Project#1

Nathan Robinson, . . .

02/19/2020

Regression

This is a group project, and students should work in a group of size 3. Include all the R code, hypothesis testing, one or two lines of explanation for any output. The report should be organized, printed, and stapled. The due date of this project is **Wednesday** 02/19/2020.

```
Movies = read.csv("C3 2008Movies.csv")
```

The 2008Movies file contains data on movies released in 2008.

1. Calculate a regression model to predict box office from run time. Interpret the R^2 value and test statistic for the slope in the context of this problem.

```
M1 = lm(BoxOfficeGross~RunTime, data=Movies)
summary(M1)
```

```
##
## Call:
## lm(formula = BoxOfficeGross ~ RunTime, data = Movies)
##
## Residuals:
##         Min         1Q      Median         3Q         Max
## -102059739  -39266026  -20290622    17164421  457025023
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3506911   24316122   0.144    0.885
## RunTime       478843     226856   2.111    0.036 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65930000 on 204 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.02137,    Adjusted R-squared:  0.01658
## F-statistic: 4.455 on 1 and 204 DF,  p-value: 0.03601
```

```
plot(resid(M1))
```
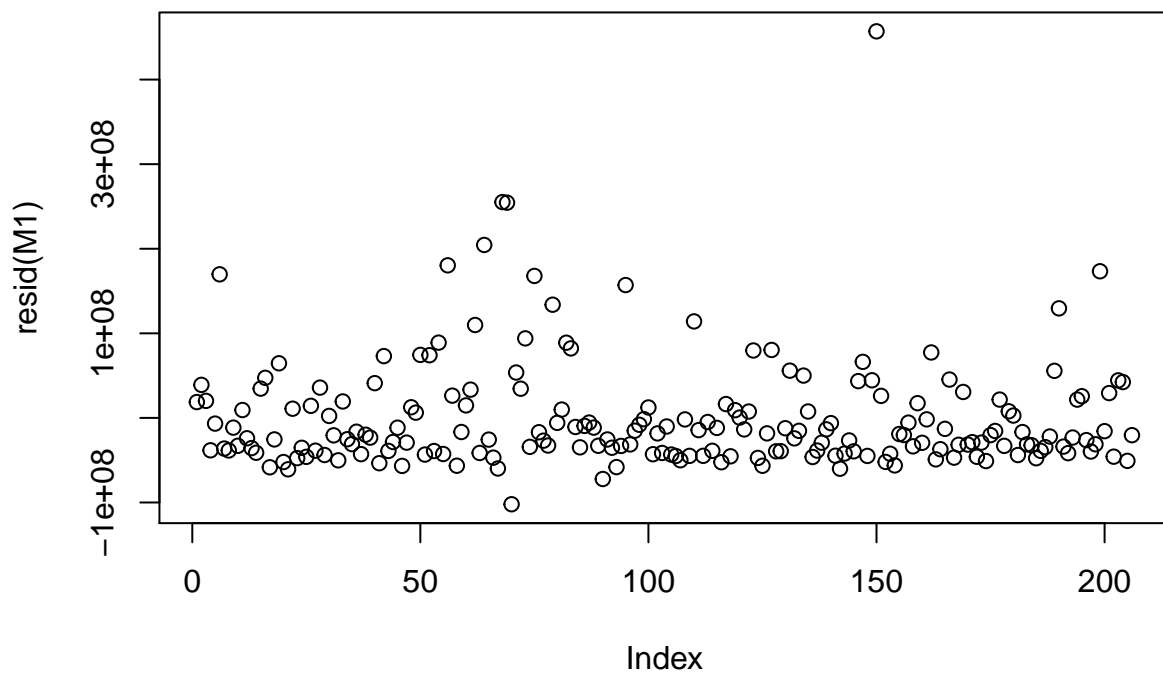
```r
qqnorm(resid(M1))
qqline(resid(M1))
```

Normal Q–Q Plot
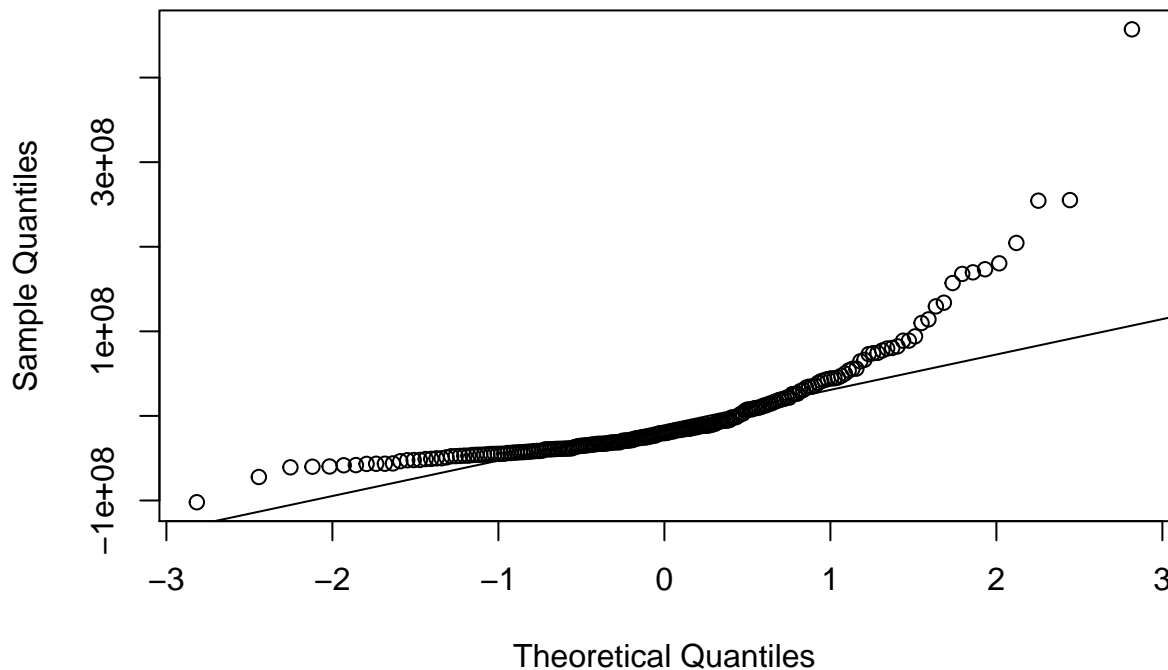
2. Create indicator variables for the genre and MPAA rating. Use the best subsets regression to determine a appropriate regression model.

```
Genre1 = as.numeric(Movies$Genre)
MPAA = as.numeric(Movies$MPAA)

library("leaps")
Model_subset = regsubsets(BoxOfficeGross~Genre + MPAA, data=Movies)
summary(Model_subset)

## Subset selection object
## Call: regsubsets.formula(BoxOfficeGross ~ Genre + MPAA, data = Movies)
## 14 Variables  (and intercept)
##                           Forced in Forced out
## GenreAdventure                FALSE      FALSE
## GenreComedy                   FALSE      FALSE
## GenreConcert/Performance      FALSE      FALSE
## GenreDocumentary              FALSE      FALSE
## GenreDrama                    FALSE      FALSE
## GenreHorror                   FALSE      FALSE
## GenreMusical                  FALSE      FALSE
## GenreRomantic Comedy          FALSE      FALSE
## GenreThriller/Suspense        FALSE      FALSE
## GenreWestern                  FALSE      FALSE
## MPAANot Rated                 FALSE      FALSE
## MPAAPG                        FALSE      FALSE
## MPAAPG-13                     FALSE      FALSE
```

```
## MPAAR                              FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##           GenreAdventure GenreComedy GenreConcert/Performance GenreDocumentary
## 1  ( 1 ) "*"            " "         " "                      " "
## 2  ( 1 ) "*"            " "         " "                      " "
## 3  ( 1 ) "*"            " "         " "                      " "
## 4  ( 1 ) "*"            " "         " "                      "*"
## 5  ( 1 ) "*"            " "         " "                      "*"
## 6  ( 1 ) " "            "*"         " "                      "*"
## 7  ( 1 ) " "            "*"         "*"                      "*"
## 8  ( 1 ) " "            "*"         "*"                      "*"
##           GenreDrama GenreHorror GenreMusical GenreRomantic Comedy
## 1  ( 1 ) " "        " "         " "          " "
## 2  ( 1 ) " "        " "         " "          " "
## 3  ( 1 ) "*"        " "         " "          " "
## 4  ( 1 ) "*"        " "         " "          " "
## 5  ( 1 ) "*"        " "         "*"          " "
## 6  ( 1 ) "*"        " "         " "          "*"
## 7  ( 1 ) "*"        " "         " "          "*"
## 8  ( 1 ) "*"        "*"         " "          "*"
##           GenreThriller/Suspense GenreWestern MPAANot Rated MPAAPG MPAAPG-13
## 1  ( 1 ) " "                    " "          " "           " "    " "
## 2  ( 1 ) " "                    " "          " "           " "    " "
## 3  ( 1 ) " "                    " "          " "           " "    " "
## 4  ( 1 ) " "                    " "          " "           " "    " "
## 5  ( 1 ) " "                    " "          " "           " "    " "
## 6  ( 1 ) "*"                    " "          " "           " "    " "
## 7  ( 1 ) "*"                    " "          " "           " "    " "
## 8  ( 1 ) "*"                    " "          " "           " "    " "
##           MPAAR
## 1  ( 1 ) " "
## 2  ( 1 ) "*"
## 3  ( 1 ) "*"
## 4  ( 1 ) "*"
## 5  ( 1 ) "*"
## 6  ( 1 ) "*"
## 7  ( 1 ) "*"
## 8  ( 1 ) "*"
```

```r
# We want row 8
# GenreComedy + GenreConcert/Performance + GenreDocumentary + GenreDrama + GenreHorror + GenreRomantic

Comedy = as.numeric(Movies$Genre == "Comedy")
Concert = as.numeric(Movies$Genre == "Concert/Performance")
Documentary = as.numeric(Movies$Genre == "Documentary")
Drama = as.numeric(Movies$Genre == "Drama")
Horror = as.numeric(Movies$Genre == "Horror")
RomCom = as.numeric(Movies$Genre == "Romantic Comedy")
Thriller = as.numeric(Movies$Genre == "Thriller/Suspense")
MPAAR = as.numeric(Movies$MPAA == "R")

Model_full= lm(BoxOfficeGross~Comedy  + Documentary + Drama + Horror + RomCom + Thriller + MPAAR, data=
summary(Model_full)
```

```
## 
## Call:
## lm(formula = BoxOfficeGross ~ Comedy + Documentary + Drama +
##     Horror + RomCom + Thriller + MPAAR, data = Movies)
## 
## Residuals:
##       Min        1Q    Median        3Q       Max
## -87403806 -34956378 -15322486  15599782 441377138
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  91938923    9290754   9.896  < 2e-16 ***
## Comedy      -32622258   12217608  -2.670 0.008213 **
## Documentary -63151227   23048950  -2.740 0.006708 **
## Drama       -50116653   13090337  -3.829 0.000173 ***
## Horror      -25167335   23297041  -1.080 0.281331
## RomCom      -40335879   22053247  -1.829 0.068901 .
## Thriller    -40776589   17191709  -2.372 0.018657 *
## MPAAR       -20904677    9660465  -2.164 0.031666 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 63550000 on 198 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.1174, Adjusted R-squared:  0.08621
## F-statistic: 3.763 on 7 and 198 DF,  p-value: 0.0007475
```

```r
Model_best = lm(BoxOfficeGross~Comedy + Documentary + Drama + Thriller + MPAAR, data=Movies)
summary(Model_best)
```

```
## 
## Call:
## lm(formula = BoxOfficeGross ~ Comedy + Documentary + Drama +
##     Thriller + MPAAR, data = Movies)
## 
## Residuals:
##       Min        1Q    Median        3Q       Max
## -78418169 -37089654 -15088470  15318455 450362775
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  82953286    8108581  10.230  < 2e-16 ***
## Comedy      -23261221   11247471  -2.068 0.039914 *
## Documentary -53919296   22656062  -2.380 0.018257 *
## Drama       -40706552   12182281  -3.341 0.000994 ***
## Thriller    -31181374   16499744  -1.890 0.060230 .
## MPAAR       -22013000    9601360  -2.293 0.022906 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 63870000 on 200 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.09966,    Adjusted R-squared:  0.07715
## F-statistic: 4.428 on 5 and 200 DF,  p-value: 0.000758
```

```
genreslist = cbind(cbind(cbind(Comedy, Documentary), Drama), Thriller)
Movies = cbind(cbind(Movies, genreslist), MPAAR)

# plot(predict(Model_best))
# plot(resid(Model_best))
```
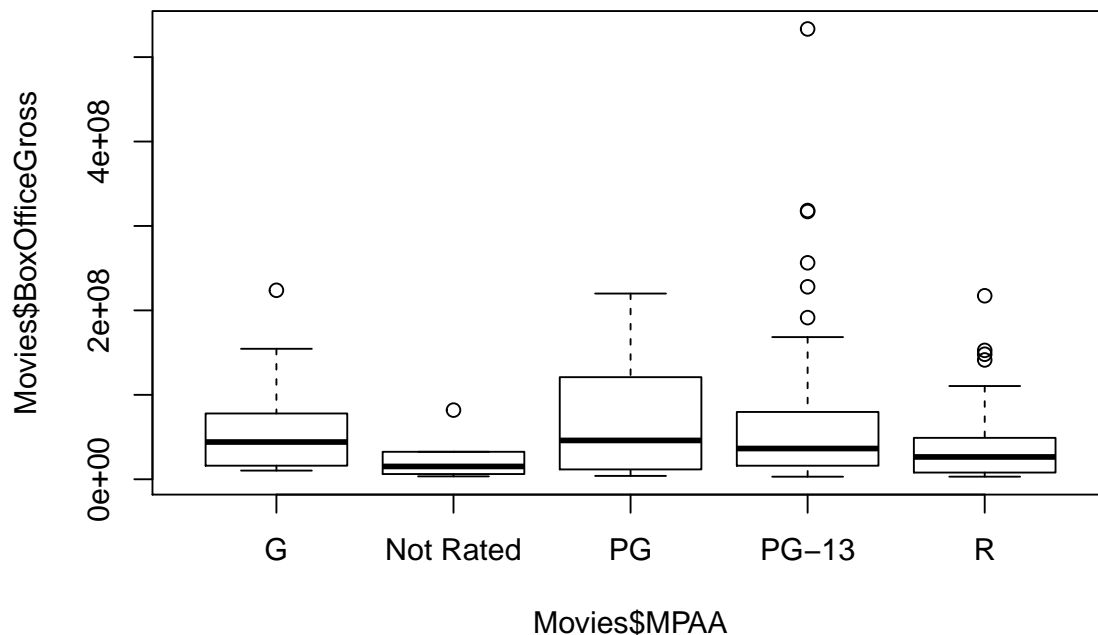
a. Validate the model assumptions.
   There is a linear relationship. By examining the coefficient of determination, it can be seen that there
   is no linear relationship.
   Multivariate normality.

```
boxplot(Movies$BoxOfficeGross~Movies$Genre)
```
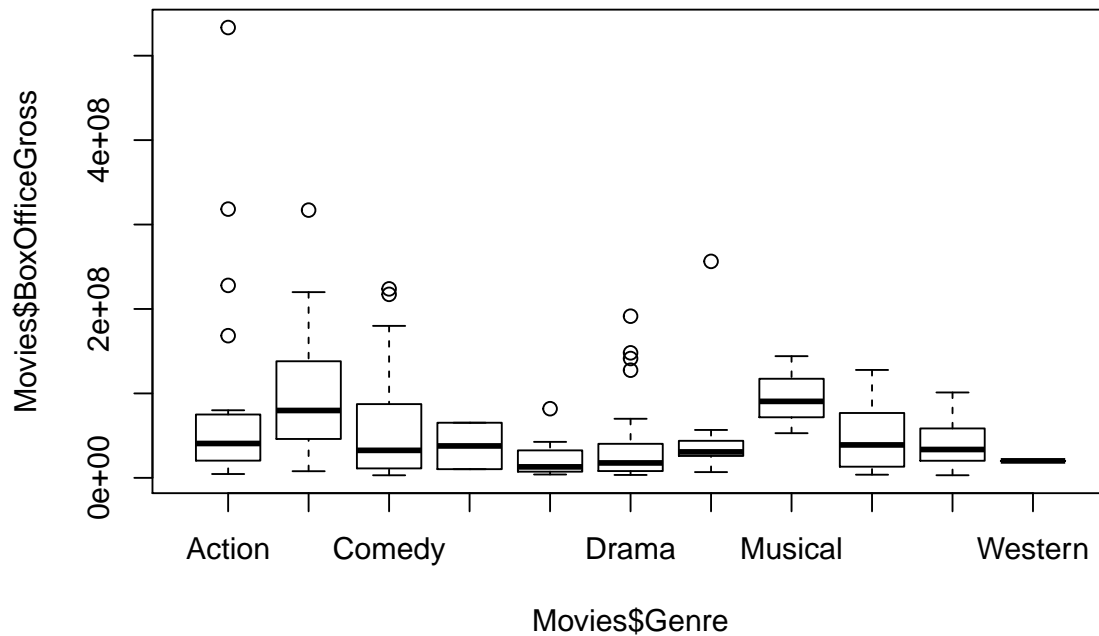


```
boxplot(Movies$BoxOfficeGross~Movies$MPAA)
```

By examining the boxplots, it can be seen that there is not multivariate normality.
Little or no multicolinearity

```
library(mctest)
imcdata = data.matrix(Movies[19:23])
imcdiag(x = imcdata, y = Movies$BoxOfficeGross)
```

```
##
## Call:
## imcdiag(x = imcdata, y = Movies$BoxOfficeGross)
##
##
## All Individual Multicollinearity Diagnostics Result
##
##                 VIF    TOL      Wi      Fi Leamer   CVIF Klein   IND1   IND2
## Comedy       1.3441 0.7440 17.2904 23.1685 0.8626 1.3089     1 0.0148 1.6390
## Documentary  1.0830 0.9233  4.1723  5.5908 0.9609 1.0547     0 0.0184 0.4908
## Drama        1.3198 0.7577 16.0711 21.5347 0.8704 1.2852     1 0.0151 1.5514
## Thriller     1.2052 0.8297 10.3115 13.8171 0.9109 1.1736     1 0.0165 1.0901
## MPAAR        1.0370 0.9643  1.8616  2.4945 0.9820 1.0099     0 0.0192 0.2287
##
## 1 --> COLLINEARITY is detected by the test
## 0 --> COLLINEARITY is not detected by the test
##
## Thriller , coefficient(s) are non-significant may be due to multicollinearity
##
## R-square of y on all x: 0.0997
##
## * use method argument to check which regressors may be the reason of collinearity
```

```
## =====================================
```

Given the F-G test, GenreComedy, GenreDrama, and GenreThriller/Suspense are all sources of multicollinearity.
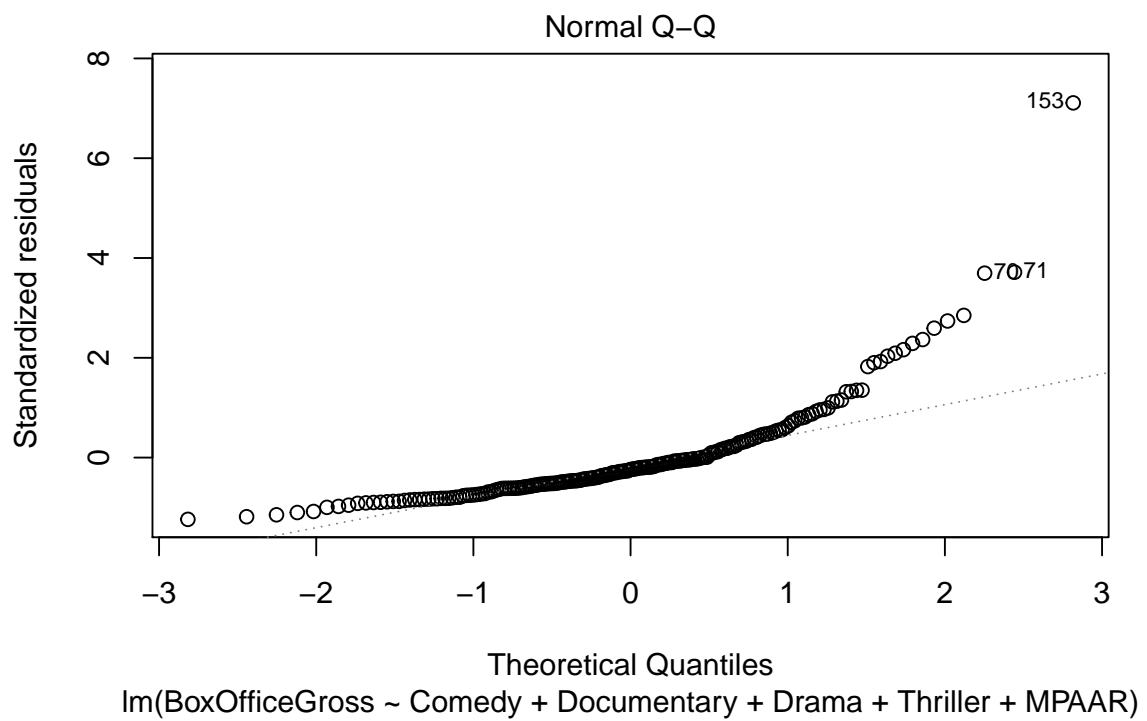
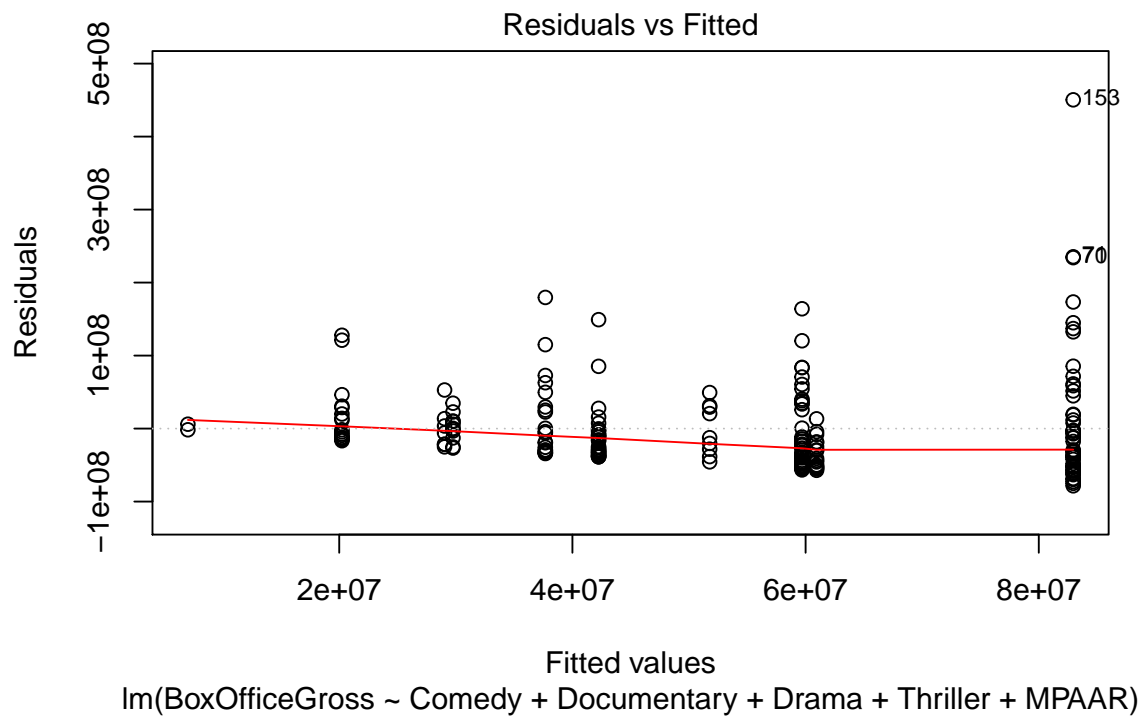Variance of error terms are similar.

```
plot(residuals(Model_best)~predict(Model_best))
abline(lm(residuals(Model_best)~predict(Model_best)))
```
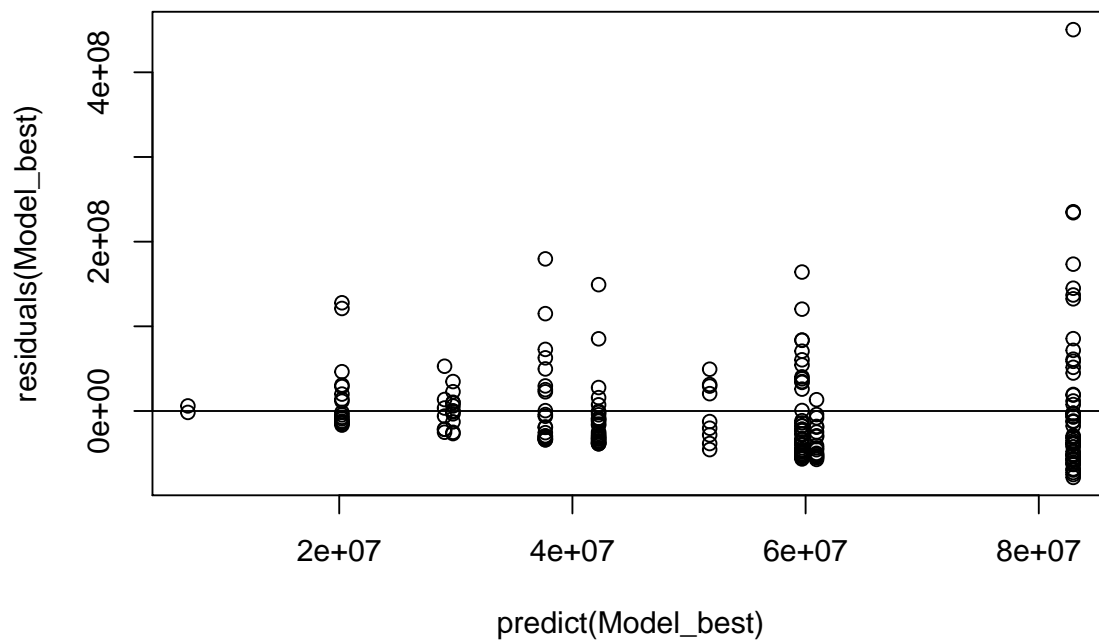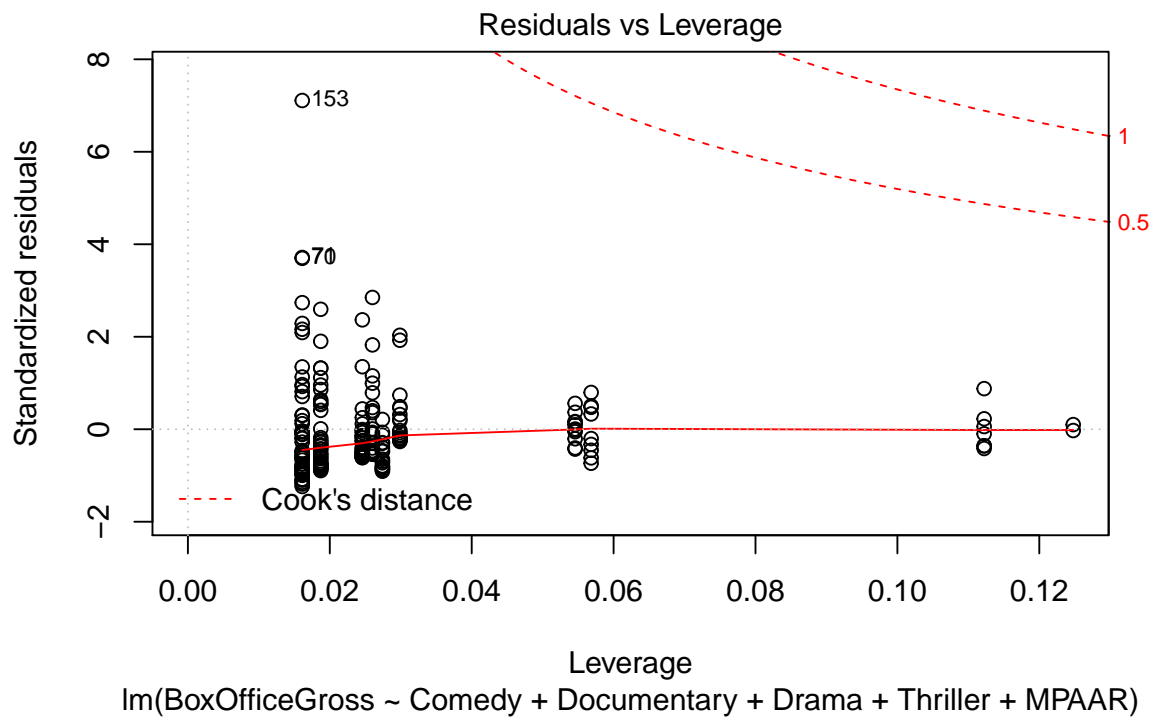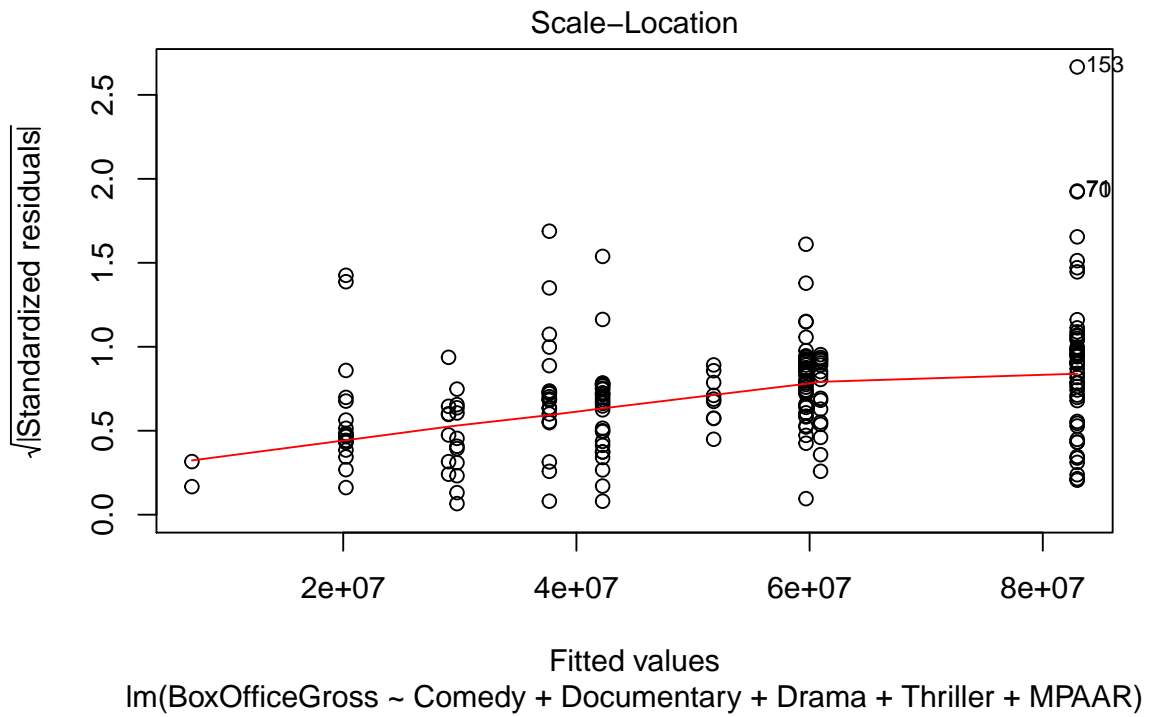

Given a plot of residuals vs predicted values, it appears that the error terms are similar and variance is not affected by how large a predicted value is.

b. Look at residual plots and check for heteroskedasticity (unequal variance), multicollinearity, correlation of errors, and outliers. Transform the data if it is appropriate.

```
plot(Model_best)
```

Residuals vs Fitted

Residuals

Fitted values
lm(BoxOfficeGross ~ Comedy + Documentary + Drama + Thriller + MPAAR)

Normal Q−Q

Standardized residuals

Theoretical Quantiles
lm(BoxOfficeGross ~ Comedy + Documentary + Drama + Thriller + MPAAR)

Scale−Location

√|Standardized residuals|

Fitted values
lm(BoxOfficeGross ~ Comedy + Documentary + Drama + Thriller + MPAAR)

Residuals vs Leverage

Standardized residuals

Cook's distance

Leverage
lm(BoxOfficeGross ~ Comedy + Documentary + Drama + Thriller + MPAAR)

```r
summary(lm(residuals(Model_best)~predict(Model_best)))
```

```
##
## Call:
## lm(formula = residuals(Model_best) ~ predict(Model_best))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -78418169 -37089654 -15088470  15318455 450362775
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -8.762e-09  1.217e+07       0        1
## predict(Model_best)  1.674e-16  2.104e-01       0        1
##
## Residual standard error: 63240000 on 204 degrees of freedom
## Multiple R-squared:  2.308e-32,  Adjusted R-squared:  -0.004902
## F-statistic: 4.709e-30 on 1 and 204 DF,  p-value: 1
```

```
boxplot(residuals(Model_best))
```
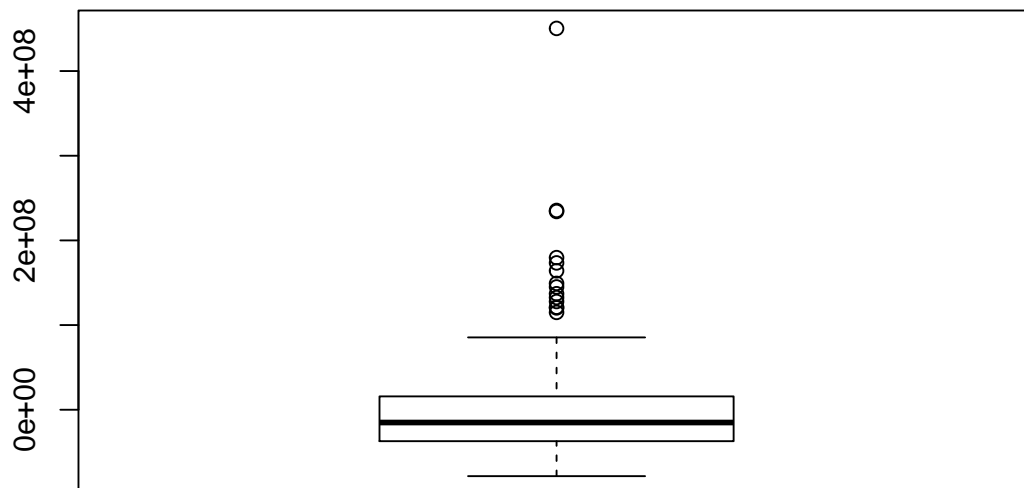

Heteroskedasticity: It was found in part a that the variance is equal, and the Residual vs. Fitted plot further shows that this is true.

Multicollinearity: It was found in part a that there are some terms which cause multicollinearity, further shown in the residual plots.

Correlation of Errors: It was found that there is no correlation for errors, given the summary statistics for the linear model for residuals vs. predicted values.

Outliers: Given the boxplot, it appears that there are quite a few outliers in the data in terms of residuals.

Transformation: Given that the independent variables in this case were all 0 or 1, it was found that most transformations would yield little to no result.

c. submit your suggested least squares regression formula along with a limited number of appropriate graphs that provide justification for your model. Describe why you believe this model is the best.

$BoxOfficeGross = 82953286 - 23261221(Comedy) - 53919296(Documentary) - 40706552(Drama) - 31181374(Thriller) + -2$

d. Test the overall model adequacy.

3. Conduct an extra sum of squares test to determine if one or more interaction terms (or quadratic terms) should be included in the model. You can choose any other terms to test.

4. Test whether average run time is the same for different Genre. Clearly show your hypothesis test.

5. Check equality of variance of run time for Genre type.