Modeling and prediction for movies

Setup

Load packages

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

Load data

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

Part 1: Data

Data provided is the random sample of 651 movies, primarily through 2 sources: Rotten Tomatoes & IMDb.

CAUSALITY: Since, this is an observational study and not experimental with random assignment, no causal inferences can be made.

GENERALIZABILITY: As the sample size is 651 and the data sources are IMDb & Rotten Tomatoes, and they have a sizable number of uSers, we can say it is a representative sample and the identified associations are generalizable.

Supporting Information: 1.IMDb has 83 million registered users.
Source:https://en.wikipedia.org/wiki/IMDb (https://en.wikipedia.org/wiki/IMDb)
2.Monthly unique visitors to the rottentomatoes.com domain is 26M global (14.4M US) according to audience measurement service Quantcast.
Source:https://en.wikipedia.org/wiki/Rotten_Tomatoes (https://en.wikipedia.org/wiki/Rotten_Tomatoes)

Concern regarding Generalizability: Since a break down of the demographics of the user data for the sites (Rotten Tomatoes, IMDb) could not be found, concern is that the users may be concentrated among few countries/or be of a paricular age range/or a particular gender. Hence the data could represent the movie perception among only that particular segment of the population.

Part 2: Research question

As casuality cannot be inferred among the variables, we aim to identify the attriButes associated to popular movies.Also, we are interested in knowing something interesting about the movies.

Part 3: Exploratory data analysis

There are 32 variables.

Popular:liked or admired by many people or by a particular person or group.

The following variables appear to be the plausible measures of a movie popularity:

Critics ratings has not been considered as we assume the popularity parameter refers to the audience. Criticallly acclaimed movies might not be popular, as critics judge the movie on the technical parameters whereas audience might not do the same.

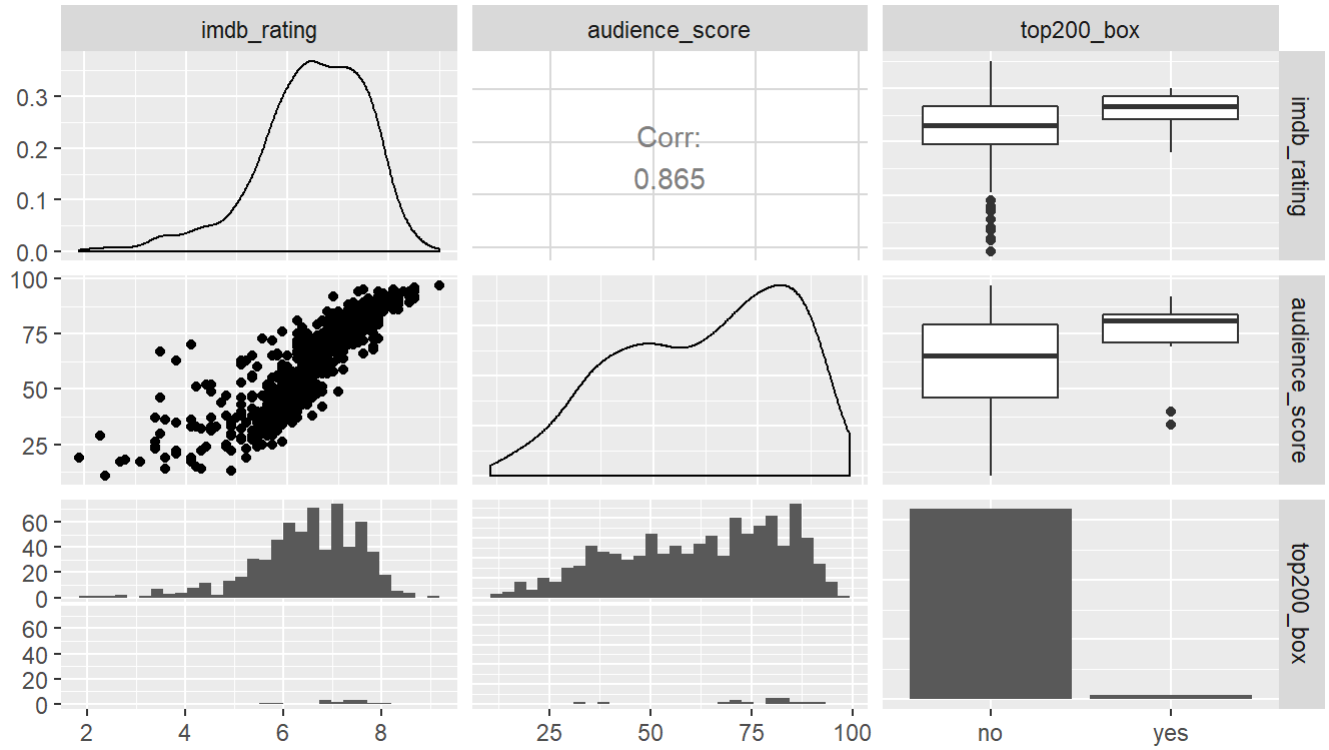
The following variables can be the plausible explanatory variables related to the popularity of the movie.

1.title_type: Type of movie (Documentary, Feature Film, TV Movie)
2.genre: Genre of movie (Action & Adventure, Comedy, Documentary, Drama, Horror, Mystery & Suspense, Other)
3.runtime: Runtime of movie (in minutes)
4.mpaa_rating: MPAA rating of the movie (G, PG, PG-13, R, Unrated)
5.studio: Studio that produced the movie
6.thtr_rel_month: Month the movie is released in theaters
7.critics_score: Critics score on Rotten Tomatoes
8.best_pic_nom: Whether or not the movie was nominated for a best picture Oscar (no, yes)
9.best_pic_win: Whether or not the movie won a best picture Oscar (no, yes)
10.best_actor_win: Whether or not one of the main actors in the movie ever won an Oscar (no, yes) – note that this is not necessarily whether the actor won an Oscar for their role in the given movie
11.best_actress win: Whether or not one of the main actresses in the movie ever won an Oscar (no, yes) – not that this is not necessarily whether the actresses won an Oscar for their role in the given movie
12.best_dir_win: Whether or not the director of the movie ever won an Oscar (no, yes) – not that this is not necessarily whether the director won an Oscar for the given movie
13.director: Director of the movie

Popularity Measure Determination:

```
ggpairs(movies,c(13, 18, 24) )
```

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

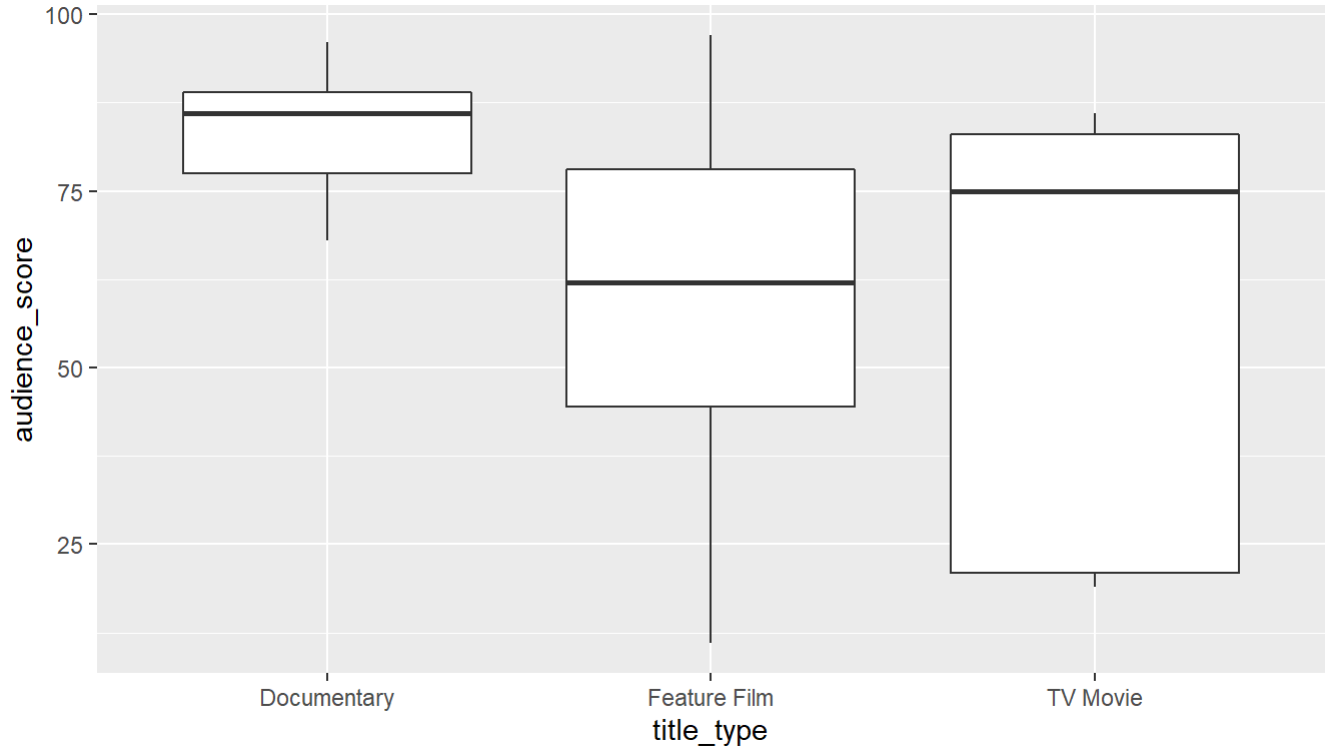


The audience_score and IMDb ratings are collinear(0.865). Hence we choose to consider only one variable i.e., audience_score. Compared to IMDb_rating, since the variability is more in audience_score. There is a significant difference between the boxplots(yes and no responses) of the top200_box variables against the audience_score and IMDb_rating.Also, Audience_score has a median as compared to IMDb_rating when we consider its association with top200 box office variable (yes).

Hence, we consider audience_score as the measure of popularity of the movie.

Audience_Score & Title Type:

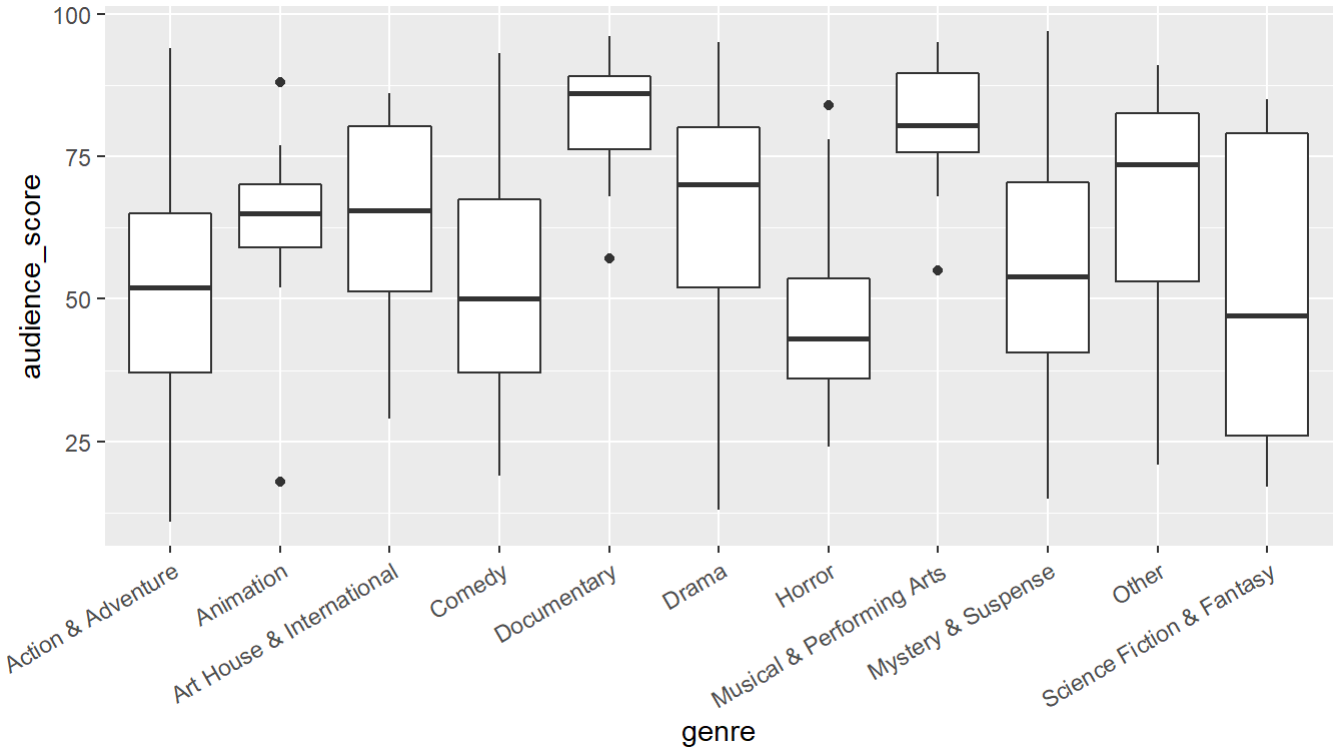
```
ggplot(movies, aes(x=title_type, y=audience_score))+geom_boxplot()
```



Documentary movies seem to have more popularity i.e., higher audience scores, as evidenced by less spread and a higher median as compared to other types(Feature Film & TV Movie).

Audience_Score & Genre:

```
ggplot(movies, aes(x=genre, y=audience_score))+geom_boxplot()+theme(axis.text.x = element_text(a
  ngle = 30, hjust = 1))
```

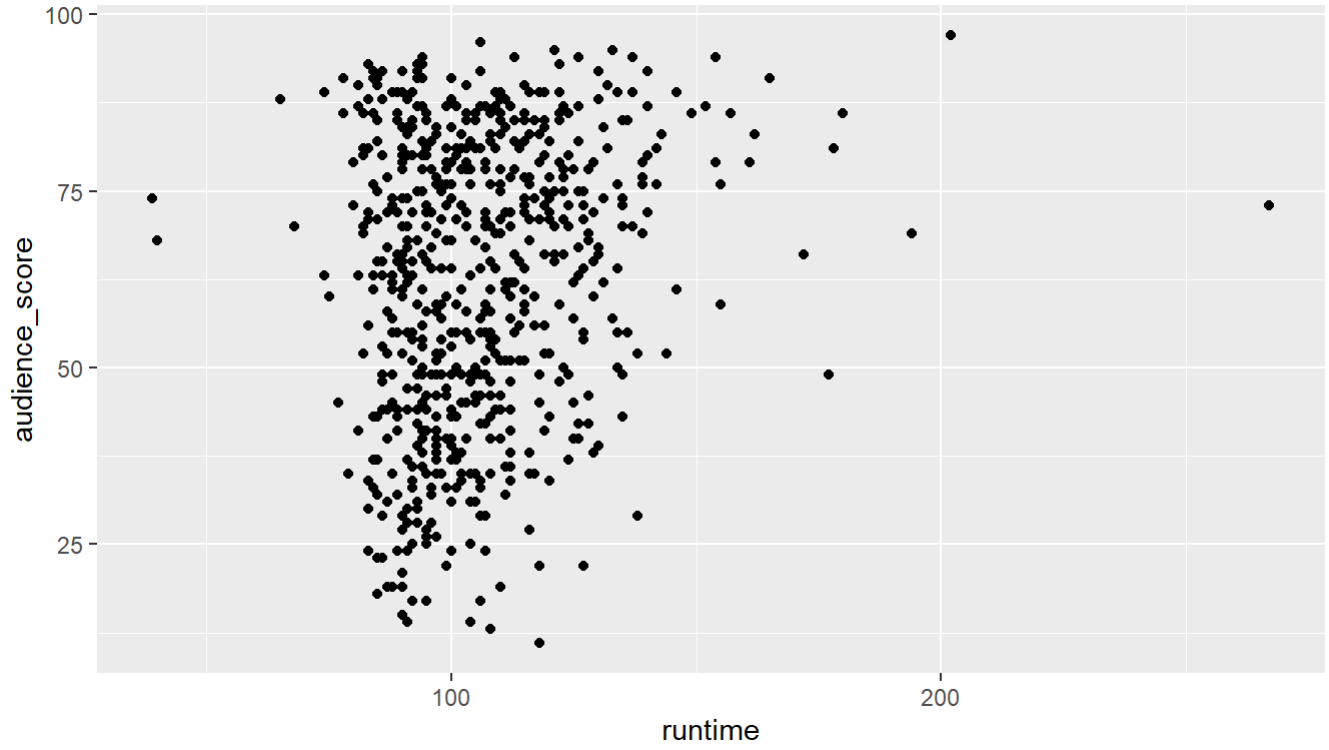


Here, the genres of Documentary followed by Musical & Performing Arts seem to have higher audience scores i.e., greater popularity (evidenced by tighter fit and higher medians). Compared to other genres, horror seems to have a lower audience score (lowest median and tight fit)

Audience_Score & Runtime:

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

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



We cannot discern any association between the variables. To reconfirm, let us check the correlation coefficient between them.

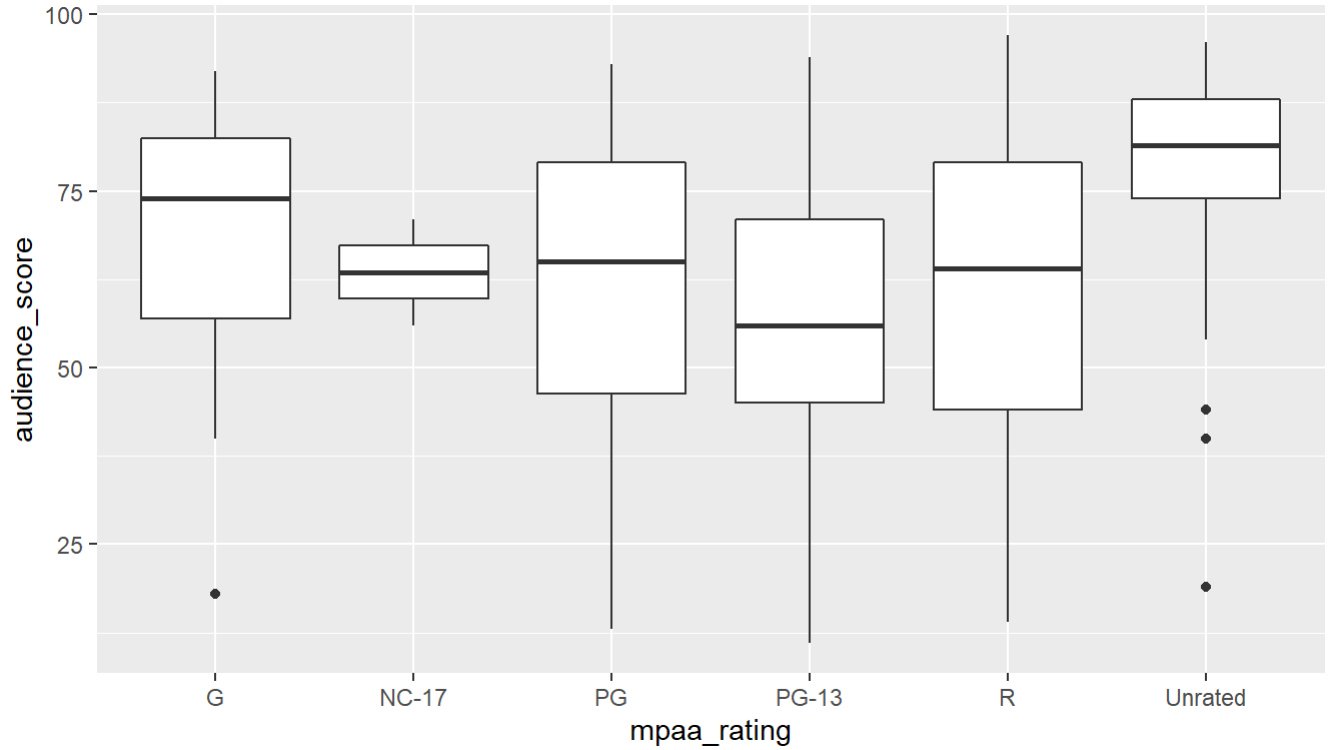
```
movies %>% filter(!(is.na(runtime))) %>% filter(!(is.na(audience_score))) %>%summarise(cor(runti
  me,audience_score))
```

```
## # A tibble: 1 x 1
##   `cor(runtime, audience_score)`
##                            <dbl>
## 1                          0.181
```

The extremely low value of correlation coefficient confirms that this attribute is not associated with popularity of the movie(audience_score).

Audience_Score & mpaa_rating:

```
ggplot(movies, aes(x=mpaa_rating, y=audience_score))+geom_boxplot()
```



Unrated movies seem to have higher popularity, based on higher median and tighter fit of the box plot as compared to others.

Audience_Score & Studio:

The idea behind trying to identify any association of movie's popularity with the studio is to find out if any studio has created a brand for itself(as producers of popular movies) among the audience.

```
length(unique(movies$studio))
```

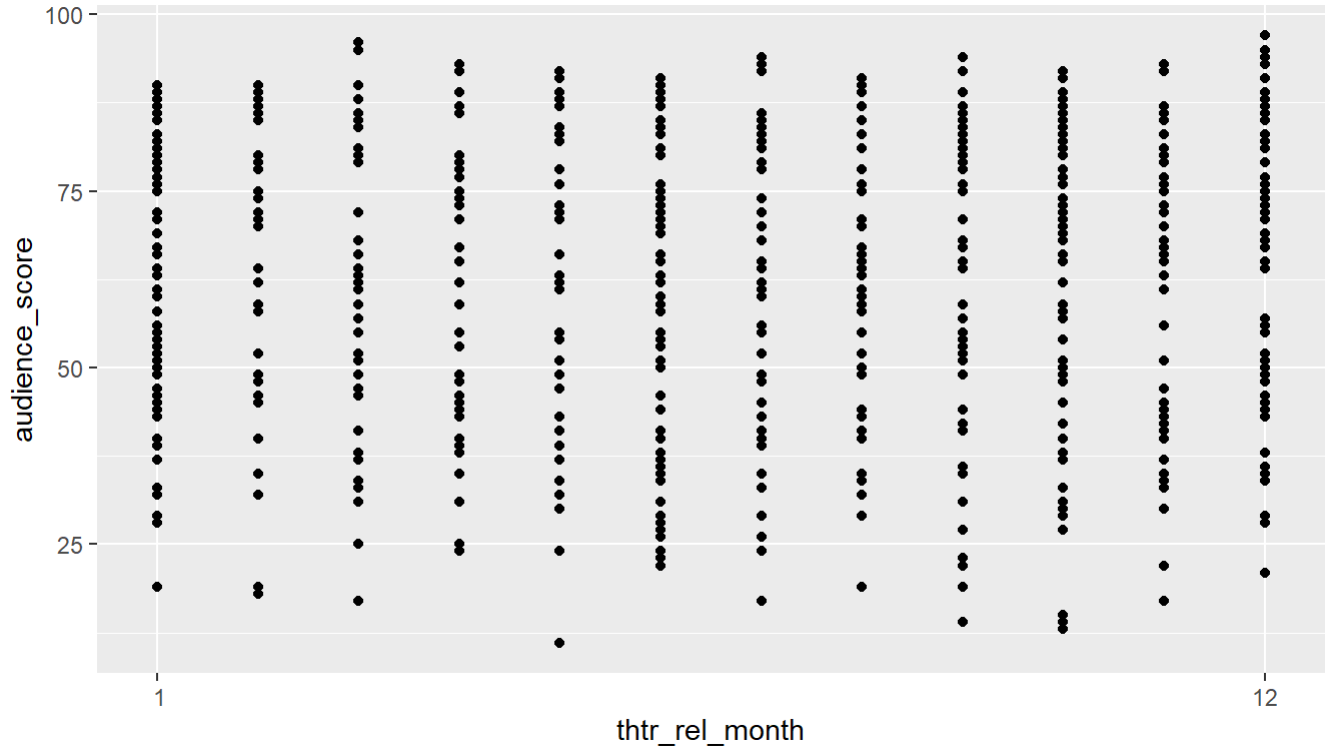
```
## [1] 212
```

There are 212 studios which have produced the 651 movies provided in this data set.The data sample is too small to identify any meaningful association between the studio of the movie and its popularity.

Audience_Score & Month of Release(thtr_rel_month):

Month is being considered here, as releasing a movie during vacations etc may lead to it being watched by a wider audience and hence might become more popular.

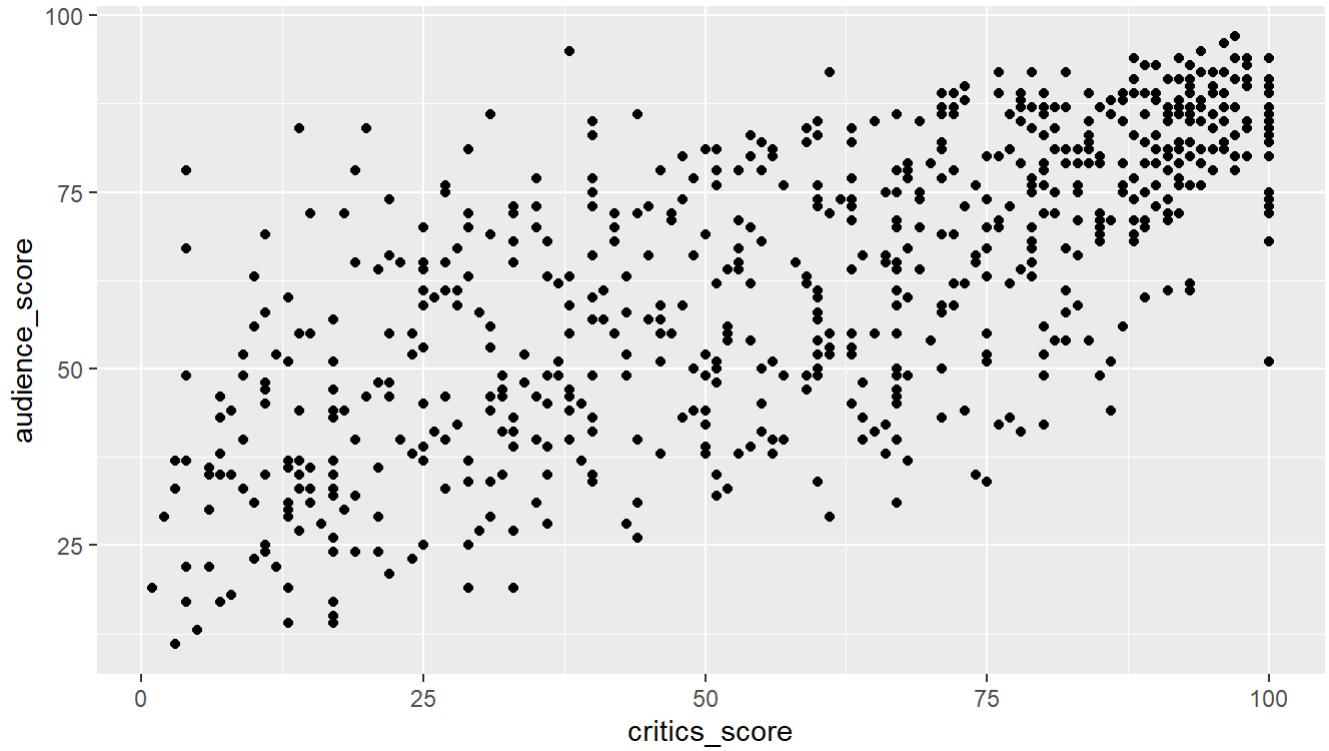
```
ggplot(movies, aes(x= thtr_rel_month, y=audience_score))+geom_point()+ scale_x_discrete(limits =
  c(1,1, 12))
```



We are unable to identify any association between the months and the popularity.

Audience_Score & Critics_Score:

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



There seems to exist a linear relationship between these variables. Let us check the correlation coefficient.

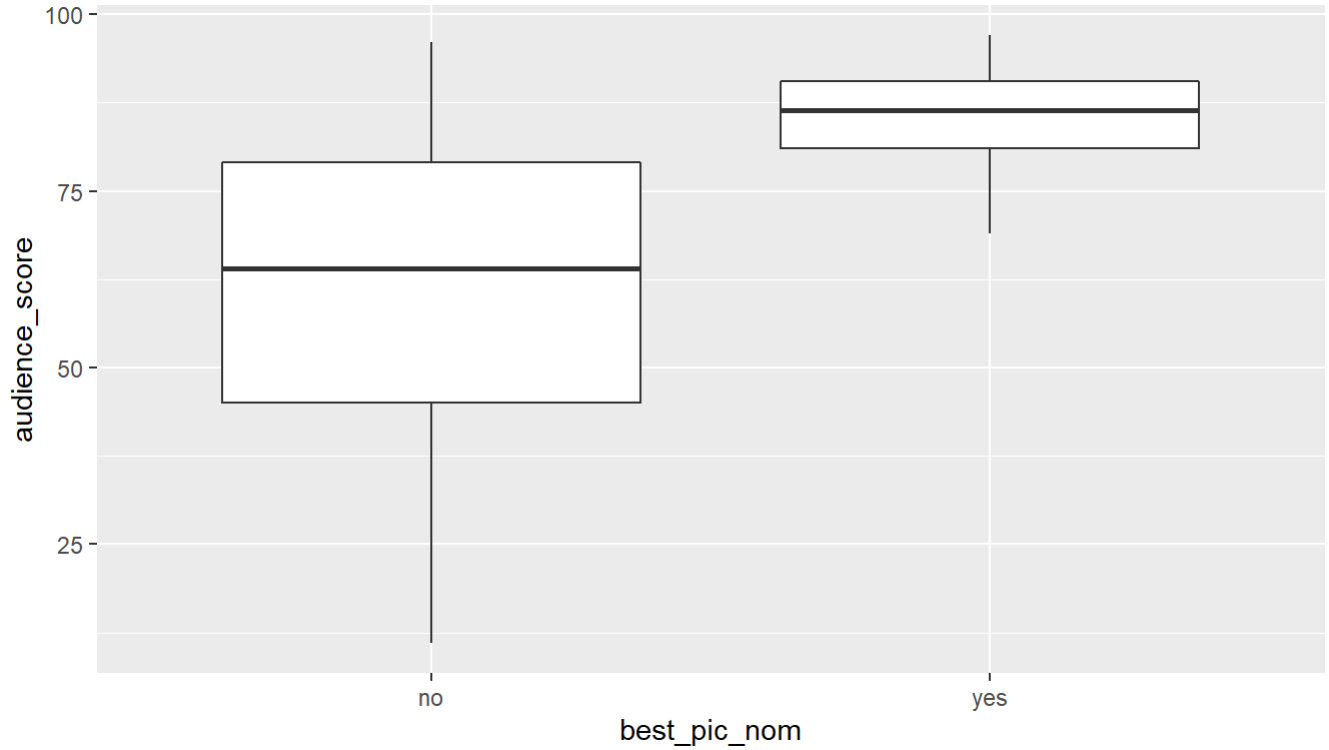
```
movies %>% filter(!(is.na(critics_score))) %>% filter(!(is.na(audience_score))) %>%summarise(cor
  (critics_score,audience_score))
```

```
## # A tibble: 1 x 1
##   `cor(critics_score, audience_score)`
##                                  <dbl>
## 1                                0.704
```

There seems to be a significant correlation between these variables. Movies with Higher critics_score tend to be associated with higher popularity.

Audience_Score & Oscar-Best Picture Nomination(best_pic_nom):

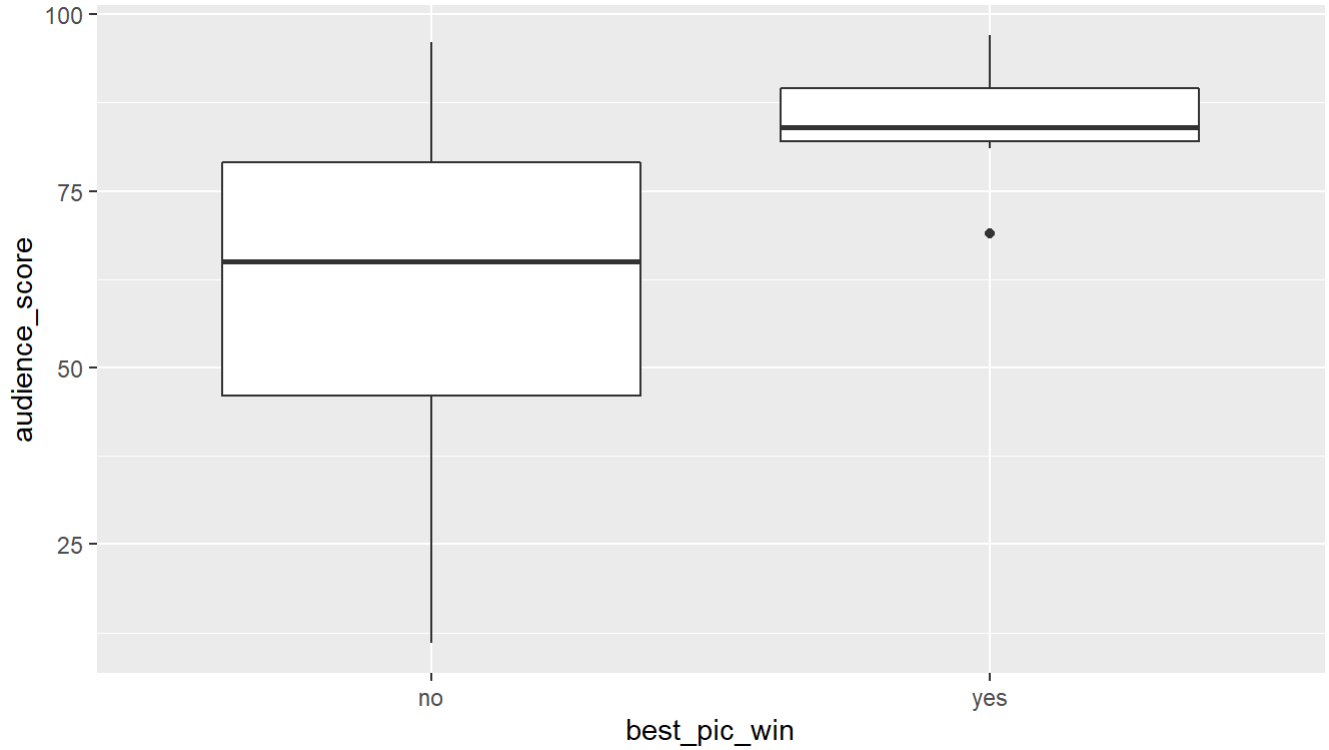
```
ggplot(movies, aes(x=best_pic_nom, y=audience_score))+geom_boxplot()
```



Movies which have received best picture nomination for Oscar seem to have higher popularity as evidenced by higher median and lower spread of the box plot.

Audience_Score & Oscar-Best Picture Winner (best_pic_win):

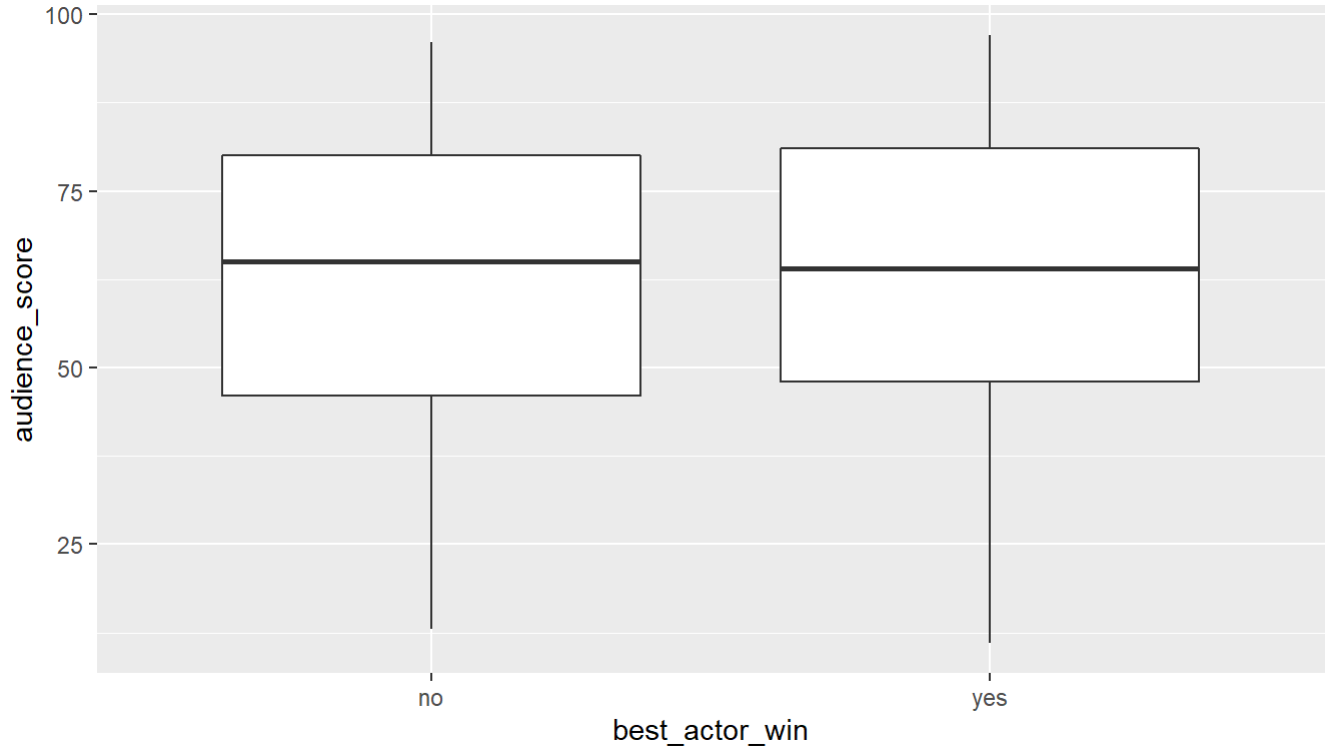
```
ggplot(movies, aes(x=best_pic_win, y=audience_score))+geom_boxplot()
```



Movies which have won an oscar for best picture seem to have higher popularity as evidenced by higher median and lower spread of the box plot.

Audience_Score & best_actor_win:

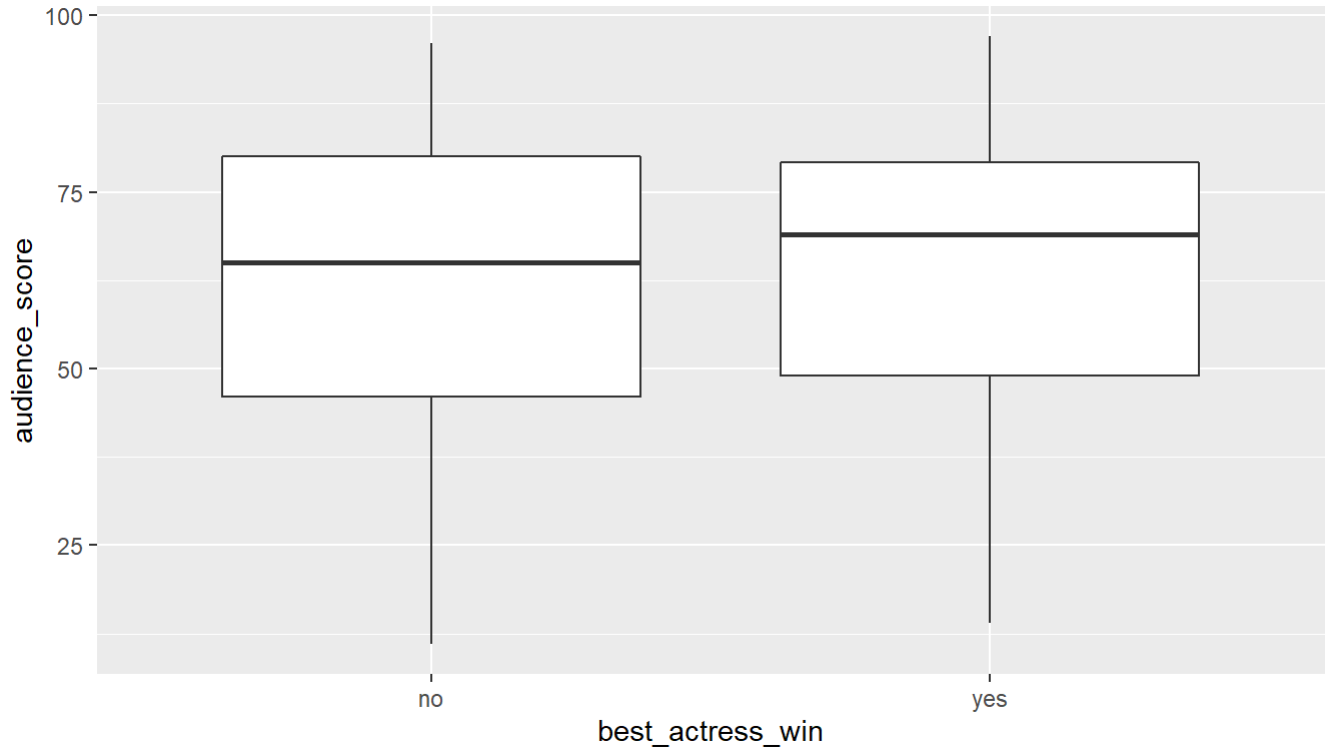
```
ggplot(movies, aes(x=best_actor_win, y=audience_score))+geom_boxplot()
```



Since there is no significant difference between the medians and the spread between the plots, movie popularity is not associated with it having an oscar winner actor in its cast.

Audience_Score & best_actress_win:

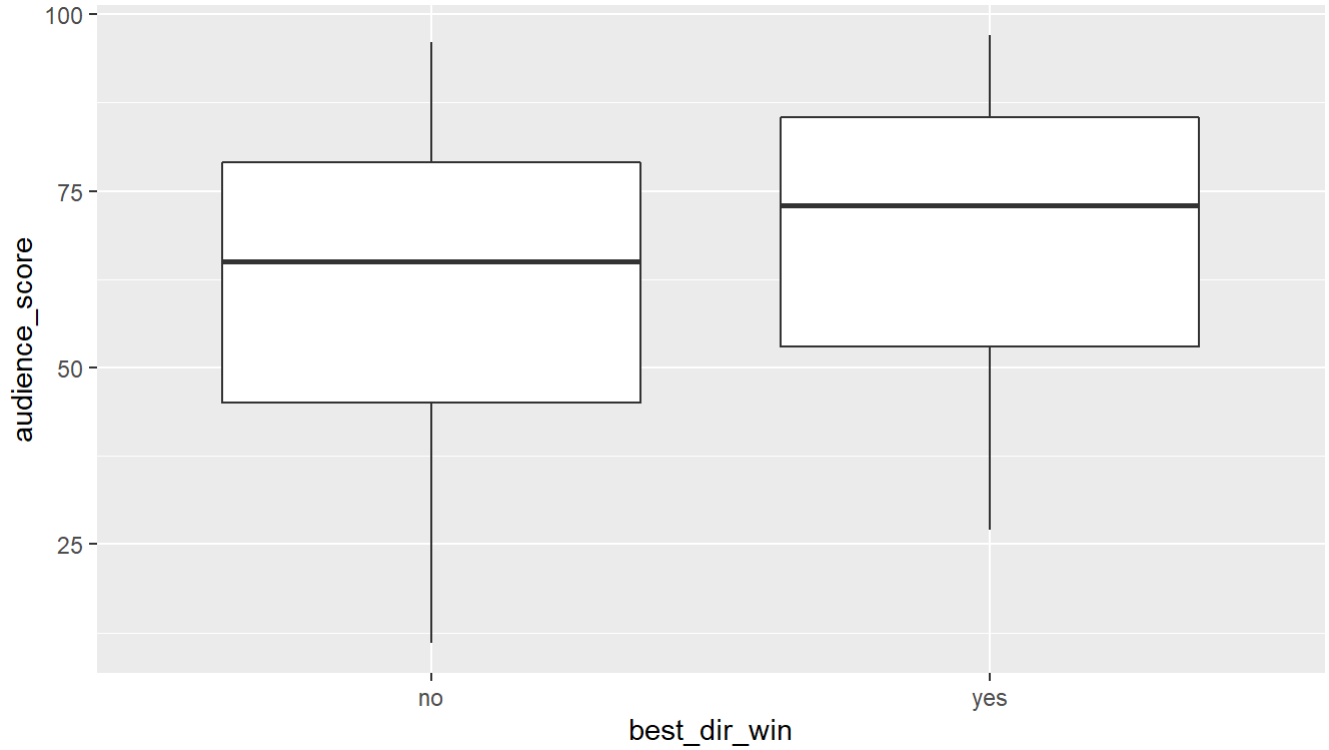
```
ggplot(movies, aes(x=best_actress_win, y=audience_score))+geom_boxplot()
```



As there is a very slight different between the medians and the spread (IQRs) between the plots, movie popularity is slightly associated with it having an oscar winner actress in its cast as the main lead. Therefore we see some value in signing an actress who has won an oscar as the main actress of the movie.

Audience_Score & best_dir_win:

```
ggplot(movies, aes(x=best_dir_win, y=audience_score))+geom_boxplot()
```



As there is a significant difference between the medians and though the spreads are same the IQR seems to be shifted upwards for movies having an oscar winning director at the helm, movie popularity is associated with it having an oscar winning director. Therefore we see some value in signing an director who has won an Oscar.

Audience_Score & Director:

The idea behind identify an association between movie popularity and its director is to check if there is any director who has been constantly producing movies with high popularity. Eg: Christopher Nolan has created a niche for himself among the audience.

```
length(unique(movies$director))
```

```
## [1] 533
```

As there are 533 unique directors within this data set containing 651 movies, the sample is too small to identify any association.

Part 4: Modeling

Our aim is to identify which movie attributes are most important in a popular movie.

```
movie_pop <- lm(audience_score ~ title_type + genre + mpaa_rating + critics_score + best_pic_nom
  + best_pic_win + best_actress_win + best_dir_win, data = movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ title_type + genre + mpaa_rating +
##     critics_score + best_pic_nom + best_pic_win + best_actress_win +
##     best_dir_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.806  -9.129   0.435   9.248  41.938
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    34.90902    6.68217   5.224 2.38e-07 ***
## title_typeFeature Film          2.06915    5.22700   0.396  0.69234
## title_typeTV Movie             -5.05182    8.19319  -0.617  0.53773
## genreAnimation                  3.98185    5.44825   0.731  0.46514
## genreArt House & International  5.75469    4.22410   1.362  0.17358
## genreComedy                    -0.79504    2.32138  -0.342  0.73210
## genreDocumentary               11.21116    5.55548   2.018  0.04401 *
## genreDrama                      2.09638    2.03293   1.031  0.30284
## genreHorror                    -9.41486    3.46459  -2.717  0.00676 **
## genreMusical & Performing Arts 11.37258    4.75982   2.389  0.01718 *
## genreMystery & Suspense        -4.15929    2.60931  -1.594  0.11144
## genreOther                      1.88567    3.99113   0.472  0.63676
## genreScience Fiction & Fantasy -6.94980    4.97851  -1.396  0.16322
## mpaa_ratingNC-17              -12.43746   10.58444  -1.175  0.24041
## mpaa_ratingPG                  -1.89996    3.84465  -0.494  0.62135
## mpaa_ratingPG-13               -2.40494    3.93121  -0.612  0.54092
## mpaa_ratingR                   -1.13420    3.80675  -0.298  0.76584
## mpaa_ratingUnrated             -2.07520    4.35707  -0.476  0.63404
## critics_score                   0.44413    0.02292  19.380  < 2e-16 ***
## best_pic_nomyes                10.56473    3.57567   2.955  0.00325 **
## best_pic_winyes                -0.60475    6.35188  -0.095  0.92418
## best_actress_winyes            -1.68775    1.82538  -0.925  0.35553
## best_dir_winyes                 0.52713    2.37893   0.222  0.82471
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.97 on 628 degrees of freedom
## Multiple R-squared:  0.5388, Adjusted R-squared:  0.5226
## F-statistic: 33.35 on 22 and 628 DF,  p-value: < 2.2e-16
```

Since p-value< 0.05, the model as a whole is significant.

Uding backwar elimiation p-vaue method, we drop the variable best_pic_win as it has the highest vale 0.92.

```
movie_pop <- lm(audience_score ~ title_type + genre + mpaa_rating + critics_score + best_pic_nom
  + best_actress_win + best_dir_win, data = movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ title_type + genre + mpaa_rating +
##     critics_score + best_pic_nom + best_actress_win + best_dir_win,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.811  -9.128   0.438   9.236  41.934
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     34.9087     6.6769   5.228 2.33e-07 ***
## title_typeFeature Film           2.0705     5.2229   0.396  0.69193
## title_typeTV Movie              -5.0566     8.1866  -0.618  0.53702
## genreAnimation                   3.9817     5.4440   0.731  0.46481
## genreArt House & International   5.7521     4.2207   1.363  0.17342
## genreComedy                     -0.8011     2.3187  -0.345  0.72985
## genreDocumentary                11.2104     5.5511   2.019  0.04386 *
## genreDrama                       2.0974     2.0313   1.033  0.30220
## genreHorror                     -9.4154     3.4619  -2.720  0.00671 **
## genreMusical & Performing Arts  11.3753     4.7560   2.392  0.01706 *
## genreMystery & Suspense         -4.1605     2.6072  -1.596  0.11105
## genreOther                       1.9067     3.9818   0.479  0.63220
## genreScience Fiction & Fantasy  -6.9457     4.9744  -1.396  0.16312
## mpaa_ratingNC-17               -12.4394    10.5761  -1.176  0.23997
## mpaa_ratingPG                   -1.8991     3.8416  -0.494  0.62122
## mpaa_ratingPG-13                -2.3977     3.9274  -0.610  0.54175
## mpaa_ratingR                    -1.1314     3.8036  -0.297  0.76623
## mpaa_ratingUnrated              -2.0747     4.3536  -0.477  0.63385
## critics_score                    0.4441     0.0229  19.396  < 2e-16 ***
## best_pic_nomyes                 10.4152     3.2099   3.245  0.00124 **
## best_actress_winyes             -1.6970     1.8213  -0.932  0.35182
## best_dir_winyes                  0.4608     2.2730   0.203  0.83940
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.96 on 629 degrees of freedom
## Multiple R-squared:  0.5388, Adjusted R-squared:  0.5234
## F-statistic: 34.99 on 21 and 629 DF,  p-value: < 2.2e-16
```

Next we drop best_dir_win variable as it has the highest value.

```
movie_pop <- lm(audience_score ~ title_type + genre + mpaa_rating + critics_score + best_pic_nom
  + best_actress_win, data = movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ title_type + genre + mpaa_rating +
##     critics_score + best_pic_nom + best_actress_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.853  -9.143   0.387   9.224  41.920
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    34.83567    6.66211   5.229 2.32e-07 ***
## title_typeFeature Film          2.11239    5.21479   0.405  0.68556
## title_typeTV Movie             -5.03838    8.17986  -0.616  0.53815
## genreAnimation                  3.96595    5.43926   0.729  0.46619
## genreArt House & International  5.71881    4.21428   1.357  0.17526
## genreComedy                    -0.81134    2.31637  -0.350  0.72626
## genreDocumentary               11.20374    5.54678   2.020  0.04382 *
## genreDrama                      2.08541    2.02888   1.028  0.30441
## genreHorror                    -9.42730    3.45872  -2.726  0.00660 **
## genreMusical & Performing Arts 11.37634    4.75236   2.394  0.01696 *
## genreMystery & Suspense        -4.15856    2.60522  -1.596  0.11094
## genreOther                      1.88078    3.97675   0.473  0.63642
## genreScience Fiction & Fantasy -6.92792    4.96985  -1.394  0.16381
## mpaa_ratingNC-17              -12.45129   10.56787  -1.178  0.23915
## mpaa_ratingPG                  -1.86367    3.83470  -0.486  0.62714
## mpaa_ratingPG-13               -2.36567    3.92121  -0.603  0.54653
## mpaa_ratingR                   -1.09537    3.79660  -0.289  0.77305
## mpaa_ratingUnrated             -2.06193    4.34987  -0.474  0.63565
## critics_score                   0.44479    0.02265  19.640  < 2e-16 ***
## best_pic_nomyes                10.48422    3.18942   3.287  0.00107 **
## best_actress_winyes            -1.68156    1.81835  -0.925  0.35544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.95 on 630 degrees of freedom
## Multiple R-squared:  0.5387, Adjusted R-squared:  0.5241
## F-statistic: 36.79 on 20 and 630 DF,  p-value: < 2.2e-16
```

Next we drop mpaa_rating. We do not drop genre as one of the values i.e., documentary is significant.

```
movie_pop <- lm(audience_score ~ title_type + genre + critics_score + best_pic_nom+ best_actress
  _win, data = movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ title_type + genre + critics_score +
##     best_pic_nom + best_actress_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.388  -9.149   0.324   9.077  42.454
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    33.04528    5.64315   5.856 7.62e-09 ***
## title_typeFeature Film          2.25485    5.15891   0.437  0.66220
## title_typeTV Movie             -4.77347    8.14731  -0.586  0.55815
## genreAnimation                  4.93068    4.95881   0.994  0.32044
## genreArt House & International  5.78342    4.10971   1.407  0.15984
## genreComedy                    -0.93890    2.29308  -0.409  0.68235
## genreDocumentary               11.08442    5.46699   2.028  0.04303 *
## genreDrama                      2.07267    1.97294   1.051  0.29386
## genreHorror                    -9.08540    3.37787  -2.690  0.00734 **
## genreMusical & Performing Arts 11.41305    4.72168   2.417  0.01592 *
## genreMystery & Suspense        -3.91784    2.53624  -1.545  0.12291
## genreOther                      1.79675    3.95193   0.455  0.64951
## genreScience Fiction & Fantasy -6.71892    4.95471  -1.356  0.17556
## critics_score                   0.44615    0.02211  20.182  < 2e-16 ***
## best_pic_nomyes                10.42411    3.17432   3.284  0.00108 **
## best_actress_winyes            -1.73871    1.80940  -0.961  0.33695
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.92 on 635 degrees of freedom
## Multiple R-squared:  0.5371, Adjusted R-squared:  0.5261
## F-statistic: 49.11 on 15 and 635 DF,  p-value: < 2.2e-16
```

Next we drop title_type variable as it has the highest value.

```
movie_pop <- lm(audience_score ~ genre + critics_score + best_pic_nom+ best_actress_win, data =
   movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ genre + critics_score + best_pic_nom +
##     best_actress_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.276  -9.157   0.277   9.167  42.416
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   35.35737    1.94888  18.142  < 2e-16 ***
## genreAnimation                 4.94668    4.95654   0.998 0.318653
## genreArt House & International  5.80001    4.10776   1.412 0.158449
## genreComedy                   -0.98981    2.28875  -0.432 0.665548
## genreDocumentary               9.02238    2.76858   3.259 0.001178 **
## genreDrama                     2.00517    1.96933   1.018 0.308970
## genreHorror                   -9.08191    3.37636  -2.690 0.007336 **
## genreMusical & Performing Arts 10.71011    4.43915   2.413 0.016118 *
## genreMystery & Suspense       -3.89909    2.53483  -1.538 0.124494
## genreOther                     1.37288    3.92885   0.349 0.726879
## genreScience Fiction & Fantasy -6.70708    4.95243  -1.354 0.176121
## critics_score                  0.44477    0.02185  20.356  < 2e-16 ***
## best_pic_nomyes               10.59477    3.16940   3.343 0.000878 ***
## best_actress_winyes           -1.77340    1.80811  -0.981 0.327064
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.91 on 637 degrees of freedom
## Multiple R-squared:  0.536,  Adjusted R-squared:  0.5266
## F-statistic: 56.61 on 13 and 637 DF,  p-value: < 2.2e-16
```

Next, we drop best_actress_win variable.

```
movie_pop <- lm(audience_score ~ genre + critics_score + best_pic_nom, data = movies)
summary(movie_pop)
```

```
##
## Call:
## lm(formula = audience_score ~ genre + critics_score + best_pic_nom,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.912  -9.413   0.263   9.303  42.404
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    35.37484    1.94874  18.153  < 2e-16 ***
## genreAnimation                  4.75334    4.95247   0.960  0.33752
## genreArt House & International  5.67764    4.10574   1.383  0.16719
## genreComedy                    -1.16710    2.28154  -0.512  0.60915
## genreDocumentary                9.00721    2.76845   3.254  0.00120 **
## genreDrama                      1.76581    1.95409   0.904  0.36652
## genreHorror                    -9.08085    3.37626  -2.690  0.00734 **
## genreMusical & Performing Arts 10.72496    4.43899   2.416  0.01597 *
## genreMystery & Suspense        -4.17514    2.51908  -1.657  0.09793 .
## genreOther                      1.23054    3.92605   0.313  0.75406
## genreScience Fiction & Fantasy -6.70347    4.95228  -1.354  0.17634
## critics_score                   0.44435    0.02184  20.342  < 2e-16 ***
## best_pic_nomyes                10.03913    3.11827   3.219  0.00135 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.91 on 638 degrees of freedom
## Multiple R-squared:  0.5353, Adjusted R-squared:  0.5266
## F-statistic: 61.25 on 12 and 638 DF,  p-value: < 2.2e-16
```

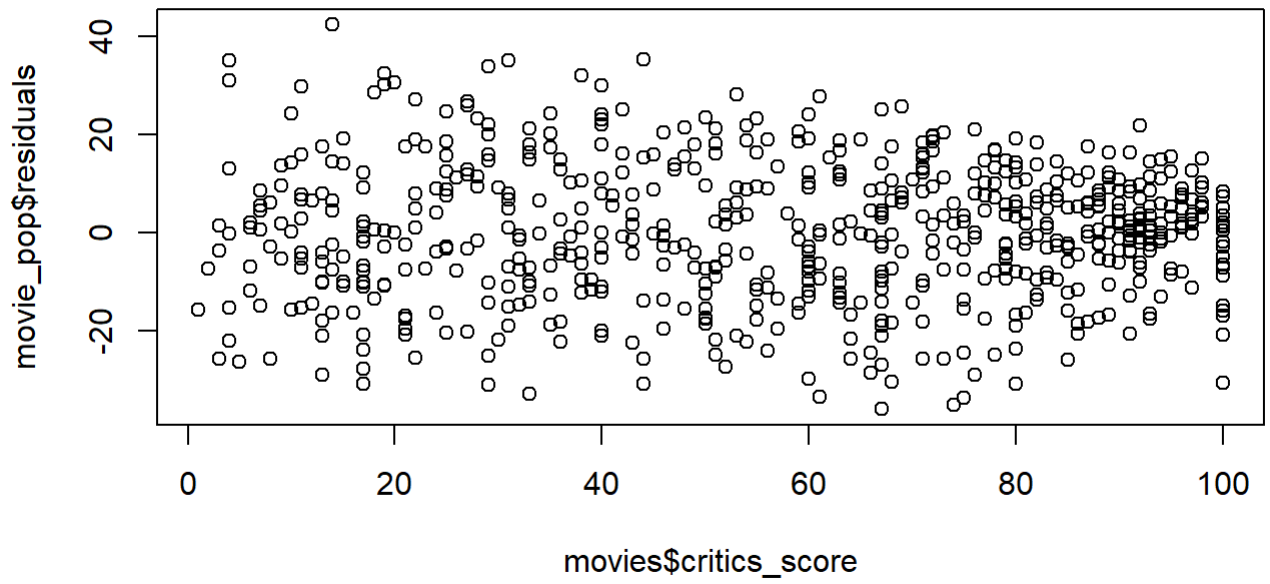
Note, there is no change in the Adjusted R-square value since last run. Also, there are no other non-significant variables to drop. Hence this is our model.

Thus, it seems the popularity of the movie is associated with its genre, critics_score and whether it has been nominated for an Oscar.

To validate it we need to check for 1.Linear relation ship between the response and explanatory variables (Applies to only numerical variables: Hence need) 2.Nearly normal residuals 3.Constant variability of residuals 4.Independence of residuals

1.Linear relation ship between the response and explanatory variables: As this applies to only numerical explanatory variables. Hence, we check this only for critic_score. Note, we consider residuals plot and a scatter plot of the variables as it allows us to consider other variables of the model.

```
movie_pop <- lm(audience_score ~ genre + critics_score + best_pic_nom, data = movies)
plot(movie_pop$residuals~ movies$critics_score)
```

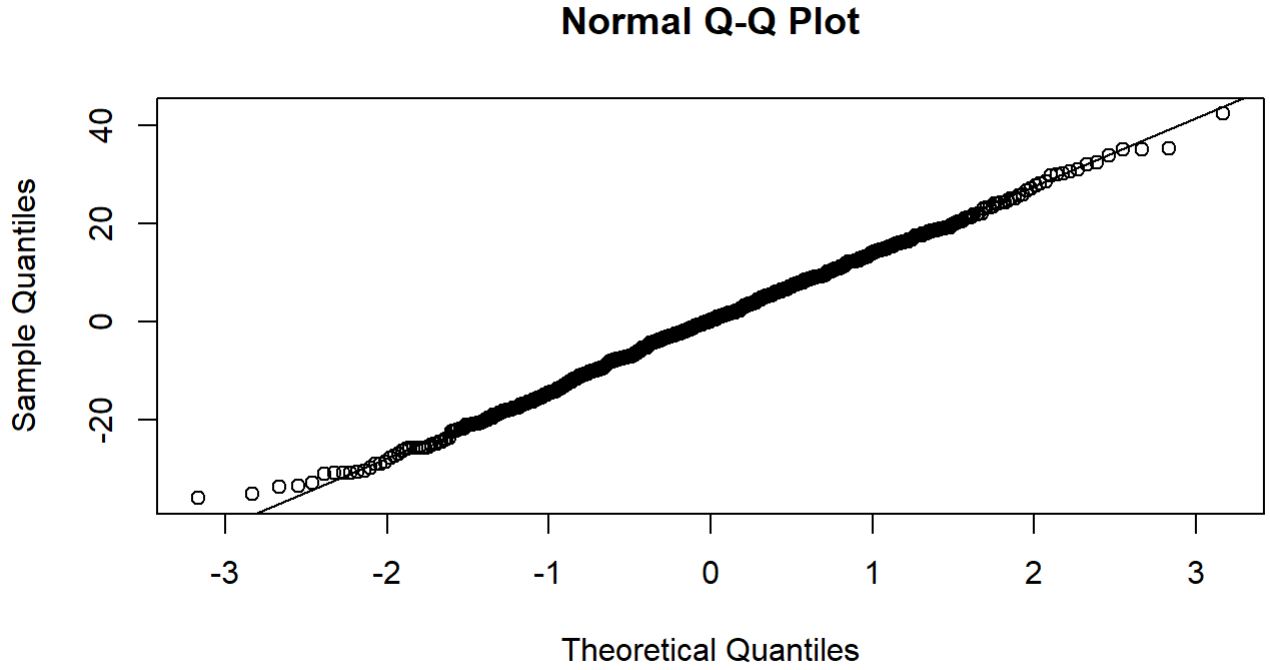


Residuals seem to be randomly scattered around 0.

2.Nearly normal residuals:

To check this

```
qqnorm(movie_pop$residuals)
qqline(movie_pop$residuals)
```



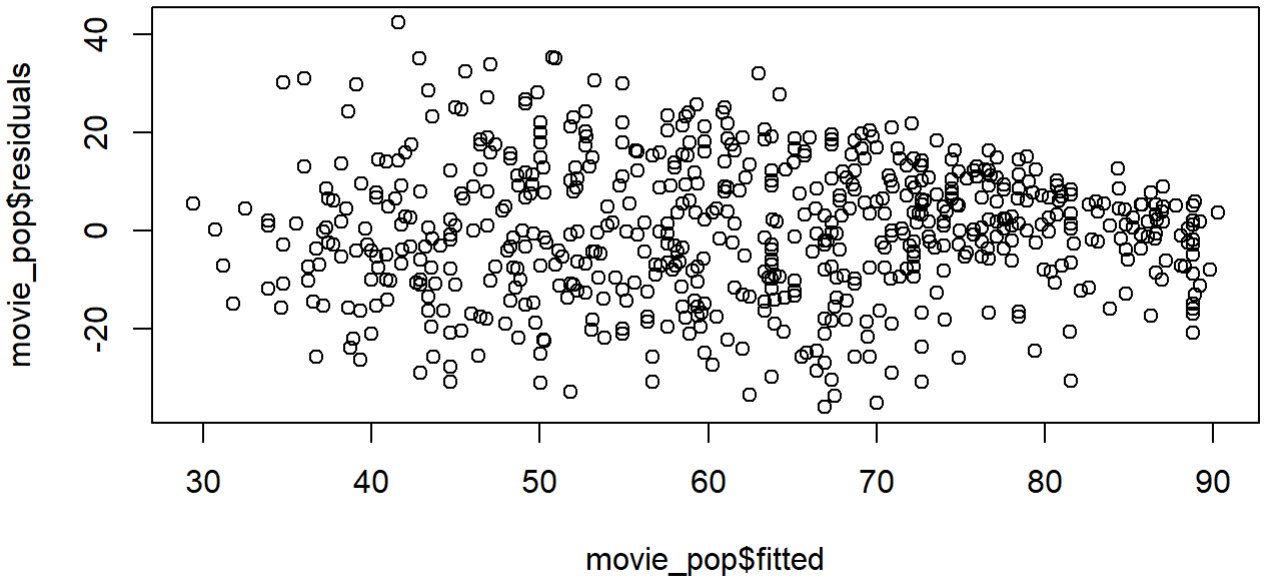
The normal probability graph indicates that this condition is satisfied.

3.Constant variability of residuals

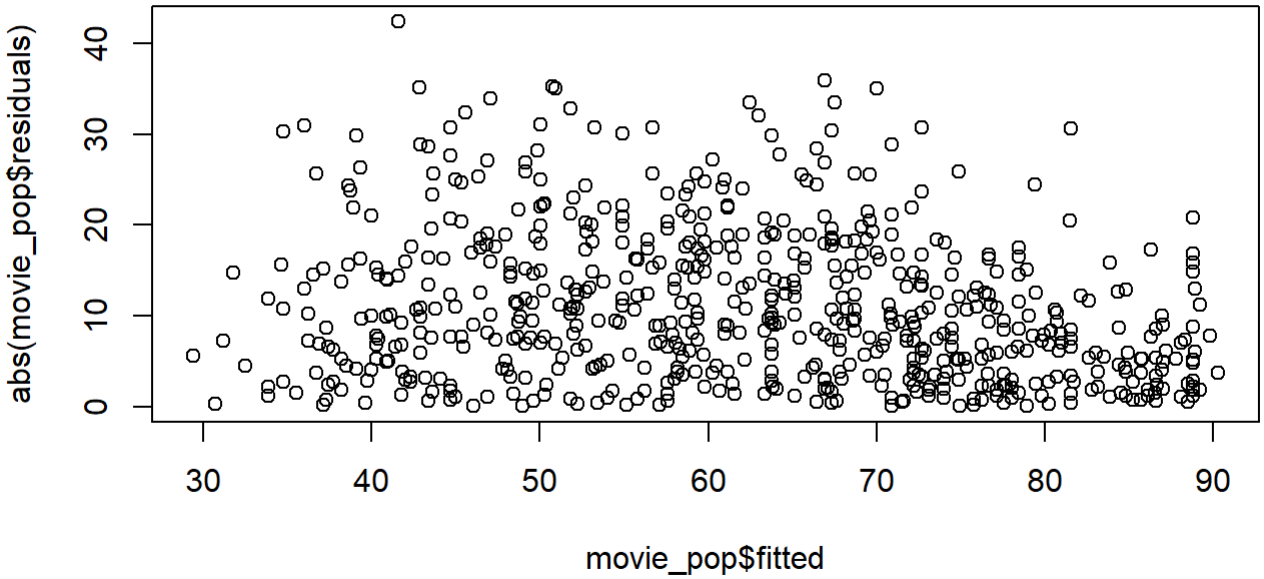
Residuals should be equally variable for high and low values of the predicted response variable.

To check this condition

```
plot(movie_pop$residuals~movie_pop$fitted)
```




```
plot(abs(movie_pop$residuals)~movie_pop$fitted)
```



The graph has a very slight fan shape, hence we to reconfirm we checked the absolute graph and through it concluded it meets the criteria.

4.Independence of residuals:

Since this a random sample, sample size is <10% of the population we can be fairly certain of the independence of the residuals.

Part 5: Prediction

So, to check the accuracy of the model, let us pick the 2016 movie La La Land which is not in the given data set. It is listed under Comedy, Drama and Musical & Performing Arts (here we consider Comedy in our prediction), has a critic score of 91 on Rotten Tomatoes, has been nominated for Oscar: Best Movie.

```
LLL<-data.frame(genre="Comedy",critics_score=91,best_pic_nom="yes")

predict(movie_pop, LLL, interval="predict")
```

```
##        fit      lwr      upr
## 1 84.68276 56.55386 112.8117
```

The predicted fit of 84.6 is pretty close to the actual audience score that is 81.

Therefore, the model is a fair predictor of movie's popularity.

Part 6: Conclusion

We started off by picking a variable(audience_score) which measures movie popularity. We then filtered through 13 variables which we considered could be associated with movie popularity. Post our exploratory data analysis, we identified 8 features that are associated with movie's popularity. We then worked on our model using backwards p-value method and finally identified 3 features which are statiscially significant and predict the movie's popularity. Further, we conducted the diagnotics for Multiple Linear Regression Model to ensure it means the criteria.We then tested our model with an independent data value and concluded the model is a good predictor.

The model can be further enhanced if we have the entire dataset. We can predict maybe even the directors/actresses etc for future movies in a bid to have some confidence that the movie will be popular. Studios may be interested in such a model.