SAP projekt - Analiza filmova na IMDb-u

Mark Sarić, Jakov Vlahović, Duje Budiselić, Kristijan Šagovac

2025-01-25

Motivacija i opis problema

IMDB je najveća javno dostupna baza podataka o filmovima, serijama, glumcima, redateljima, scenaristima i mnoštvu drugih informacija vezanih uz kinematografiju. Osim samog prikaza podataka, IMDB dopušta korisnicima i ocjenjivanje te komentiranje filmova i serija. Analizom takvog skupa podataka moguće je doći do raznih zanimljivih zaključaka kao što su preferencije korisnika ili koje značajke čine film popularnim.

Opis skupa podataka

Podatci se sastoje od opisa 5044 filmova preuzetih s internetskih stranica IMDB-a. Uz svaki film dane su i općenite informacije poput žanra filma, godine izdanja, zemlje podrijetla ili imena redatelja. Dodatno, dani su i metrički podatci kao što su budžet filma, zarada filma, IMDB ocjena i slično.

Istraživačka pitanja

Pitanje 1: Postoji li razlika u zaradi filmova namijenjenih za opću publiku (PG) i filmova namijenjenih za osobe određene dobi (R)?

Učitajmo podatke.

```
filmovi = read.csv("Datasets/movie_IMDB.csv")
dim(filmovi)
```

```
## [1] 5043   28
```

```
head(filmovi)
```

```
##   color     director_name num_critic_for_reviews duration
## 1 Color     James Cameron                   723      178
## 2 Color    Gore Verbinski                   302      169
## 3 Color        Sam Mendes                   602      148
## 4 Color Christopher Nolan                   813      164
## 5             Doug Walker                    NA       NA
## 6 Color    Andrew Stanton                   462      132
##   director_facebook_likes actor_3_facebook_likes     actor_2_name
## 1                       0                    855 Joel David Moore
## 2                     563                   1000    Orlando Bloom
```

```
## 3                      0                   161     Rory Kinnear
## 4                  22000                 23000   Christian Bale
## 5                    131                    NA       Rob Walker
## 6                    475                   530  Samantha Morton
##   actor_1_facebook_likes     gross                          genres
## 1                   1000 760505847 Action|Adventure|Fantasy|Sci-Fi
## 2                  40000 309404152        Action|Adventure|Fantasy
## 3                  11000 200074175       Action|Adventure|Thriller
## 4                  27000 448130642                 Action|Thriller
## 5                    131        NA                    Documentary
## 6                    640  73058679         Action|Adventure|Sci-Fi
##      actor_1_name                                          movie_title
## 1     CCH Pounder                                              Avatar
## 2     Johnny Depp              Pirates of the Caribbean: At World's End
## 3 Christoph Waltz                                               Spectre
## 4       Tom Hardy                                The Dark Knight Rises
## 5     Doug Walker Star Wars: Episode VII - The Force Awakens
## 6    Daryl Sabara                                           John Carter
##   num_voted_users cast_total_facebook_likes         actor_3_name
## 1          886204                     4834            Wes Studi
## 2          471220                    48350       Jack Davenport
## 3          275868                    11700     Stephanie Sigman
## 4         1144337                   106759 Joseph Gordon-Levitt
## 5               8                      143
## 6          212204                     1873         Polly Walker
##   facenumber_in_poster
## 1                    0
## 2                    0
## 3                    1
## 4                    0
## 5                    0
## 6                    1
##                                                 plot_keywords
## 1                          avatar|future|marine|native|paraplegic
## 2     goddess|marriage ceremony|marriage proposal|pirate|singapore
## 3                            bomb|espionage|sequel|spy|terrorist
## 4 deception|imprisonment|lawlessness|police officer|terrorist plot
## 5
## 6               alien|american civil war|male nipple|mars|princess
##                                    movie_imdb_link num_user_for_reviews
## 1 http://www.imdb.com/title/tt0499549/?ref_=fn_tt_tt_1                3054
## 2 http://www.imdb.com/title/tt0449088/?ref_=fn_tt_tt_1                1238
## 3 http://www.imdb.com/title/tt2379713/?ref_=fn_tt_tt_1                 994
## 4 http://www.imdb.com/title/tt1345836/?ref_=fn_tt_tt_1                2701
## 5 http://www.imdb.com/title/tt5289954/?ref_=fn_tt_tt_1                  NA
## 6 http://www.imdb.com/title/tt0401729/?ref_=fn_tt_tt_1                 738
##   language country content_rating    budget title_year actor_2_facebook_likes
## 1  English     USA          PG-13 237000000       2009                    936
## 2  English     USA          PG-13 300000000       2007                   5000
## 3  English      UK          PG-13 245000000       2015                    393
## 4  English     USA          PG-13 250000000       2012                  23000
## 5                                       NA         NA                     12
## 6  English     USA          PG-13 263700000       2012                    632
##   imdb_score aspect_ratio movie_facebook_likes
```

```
## 1        7.9         1.78               33000
## 2        7.1         2.35                   0
## 3        6.8         2.35               85000
## 4        8.5         2.35              164000
## 5        7.1           NA                   0
## 6        6.6         2.35               24000
```

Podaci se sastoje od 5043 filmova i 28 kategorija koje ih opisuju (varijabli). Koje to sve kategorije opisuju fimlove?

```
names(filmovi)
```

```
##  [1] "color"                     "director_name"
##  [3] "num_critic_for_reviews"    "duration"
##  [5] "director_facebook_likes"   "actor_3_facebook_likes"
##  [7] "actor_2_name"              "actor_1_facebook_likes"
##  [9] "gross"                     "genres"
## [11] "actor_1_name"              "movie_title"
## [13] "num_voted_users"           "cast_total_facebook_likes"
## [15] "actor_3_name"              "facenumber_in_poster"
## [17] "plot_keywords"             "movie_imdb_link"
## [19] "num_user_for_reviews"      "language"
## [21] "country"                   "content_rating"
## [23] "budget"                    "title_year"
## [25] "actor_2_facebook_likes"    "imdb_score"
## [27] "aspect_ratio"              "movie_facebook_likes"
```

Prvo pogledajmo koje se sve različite vrijednosti pojavljuju u stupcu "content_rating".

```
table(filmovi$content_rating)
```

```
##
##            Approved         G        GP         M     NC-17 Not Rated    Passed
##       303        55       112         6         5         7       116         9
##        PG     PG-13         R     TV-14      TV-G     TV-MA     TV-PG      TV-Y
##       701      1461      2118        30        10        20        13         1
##     TV-Y7   Unrated         X
##         1        62        13
```

```
length(unique(filmovi$content_rating))
```

```
## [1] 19
```

Vidimo da postoji više varijanti ocjena sadržaja namijenjenih za opću publiku: PG, PG-13 i TV-PG, dok sadržaj namijenjen osobama određene dobi ima samo jednu ocjenu, R. Budući da nas interesira razlika u zaradi filmova, pod filmove namijenjene općoj pubilci smatrat ćemo PG i PG-13, a TV-PG nećemo uzimati u obzir jer je on namijenjen televizijsikm emisijama i serijama.

Objedinimo filmove ocijenjene s PG-13 i PG.

```
filmovi1 = filmovi
filmovi1$content_rating[filmovi1$content_rating == "PG-13"] = "PG"
```

```
table(filmovi1$content_rating)
```

```
##
##            Approved         G        GP         M     NC-17 Not Rated    Passed
##       303        55       112         6         5         7       116         9
##        PG         R     TV-14      TV-G     TV-MA     TV-PG      TV-Y     TV-Y7
##      2162      2118        30        10        20        13         1         1
##   Unrated         X
##        62        13
```

```
length(unique(filmovi1$content_rating))
```

```
## [1] 18
```

Sada možemo napraviti dvije varijable u koje ćemo spremiti podatke o filmovima kojima je "content_rating" ili PG ili R.

```
filmovi_PG = filmovi1[filmovi1$content_rating == "PG",]
filmovi_R = filmovi1[filmovi1$content_rating == "R",]
dim(filmovi_PG)
```

```
## [1] 2162   28
```

```
dim(filmovi_R)
```

```
## [1] 2118   28
```

U novim tablicama trebamo očistiti stupac zarade ("gross") od Na vrijednosti kako ne bi bilo problema pri daljnjem računanju i provođenju statističkih testova.

```
filmovi_PG_clean = na.omit(filmovi_PG)
filmovi_R_clean = na.omit(filmovi_R)
dim(filmovi_PG_clean)
```

```
## [1] 1880   28
```

```
dim(filmovi_R_clean)
```

```
## [1] 1709   28
```

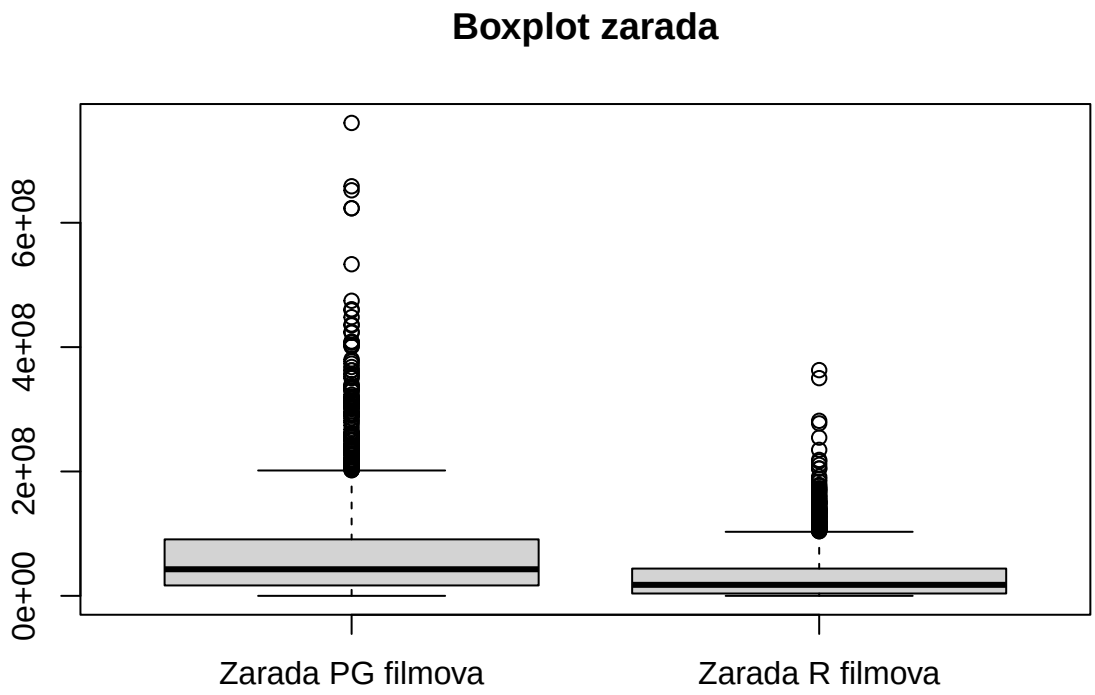
```
cat('Prosječna zarada filmova namijenjenih za opću publiku (PG) iznosi ',
    mean(filmovi_PG_clean$gross), '€\n')
```

```
## Prosječna zarada filmova namijenjenih za opću publiku (PG) iznosi  71274507 €
```

```r
cat('Prosječna zarada filmova namijenjenih za osobe određene dobi (R) iznosi ',
    mean(filmovi_R_clean$gross), '€\n')
```

```
## Prosječna zarada filmova namijenjenih za osobe određene dobi (R) iznosi  31991790 €
```

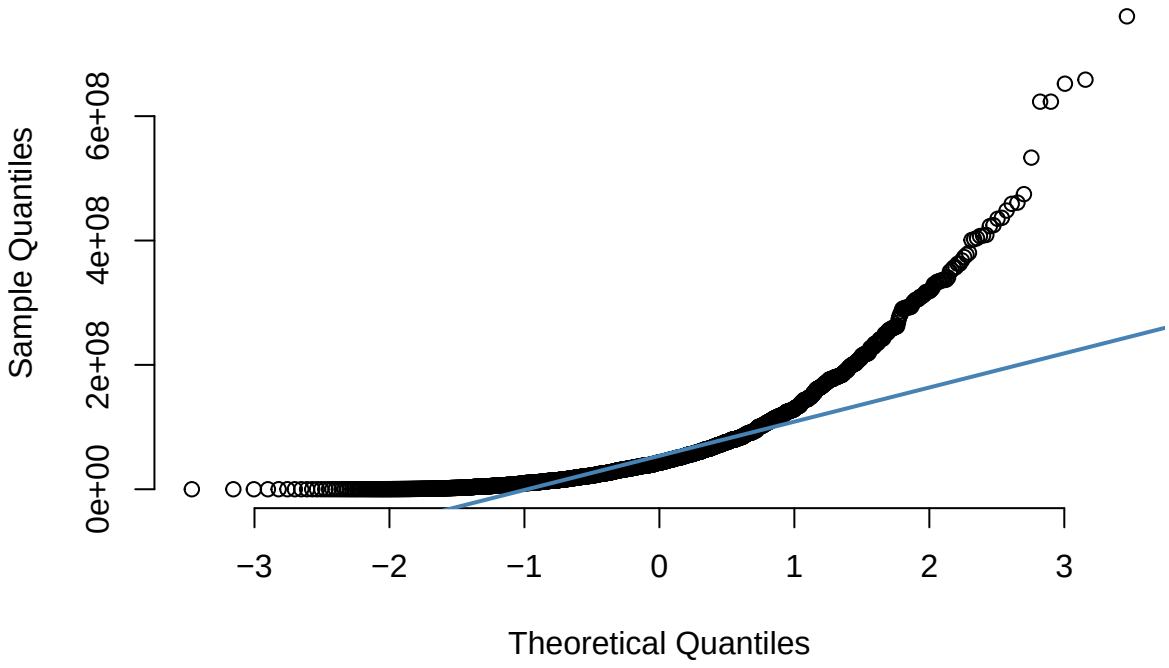
```r
boxplot(filmovi_PG_clean$gross, filmovi_R_clean$gross,
        names = c('Zarada PG filmova','Zarada R filmova'),
        main='Boxplot zarada')
```



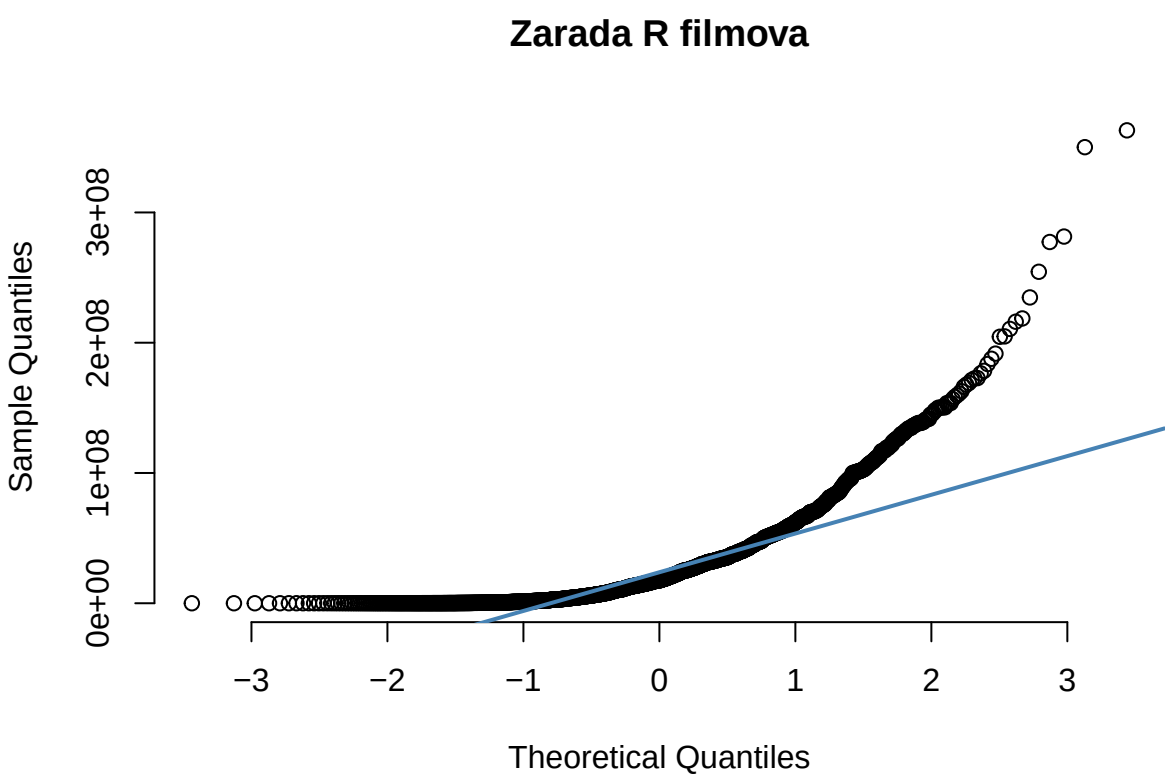
Postoje indikacije da bi filmovi namijenjeni općoj publici trebali zarađivati više od filmova namijenjenoj pubilci određene dobi, tj. postoji razlika u zaradi filmova, a to možemo ispitati t-testom.

Kako bi proveli t-test, moramo najprije provjeriti zadovoljavaju li naši uzorci (filmovi_PG_clean i filmovi_R_clean) pretpostavke normalnosti i nezavisnosti. Obzirom da razmatramo dva uzorka koji imaju različite ocjene sadržaja, možemo pretpostaviti njihovu nezavisnost. Sada moramo ispitati normalnost podataka koje koristimo, a to ćemo napraviti qqplot-ovima.

```r
qqnorm(filmovi_PG_clean$gross, pch = 1, frame = FALSE,main='Zarada PG filmova')
qqline(filmovi_PG_clean$gross, col = "steelblue", lwd = 2)
```




```
qqnorm(filmovi_R_clean$gross, pch = 1, frame = FALSE,main='Zarada R filmova')
qqline(filmovi_R_clean$gross, col = "steelblue", lwd = 2)
```



Iz danih qqplot-ova vidljivo je da podaci nisu normalno distribuirani, a to ćemo i provjeriti pomoću Lilliefor testa.

$$H_0 : \text{Podaci su normalno distribuirani}$$
$$H_1 : \text{Podaci nisu normalno distribuirani}$$

```
lillie.test(filmovi_PG_clean$gross)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi_PG_clean$gross
## D = 0.1993, p-value < 2.2e-16
```

```
lillie.test(filmovi_R_clean$gross)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi_R_clean$gross
## D = 0.21356, p-value < 2.2e-16
```

p-vrijednost je značajno manja od 0.05 tako da odbacujemo nultu hipotezu u korist alternativne.

Iako podaci ne dopuštaju provedbu parametarskog t-testa, provest ćemo ga kako bi kasnije mogli bolje objasniti dobivene rezultate neparametarskog testa.

Kako bi znali koji t-test moramo provesti, prvo moramo ispitati jednakost varijanci uzoraka.

Hipoteze testa jednakosti varijanci glase:

$$H_0 : \sigma_1^2 = \sigma_2^2$$
$$H_1 : \sigma_1^2 \neq \sigma_2^2$$

```
var.test(filmovi_PG_clean$gross, filmovi_R_clean$gross)
```

```
##
##  F test to compare two variances
##
## data:  filmovi_PG_clean$gross and filmovi_R_clean$gross
## F = 4.3927, num df = 1879, denom df = 1708, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  4.003442 4.818903
## sample estimates:
## ratio of variances
##           4.392745
```

Vrlo mala p-vrijednost nam govori kako ćemo odaciti našu hipotezu da su varijance naša dva uzorka jednake. Sada znamo da provodimo t-test u slučaju kad su varijance nepoznate i različite, a hipoteze t-testa glase:

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 \neq \mu_2$$

```
t.test(filmovi_PG_clean$gross, filmovi_R_clean$gross, alt="two.sided", var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  filmovi_PG_clean$gross and filmovi_R_clean$gross
## t = 18.04, df = 2748.3, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  35012935 43552498
## sample estimates:
## mean of x mean of y
##  71274507  31991790
```

Zbog jako male p-vrijednosti možemo odbaciti nultu hipotezu u korist alternativne hipoteze, odnosno možemo reći da postoji značajna razlika u zaradi filmova namjenjenih za opću publiku i filmova namijenjenih određenoj dobi.

Kada želimo testirati jednakost srednjih vrijednosti dvaju uzoraka čiji podaci nisu normalno distribuirani, ali uzorci su nezavisni, koristimo Wilcoxonov test zbrajanja rangova.

Kao i za t-test, testiramo hipoteze:

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 \neq \mu_2$$

I provodimo Wilcoxonov test zbrajanja rangova.

```
wilcox.test(filmovi_PG_clean$gross, filmovi_R_clean$gross, alternative = "two.sided")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  filmovi_PG_clean$gross and filmovi_R_clean$gross
## W = 2186206, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Rezultat testa je, kao i kod t-test, jako mala p-vrijednost te odbijamo nultu hipotezu u korist alternativne hipoteze, tj. postoji značajna razlika u zaradi filmova najimenjenih za opću publiku i filmova namijenjenih za osobe određene dobi.

Pošto uzorci ne dolaze iz normalne ditribucije, Wilcoxonov test zbrajanja rangova je superiorniji u odnosu na t-test, no naši podaci toliko očito pokazuju razliku u zaradi PG filmova i R filmova da su oba testa dali jednaku, izrazito malu, p-vrijednost.

Dodatno, iz boxplot grafa zarada mogli bismo zaključiti da PG filmovi imaju veću zaradu od R filmova pa provjerimo to još jednim Wilcoxonovim testom zbrajanja rangova.

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 > \mu_2$$

```
wilcox.test(filmovi_PG_clean$gross, filmovi_R_clean$gross, alternative = "greater")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  filmovi_PG_clean$gross and filmovi_R_clean$gross
## W = 2186206, p-value < 2.2e-16
## alternative hypothesis: true location shift is greater than 0
```

Odbacujemo H_0 u korist H_1 , odnosno možemo reći da filmovi namijenjeni za opću publiku u prosjeku imaju zntno veću zaradu od filmova namijenjenih osobama određene dobi.

Pitanje 2: Postoji li razlika u budžetima filmova s obzirom na njihov žanr?

Ovo istraživačko pitanje mogli bismo testirati jednofaktorskim ANOVA testom.

```
IMDB <- unique(filmovi)
IMDB %>% group_by(genres) %>% summarise(  # broj filmova po žanrovima
          count = n())
```

```
## # A tibble: 914 x 2
##    genres                                                    count
##    <chr>                                                     <int>
##  1 Action                                                       11
```

```
##  2 Action|Adventure                                                          10
##  3 Action|Adventure|Animation|Comedy|Crime|Family|Fantasy                    1
##  4 Action|Adventure|Animation|Comedy|Drama|Family|Fantasy|Thriller           1
##  5 Action|Adventure|Animation|Comedy|Drama|Family|Sci-Fi                     2
##  6 Action|Adventure|Animation|Comedy|Family                                  6
##  7 Action|Adventure|Animation|Comedy|Family|Fantasy                          4
##  8 Action|Adventure|Animation|Comedy|Family|Fantasy|Sci-Fi                   2
##  9 Action|Adventure|Animation|Comedy|Family|Sci-Fi                           3
## 10 Action|Adventure|Animation|Comedy|Fantasy                                 1
## # i 904 more rows
```

```
IMDB = IMDB[complete.cases(IMDB$budget), ] # izbacivanje NA vrijednosti
IMDB %>% group_by(genres) %>% summarise( # broj filmova po žanrovima
          count = n())
```

```
## # A tibble: 859 x 2
##    genres                                                          count
##    <chr>                                                           <int>
##  1 Action                                                             10
##  2 Action|Adventure                                                   10
##  3 Action|Adventure|Animation|Comedy|Crime|Family|Fantasy              1
##  4 Action|Adventure|Animation|Comedy|Drama|Family|Fantasy|Thriller     1
##  5 Action|Adventure|Animation|Comedy|Drama|Family|Sci-Fi               2
##  6 Action|Adventure|Animation|Comedy|Family                            5
##  7 Action|Adventure|Animation|Comedy|Family|Fantasy                    4
##  8 Action|Adventure|Animation|Comedy|Family|Fantasy|Sci-Fi             2
##  9 Action|Adventure|Animation|Comedy|Family|Sci-Fi                     3
## 10 Action|Adventure|Animation|Comedy|Fantasy                           1
## # i 849 more rows
```

```
dim(IMDB)
```

```
## [1] 4511   28
```

Varijabla genres je u zapisu x|y|z|... Na primjer, film Avatar ima više žanrova, a to su Action|Adventure|Fantasy|Sci-Fi. Stoga ćemo ih razdvojiti jer želimo da se budžeti filmova gledaju pod svakim njihovim žanrom.

```
IMDB = IMDB %>%
  separate_rows(genres, sep = "\\|")
dim(IMDB)
```

```
## [1] 13136    28
```

Sada se u podatkvnom okviru nalazi 13136 filmova, a svaki film ima 28 varijabli.

```
unique_genres_count = IMDB %>%
  distinct(genres) %>%
  count()

unique_genres_count$n
```

```
## [1] 26
```

Postoji 26 žanrova.

```
unique(IMDB$genres) # žanrovi filmova
```

```
##  [1] "Action"      "Adventure"   "Fantasy"     "Sci-Fi"      "Thriller"
##  [6] "Romance"     "Animation"   "Comedy"      "Family"      "Musical"
## [11] "Mystery"     "Western"     "Drama"       "History"     "Sport"
## [16] "Crime"       "Horror"      "War"         "Biography"   "Music"
## [21] "Documentary" "Game-Show"   "Reality-TV"  "Short"       "Film-Noir"
## [26] "News"
```

```
IMDB$genres = factor(IMDB$genres, levels = c('Action', 'Adventure', 'Animation', 'Biography', 'Comedy',
                                             'Documentary', 'Drama', 'Family', 'Fantasy', 'Film-No
                                             'History', 'Horror', 'Music', 'Musical', 'Mystery', '
                                             'Reality-TV', 'Romance', 'Sci-Fi', 'Short', 'Sport',
                                             'War', 'Western'))
```

```
genre_counts = table(IMDB$genres) # broj filmova po žanru
print(genre_counts)
```

```
##
##      Action   Adventure   Animation   Biography      Comedy       Crime
##        1092         876         223         271        1674         801
## Documentary       Drama      Family     Fantasy   Film-Noir   Game-Show
##          95        2262         506         568           5           1
##     History      Horror       Music     Musical     Mystery        News
##         192         509         180         122         438           2
##  Reality-TV     Romance      Sci-Fi       Short       Sport    Thriller
##           1         999         571           3         161        1297
##         War     Western
##         199          88
```

Uočavamo da neki žanrovi imaju mali broj filmova. To su "Game-Show", "News", "Reality-TV" i "Short". Te žanrove ćemo izbaciti iz dataseta. Na primjer, za "Reality-TV" imamo samo jedan primjer i da je taj primjer imao veliki budžet došli bi do zaključka da bi svaki "Reality-TV" imao takav "budget" što nije nužno istina. Također za te žanrove nećemo moći provesti testove potrebne za dokazivanje pretpostavke za ANOVA-u zbog tako male veličine uzorka.

```
excluded_genres = c("Game-Show", "News", "Reality-TV", "Short")
# Izbacivanje određene žanrove
IMDB = IMDB[!(IMDB$genres %in% excluded_genres), ]
dim(IMDB)
```

```
## [1] 13129    28
```

```
genre_counts = table(IMDB$genres) # broj filmova po žanru
print(genre_counts)
```

```
##
##      Action   Adventure   Animation   Biography      Comedy       Crime
##        1092         876         223         271        1674         801
```

```
## Documentary       Drama      Family     Fantasy   Film-Noir   Game-Show
##          95        2262         506         568           5           0
##     History      Horror       Music     Musical     Mystery        News
##         192         509         180         122         438           0
##  Reality-TV     Romance      Sci-Fi       Short       Sport    Thriller
##           0         999         571           0         161        1297
##         War     Western
##         199          88
```

Sada imamo 13129 filmova, a svaki film ima 28 varijabli.

```
unique_genres_count = IMDB %>%
  distinct(genres) %>%
  count()
unique_genres_count$n
```

```
## [1] 22
```

Kako su vrijednosti budžeta jako velike smanjit ćemo ih pomoću log() funkcije. Ovo radimo kako bismo lakše mogli očitati neke pretpostavke s grafova.

```
# Provođenje log() nad budžet vrijednostima
IMDB$budget_log = log(IMDB$budget)
IMDB$budget = IMDB$budget_log
IMDB$budget_log = NULL
```

"Budget" -> originalna vrijednost

```
# Vračanje na originalnu vrijednost
IMDB_org = IMDB
IMDB_org$budget_exp = exp(IMDB$budget)
IMDB_org$budget = IMDB_org$budget_exp
IMDB_org$budget_exp = NULL
```

Dataset je sada spreman, pa možemo započeti sa ANOVA postupkom.

Pretpostavke ANOVA-e su:

- nezavisnost pojedinih podataka u uzorcima,
- normalna razdioba podataka,
- homogenost varijanci među populacijama.

Normalna razdioba podataka

Provjera normalnosti može se za svaku pojedinu grupu napraviti KS testom ili Lillieforsovom inačicom KS testa. Također možemo iz histograma zaključiti normalnost.

```r
require(nortest)
lillie.test(IMDB$budget)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget
## D = 0.10865, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Action'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Action"]
## D = 0.11795, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Adventure'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Adventure"]
## D = 0.07974, p-value = 2.229e-14
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Fantasy'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Fantasy"]
## D = 0.091484, p-value = 2.825e-12
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Sci-Fi'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Sci-Fi"]
## D = 0.11553, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Thriller'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Thriller"]
## D = 0.11595, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Documentary'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Documentary"]
## D = 0.075896, p-value = 0.1956
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Romance'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Romance"]
## D = 0.12108, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Animation'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Animation"]
## D = 0.13263, p-value = 3.196e-10
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Comedy'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Comedy"]
## D = 0.11108, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Family'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Family"]
## D = 0.10958, p-value = 6.089e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Musical'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Musical"]
## D = 0.10928, p-value = 0.001104
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Mystery'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Mystery"]
## D = 0.12246, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Western'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Western"]
## D = 0.089362, p-value = 0.0795
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Drama'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Drama"]
## D = 0.10689, p-value < 2.2e-16
```

```r
lillie.test(IMDB$budget[IMDB$genres=='History'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "History"]
## D = 0.10383, p-value = 3.071e-05
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Sport'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Sport"]
## D = 0.17149, p-value = 1.486e-12
```

```r
lillie.test(IMDB$budget[IMDB$genres=='Crime'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Crime"]
## D = 0.13217, p-value < 2.2e-16
```

```
lillie.test(IMDB$budget[IMDB$genres=='Horror'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Horror"]
## D = 0.10867, p-value = 9.236e-16
```

```
lillie.test(IMDB$budget[IMDB$genres=='War'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "War"]
## D = 0.090474, p-value = 0.0004359
```

```
lillie.test(IMDB$budget[IMDB$genres=='Biography'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Biography"]
## D = 0.12999, p-value = 5.495e-12
```

```
lillie.test(IMDB$budget[IMDB$genres=='Music'])
```

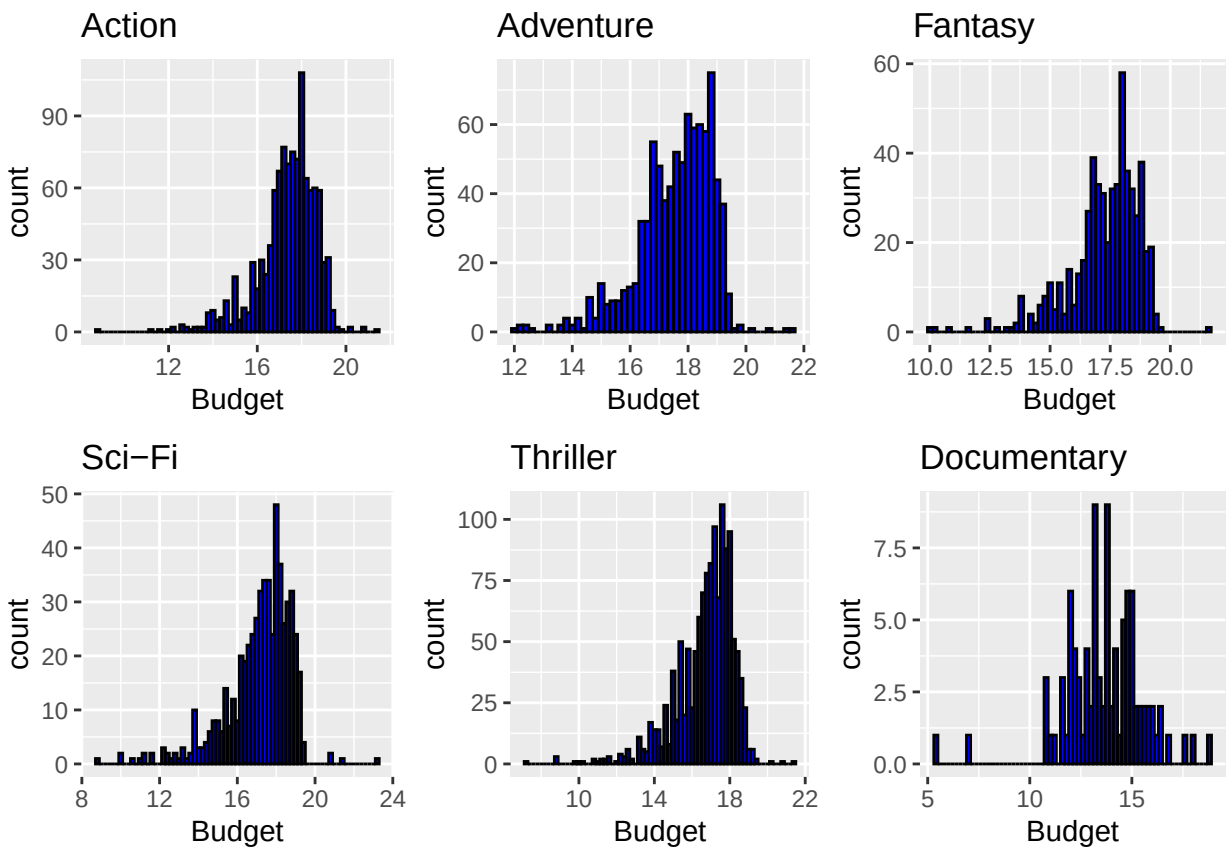
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Music"]
## D = 0.14977, p-value = 1.385e-10
```

```
lillie.test(IMDB$budget[IMDB$genres=='Film-Noir'])
```

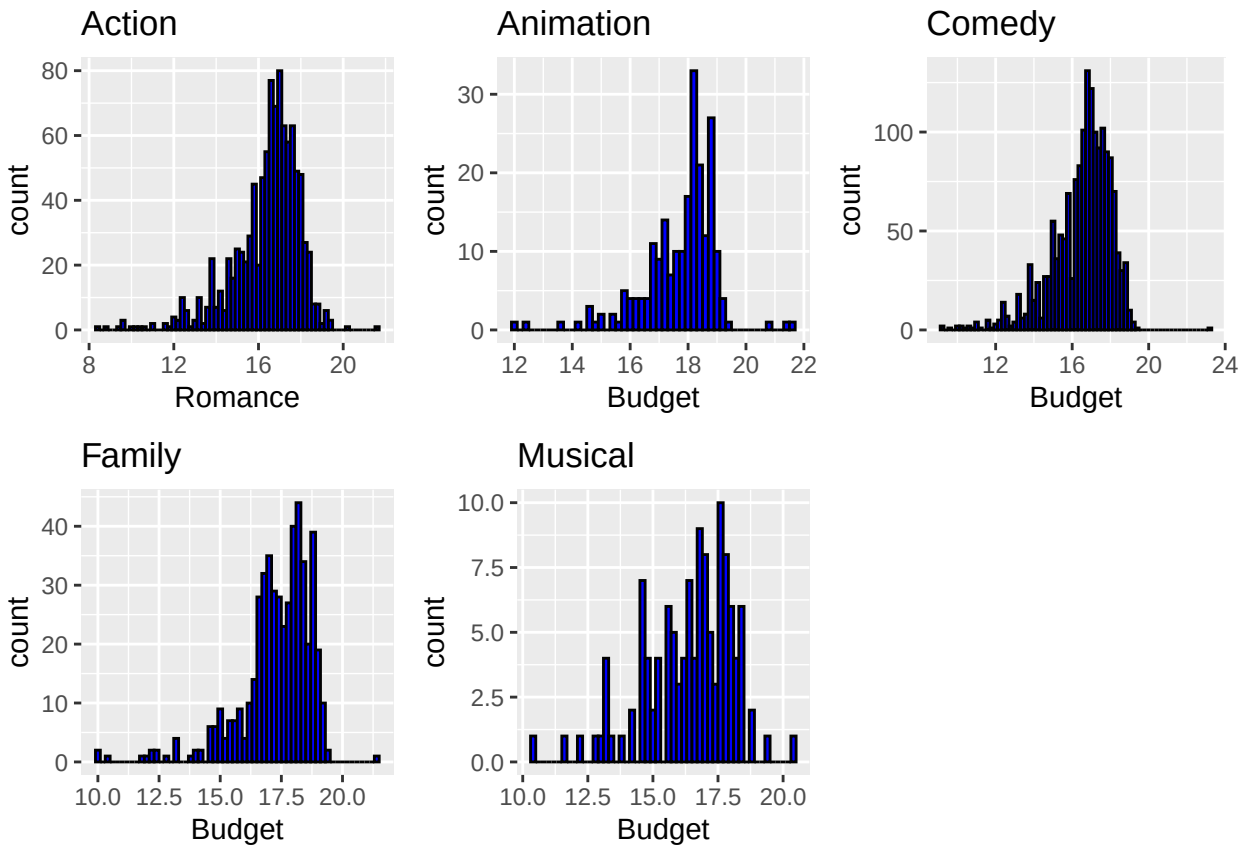
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB$budget[IMDB$genres == "Film-Noir"]
## D = 0.32358, p-value = 0.09453
```

```
graf1 = ggplot(data = IMDB[IMDB$genres == 'Action', ], aes(x = budget)) + geom_histogram(binwidth = 0.2
  labs(title = "Action", x = "Budget")

graf2 = ggplot(data = IMDB[IMDB$genres == 'Adventure', ], aes(x = budget)) + geom_histogram(binwidth = 
  labs(title = "Adventure", x = "Budget")

graf3 = ggplot(data = IMDB[IMDB$genres == 'Fantasy', ], aes(x = budget)) + geom_histogram(binwidth = 0.1
  labs(title = "Fantasy", x = "Budget")
```

```
graf4 = ggplot(data = IMDB[IMDB$genres == 'Sci-Fi', ], aes(x = budget)) + geom_histogram(binwidth = 0.2
  labs(title = "Sci-Fi", x = "Budget")

graf5 = ggplot(data = IMDB[IMDB$genres == 'Thriller', ], aes(x = budget)) + geom_histogram(binwidth = 0
  labs(title = "Thriller", x = "Budget")

graf6 = ggplot(data = IMDB[IMDB$genres == 'Documentary', ], aes(x = budget)) + geom_histogram(binwidth =
  labs(title = "Documentary", x = "Budget")

graf7 = ggplot(data = IMDB[IMDB$genres == 'Romance', ], aes(x = budget)) + geom_histogram(binwidth = 0.1
  labs(title = "Action", x = "Romance")

graf8 = ggplot(data = IMDB[IMDB$genres == 'Animation', ], aes(x = budget)) + geom_histogram(binwidth = 
  labs(title = "Animation", x = "Budget")

graf9 = ggplot(data = IMDB[IMDB$genres == 'Comedy', ], aes(x = budget)) + geom_histogram(binwidth = 0.2
  labs(title = "Comedy", x = "Budget")

graf10 = ggplot(data = IMDB[IMDB$genres == 'Family', ], aes(x = budget)) + geom_histogram(binwidth = 0.1
  labs(title = "Family", x = "Budget")

graf11 = ggplot(data = IMDB[IMDB$genres == 'Musical', ], aes(x = budget)) + geom_histogram(binwidth = 0
  labs(title = "Musical", x = "Budget")



grid.arrange(graf1, graf2, graf3, graf4, graf5, graf6, ncol = 3)
```

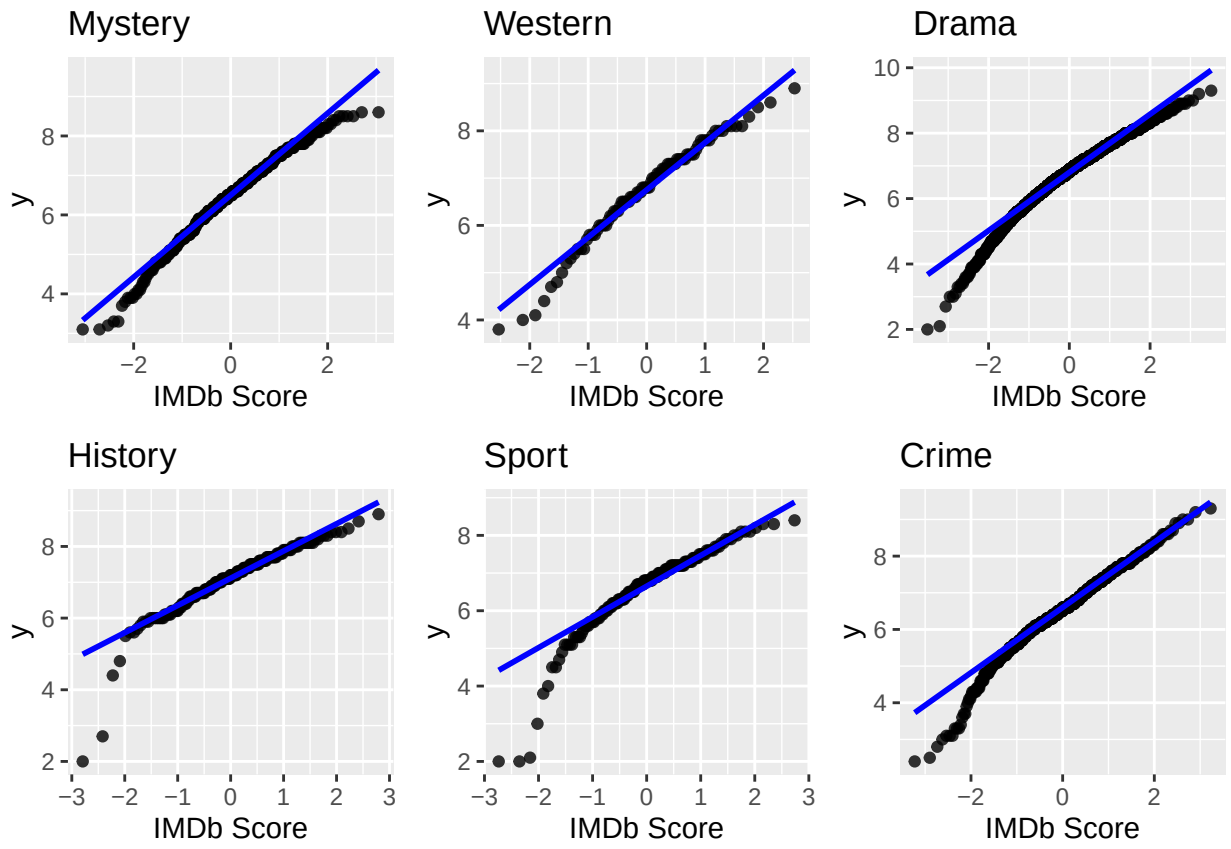



```
grid.arrange(graf7, graf8, graf9, graf10, graf11, ncol = 3)
```

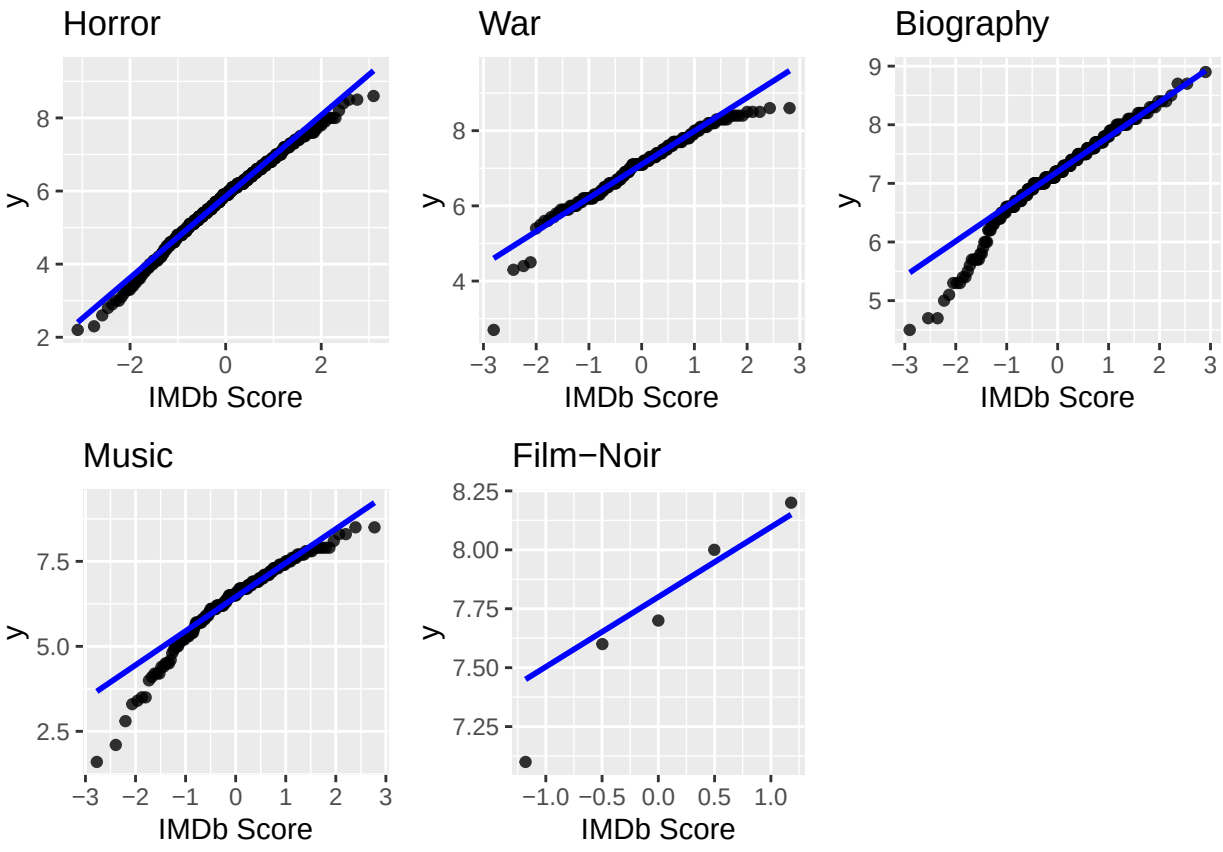



```
graf12 = ggplot(data = IMDB[IMDB$genres == 'Mystery', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8
  labs(title = "Mystery", x = "IMDb Score")

graf13 = ggplot(data = IMDB[IMDB$genres == 'Western', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8
  labs(title = "Western", x = "IMDb Score")

graf14 = ggplot(data = IMDB[IMDB$genres == 'Drama', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8)
  labs(title = "Drama", x = "IMDb Score")

graf15 = ggplot(data = IMDB[IMDB$genres == 'History', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8
  labs(title = "History", x = "IMDb Score")

graf16 = ggplot(data = IMDB[IMDB$genres == 'Sport', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8)
  labs(title = "Sport", x = "IMDb Score")

graf17 = ggplot(data = IMDB[IMDB$genres == 'Crime', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8)
  labs(title = "Crime", x = "IMDb Score")

graf18 = ggplot(data = IMDB[IMDB$genres == 'Horror', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8
  labs(title = "Horror", x = "IMDb Score")

graf19 = ggplot(data = IMDB[IMDB$genres == 'War', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) +
  labs(title = "War", x = "IMDb Score")

graf20 = ggplot(data = IMDB[IMDB$genres == 'Biography', ], aes(sample = imdb_score)) + geom_qq(alpha =
  stat_qq_line(color = "blue", linewidth = 1) + labs(title = "Biography", x = "IMDb Score")
```

```
graf21 = ggplot(data = IMDB[IMDB$genres == 'Music', ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8)
  labs(title = "Music", x = "IMDb Score")

graf22 = ggplot(data = IMDB[IMDB$genres == 'Film-Noir', ], aes(sample = imdb_score)) + geom_qq(alpha = 
  stat_qq_line(color = "blue", linewidth = 1) + labs(title = "Film-Noir", x = "IMDb Score")

grid.arrange(graf12, graf13, graf14, graf15, graf16, graf17,ncol = 3)
```




```
grid.arrange(graf18, graf19, graf20, graf21, graf22, ncol = 3)
```




```
require(nortest)
lillie.test(IMDB_org$budget)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget
## D = 0.42167, p-value < 2.2e-16
```

```
lillie.test(IMDB_org$budget[IMDB_org$genres=='Action'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Action"]
## D = 0.25292, p-value < 2.2e-16
```

```
lillie.test(IMDB_org$budget[IMDB_org$genres=='Adventure'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Adventure"]
## D = 0.26875, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Fantasy'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Fantasy"]
## D = 0.28566, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Sci-Fi'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Sci-Fi"]
## D = 0.43442, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Thriller'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Thriller"]
## D = 0.30387, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Documentary'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Documentary"]
## D = 0.39422, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Romance'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Romance"]
## D = 0.35948, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Animation'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Animation"]
## D = 0.3317, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Comedy'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Comedy"]
## D = 0.44888, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Family'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Family"]
## D = 0.27551, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Musical'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Musical"]
## D = 0.30712, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Mystery'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Mystery"]
## D = 0.17041, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Western'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Western"]
## D = 0.22945, p-value = 1.946e-12
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Drama'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Drama"]
## D = 0.4493, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='History'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "History"]
## D = 0.25733, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Sport'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Sport"]
## D = 0.1698, p-value = 2.712e-12
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Crime'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Crime"]
## D = 0.40802, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Horror'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Horror"]
## D = 0.46952, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='War'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "War"]
## D = 0.38759, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Biography'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Biography"]
## D = 0.2042, p-value < 2.2e-16
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Music'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Music"]
## D = 0.17551, p-value = 8.547e-15
```

```r
lillie.test(IMDB_org$budget[IMDB_org$genres=='Film-Noir'])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  IMDB_org$budget[IMDB_org$genres == "Film-Noir"]
## D = 0.32221, p-value = 0.09806
```

```r
#hist(IMDB_org$budget[IMDB_org$genres=='Action']) Nisam radio grafove za org vrijednost jer nisu čitlji
```

#Analiza Lillieforsovom inačicom KS testa: Lillieforsovom inačicom KS testa za oba dataseta, odnosno za dataset s budgetom s originalnim vrijednostima i budžet s logaritamskim vrijednostima su jako slični. Ako je p-vrijednost mala tj. manja od 0.05 zaključujemo da za te žanrove normalnost nije potvrđena, a ako je veći od 0.05 normalnost je potvrđena. Većina žanrova ima p-vrijednost manju od 0.05, što znači da su podaci o budžetu u tim žanrovima statistički različiti od normalne distribucije. Ovo dokazujemo i grafički. Histogrami pokazuju asimetriju u distribuciji, a na QQ grafu da su distribucije normalne, ne bi bilo izrazitog zakrivljenja kao na nekim žanrovima, nego bi podaci bili više posloženi na plavoj liniji.

homogenost varijanci među populacijama

Što se tiče homogenosti varijanci različitih populacija, potrebno je testirati:

$$H_0 : \sigma_1^2 = \sigma_2^2 = \ldots = \sigma_k^2$$
$$H_1 : \text{barem dvije varijance nisu iste.}$$

Navedenu hipotezu možemo testirati Bartlettovim testom.

```r
bartlett.test(IMDB$budget ~ IMDB$genres)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  IMDB$budget by IMDB$genres
## Bartlett's K-squared = 207.18, df = 21, p-value < 2.2e-16
```

```r
var((IMDB$budget[IMDB$genres=='Film-Noir']))
```

```
## [1] 0.07183028
```

```r
var((IMDB$budget[IMDB$genres=='Animation']))
```

```
## [1] 1.586785
```

```r
var((IMDB$budget[IMDB$genres=='Drama']))
```

```
## [1] 2.651133
```

```r
var((IMDB$budget[IMDB$genres=='Music']))
```

```
## [1] 2.291082
```

```r
var((IMDB$budget[IMDB$genres=='Horror']))
```

```
## [1] 3.249019
```

```r
var((IMDB$budget[IMDB$genres=='Documentary']))
```

```
## [1] 3.806051
```

```r
bartlett.test(IMDB_org$budget ~ IMDB_org$genres)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  IMDB_org$budget by IMDB_org$genres
## Bartlett's K-squared = 12861, df = 21, p-value < 2.2e-16
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Film-Noir']))
```

```
## [1] 208083467626
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Animation']))
```

```
## [1] 5.037497e+16
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Drama']))
```

```
## [1] 7.812566e+16
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Music']))
```

```
## [1] 4.407868e+14
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Horror']))
```

```
## [1] 2.928333e+17
```

```r
var((IMDB_org$budget[IMDB_org$genres=='Documentary']))
```

```
## [1] 3.193342e+14
```

Analiza Bartlettovog testa: Ako je p-value mala tj. manji od 0.05 zaključujemo da varijance između žanrova nisu jednake. U oba testa je p-value manji od 0.05. Najmanju varijancu ima žanr 'Film-Noir' što znači da filmovi toga žanra imaju slične budžete. Naravno, veličina uzorka od 'Film-Noir' je malena, pa je to mogući uzrok male vrijednosti pa izdvajamo žanr 'Animation' kojem je vrijednost varijance također mala. Najveću varijancu ima žanr 'Documentary', što ukazuje na širok raspon u budžetima.

Iz testova iznad zaključujemo da normalnost i homogenost nisu potvrđeni. Radi tog zaključka provest ćemo neparametarsku alternativu ANOVA testa -> Kruskal-Wallis test.

```r
kw_log = kruskal.test(budget ~ genres, data = IMDB)
kw_log
```

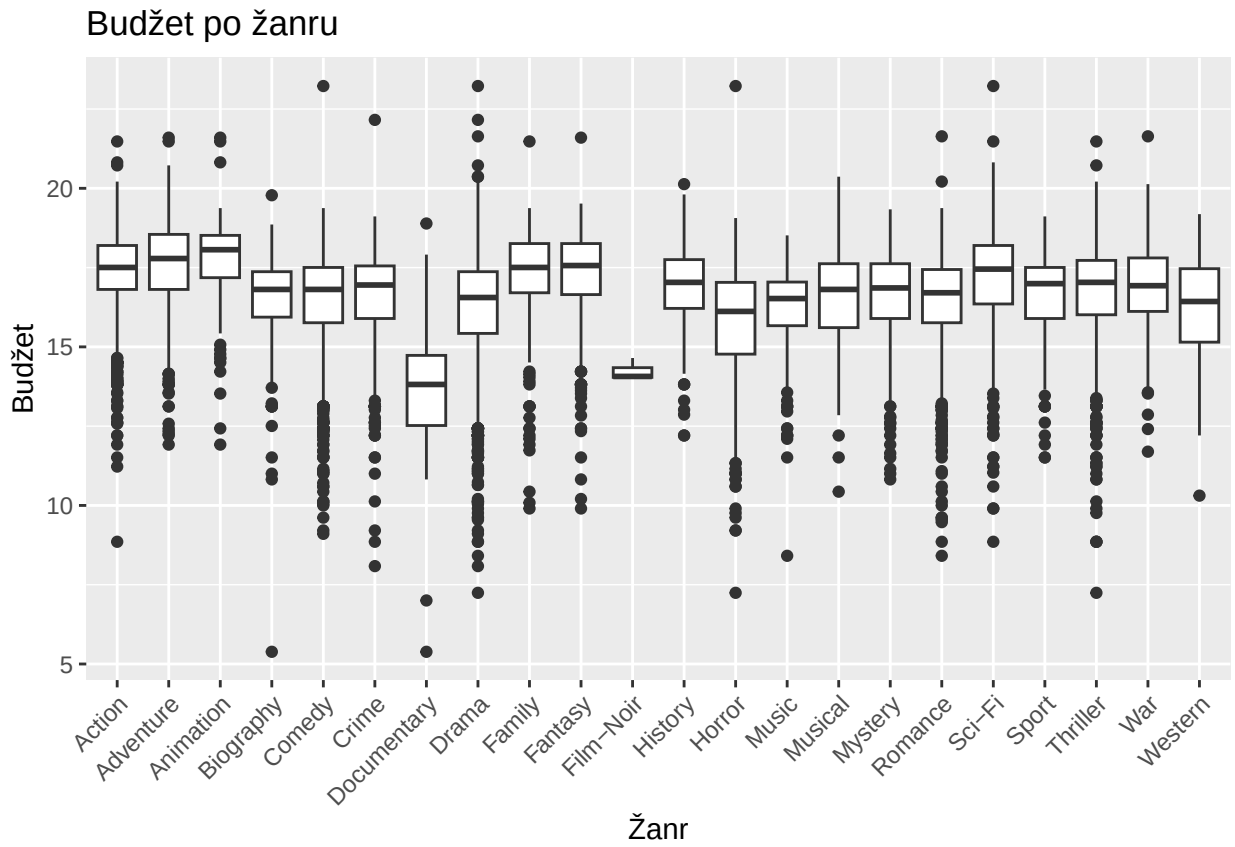
```
##
##  Kruskal-Wallis rank sum test
##
## data:  budget by genres
## Kruskal-Wallis chi-squared = 1637.9, df = 21, p-value < 2.2e-16
```

```r
kw_org = kruskal.test(budget ~ genres, data = IMDB_org)
kw_org
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  budget by genres
## Kruskal-Wallis chi-squared = 1637.9, df = 21, p-value < 2.2e-16
```

#ANOVA

```r
ggplot(IMDB, aes(x = genres, y = budget)) + geom_boxplot() + labs(x = "Žanr", y = "Budžet", title = "Bu
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```




```
a = aov(IMDB$budget ~ IMDB$genres)
summary(a)
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## IMDB$genres    21   3867  184.15   78.47 <2e-16 ***
## Residuals   13107  30761    2.35
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#boxplot(IMDB_org$budget ~ IMDB_org$genres)

a = aov(IMDB_org$budget ~ IMDB_org$genres)
summary(a)
```

```
##                    Df    Sum Sq   Mean Sq F value   Pr(>F)
## IMDB_org$genres    21 4.480e+18 2.133e+17   3.872 5.25e-09 ***
## Residuals       13107 7.222e+20 5.510e+16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model = lm(budget ~ genres, data = IMDB)
summary(model)
```

```
##
```

```
## Call:
## lm(formula = budget ~ genres, data = IMDB)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.1189  -0.6926   0.3078   1.0235   7.4983
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       17.34131    0.04636 374.066  < 2e-16 ***
## genresAdventure    0.21691    0.06949   3.122 0.001802 **
## genresAnimation    0.38272    0.11258   3.400 0.000677 ***
## genresBiography   -0.83788    0.10397  -8.059 8.35e-16 ***
## genresComedy      -0.84096    0.05959 -14.112  < 2e-16 ***
## genresCrime       -0.74261    0.07127 -10.420  < 2e-16 ***
## genresDocumentary -3.68815    0.16387 -22.507  < 2e-16 ***
## genresDrama       -1.06561    0.05645 -18.877  < 2e-16 ***
## genresFamily      -0.04484    0.08238  -0.544 0.586276
## genresFantasy     -0.06657    0.07925  -0.840 0.400966
## genresFilm-Noir   -3.11364    0.68668  -4.534 5.83e-06 ***
## genresHistory     -0.52821    0.11989  -4.406 1.06e-05 ***
## genresHorror      -1.61359    0.08222 -19.626  < 2e-16 ***
## genresMusic       -1.18016    0.12324  -9.576  < 2e-16 ***
## genresMusical     -0.96406    0.14624  -6.592 4.50e-11 ***
## genresMystery     -0.74728    0.08664  -8.625  < 2e-16 ***
## genresRomance     -0.94665    0.06707 -14.114  < 2e-16 ***
## genresSci-Fi      -0.22747    0.07912  -2.875 0.004045 **
## genresSport       -0.79295    0.12933  -6.131 8.97e-10 ***
## genresThriller    -0.64059    0.06292 -10.181  < 2e-16 ***
## genresWar         -0.53104    0.11808  -4.497 6.94e-06 ***
## genresWestern     -1.09752    0.16976  -6.465 1.05e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.532 on 13107 degrees of freedom
## Multiple R-squared:  0.1117, Adjusted R-squared:  0.1103
## F-statistic: 78.47 on 21 and 13107 DF,  p-value: < 2.2e-16
```

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: budget
##              Df  Sum Sq Mean Sq F value    Pr(>F)
## genres       21  3867.1 184.149  78.466 < 2.2e-16 ***
## Residuals 13107 30760.5   2.347
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Linearni model istovjetan je ANOVA modelu. Zaključci u oba slučaja isti.

Zaključak

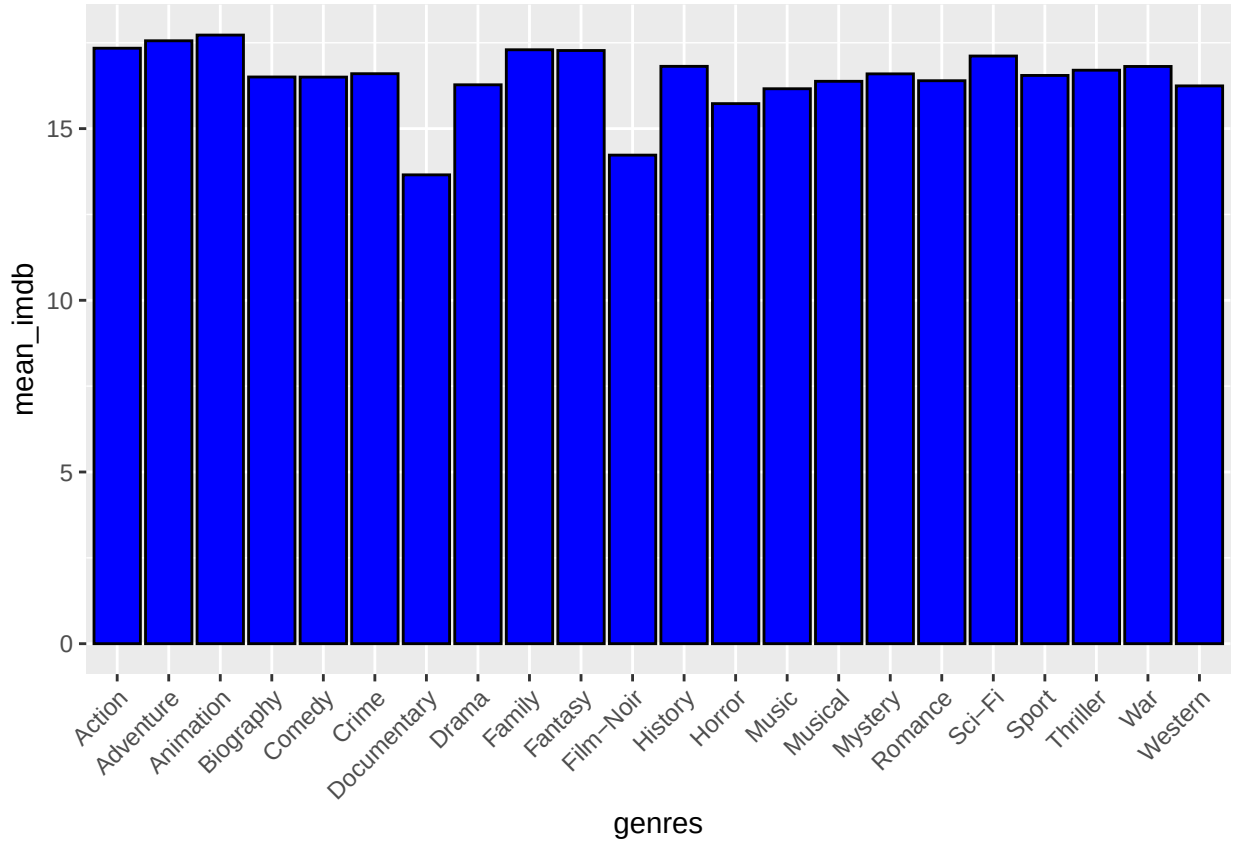
H_0 : Distribucije budgeta isti je za sve žanrove.

H_1 : Postoji razlika u distribuciji budžeta među barem dva žanra.

Prvo što uočavamo je da provedeni Kruskal-Wallis test daje sličan rezultat kao ANOVA test iako nisu bile zadovoljene pretpostavke "normalna razdioba podataka" i "homogenost varijanci među populacijama". Zaključak koji donosimo na temelju Kruskal-Wallis testa je da je p-vrijednost izuzetno mala. To nam govori da postoji značajna razlika u distribuciji budžeta među žanrovima tj. da se budžeti razlikuju među različitim žanrovima.

Ovim grafovom također uočavamo tu razliku budžeta između žanrova, odnosno razlika prosjeka budžeta.

```
IMDB_mean = IMDB %>%
  group_by(genres) %>%
  summarise(mean_imdb = mean(budget, na.rm = TRUE))

ggplot(IMDB_mean, aes(x = genres, y = mean_imdb)) + geom_bar(stat = "identity", fill = "blue", color =
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Pitanje 3: Možemo li temeljem danih varijabli predvidjeti IMDB ocjenu nekog filma?

```
imdb <- filmovi
imdb <- unique(imdb)
```

Gledat ćemo možemo li predvidjeti ocjenu nekog filma s obzirom na sljedeće varijable te ćemo probati napraviti model linearne regresije:

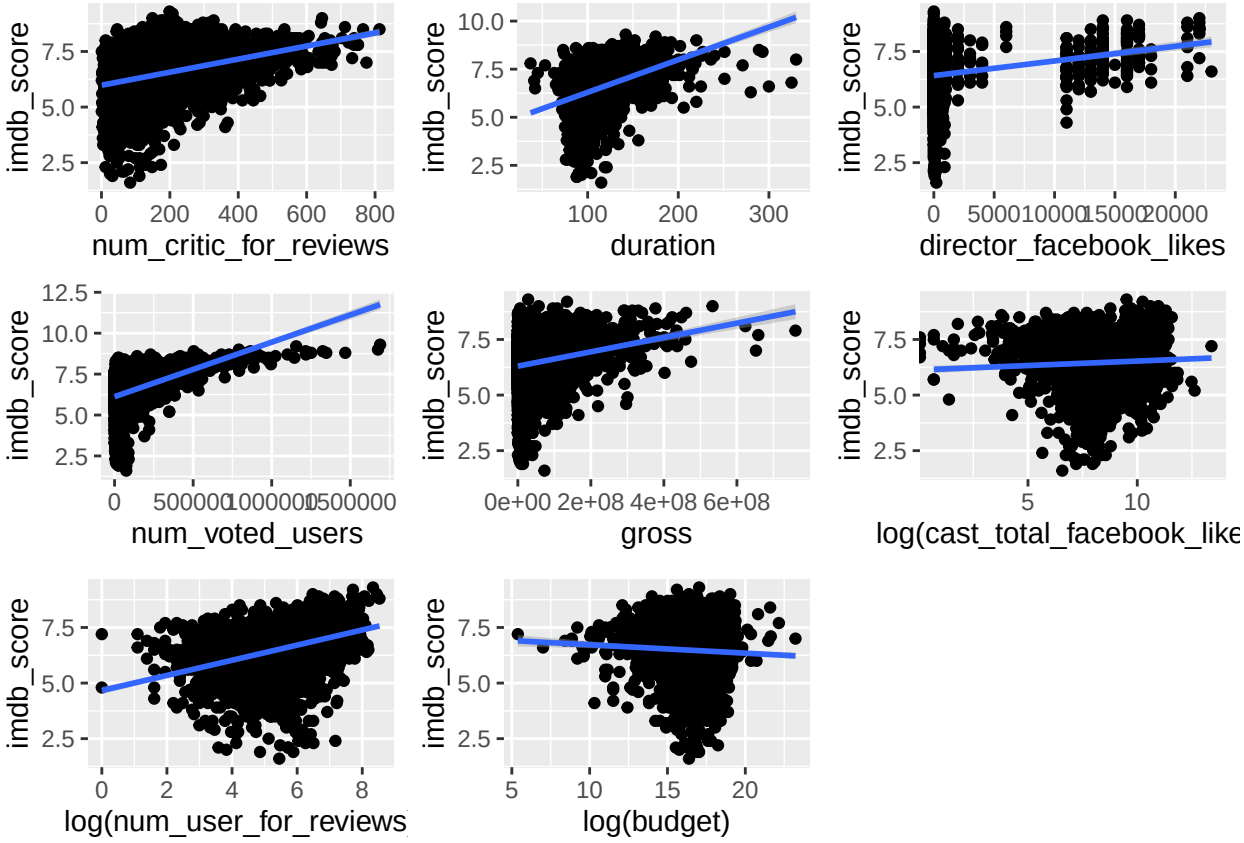
- **num_critic_for_reviews**
- **duration**:
- **director_facebook_likes**:
- **num_voted_users**:
- **gross**:
- **cast_total_facebook_likes**
- **num_user_for_reviews**:
- **budget**:
- **movie_facebook_likes**:
- **color**

Pogledajmo kako izgledaju njihovi odnosi grafički sa zavisnom varijablom imdb_score. Plave linije predstavljaju vrijednosti koje bi dale pojedine jednostavne linearne regresije.

```
imdb <- imdb %>% select(
    num_critic_for_reviews,
    duration,
    director_facebook_likes,
    num_voted_users,
    gross,
    cast_total_facebook_likes,
    num_user_for_reviews,
    budget,
    movie_facebook_likes,
    color,
    aspect_ratio,
    imdb_score
  )

imdb <- na.omit(imdb)

g1 <- ggplot(imdb,aes(x=num_critic_for_reviews,y=imdb_score)) + geom_point() + geom_smooth(formula = y
g2 <- ggplot(imdb,aes(x=duration,y=imdb_score)) + geom_point() + geom_smooth(formula = y ~ x,method="lm
g3 <- ggplot(imdb,aes(x=director_facebook_likes,y=imdb_score)) + geom_point() + geom_smooth(formula = y
g4 <- ggplot(imdb,aes(x=num_voted_users,y=imdb_score)) + geom_point() + geom_smooth(formula = y ~ x,met
g5 <- ggplot(imdb,aes(x=gross,y=imdb_score)) + geom_point() + geom_smooth(formula = y ~ x,method="lm")
```

```
g6 <- ggplot(imdb,aes(x=log(cast_total_facebook_likes),y=imdb_score)) + geom_point() + geom_smooth(formu
g7 <- ggplot(imdb,aes(x=log(num_user_for_reviews),y=imdb_score)) + geom_point() + geom_smooth(formula =
g8 <- ggplot(imdb,aes(x=log(budget),y=imdb_score)) + geom_point() + geom_smooth(formula = y ~ x,method=

grid.arrange(g1,g2,g3,g4,g5,g6,g7,g8,nrow=3)
```

```
## Warning: Removed 5 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```



Iz grafova se može zaključiti da će varijable poput num_critic_for_reviews, duration i gross imati jači utjecaj, dok će varijable poput budget i cast_total_facebook_likes imati slabiji.

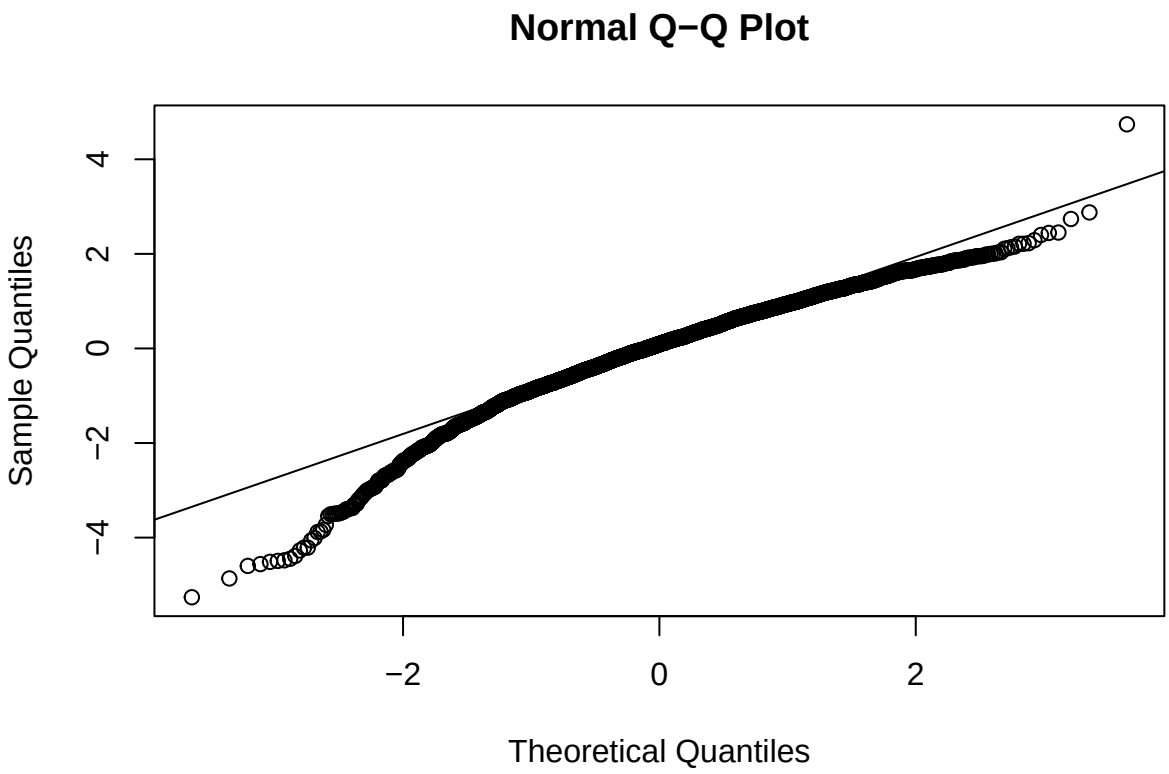
Sada ćemo napraviti model u koji ćemo uključiti sve varijable.

```
model <- lm(imdb_score ~. ,data=imdb)
```

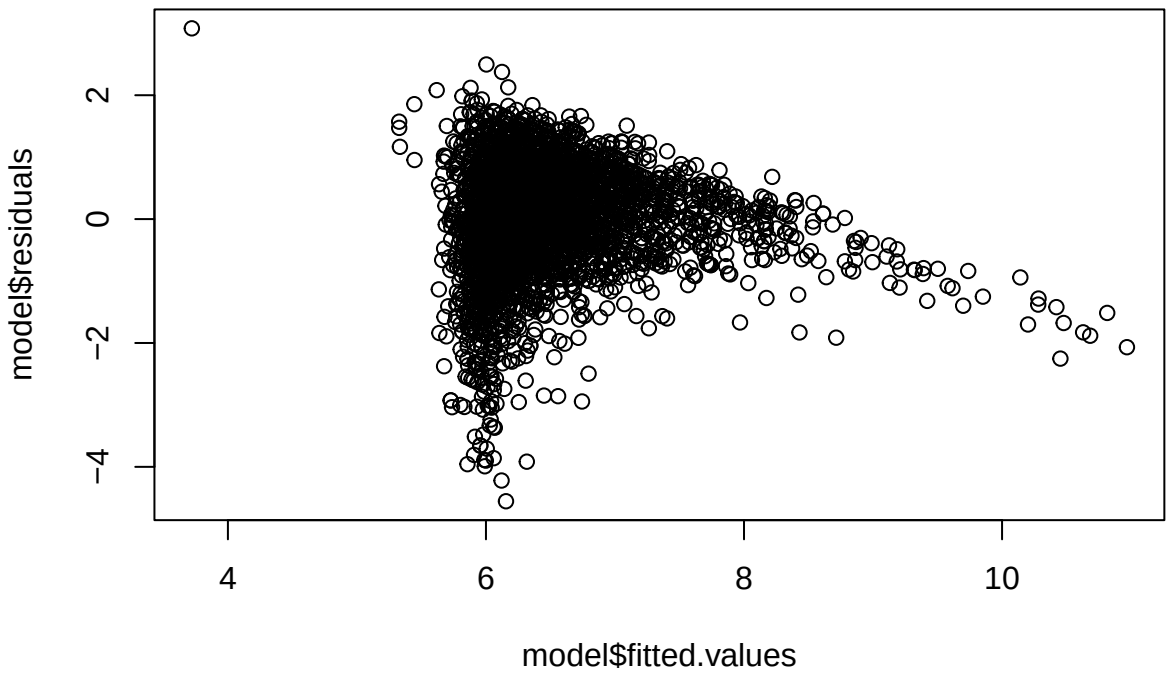
Da bismo analizirali model moramo provjeriti jesu li zadovoljene pretpostavke: regresori ne smiju biti jako korelirani te mora vrijediti normalnost reziduala. To ćemo provjeriti uz pomoć histograma, q-q plota i Lilliefors testa.

```
hist(model$residuals)
```


```
qqnorm(rstandard(model))
qqline(rstandard(model))
```




```
plot(model$fitted.values,model$residuals)
```



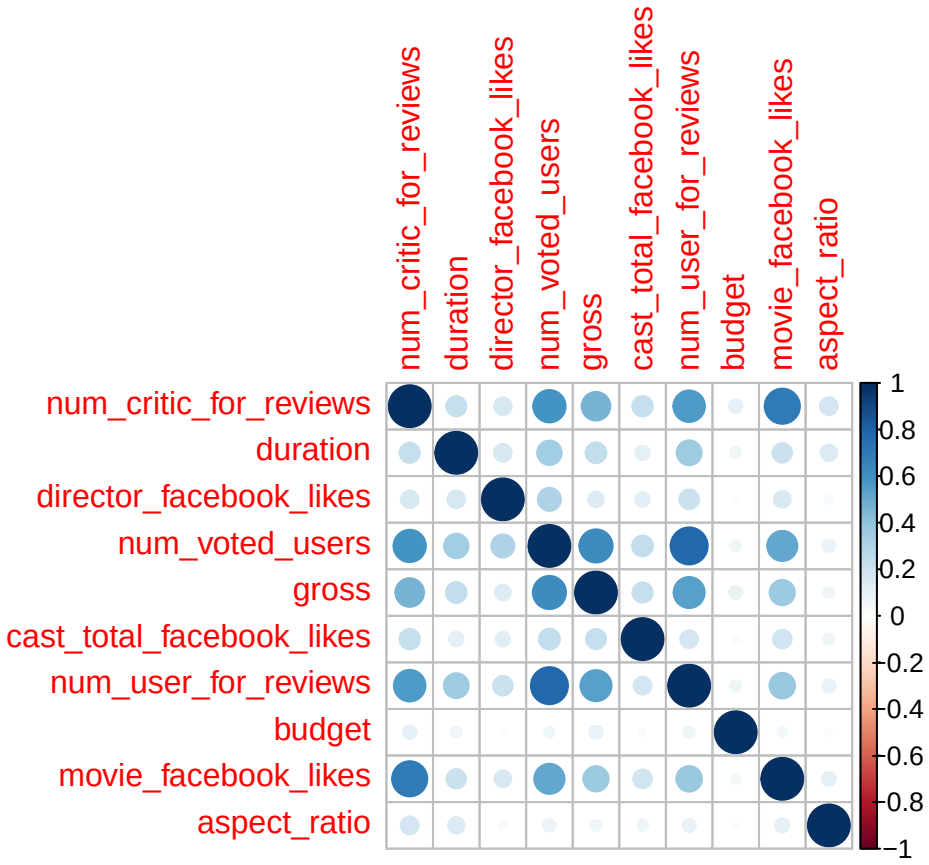

```
lillie.test(rstandard(model))
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  rstandard(model)
## D = 0.047746, p-value < 2.2e-16
```

Vidimo nenormalnost reziduala i da odstupanje nije zanemarivo. No s obzirom na veliki broj opservacija i činjenicu da je t-test robustan na nenormalnost, probat ćemo donijeti neke zaključke.

Pogledajmo koje varijable su jako korelirane.

```
imdb %>% select(-imdb_score,-color) %>% cor %>% corrplot
```



Vidimo da su varijable num_critic_for_reviews i movie_facebook_likes te num_voted_users i num_user_for_reviews jako korelirane. Zbog toga ćemo izbaciti varijable movie_facebook_likes te num_user.

```
fit.num_critic = lm(imdb_score~ num_critic_for_reviews,data=imdb)
fit.movie_face = lm(imdb_score~ movie_facebook_likes ,data=imdb)

summary(fit.num_critic)
```

```
##
## Call:
## lm(formula = imdb_score ~ num_critic_for_reviews, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6280 -0.5663  0.0691  0.6782  2.7339
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            5.9810978  0.0268853  222.47   <2e-16 ***
## num_critic_for_reviews 0.0029399  0.0001301   22.59   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9898 on 3780 degrees of freedom
## Multiple R-squared:  0.1189, Adjusted R-squared:  0.1187
```

```
## F-statistic: 510.3 on 1 and 3780 DF,  p-value: < 2.2e-16
```

```
summary(fit.movie_face)
```

```
##
## Call:
## lm(formula = imdb_score ~ movie_facebook_likes, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5992 -0.5529  0.1081  0.6803  2.4665
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.339e+00  1.792e-02  353.86   <2e-16 ***
## movie_facebook_likes 1.387e-05  7.673e-07   18.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.012 on 3780 degrees of freedom
## Multiple R-squared:  0.07955,    Adjusted R-squared:  0.07931
## F-statistic: 326.7 on 1 and 3780 DF,  p-value: < 2.2e-16
```

Varijabla num_critic_for_reviews objašnjava više varijablinosti te ćemo stoga uzeti nju. Napravimo isto za drugi par.

```
fit.num_voted = lm(imdb_score~ num_voted_users,data=imdb)
fit.num_users = lm(imdb_score~ num_user_for_reviews ,data=imdb)

summary(fit.num_voted)
```

```
##
## Call:
## lm(formula = imdb_score ~ num_voted_users, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7679 -0.5231  0.0801  0.6443  2.3050
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     6.120e+00  1.831e-02   334.3   <2e-16 ***
## num_voted_users 3.331e-06  9.975e-08    33.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9266 on 3780 degrees of freedom
## Multiple R-squared:  0.2278, Adjusted R-squared:  0.2276
## F-statistic:  1115 on 1 and 3780 DF,  p-value: < 2.2e-16
```

```
summary(fit.num_users)
```

```
##
## Call:
## lm(formula = imdb_score ~ num_user_for_reviews, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8766 -0.5461  0.1270  0.7062  2.2690
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.192e+00  2.090e-02  296.23   <2e-16 ***
## num_user_for_reviews 8.292e-04  3.963e-05   20.92   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9983 on 3780 degrees of freedom
## Multiple R-squared:  0.1038, Adjusted R-squared:  0.1036
## F-statistic: 437.8 on 1 and 3780 DF,  p-value: < 2.2e-16
```

Varijabla num_voted_users objašnjava više varijabilnosti nego num_user_for_reviews.

Sada ćemo provesti višestruku linearnu regresiju. Varijabla color je nominalna kategorijska varijabla stoga treba napraviti dummy varijable, no to funkcija lm odradi automatski.

```
model <- lm(formula =  imdb_score ~ num_critic_for_reviews + duration + director_facebook_likes + num_v

summary(model)
```

```
##
## Call:
## lm(formula = imdb_score ~ num_critic_for_reviews + duration +
##     director_facebook_likes + num_voted_users + gross + cast_total_facebook_likes +
##     budget + color + aspect_ratio, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7282 -0.4976  0.1027  0.6147  3.0559
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 5.246e+00  6.315e-01   8.308  < 2e-16 ***
## num_critic_for_reviews      1.058e-03  1.485e-04   7.128 1.22e-12 ***
## duration                    1.093e-02  6.788e-04  16.101  < 2e-16 ***
## director_facebook_likes     7.152e-06  4.955e-06   1.443 0.148990
## num_voted_users             2.956e-06  1.426e-07  20.730  < 2e-16 ***
## gross                      -2.405e-09  2.695e-10  -8.925  < 2e-16 ***
## cast_total_facebook_likes -9.751e-07  7.867e-07  -1.240 0.215223
## budget                     -4.353e-11  6.402e-11  -0.680 0.496544
## color Black and White       4.333e-01  6.268e-01   0.691 0.489420
## colorColor                 -1.584e-02  6.220e-01  -0.025 0.979681
## aspect_ratio               -1.582e-01  4.163e-02  -3.800 0.000147 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.8791 on 3771 degrees of freedom
## Multiple R-squared:  0.3066, Adjusted R-squared:  0.3048
## F-statistic: 166.8 on 10 and 3771 DF,  p-value: < 2.2e-16
```

Vidimo da varijable budget, cast_total_facebook_likes i director_facebook_likes potencijalno nisu toliko korisne u modelu te ih možda možemo izbaciti.

```
model <- lm(formula =  imdb_score ~ num_critic_for_reviews + duration + num_voted_users + gross + color

summary(model)
```

```
##
## Call:
## lm(formula = imdb_score ~ num_critic_for_reviews + duration +
##     num_voted_users + gross + color + aspect_ratio, data = imdb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7204 -0.4948  0.1025  0.6157  3.0788
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             5.239e+00  6.316e-01   8.296  < 2e-16 ***
## num_critic_for_reviews  1.034e-03  1.475e-04   7.014 2.73e-12 ***
## duration                1.097e-02  6.756e-04  16.235  < 2e-16 ***
## num_voted_users         2.993e-06  1.386e-07  21.592  < 2e-16 ***
## gross                  -2.469e-09  2.675e-10  -9.230  < 2e-16 ***
## color Black and White   4.432e-01  6.269e-01   0.707 0.479562
## colorColor             -1.430e-02  6.220e-01  -0.023 0.981666
## aspect_ratio           -1.595e-01  4.162e-02  -3.832 0.000129 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8792 on 3774 degrees of freedom
## Multiple R-squared:  0.3059, Adjusted R-squared:  0.3046
## F-statistic: 237.6 on 7 and 3774 DF,  p-value: < 2.2e-16
```

Prilagođeni koeficijent determinacije se povećao za vrlo mali iznos. Model objašnjava 30.46% varijacije što, s obzirom na primjenu, možemo reći da je dobar rezultat. F-test upućuje da je model značajan. Ako želimo veći prilagođeni koeficijent determinacije, moramo uključiti sve varijable, no time ćemo izgubiti na interpretaciji rezultata. Bitno je napomenuti da kod ovog modela ne vrijede pretpostavke za linearnu regresiju. Iz grafa u kojem gledamo odnos reziduala i procijenjenih vrijednosti vidi se heteroskedastičnost što može uzrokovati greške kod p-vrijednosti. Za ove podatke potreban je drugačiji model, a ne linearna regresija.

Pitanje 4: Razlikuju li se ocjene s obzirom na vrijeme premijere?

Pošto nam za analizu ovog pitanja ne trebaju oni filmovi koji u stupcima "title_year" ili "imdb_score" imaju nedostajuću vrijednost, najprije ćemo njih izbaciti iz okvira. Također želimo izbaciti potencijalne duplikate.

```
filmovi4 <- filter(filmovi, !is.na(title_year) & !is.na(imdb_score))
filmovi4 <- unique(filmovi4)
```

Uzevši u obzir da u datasetu postoji 91 različita godina (te je shodno tome za neke od njih vrlo malen uzorak filmova), analizirat ćemo filmove po desetljećima u kojima su izašli. Dodat ćemo stupac "decade" u podatkovni okvir.

```
filmovi4 <- mutate(filmovi4, decade = title_year - title_year %% 10)
```

Provjerimo sada veličinu uzorka za svako desetljeće:

```
count(filmovi4, decade)
```

```
##    decade    n
## 1    1910    1
## 2    1920    5
## 3    1930   15
## 4    1940   25
## 5    1950   28
## 6    1960   72
## 7    1970  112
## 8    1980  287
## 9    1990  782
## 10   2000 2083
## 11   2010 1481
```

Pošto u 1910-ima postoji samo jedan film, izbacit ćemo ga iz ove analize.

```
filmovi4 <- filter(filmovi4, decade != 1910)
```

Također, primjećujemo da su veličine uzoraka za neka desetljeća manja od 30. Kako bismo proveli ANOVA postupak, najprije moramo provjeriti jesu li ocjene po desetljećima normalno distribuirane. To možemo učiniti upotrebom Lillieforsovog testa, koji je inačica Kolmogorov-Smirnovljevog testa, ali se može upotrijebiti i ako varijanca i aritmetička sredina nisu poznate. Početna hipoteza testa je da je razdioba koju testiramo normalna, a alternativna je suprotna početnoj.

```
require(nortest)
lillie.test(filmovi4$imdb_score[filmovi4$decade==1920])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1920]
## D = 0.31048, p-value = 0.1163
```

```
lillie.test(filmovi4$imdb_score[filmovi4$decade==1930])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1930]
## D = 0.24195, p-value = 0.0183
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1940])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1940]
## D = 0.085221, p-value = 0.9108
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1950])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1950]
## D = 0.17239, p-value = 0.03248
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1960])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1960]
## D = 0.11605, p-value = 0.0176
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1970])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1970]
## D = 0.090583, p-value = 0.02452
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1980])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1980]
## D = 0.090417, p-value = 6.38e-06
```

```r
lillie.test(filmovi4$imdb_score[filmovi4$decade==1990])
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 1990]
## D = 0.060573, p-value = 3.682e-07
```

```
lillie.test(filmovi4$imdb_score[filmovi4$decade==2000])
```

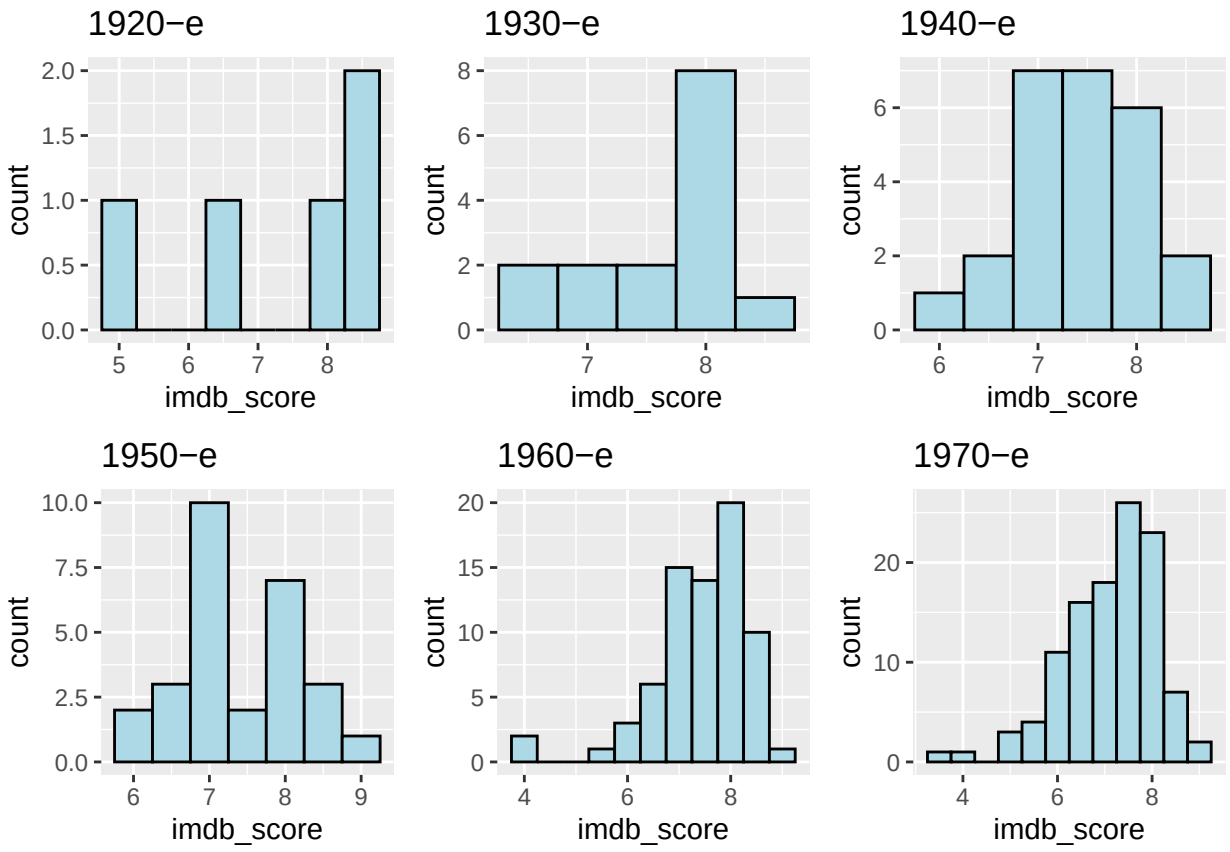
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 2000]
## D = 0.071725, p-value < 2.2e-16
```

```
lillie.test(filmovi4$imdb_score[filmovi4$decade==2010])
```

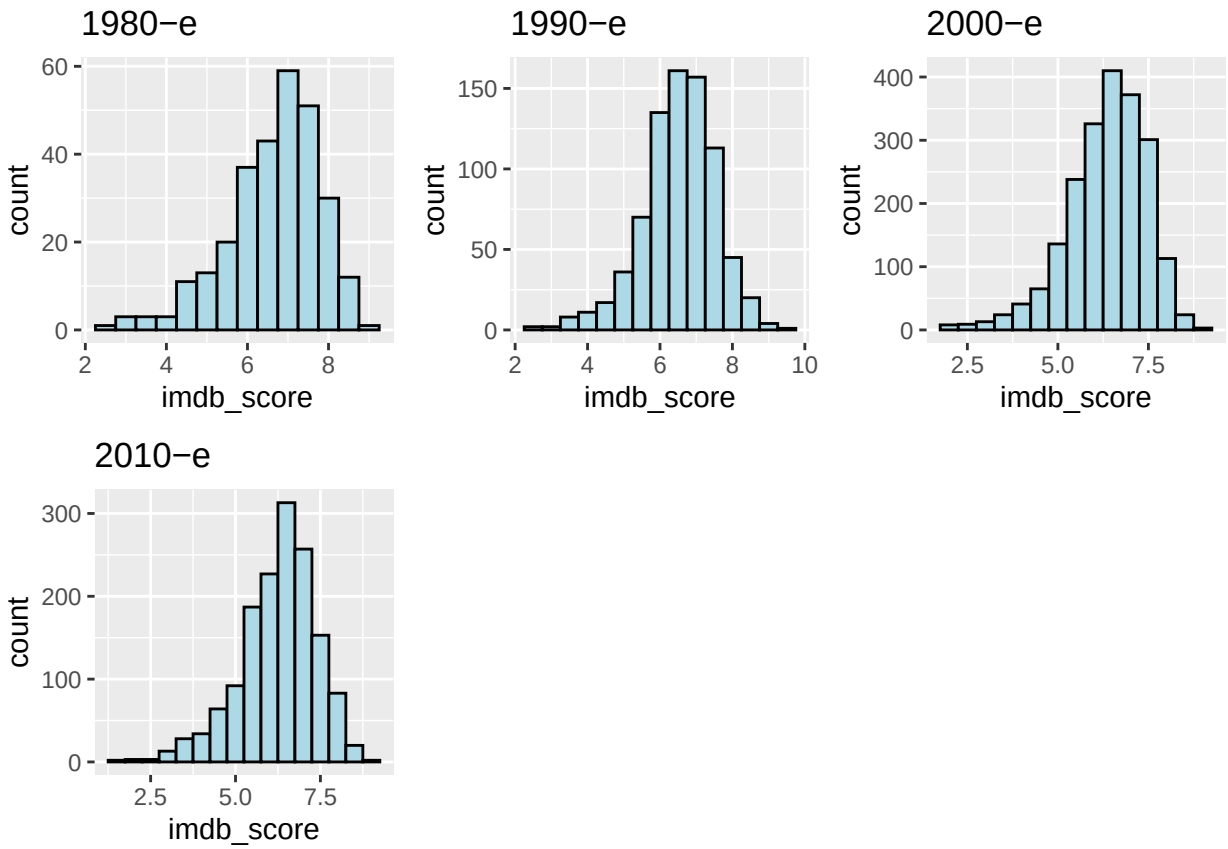
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  filmovi4$imdb_score[filmovi4$decade == 2010]
## D = 0.077139, p-value < 2.2e-16
```

Uočimo da su p-vrijednosti za gotovo sva desetljeća iznimno male. Za sva desetljeća s p-vrijednostima manjima od 0.05 odbacujemo početnu hipotezu, odnosno zaključujemo da razdiobe ocjena filmovima po tim desetljećima nisu normalno distribuirane. Pogledajmo njihove razdiobe na sljedećim grafovima.

```
graf0 <- ggplot(data = filmovi4[filmovi4$decade == 1920, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1920-e")
graf1 <- ggplot(data = filmovi4[filmovi4$decade == 1930, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1930-e")
graf2 <- ggplot(data = filmovi4[filmovi4$decade == 1940, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title ="1940-e")
graf3 <- ggplot(data = filmovi4[filmovi4$decade == 1950, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1950-e")
graf4 <- ggplot(data = filmovi4[filmovi4$decade == 1960, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1960-e")
graf5 <- ggplot(data = filmovi4[filmovi4$decade == 1970, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1970-e")
graf6 <- ggplot(data = filmovi4[filmovi4$decade == 1980, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1980-e")
graf7 <- ggplot(data = filmovi4[filmovi4$decade == 1990, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "1990-e")
graf8 <- ggplot(data = filmovi4[filmovi4$decade == 2000, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "2000-e")
graf9 <- ggplot(data = filmovi4[filmovi4$decade == 2010, ], aes(x = imdb_score)) +
    geom_histogram(binwidth = 0.5, fill = "lightblue", color = "black") +
    labs(title = "2010-e")
grid.arrange(graf0, graf1, graf2, graf3, graf4, graf5, ncol = 3)
```

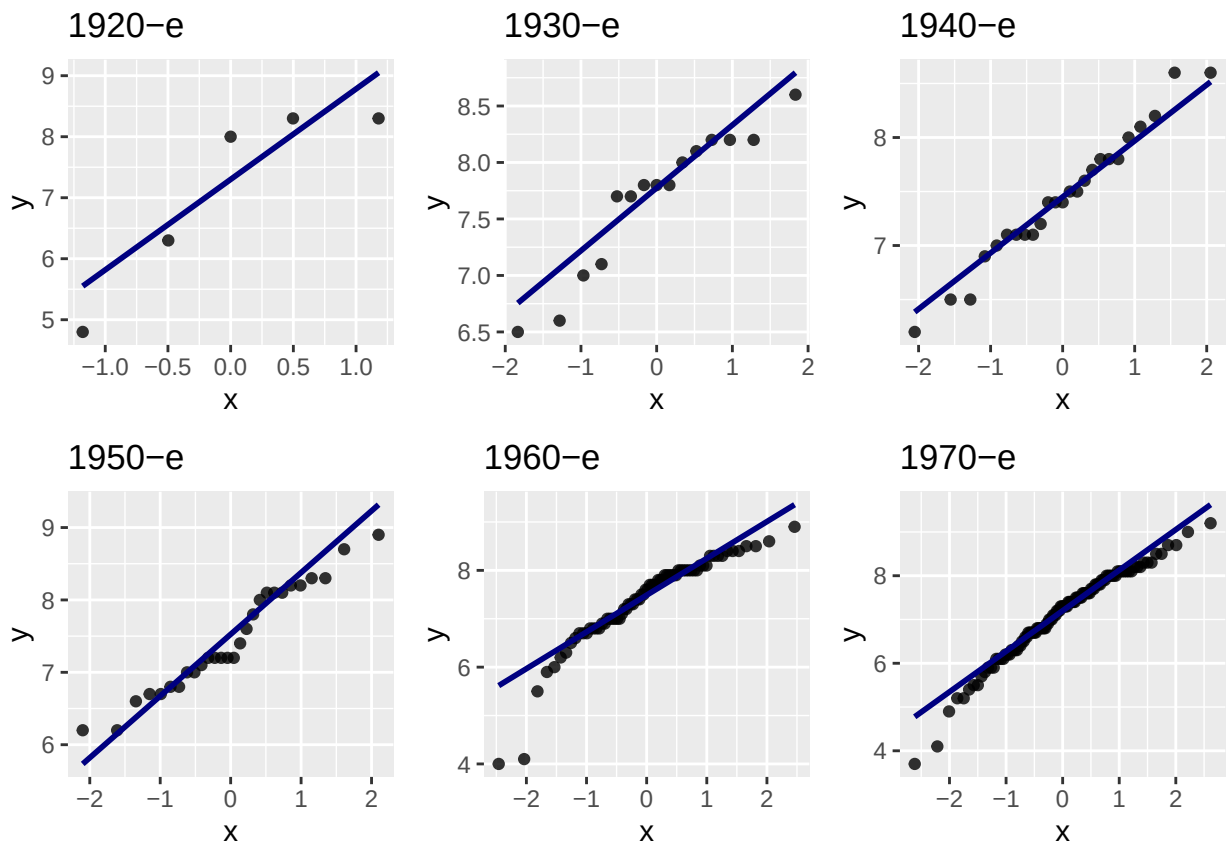



```
grid.arrange(graf6, graf7, graf8, graf9, ncol = 3)
```

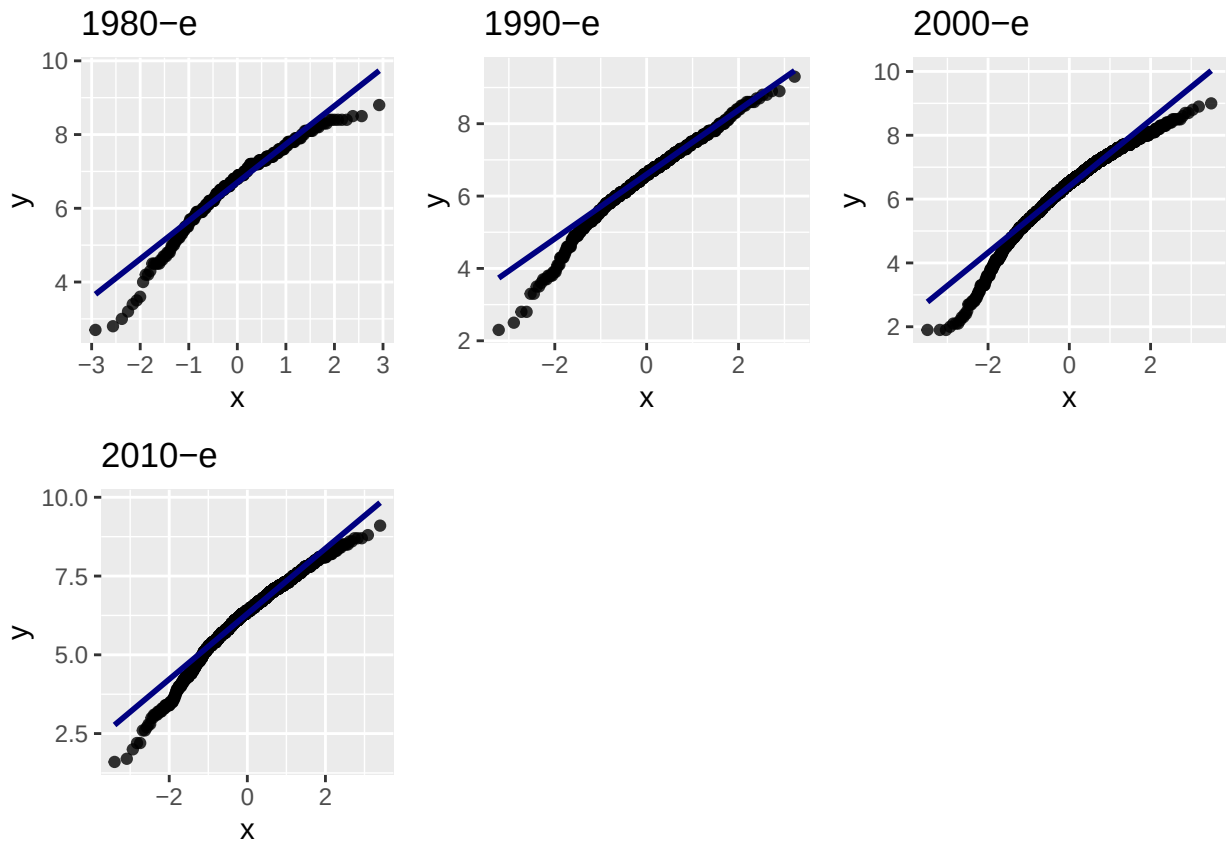


Histogrami pokazuju asimetriju u distribuciji. To ćemo još bolje prikazati na sljedećim QQ grafovima:

```
graf0 <- ggplot(filmovi4[filmovi4$decade==1920, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf1 <- ggplot(filmovi4[filmovi4$decade==1930, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf2 <- ggplot(filmovi4[filmovi4$decade==1940, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf3 <- ggplot(filmovi4[filmovi4$decade==1950, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf4 <- ggplot(filmovi4[filmovi4$decade==1960, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf5 <- ggplot(filmovi4[filmovi4$decade==1970, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf6 <- ggplot(filmovi4[filmovi4$decade==1980, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf7 <- ggplot(filmovi4[filmovi4$decade==1990, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf8 <- ggplot(filmovi4[filmovi4$decade==2000, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st
graf9 <- ggplot(filmovi4[filmovi4$decade==2010, ], aes(sample = imdb_score)) + geom_qq(alpha = 0.8) + st

grid.arrange(graf0, graf1, graf2, graf3, graf4, graf5, ncol = 3)
```




```
grid.arrange(graf6, graf7, graf8, graf9, ncol = 3)
```



Kada bi distribucije bile normalne, podatci na QQ grafu bi bili posloženi približno u ravnoj liniji, dok na ovim grafovima primjećujemo zakrivljenost tih linija. Time smo i grafički prikazali kako distribucije ocjena filmova po desetljećima nisu normalne.

Učinimo još Bartlettov test za provjeru homogenosti varijanci. Početna hipoteza je da su varijance za sva desetljeća jednake, dok je alternativna hipoteza suprotna.

```
bartlett.test(filmovi4$imdb_score ~ filmovi4$decade)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  filmovi4$imdb_score by filmovi4$decade
## Bartlett's K-squared = 42.686, df = 9, p-value = 2.462e-06
```

Primjećujemo da je i u ovom testu p-vrijednost jako mala, zbog čega odbacujemo početnu hipotezu i zaključujemo kako varijance nisu jednake. S obzirom da razdiobe po većini desetljeća nisu ni približno normalne, a varijance nisu homogene, trebamo upotrijebiti neparametarsku alternativu ANOVA postupku, a to je Kruskal-Wallisov test. Početna hipoteza testa je da su aritmetičke sredine ocjena u svim desetljećima jednake. Alternativna hipoteza je da se barem jedna od tih sredina razlikuje.

```
kruskal.test(imdb_score ~ decade, data = filmovi4)
```

```
##
##  Kruskal-Wallis rank sum test
```

```
##
## data:  imdb_score by decade
## Kruskal-Wallis chi-squared = 250.22, df = 9, p-value < 2.2e-16
```

Izuzetno mala p-vrijednost nas upućuje da odbacimo početnu hipotezu. Time zaključujemo da se IMDB ocjene filmova razlikuju s obzirom na vrijeme premijere filma. Promotrimo na koji način se razlikuju, odnosno na koji način ocjena ovisi o vremenu premijere filma.

Faktorizirajmo stupac za desetljeće:

```
decades <- paste(seq(1920, 2010, by=10), seq(1929, 2019, by=10), sep="-")
filmovi4$decade <- factor(filmovi4$decade, levels = seq(1920, 2010, 10),
                          labels = decades)
```

Napravimo linearni model za dane podatke:

```
model = lm(imdb_score ~ decade, data = filmovi4)
summary(model)
```

```
##
## Call:
## lm(formula = imdb_score ~ decade, data = filmovi4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6516 -0.6159  0.1449  0.7484  2.8484
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.14000    0.48672  14.670   <2e-16 ***
## decade1930-1939  0.54667    0.56201   0.973   0.3308
## decade1940-1949  0.30400    0.53317   0.570   0.5686
## decade1950-1959  0.31714    0.52839   0.600   0.5484
## decade1960-1969  0.26000    0.50333   0.517   0.6055
## decade1970-1979 -0.01946    0.49746  -0.039   0.9688
## decade1980-1989 -0.49505    0.49094  -1.008   0.3133
## decade1990-1999 -0.62414    0.48827  -1.278   0.2012
## decade2000-2009 -0.78489    0.48730  -1.611   0.1073
## decade2010-2019 -0.88841    0.48754  -1.822   0.0685 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.088 on 4880 degrees of freedom
## Multiple R-squared:  0.04613,    Adjusted R-squared:  0.04437
## F-statistic: 26.22 on 9 and 4880 DF,  p-value: < 2.2e-16
```

Prema procjeni ovog modela, prosječna IMDB ocjena filmova raste po desetljećima do 1930-ih, nakon kojih počinje padati. Pogledajmo kakav rezultat bismo dobili kad bismo prema njemu izvršili ANOVA postupak.

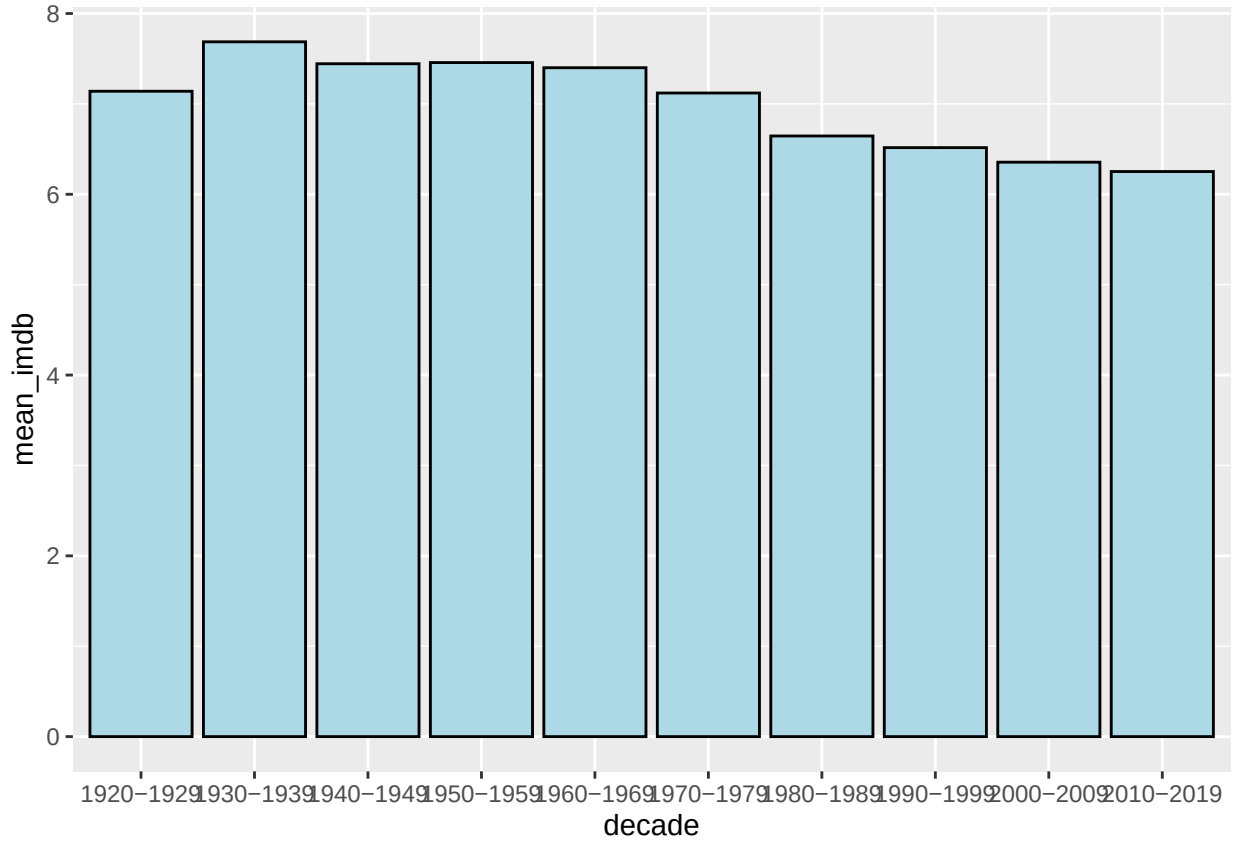
```
anova(model)
```

```
## Analysis of Variance Table
```

```
##
## Response: imdb_score
##             Df Sum Sq Mean Sq F value    Pr(>F)
## decade       9  279.5 31.0573   26.22 < 2.2e-16 ***
## Residuals 4880 5780.2  1.1845
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Iz navedenog postupka vidimo kako je p-vrijednost ponovno izuzetno malena, odnosno da se prosječna ocjena filmova razlikuje s obzirom na desetljeće. Rezultat testa slaže se s gore provedenim neparametarskim postupkom. Pogledajmo sada podatke po desetljećima na određenim grafovima kako bismo vizualno predočili tu razliku. Najprije ćemo prikazati histogram prosječnih ocjena po desetljećima.

```
f4_mean_scores <- filmovi4 %>%
  group_by(decade) %>%
  summarise(mean_imdb = mean(imdb_score, na.rm = TRUE))

ggplot(f4_mean_scores, aes(x = decade, y = mean_imdb)) + geom_bar(stat = "identity", fill = "lightblue"
```



Kao što je gornji model pokazao u svojim očekivanim vrijednostima, nakon 1930-ih primjećuje se pad prosječne ocjene. Prikažimo sada box-plot dijagram razdioba:

```
ggplot(filmovi4, aes(x = decade, y = imdb_score)) +
  geom_boxplot() +
  labs(x = "Desetljeće", y = "IMDb ocjena", title = "IMDb ocjene po desetljećima")
```

Primjetimo da osim što medijan načelno pada s obzirom na desetljeće, kasnija desetljeća imaju nekoliko stršećih vrijednosti sa jako niskim ocjenama, što nas upućuje da konačno razmislimo o pitanju: znači li ovo uistinu da su stariji filmovi i bolji? Pogledajmo kako se prosječne ocjene odnose prema broju filmova po desetljeću koji su uzeti u obzir. Već smo na početku ispisali te brojeve radi određivanja veličine uzoraka, ali radi jednostavnosti ćemo ih prikazati i ovdje.

```
film_count <- filmovi4 %>%
  group_by(decade) %>%
  summarise(
    count = n(),
    avg_score = mean(imdb_score)
  )

film_count
```

```
## # A tibble: 10 x 3
##    decade    count avg_score
##    <fct>     <int>     <dbl>
##  1 1920-1929     5      7.14
##  2 1930-1939    15      7.69
##  3 1940-1949    25      7.44
##  4 1950-1959    28      7.46
##  5 1960-1969    72      7.4
##  6 1970-1979   112      7.12
##  7 1980-1989   287      6.64
##  8 1990-1999   782      6.52
```

```
##  9 2000-2009  2083      6.36
## 10 2010-2019  1481      6.25
```

Dodajmo ih u tablicu filmova zajedno s prosjecima po desetljećima:

```
filmovi4 <- filmovi4 %>%
  left_join(film_count, by = "decade")
```

Napravimo model koji će pokušati predvidjeti prosječnu ocjenu filma po desetljeću s obzirom na broj filmova iz tog desetljeća.

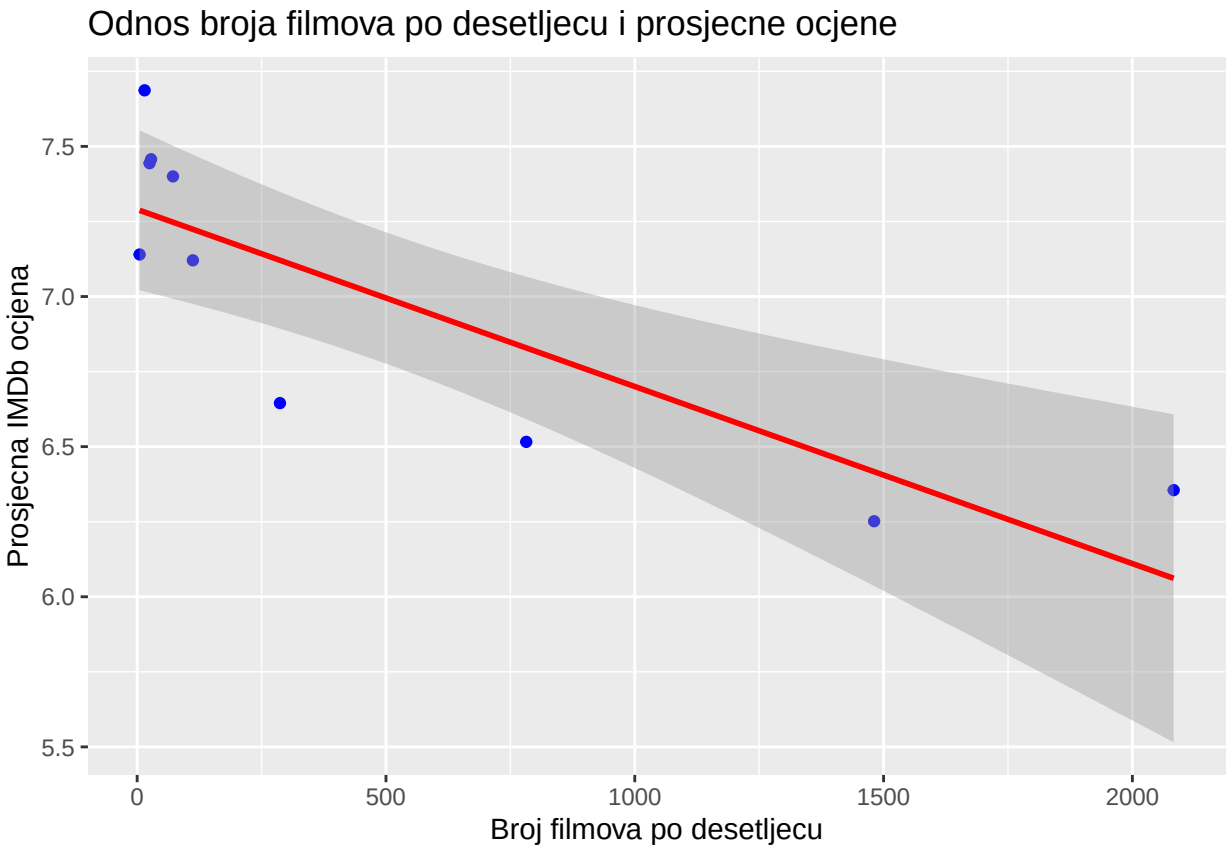
```
model_with_count <- lm(avg_score ~ count, data = filmovi4)

summary(model_with_count)
```

```
##
## Call:
## lm(formula = avg_score ~ count, data = filmovi4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.16475 -0.16475 -0.07962  0.09418  0.89188
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.799e+00  6.095e-03  1115.4   <2e-16 ***
## count       -2.582e-04  3.769e-06   -68.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1708 on 4888 degrees of freedom
## Multiple R-squared:  0.4898, Adjusted R-squared:  0.4897
## F-statistic:  4692 on 1 and 4888 DF,  p-value: < 2.2e-16
```

Primjetimo da s obzirom na dane podatke, prosječna ocjena filmova iz pojedinog desetljeća uistinu ovisi o broju filmova uzetih u obzir iz tog desetljeća. Osim toga, model objašnjava gotovo polovicu varijance među prosječnim vrijednostima, iz čega možemo zaključiti da postoji trend smanjenja prosječnih ocjena po desetljeću što je više filmova u njemu. Prikažimo to i grafički.

```
ggplot(film_count, aes(x = count, y = avg_score)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(
    title = "Odnos broja filmova po desetljeću i prosječne ocjene",
    x = "Broj filmova po desetljeću",
    y = "Prosječna IMDb ocjena"
  )
```

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



Razlog ovakvog silaznog trenda može biti laka dostupnost današnjih filmova. Time je dostupno i više lošijih filmova, dok su od starih filmova dostupni samo oni koji su okarakterizirani kao klasici (dok se oni koji su bili smatrani lošijima u tadašnje vrijeme nisu sačuvali). Osim toga, moguće je da današnje filmove gleda šira publika raznovrsnijih filmskih ukusa, čime dolazi do veće varijabilnosti u korisničkim ocjenama.