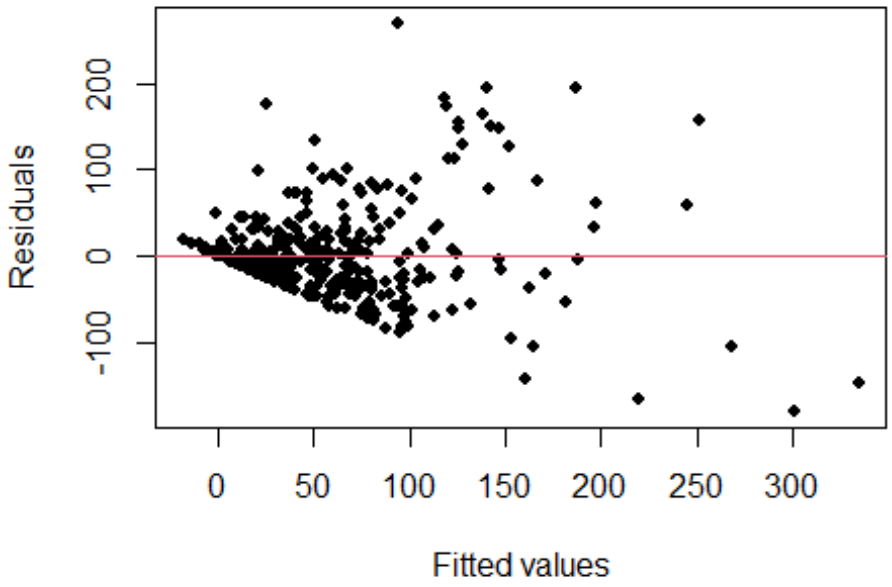
```
vif(ols1)
```

```
##        year     runtime       votes genreComedy  genreDrama   avg_score
##    1.187225    1.243754    1.548170    2.033231    1.978300    1.346017
```

As you can see all the value of the VIF are small so there isn't a collinearity issue.

DIAGNOSTIC

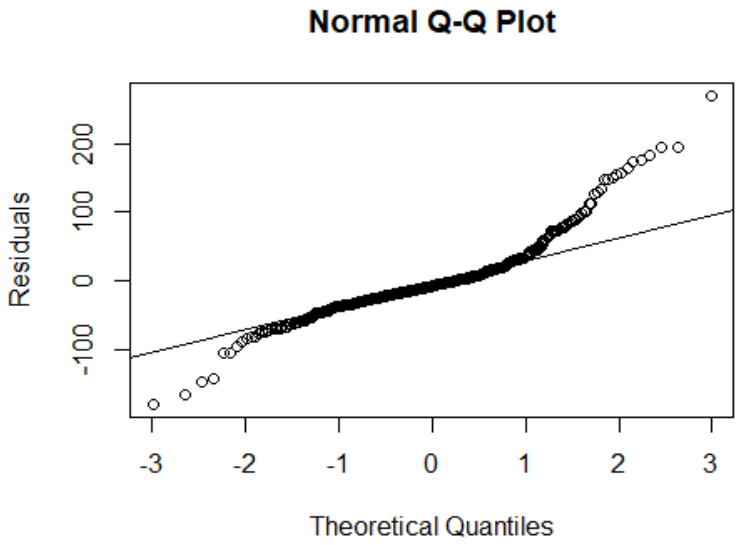
Diagnostic is used to check the assumption of the linear model. First I will check the constant variance assumption for the errors.



As you can see from the graph above the variance of the residual isn't constant. Then I will check the normality assumption of the errors with the Shapiro-Wilk test:

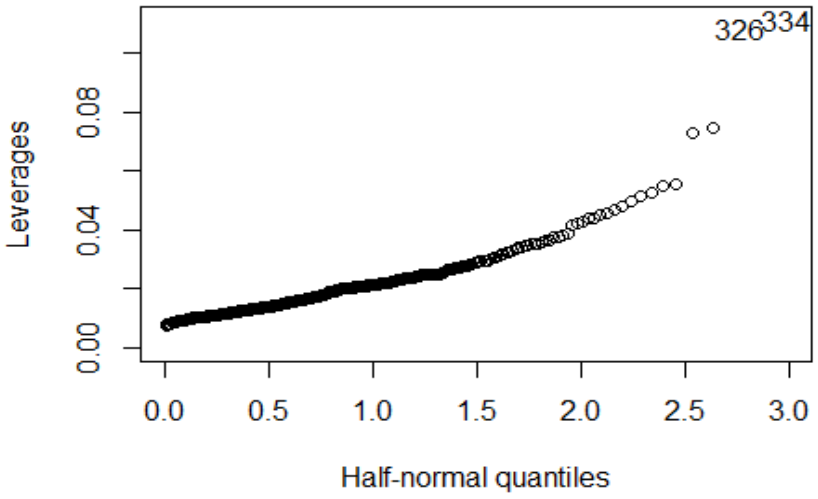
```
shapiro.test(residuals(ols1))

##
##  Shapiro-Wilk normality test
##
## data:  residuals(ols1)
## W = 0.90066, p-value = 1.681e-14
```

The Shapiro-Wilk test confirms this hypothesis: the errors aren't distributed as a normal function. Even graphically using the q-q plot it can be seen the non-normality distribution



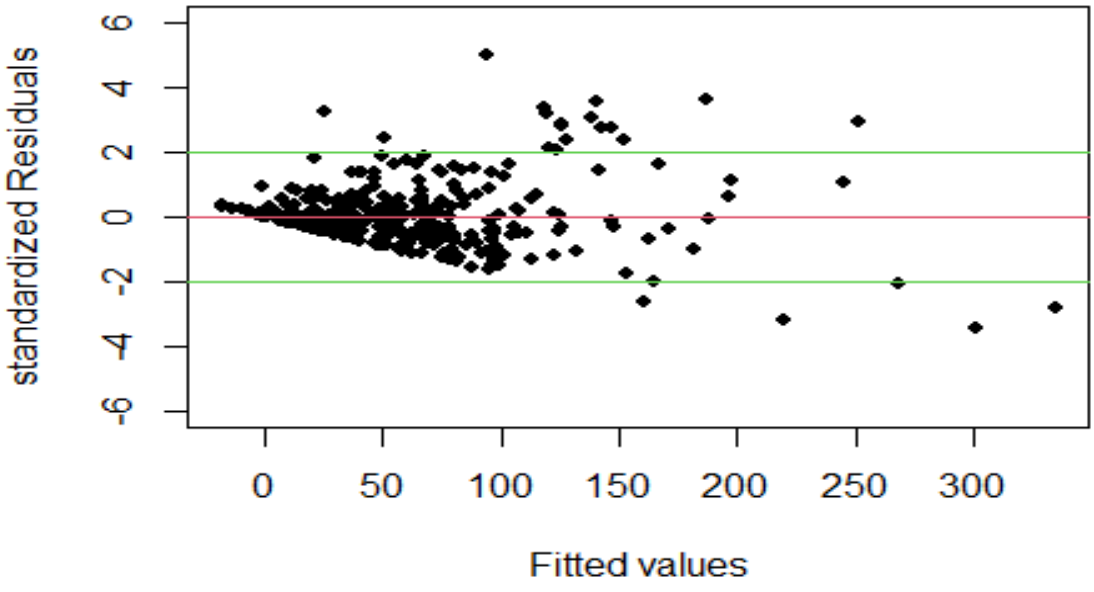
There is a big long tail issue.

Now I will check for large leverage points using halfnorm function.



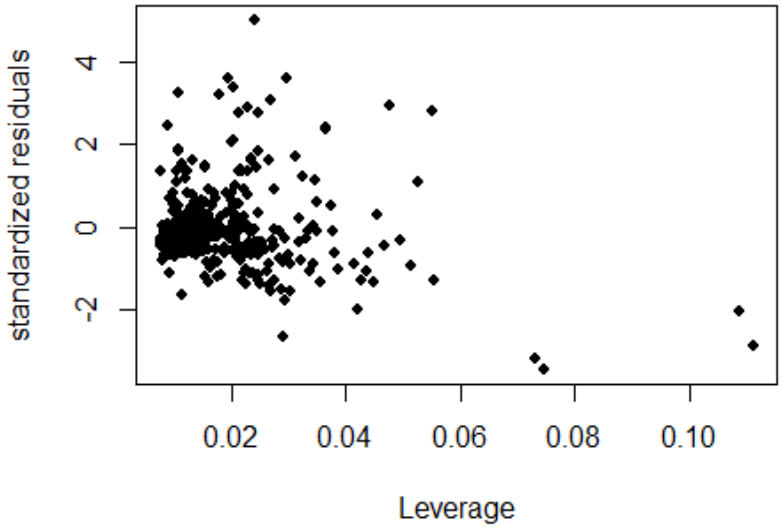
As you can see thera are a lot of leverage points(at least four).

Now I will check for outliers. I will use the standardize residual



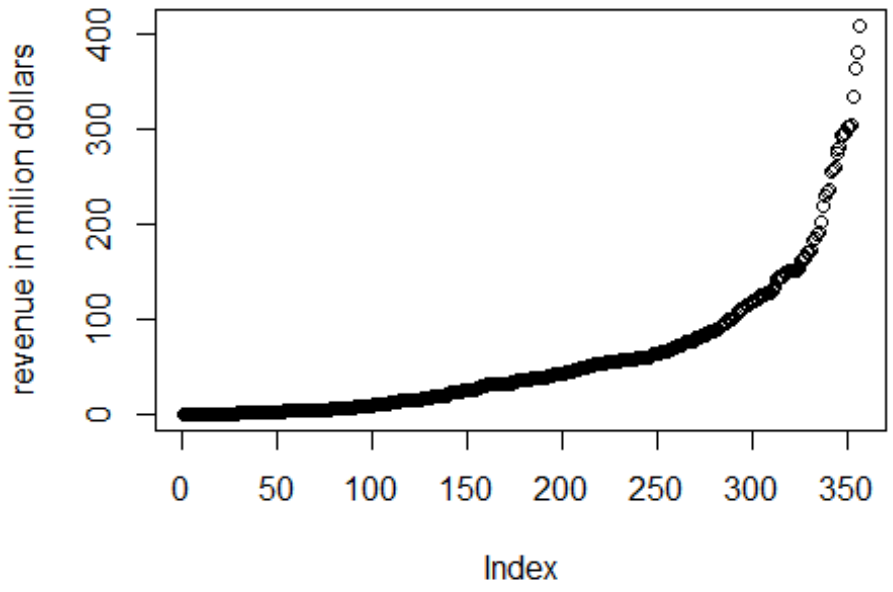
It is an outlier a point that it is outside the green lines, so it lies in the extreme 5% of the distribution. There are a lot of outliers in this graphs both in the right tail and the left tail

And lastly I will check for influential points:

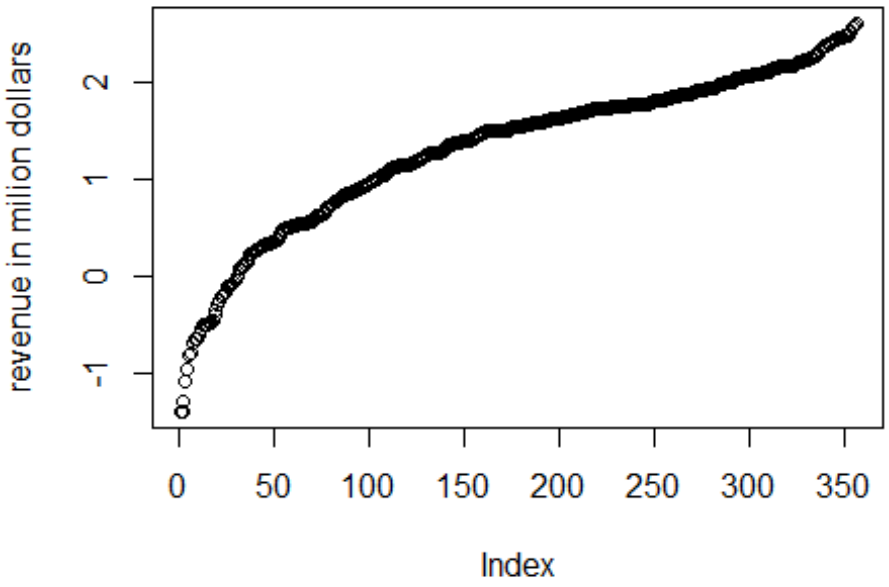


There are at least four influential points. Thanks to the diagnostics I will adjust the model in order to improve it. First of all because from the plot I noticed an exponential pattern of the revenue variable so I will use the logarithm (with base 10) of the revenue in order to linearize it:

```
plot(movies$revenue, ylab="revenue in milion dollars")
```




```
movies2 <- cbind(log_revenue = log(movies$revenue,10),movies[,-c(1)])

plot(movies2$log_revenue, ylab="revenue in milion dollars")
```



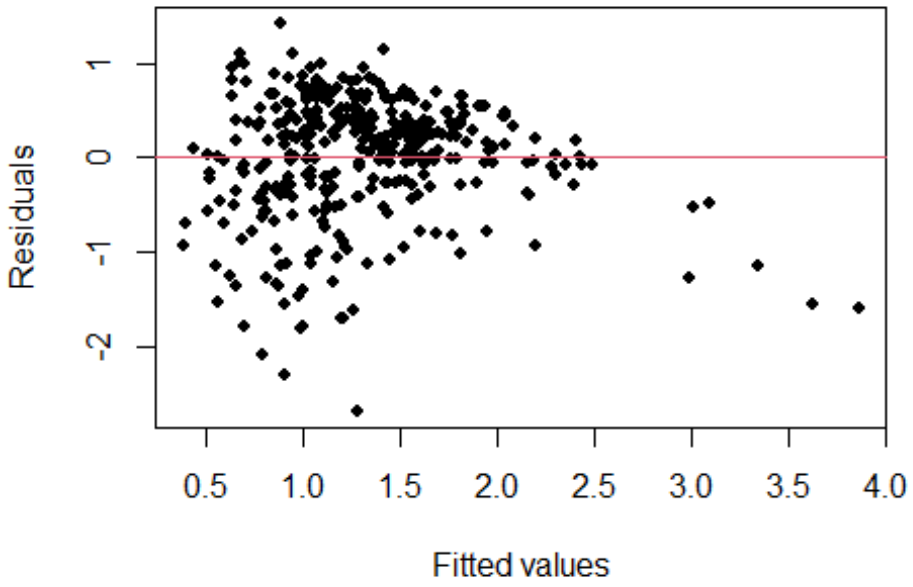
Now the revenue variable is more like to a line than the previoius one.

So I will re-run all the diagnostic analysis that I have done with the previous model

```
ols4 <- lm(log_revenue ~ ., data=movies2)

 summary(ols4)
```

```
##
## Call:
## lm(formula = log_revenue ~ ., data = movies2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6809 -0.3086  0.1480  0.4283  1.4189
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.8580190  0.2970615   6.255 1.17e-09 ***
## year         0.0043489  0.0119369   0.364    0.716
## runtime      0.0028703  0.0020350   1.410    0.159
## votes        0.0027659  0.0002564  10.787  < 2e-16 ***
## genreComedy -0.0891790  0.1001207  -0.891    0.374
## genreDrama  -0.4301373  0.0998660  -4.307 2.15e-05 ***
## avg_score   -0.1768978  0.0326977  -5.410 1.17e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6536 on 349 degrees of freedom
## Multiple R-squared:  0.3723, Adjusted R-squared:  0.3615
## F-statistic:  34.5 on 6 and 349 DF,  p-value: < 2.2e-16
```

Check the constant variance assumption for the errors:

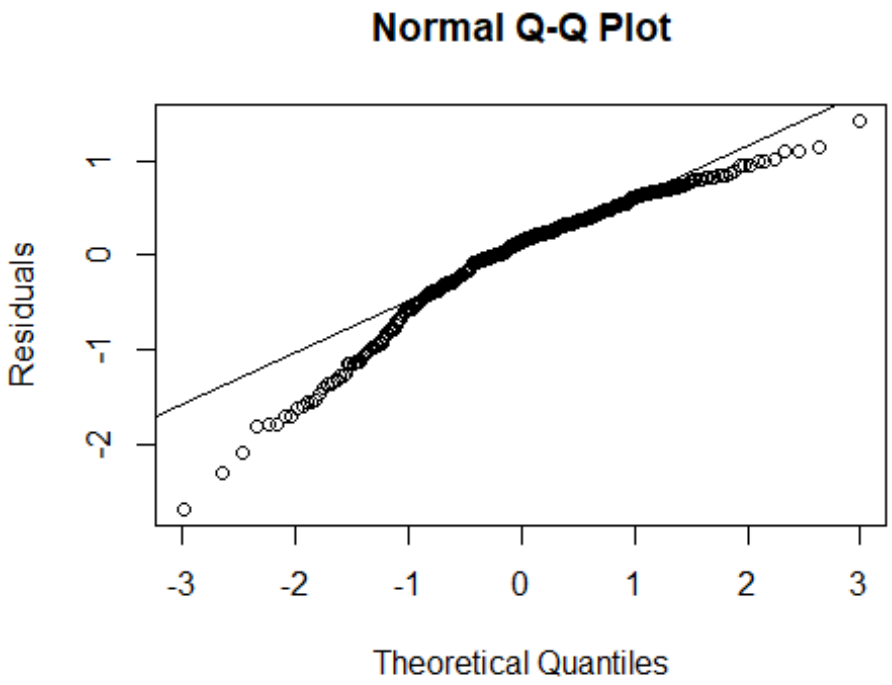


Now the graph looks like better and the variance is almost constat Check the normality assumption using Shapiro-Wilk test:

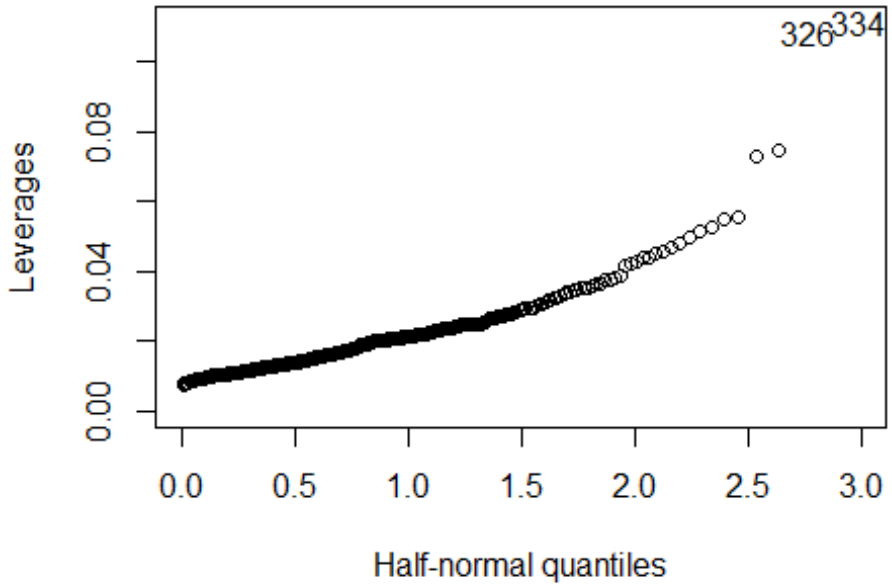
```
shapiro.test(residuals(ols4))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(ols4)
## W = 0.93572, p-value = 2.813e-11
```

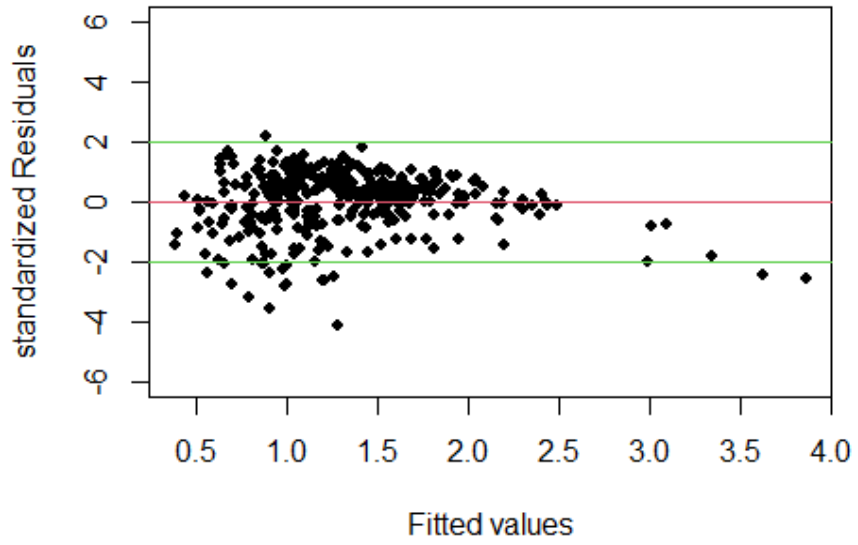
The Shapiro-Wilk test confirms that the errors aren't distributed as a normal function. Even graphically using the q-q plot it can be seen the non-normality distribution of the data



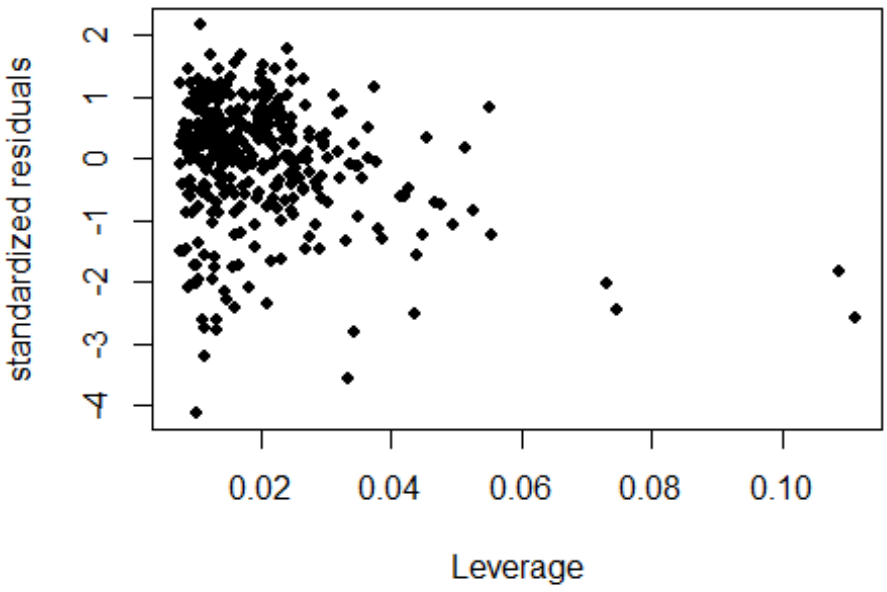
Check for large leverage points:



Check for outliers:

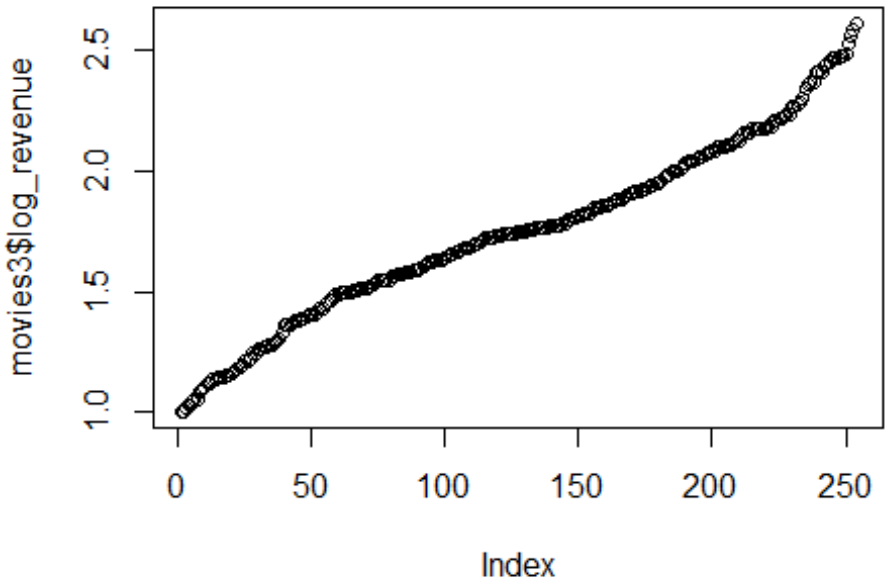


Check for influential points:



Now the model fits much better the data but there are still some issues, in particular in the left tail, so I will impove the databese removing the outliers and the most influencial points.

```
movies3 <- movies2[movies2$log_revenue>1,]

plot(movies3$log_revenue)
```



I have decided to eliminate the movies with a turnover<10 million in log scale<1 because they have given more problems in diagnostics (don't fit very well with the model, long tail problem)

So the final model will be:

$$\text{log\_revenue} = \beta 0+\beta 1 \textit{year}+\beta 2 \text{runtime}+\beta 3 \textit{votes}+\beta 4 \text{genre}+\beta 5 \textit{avg\_score}$$

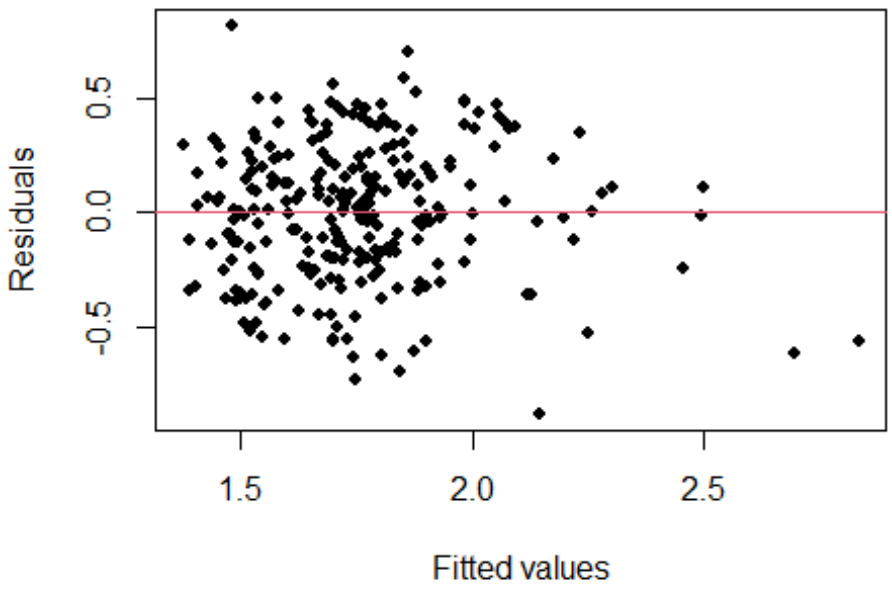
or equivalently:

$$\textit{revenue} = 10^{(\beta 0+\beta 1 \text{year}+\beta 2 \textit{runtime}+\beta 3 \text{votes}+\beta 4 \textit{genre}+\beta 5 \text{avg\_score})}$$

```
ols5 <- lm(log_revenue ~ ., data=movies3)

 summary(ols5)

##
## Call:
## lm(formula = log_revenue ~ ., data = movies3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.88021 -0.20749  0.01236  0.22232  0.81850
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.8199840  0.1823313   9.982  < 2e-16 ***
## year         0.0076840  0.0066923   1.148  0.25200
## runtime      0.0020264  0.0012136   1.670  0.09623 .
## votes        0.0010384  0.0001431   7.257 5.11e-12 ***
## genreComedy -0.1506780  0.0543474  -2.772  0.00599 **
## genreDrama  -0.3294246  0.0543529  -6.061 5.03e-09 ***
## avg_score   -0.0608272  0.0191864  -3.170  0.00171 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.3067 on 247 degrees of freedom
## Multiple R-squared:  0.3542, Adjusted R-squared:  0.3385
## F-statistic: 22.58 on 6 and 247 DF,  p-value: < 2.2e-16
```

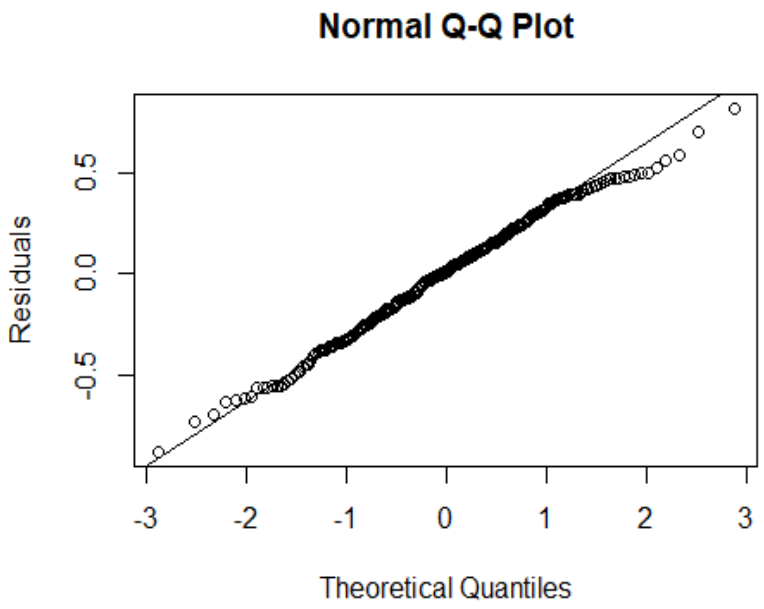
Now I will perform the diagnostic check to see the improvements of the model Check the constant variance assumption for the errors.



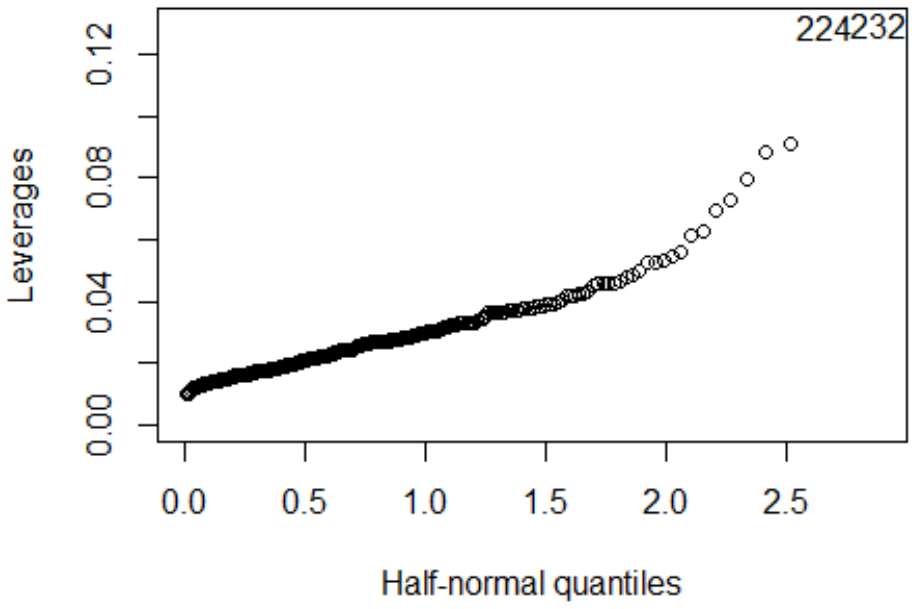
Check the normality assumption:

```
shapiro.test(residuals(ols5))
```

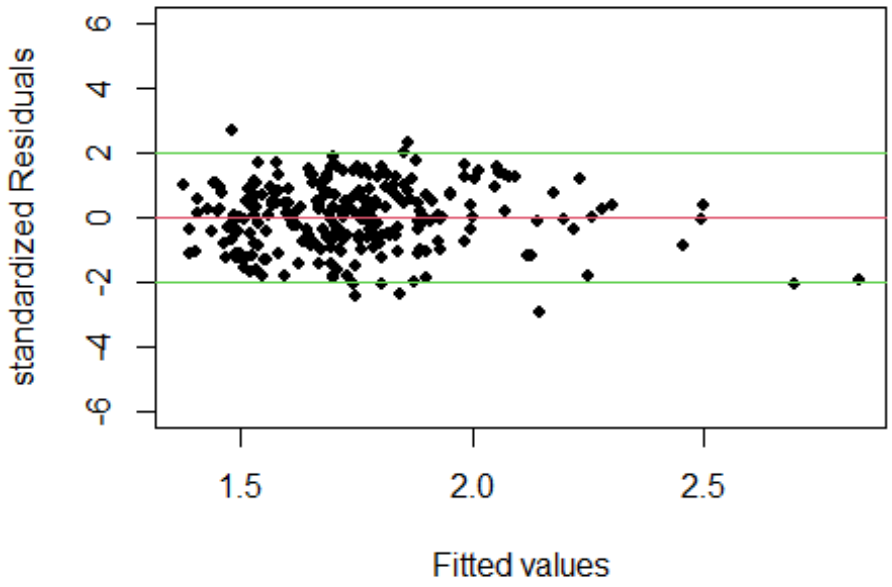
```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(ols5)
## W = 0.99276, p-value = 0.2527
```



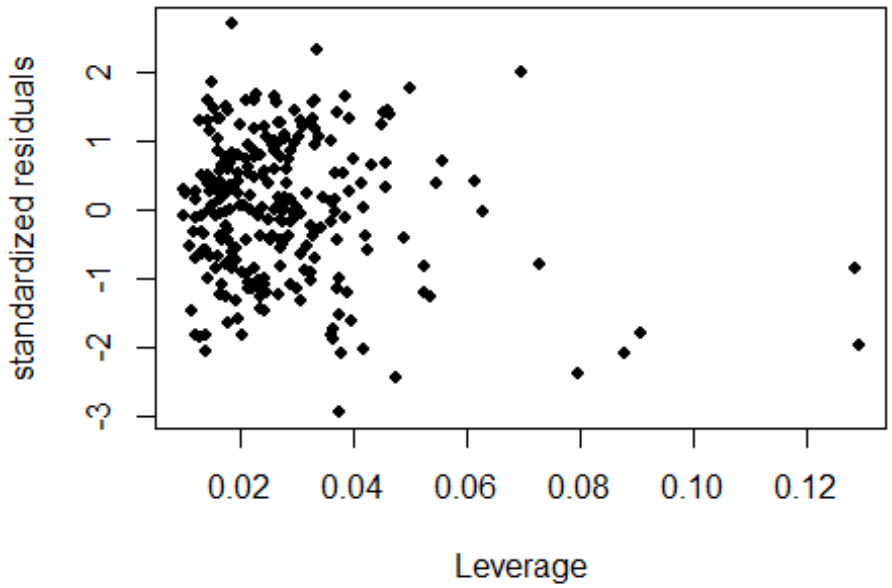
Check for large leverage points:



Check for outliers:



Check for influential points:



Now the assumption in particular the normality assumption of the errors are true.

Due to the fact that I have transformed the variable revenue from linear to logarithmic now every one unit increase of any predictors is an exponential increase of the type e ^β. The uncertainties of the betas are very low in fact in this model the sigma is only 0.3067282, the first model had 54.64, a big improvement. As I said before the sigma of the model is:

```
summ5$sigma
```

```
## [1] 0.3067282
```

and the related adjusted R^2 is:

```
summ5$adj.r.squared
```

```
## [1] 0.3385138
```

noticed that in the first model was 0.4475 and in the second one 0.3385, but in the first case the assumptions on the error aren't respected. So this means that roughly 34% of the variability of the response is explainby the model.

ANOVA TEST

```
anova(ols5,ols)
```

```
## Analysis of Variance Table
##
## Model 1: log_revenue ~ year + runtime + votes + genre + avg_score
## Model 2: log_revenue ~ 1
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    247 23.238
## 2    253 35.984 -6   -12.745 22.579 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As you can see the test where the betas are tested all together with null hypotesis all betas=0 gives a significant result, so exist at least a beta different from zero. Now I will run a test to test if each of the individually beta are different from zero.

```
anova(ols5)
```

```
## Analysis of Variance Table
##
## Response: log_revenue
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## year        1  0.1177  0.1177  1.2512  0.264402
## runtime     1  1.6718  1.6718 17.7701 3.499e-05 ***
## votes       1  5.1372  5.1372 54.6037 2.272e-12 ***
## genre       2  4.8731  2.4365 25.8980 6.156e-11 ***
## avg_score   1  0.9456  0.9456 10.0510  0.001715 **
## Residuals 247 23.2383  0.0941
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As you can see only the intercept takes individually has a p-value grater than 5% so it is legit check if there is or there are better model using ANOVA test. Now I will compare the model with the best model for the BIC criteria but now the response variable is a logarithmic:

$$\text{revenue} = \beta 0+\beta 1 votes+\beta 2 \text{genre}+\beta 3*\text{avg\_score}$$

```
anova(ols5,ols6)
```

```
## Analysis of Variance Table
##
## Model 1: log_revenue ~ year + runtime + votes + genre + avg_score
```

```
## Model 2: log_revenue ~ (year + runtime + votes + genre + avg_score) -
##     year - runtime
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    247 23.238
## 2    249 23.632 -2   -0.3939 2.0934 0.1255
```

At a firs seen the two model are equal due to high p-value of the ANOVA test, and so the secon model tha are less complex could be better but if we compare the r^2 adjusted and the sigma of the two test the full model is better. We have r^2 adjusted of BIC model

```
summ6$adj.r.squared

## [1] 0.3327046
```

that is lower than 0.3385138 the r^2 adjusted of full model, and the sigma of the BIC model is

```
summ6$sigma

## [1] 0.3080721
```

that is greater than 0.3067 the sigma of the full model.

NEW OBSERVATION

Now I suppose that I have a new observation of my regressor:

```
new_ob <- data.frame(year= 4, runtime=120,
votes=70,avg_score=6.2,genre="Adventure",log_revenue=1.55)
```

with the current model the expected and its uncertaintie of the log revenue will be:

```
predict_value

##        fit      lwr      upr
## 1 1.789451 1.676665 1.902238
```

and so the expected revenue and its uncertaintie

```
10**(predict_value)

##        fit      lwr      upr
## 1 61.58167 47.49687 79.84321
```

56.38254 millions of dollars +- a confidence interval of 14.08 million dollars

SIMULATED DATA

Now I will simulated 20 new data points from the multiple linear model fitted, assuming the estimated parameters as the true parameters.

```
p<-20
genre =NULL
for (s in 1:p){
  genre[s] <-sample(c("Adventure","Comedy","Drama"),1,replace =1)
}
p = 20
new_data <- data.frame(year= runif(p,1,11),
avg_score=rnorm(p,6.5,1),runtime=rnorm(p,105,10)
                       , votes=rnorm(p,150,45),
```

```
                      genre =genre
                      )
```

```
predict_data
```

```
##          fit      lwr      upr
## 1  1.364151 1.262590 1.465713
## 2  1.903299 1.803529 2.003069
## 3  1.836799 1.732113 1.941485
## 4  1.713532 1.621388 1.805675
## 5  1.675052 1.596673 1.753431
## 6  1.595715 1.499431 1.691999
## 7  1.579373 1.498362 1.660383
## 8  1.461728 1.385708 1.537748
## 9  1.437298 1.316345 1.558251
## 10 1.639713 1.572018 1.707408
## 11 1.755563 1.689476 1.821650
## 12 1.813184 1.711035 1.915332
## 13 1.853324 1.744923 1.961725
## 14 1.449398 1.368216 1.530580
## 15 1.581237 1.503953 1.658521
## 16 1.511773 1.427019 1.596526
## 17 1.511627 1.439031 1.584224
## 18 1.542705 1.437963 1.647447
## 19 1.637624 1.542552 1.732697
## 20 1.853034 1.739169 1.966899
```

and the corresponding revenue

```
10**(predict_data )
```

```
##          fit      lwr       upr
## 1  23.12871 18.30586  29.22219
## 2  80.03856 63.61057 100.70922
## 3  68.67504 53.96514  87.39460
## 4  51.70490 41.82040  63.92565
## 5  47.32080 39.50694  56.68012
## 6  39.41986 31.58137  49.20385
## 7  37.96408 31.50376  45.74918
## 8  28.95530 24.30572  34.49434
## 9  27.37146 20.71785  36.16191
## 10 43.62275 37.32658  50.98095
## 11 56.95908 48.91884  66.32080
## 12 65.04046 51.40852  82.28716
## 13 71.33844 55.58052  91.56396
## 14 28.14479 23.34617  33.92972
## 15 38.12742 31.91196  45.55346
## 16 32.49171 26.73122  39.49356
## 17 32.48085 27.48090  38.39050
## 18 34.89032 27.41338  44.40657
## 19 43.41347 34.87801  54.03775
## 20 71.29087 54.84906  92.66136
```