Popularnost Spotify pjesama

Projekt iz predmeta *Statistička analiza podataka*

Lucija Aleksić, Domagoj Matošević, Maria Fain, Matko Barbić

U datasetu *SpotifyDB* svaki redak predstavlja pojedinu pjesmu. Svaka pjesma je opisana s 18 varijabli.

```
dataset <- read.csv(file = "SpotifyDB.csv")
```

Osnovne informacije o samom datasetu:

```
# Broj redaka(pjesama), broj stupaca
dim(dataset)
```

```
## [1] 232725     18
```

```
# Imena stupaca
names(dataset)
```

```
##  [1] "ï..genre"         "artist_name"      "track_name"       "track_id"
##  [5] "popularity"       "acousticness"     "danceability"     "duration_ms"
##  [9] "energy"           "instrumentalness" "key"              "liveness"
## [13] "loudness"         "mode"             "speechiness"      "tempo"
## [17] "time_signature"   "valence"
```

- ***genre*** žanr glazbe
- ***artist_name*** ime izvođača
- ***track_name*** ime pjesme
- ***track_id*** jedinstveni identifikator pjesme
- ***popularity*** popularnost pojedine pjesme, poprima vrijednosti 0-100, gdje je 100 najpopularnija
- ***acousticness*** mjera koja govori o akustičnosti pjesme, poprima vrijednosti 0-1, gdje je 1 visoka akustičnost
- ***danceability*** mjera koja govori o pjesnosti pjesme, poprima vrijednosti 0-1, gdje je 1 visoka plesnost
- ***duration_ms*** trajanje pjesme u milisekundama
- ***energy*** mjera koja govori o intenzitetu i aktivnosti pjesme, poprima vrijednosti 0-1, gdje je 1 visoka energija
- ***instrumentalness*** mjera koja govori o omjeru ljudskog glasa i instrumentala, poprima vrijednosti 0-1, gdje je 1 potpuno instrumentalna pjesma bez vokala
- ***key*** tonalitet pjesme
- ***liveness*** prisutnost publike u nastupu, poprima vrijednosti 0-1, gdje je 1 potpuna živost
- ***loudness*** sveukupna glasnoća u dB
- ***mode*** tonski rod ljestvice, dur ili mol
- ***speechiness*** prisutnost izgovorenih riječi u pjesmi, poprima vrijednosti 0-1, gdje je 0 niska govorljivost npr. podcast
- ***tempo*** tempo pjesme izražen u BPM (beats per minute)

- ***time_signature*** zapis mjere vremena
- ***valence*** pozitivnost pjesme, poprima vrijednosti 0-1, gdje je 1 jako sretna i euforična pjesma

---

Deskriptivna statistika

Ovdje ćemo prikazati i opisati varijable koje ćemo najčešće koristiti u daljnjem projektu kako bismo se bolje upoznali. Izračunat ćemo mjere centralne tendencije, rasipanja te po potrebi ih vizualizirati na razne načine.

```
summary(dataset)
```

Osnovna deskriptivna statistika

```
##    ï..genre         artist_name         track_name          track_id
##  Length:232725      Length:232725      Length:232725      Length:232725
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    popularity      acousticness     danceability     duration_ms
##  Min.   :  0.00   Min.   :0.0000   Min.   :0.0569   Min.   :  15387
##  1st Qu.: 29.00   1st Qu.:0.0376   1st Qu.:0.4350   1st Qu.: 182857
##  Median : 43.00   Median :0.2320   Median :0.5710   Median : 220427
##  Mean   : 41.13   Mean   :0.3686   Mean   :0.5544   Mean   : 235122
##  3rd Qu.: 55.00   3rd Qu.:0.7220   3rd Qu.:0.6920   3rd Qu.: 265768
##  Max.   :100.00   Max.   :0.9960   Max.   :0.9890   Max.   :5552917
##      energy         instrumentalness        key              liveness
##  Min.   :2.03e-05   Min.   :0.0000000   Length:232725      Min.   :0.00967
##  1st Qu.:3.85e-01   1st Qu.:0.0000000   Class :character   1st Qu.:0.09740
##  Median :6.05e-01   Median :0.0000443   Mode  :character   Median :0.12800
##  Mean   :5.71e-01   Mean   :0.1483012                      Mean   :0.21501
##  3rd Qu.:7.87e-01   3rd Qu.:0.0358000                      3rd Qu.:0.26400
##  Max.   :9.99e-01   Max.   :0.9990000                      Max.   :1.00000
##     loudness           mode            speechiness         tempo
##  Min.   :-52.457   Length:232725      Min.   :0.0222   Min.   : 30.38
##  1st Qu.:-11.771   Class :character   1st Qu.:0.0367   1st Qu.: 92.96
##  Median : -7.762   Mode  :character   Median :0.0501   Median :115.78
##  Mean   : -9.570                      Mean   :0.1208   Mean   :117.67
##  3rd Qu.: -5.501                      3rd Qu.:0.1050   3rd Qu.:139.05
##  Max.   :  3.744                      Max.   :0.9670   Max.   :242.90
##  time_signature        valence
##  Length:232725      Min.   :0.0000
##  Class :character   1st Qu.:0.2370
##  Mode  :character   Median :0.4440
##                     Mean   :0.4549
##                     3rd Qu.:0.6600
##                     Max.   :1.0000
```

Tempo

```
# Aritmetička sredina
mean(dataset$tempo)
```

```
## [1] 117.6666
```

```
# Podrezana aritmetička sredina
mean(dataset$tempo, trim=0.2)
```

```
## [1] 115.2805
```

```
# Medijan
median(dataset$tempo)
```

```
## [1] 115.778
```

```
# Najbrža pjesma
max(dataset$tempo)
```

```
## [1] 242.903
```

```
# Ime najbrže pjesme
cat(dataset[which.max(dataset$tempo), ]$track_name, ",",
    dataset[which.max(dataset$tempo), ]$artist, "\n")
```

```
## Call The Doctor , J.J. Cale
```

```
# Najsporija pjesma
min(dataset$tempo)
```

```
## [1] 30.379
```

```
# Ime najsporije pjesme
cat(dataset[which.min(dataset$tempo), ]$track_name , ",",
    dataset[which.min(dataset$tempo), ]$artist, "\n")
```

```
## Voices of Winter , Tomoki Miyoshi
```

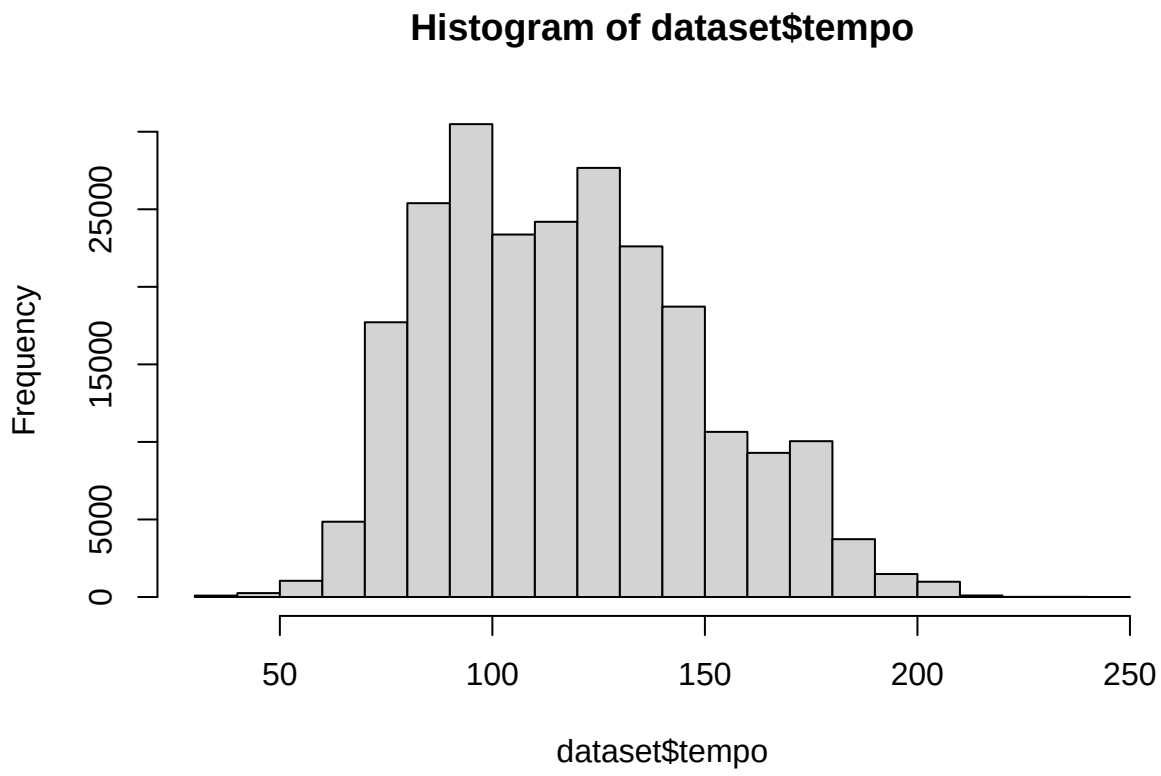
```
# Varijanca i standardna devijacija
var(dataset$tempo)
```

```
## [1] 954.7424
```

```
sd(dataset$tempo)
```

```
## [1] 30.89891
```

```
hist(dataset$tempo, breaks = 20)
```

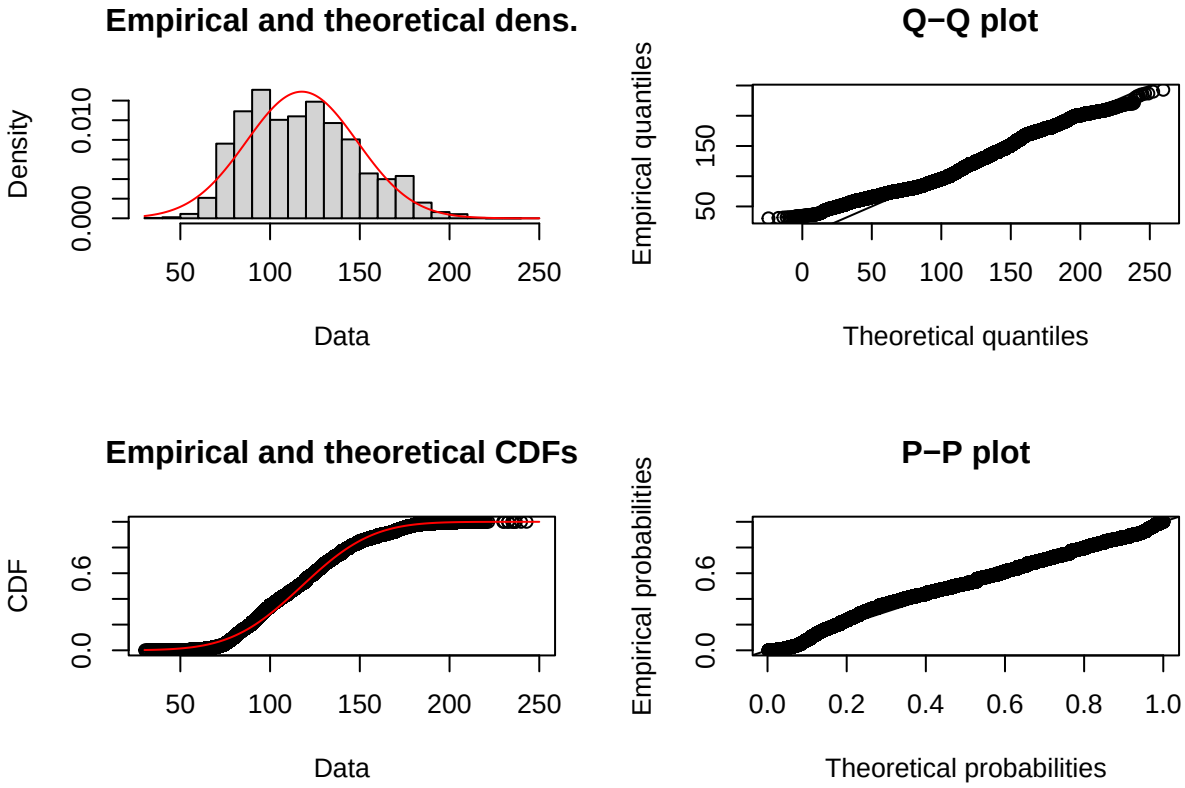



```
# Ispitivanje normalnosti tempa
library(fitdistrplus)
```

```
## Loading required package: MASS
```

```
## Loading required package: survival
```

```
FIT <- fitdistrplus::fitdist(dataset$tempo, "norm")
plot(FIT)
```



Danceability

```
# Aritmetička sredina
mean(dataset$danceability)
```

```
## [1] 0.5543645
```

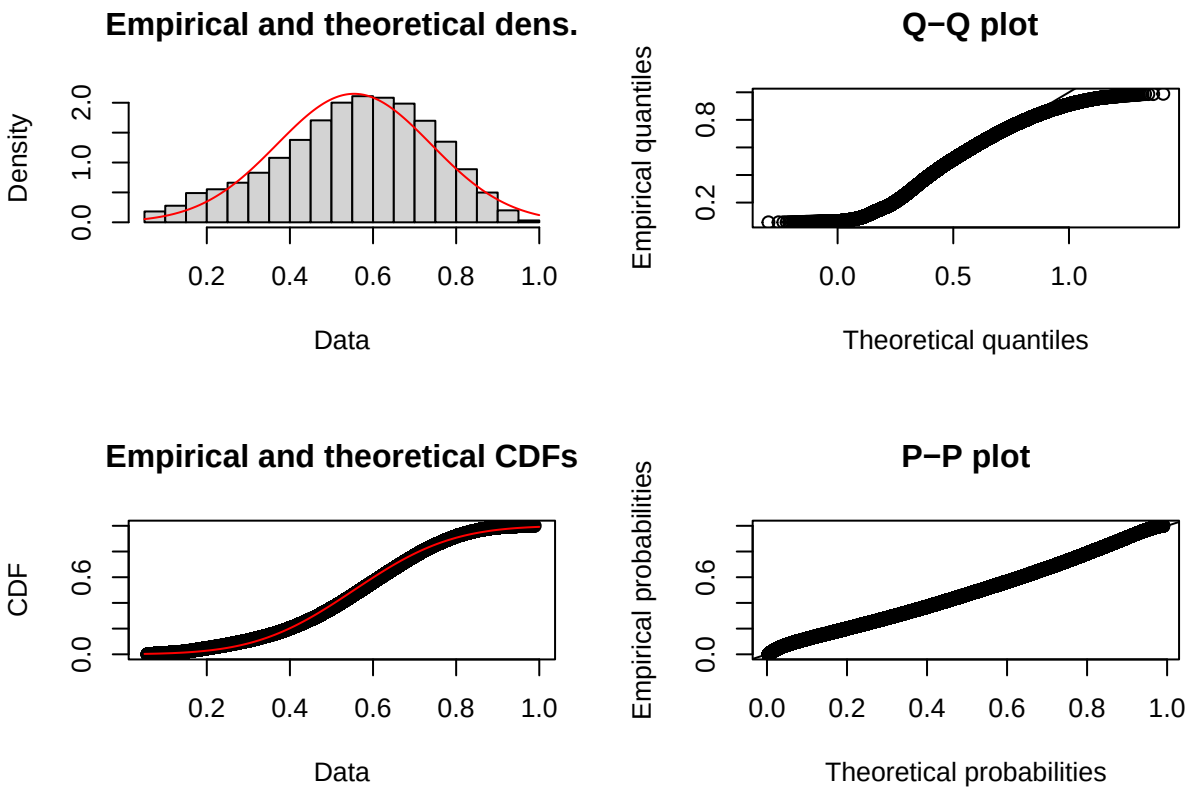
```
# Podrezana aritmetička sredina
mean(dataset$danceability, trim=0.2)
```

```
## [1] 0.5675531
```

```
# Medijan
median(dataset$danceability)
```

```
## [1] 0.571
```

```
# Ispitivanje normalnosti danceability-a
library(fitdistrplus)
FIT <- fitdistrplus::fitdist(dataset$danceability, "norm")
plot(FIT)
```




```
# U kojem žanru su pjesme najplesnje?
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
```

```
dataset %>% group_by(ï..genre) %>% summarise(
  average = mean(danceability)
) -> avgByGenre
avgByGenre[which.max(avgByGenre$average), ]$ï..genre
```

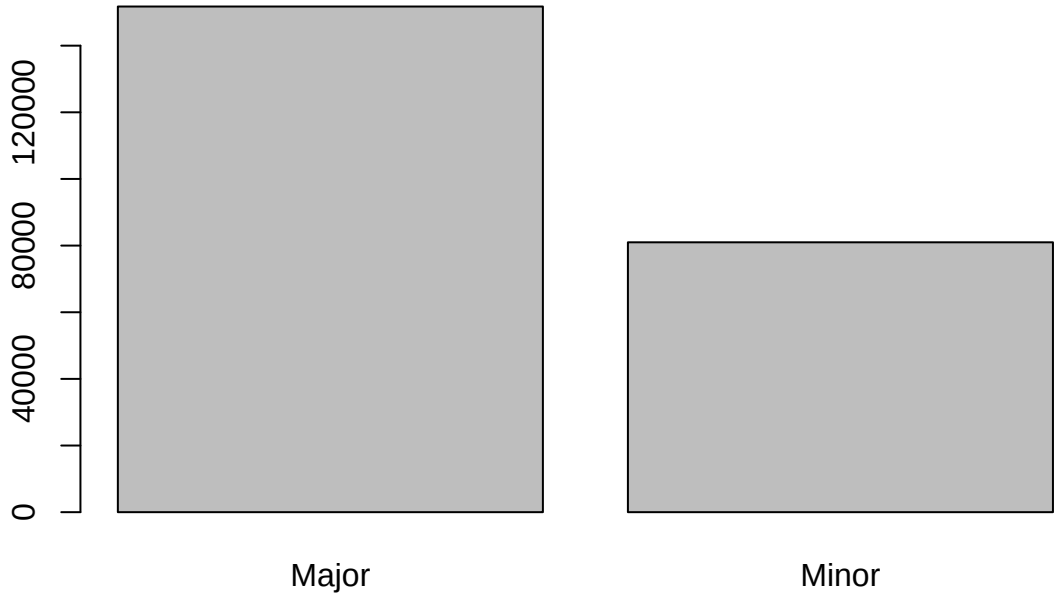
```
## [1] "Reggaeton"
```

Mode

```
table(dataset$mode)
```

```
##
##  Major  Minor
## 151744  80981
```

```
barplot(table(dataset$mode))
```



Popularity

```
summary(dataset$popularity)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   29.00   43.00   41.13   55.00  100.00
```

```
# Aritmetička sredina
mean(dataset$popularity)
```

```
## [1] 41.1275
```

```r
# Podrezana aritmetička sredina
mean(dataset$popularity, trim=0.2)
```

```
## [1] 42.39398
```

```r
# Medijan
median(dataset$popularity)
```

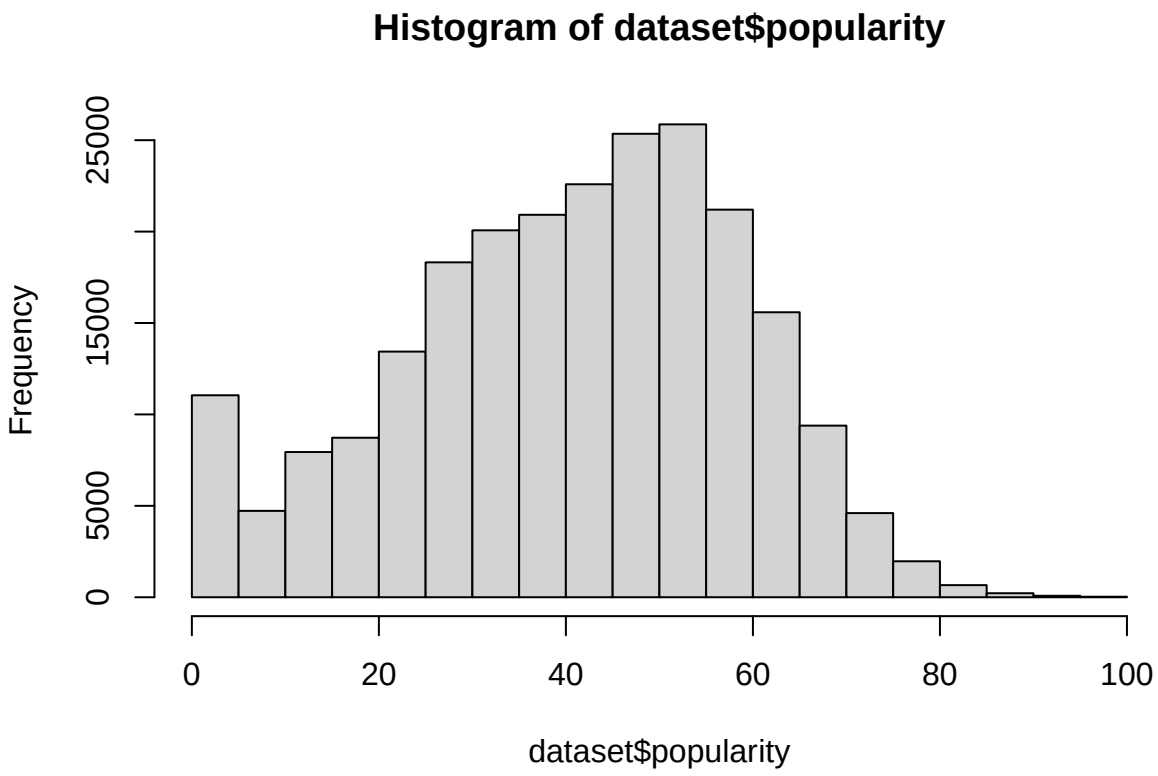
```
## [1] 43
```

Top 5 pjesama

```r
max = max(dataset$popularity)

# Ljestvica 5 najpopularnijih pjesama
for (i in 0:4) {
  song = dataset[which(dataset$popularity == (max - i)),]
  results = distinct(song[,c(2,3)])
  cat("----------------------", i+1, "----------------------", "\n")
      print.data.frame(results)
}
```

```
## ---------------------- 1 ----------------------
##     artist_name track_name
## 1 Ariana Grande    7 rings
## ---------------------- 2 ----------------------
##     artist_name                                track_name
## 1 Ariana Grande break up with your girlfriend, i'm bored
## 2   Post Malone                                     Wow.
## ---------------------- 3 ----------------------
##    artist_name track_name
## 1 Daddy Yankee  Con Calma
## ---------------------- 4 ----------------------
##   artist_name                                  track_name
## 1      Halsey                                 Without Me
## 2     Ava Max                           Sweet but Psycho
## 3 Post Malone Sunflower - Spider-Man: Into the Spider-Verse
## 4   Sam Smith        Dancing With A Stranger (with Normani)
## 5  Marshmello                                     Happier
## 6 Pedro CapÃ³                                Calma - Remix
## ---------------------- 5 ----------------------
##   artist_name                                    track_name
## 1    DJ Snake Taki Taki (with Selena Gomez, Ozuna & Cardi B)
## 2     J. Cole                                  MIDDLE CHILD
## 3   Lady Gaga                                       Shallow
## 4    Anuel Aa                                       Secreto
```

```r
hist(dataset$popularity, breaks = 20)
```



Zanr

```
table(dataset$ï..genre)
```

```
##
##          A Capella        Alternative              Anime              Blues
##                119               9263               8936               9023
##   Children's Music Childrenâ\200\231s Music          Classical             Comedy
##               5403               9353               9256               9681
##            Country              Dance         Electronic               Folk
##               8664               8701               9377               9299
##            Hip-Hop              Indie               Jazz              Movie
##               9295               9543               9441               7806
##              Opera                Pop                R&B                Rap
##               8280               9386               8992               9232
##             Reggae           Reggaeton               Rock                Ska
##               8771               8927               9272               8874
##               Soul         Soundtrack              World
##               9089               9646               9096
```

```
genreFreq <- data.frame(table(dataset$ï..genre))

# Žanr s najviše pjesama
genreFreq[which.max(genreFreq$Freq), ]$Var1
```

```
## [1] Comedy
```

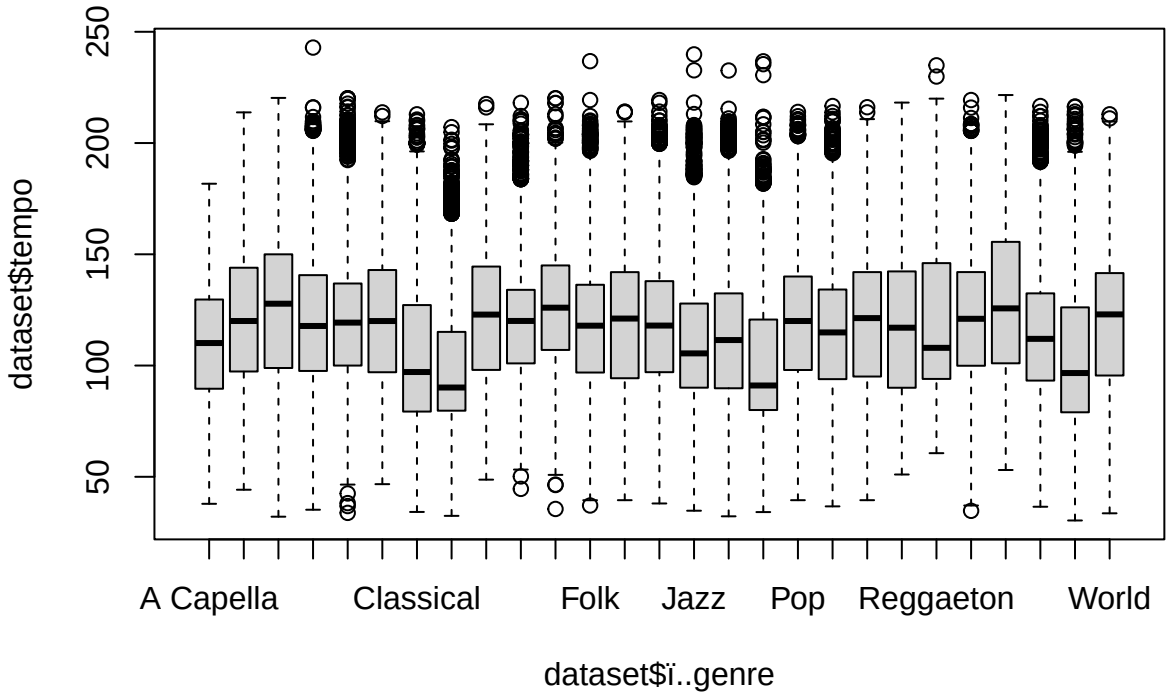
```
## 27 Levels: A Capella Alternative Anime Blues ... World
```

```
# Žanr s najmanje pjesama
genreFreq[which.min(genreFreq$Freq), ]$Var1
```

```
## [1] A Capella
## 27 Levels: A Capella Alternative Anime Blues ... World
```

Imaju li neki zanrovi znacajno razlicit tempo?

```
## Box-plot za tempo
boxplot(dataset$tempo ~ dataset$ï..genre, data = dataset)
```



Iz pravokutnog dijagrama možemo pretpostaviti da postoje značajne razlike u tempu žanrova.Na primjer, možemo vidjeti da žanr Electronic ima prosječno značajno veću vrijednost varijable tempo od žanra Comedy. Kako bi dokazali da postoji razlika koristit ćemo ANOVA test.

```
## Anova
anovaTable <- aov(formula = dataset$tempo ~ dataset$ï..genre, data = dataset)
summary(anovaTable)
```

```
##                      Df    Sum Sq Mean Sq F value Pr(>F)
## dataset$ï..genre     26  13895960  534460   597.1 <2e-16 ***
```

```
## Residuals        232698 208295517     895
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Možemo vidjedi da postoji značajna razlika među tempima žanrova pri značajnosti mnogo manjoj od 0.05. Kako bi našli razliku između pojedinih žanrova koristiti ćemo se Tukey-evim testom.

```
## Tukey post hoc test
tukey <- TukeyHSD(anovaTable)
print(tukey$`dataset$ï..genre`[0:15, ])
```

```
##                                      diff         lwr       upr        p adj
## Alternative-A Capella          11.0155357   0.8126606 21.218411 0.0173468600
## Anime-A Capella                15.1102069   4.9049643 25.315449 0.0000151325
## Blues-A Capella                 9.6186877  -0.5859083 19.823284 0.0984460465
## Children's Music-A Capella      9.6123382  -0.6366597 19.861336 0.1037100578
## Childrenâ\200\231s Music-A Capella  10.4121547   0.2099023 20.614407 0.0385251875
## Classical-A Capella            -7.1771430 -17.3800670  3.025781 0.6563483042
## Comedy-A Capella              -13.2834612 -23.4835420 -3.083380 0.0004786782
## Country-A Capella              11.8954699   1.6881222 22.102817 0.0048274093
## Dance-A Capella                 9.2769690  -0.9300845 19.484023 0.1413021833
## Electronic-A Capella           14.3270170   4.1249287 24.529105 0.0000703276
## Folk-A Capella                  7.2299323  -2.9726922 17.432557 0.6409987379
## Hip-Hop-A Capella               9.2720889  -0.9305634 19.474741 0.1414277996
## Indie-A Capella                 7.7718645  -2.4291118 17.972841 0.4801892947
## Jazz-A Capella                  0.2647082  -9.9369468 10.466363 1.0000000000
## Movie-A Capella                 2.4278203  -7.7871250 12.642766 0.9999999893
```

S obzirom na to da postoji puno kombinacija vizualizirati ćemo samo pet rezultata koji su međusobno najviše različiti.

```
## Sortiranje rezultata tukey-evog testa
tukeyData <- as.data.frame(tukey$`dataset$ï..genre`)
## plot(tukey)

orderedTukey <- tukeyData[order(abs(tukeyData[, 1]), decreasing = TRUE),]

print(orderedTukey[0:5, ])
```

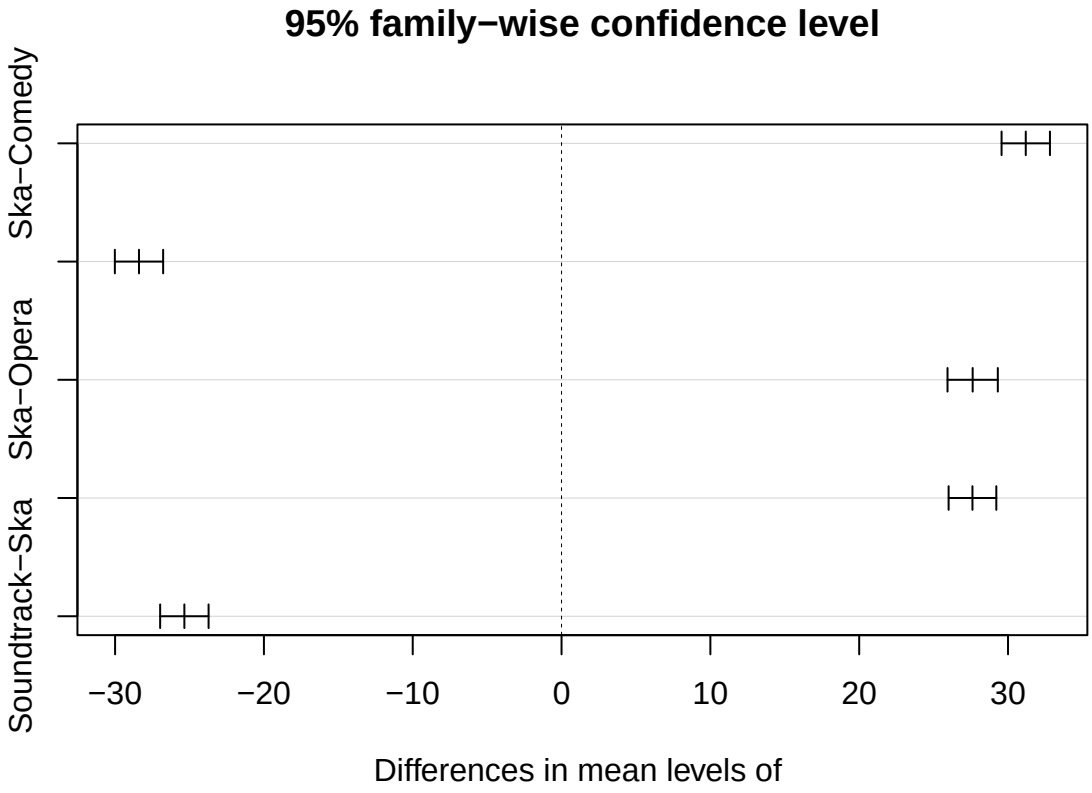
```
##                         diff       lwr       upr p adj
## Ska-Comedy          31.19213  29.56683  32.81744     0
## Comedy-Anime       -28.39367 -30.01603 -26.77131     0
## Ska-Opera           27.62465  25.93486  29.31443     0
## Electronic-Comedy   27.61048  26.00808  29.21288     0
## Soundtrack-Ska     -25.34411 -26.97083 -23.71740     0
```

```
## Funkcija za crtanje grafa za prikaz top pet rezultata sa najvećom razlikom u prosječnoj vrijednosti
tukeyPlot <- function (x, ...)
{
  for (i in seq_along(x)) {
    xi <- x[[i]][, -4L, drop = FALSE]
    yvals <- nrow(xi):1L
```

```
    dev.hold()
    on.exit(dev.flush())
    plot(c(xi[, "lwr"], xi[, "upr"]), rep.int(yvals,
                                              2L), type = "n", axes = FALSE, xlab = "",
         ylab = "", main = NULL, ...)
    axis(1, ...)
    axis(2, at = nrow(xi):1, labels = dimnames(xi)[[1L]],
         srt = 0, ...)
    abline(h = yvals, lty = 1, lwd = 0.5, col = "lightgray")
    abline(v = 0, lty = 2, lwd = 0.5, ...)
    segments(xi[, "lwr"], yvals, xi[, "upr"],
             yvals, ...)
    segments(as.vector(xi), rep.int(yvals - 0.1, 3L), as.vector(xi),
             rep.int(yvals + 0.1, 3L), ...)
    title(main = paste0(format(100 * attr(x, "conf.level"),
                               digits = 2L), "95% family-wise confidence level\n"),
          xlab = paste("Differences in mean levels of",
                       names(x)[i]))
    box()
    dev.flush()
    on.exit()
  }
}
```

Vizualizacija rezultata

```
## Crtanje sortiranih rezultata
tukeyPlot(list(data.matrix(orderedTukey[0:5, ])))
```



Možemo vidjeti da je najveća razlika između žanrova Ska i Comedy, a kao što smo i pretpostavili iz pravokutnog dijagrama postoji i značajna razlika između žanrova Electronic i Comedy.

Jesu li pjesme pisane u duru "plesnije" od onih pisane u molu?

```
cat('Major označava pjesme u duru kojih ima', sum(table(dataset$mode)[c(1)]), ', a minor označava pjesm
```

```
## Major oznacava pjesme u duru kojih ima 151744 , a minor oznacava pjesme u molu kojih ima 151744
```

```
cat('Zbroj tih vrijednosti jednak je broju redaka naše baze (', sum(table(dataset$mode)), ') što znači
```

```
## Zbroj tih vrijednosti jednak je broju redaka naše baze ( 232725 ) što znaci da svaka pjesma ima defi
```

```
table(dataset$mode)
```

```
##
##  Major  Minor
## 151744  80981
```

```
#definiramo pjesme u duru i pjesme u molu
major_songs = dataset[dataset$mode == "Major",]
minor_songs = dataset[dataset$mode == "Minor",]
```

Izračun očekivanja plesnosti za pjesme pisane u duru i pjesme pisane u molu.

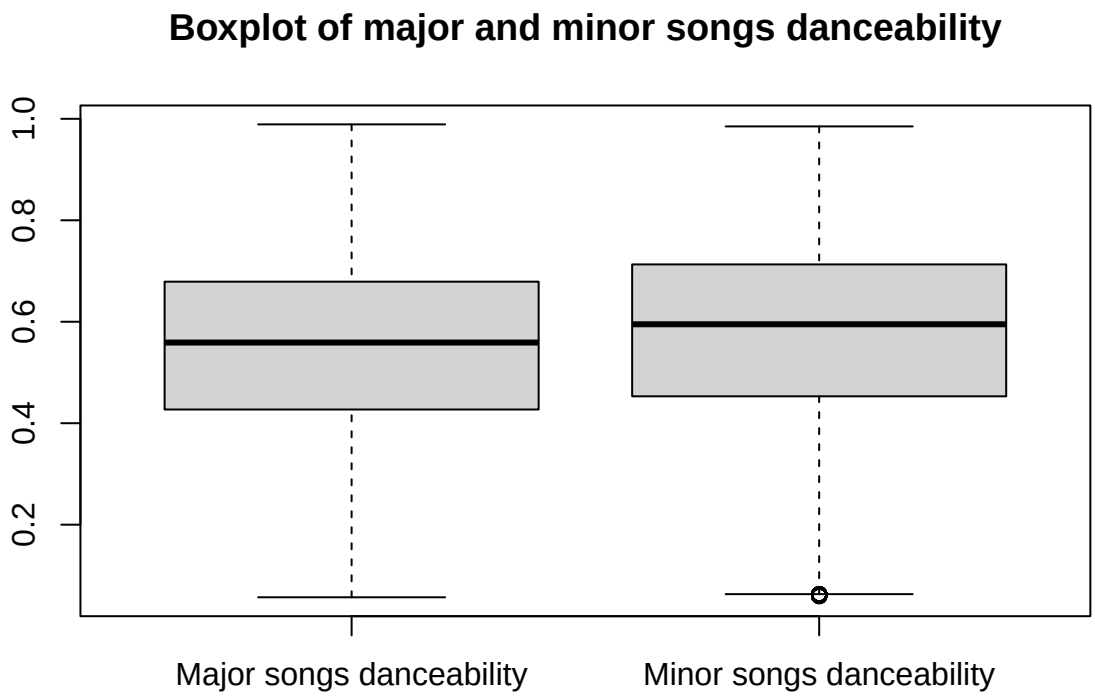
```
cat('Očekivana plesnost pjesama pisanih u duru iznosi ', mean(major_songs$danceability), '\n')
```

```
## Ocekivana plesnost pjesama pisanih u duru iznosi  0.5459727
```

```
cat('Očekivana plesnost pjesama pisanih u molu iznosi ', mean(minor_songs$danceability), '\n')
```

```
## Ocekivana plesnost pjesama pisanih u molu iznosi  0.5700892
```

```
boxplot(major_songs$danceability, minor_songs$danceability,
        names = c('Major songs danceability','Minor songs danceability'),
        main='Boxplot of major and minor songs danceability')
```



Postoje indikacije da bi pjesme pisane u molu trebale biti "plesnije" od onih pisanih u duru.

Da bi provjerili jesu li pjesme pisane u duru "plesnije" od onih pisanih u molu koristit ćemo t-test za dva uzorka. Također ćemo provjeriti jednakost njihovih varijanci da bi utvrdili koji točno t-test možemo koristiti jer razlikujemo t-test s pretpostavkom jednakosti varijanci i t-test s pretpostavkom nejednakosti varijanci.

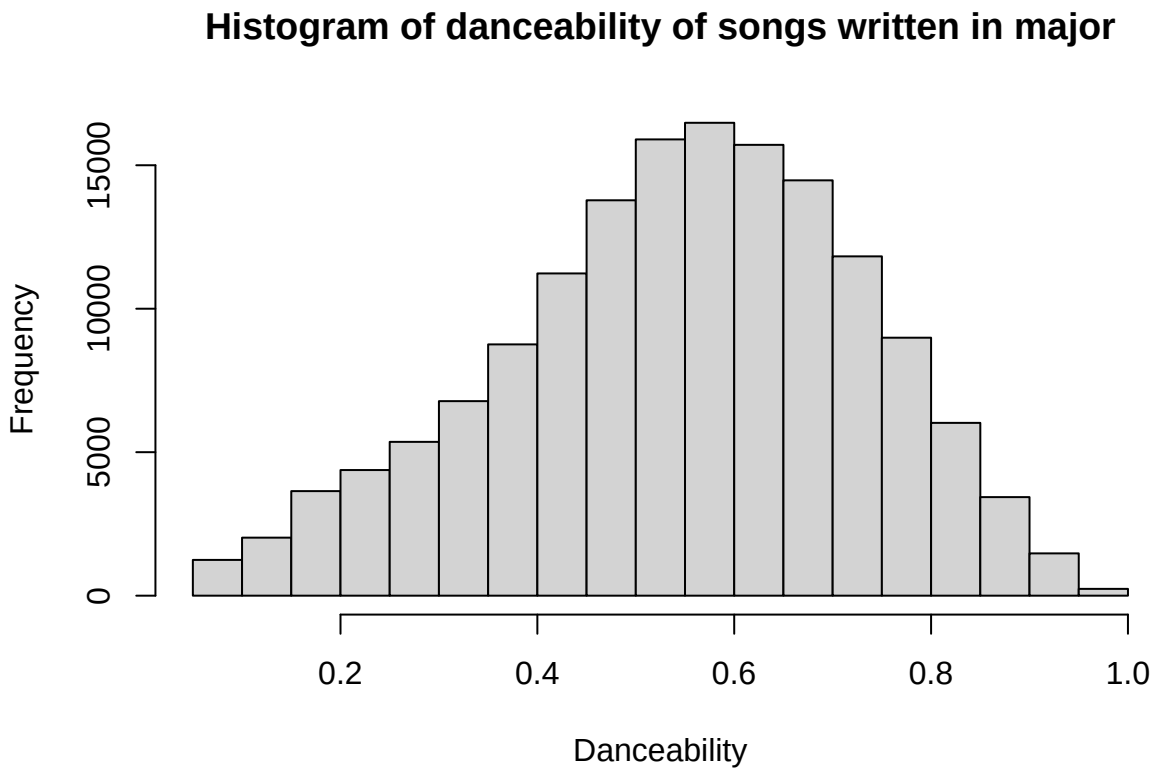
Hipoteze tada glase ovako:

$$H_0 : \mu_1 = \mu_2$$
$$H_1 : \mu_1 < \mu_2 \quad , \quad \mu_1 > \mu_2 \quad , \quad \mu_1 \neq \mu_2$$

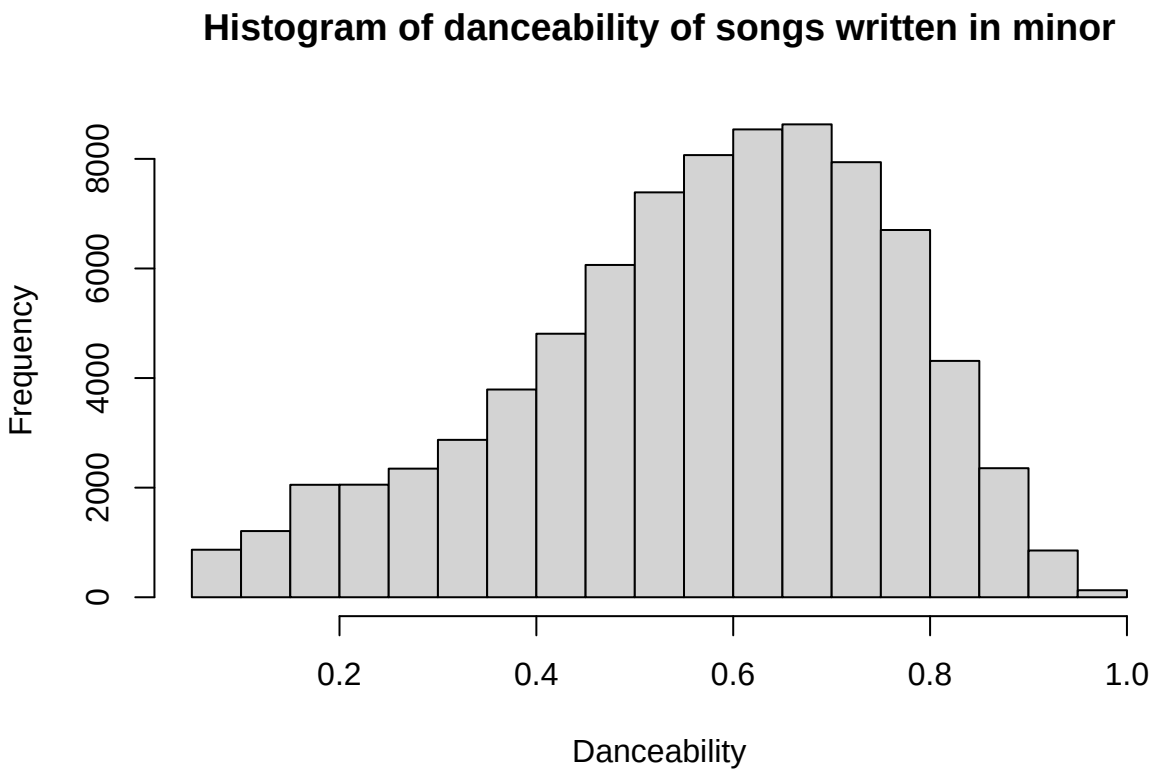
μ_1 i μ_2 označavaju očekivanja plesnosti pjesama pisanih u duru i pjesama pisanih u molu, respektivno.

Kako bismo mogli provesti test, moramo najprije provjeriti pretpostavke normalnosti i nezavisnosti uzorka. Obzirom da razmatramo pjesme pisane u drukčijim tonskim rodovima, možemo pretpostaviti njihovu nezavisnost. Sljedeći korak je provjeriti normalnost podataka koju ćemo provjeriti histogramom.

```
hist(major_songs$danceability,
     main='Histogram of danceability of songs written in major',
     xlab='Danceability')
```




```
hist(minor_songs$danceability,
     main='Histogram of danceability of songs written in minor',
     xlab='Danceability')
```



Histogrami nam pokazuju da su podaci normalne distribucije.

Zatim ispitujemo jednakost varijanci da utvrdimo koji t-test ćemo koristiti.

Pogledajmo vrijednosti varijanci naših uzoraka.

```
var(major_songs$danceability)
```

```
## [1] 0.03344804
```

```
var(minor_songs$danceability)
```

```
## [1] 0.03594989
```

Moramo ispitati jesu li značajno različite.

Ako imamo dva nezavisna slučajna uzorka $X_1^1, X_1^2, \dots X_1^{n_1}$ i $X_2^1, X_2^2, \dots, X_2^{n_2}$ koji dolaze iz normalnih distribucija s varijancama σ_1^2 i σ_2^2, tada slučajna varijabla

$$F = \frac{S_{X_1}^2/\sigma_1^2}{S_{X_2}^2/\sigma_2^2}$$

ima Fisherovu distribuciju s $(n_1 - 1, n_2 - 1)$ stupnjeva slobode, pri čemu vrijedi:

$$S_{X_1}^2 = \frac{1}{n_1 - 1}\sum_{i=1}^{n_1}(X_1^i - \bar{X}_1)^2, \quad S_{X_2}^2 = \frac{1}{n_2 - 1}\sum_{i=1}^{n_2}(X_2^i - \bar{X}_2)^2.$$

Hipoteze testa jednakosti varijanci glase:

$$H_0 : \sigma_1^2 = \sigma_2^2$$
$$H_1 : \sigma_1^2 < \sigma_2^2 \quad , \quad \sigma_1^2 > \sigma_2^2 \quad , \quad \sigma_1^2 \neq \sigma_2^2$$

Ispitajmo jednakost varijanci naših danih uzoraka.

```
var.test(major_songs$danceability, minor_songs$danceability)
```

```
##
##  F test to compare two variances
##
## data:  major_songs$danceability and minor_songs$danceability
## F = 0.93041, num df = 151743, denom df = 80980, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.9192413 0.9416879
## sample estimates:
## ratio of variances
##          0.9304074
```

P-vrijednost od 2.2e-16 nam govori da ćemo odbaciti hipotezu H_0 u korist H_1 i zaključiti da varijance naših dvaju uzoraka nisu jednake te u skladu s time dalje provoditi prikladan test.

Koristimo testnu statistiku

$$T' = \frac{\bar{X}_1 - \bar{X}_2 - (\mu_1 - \mu_2)}{\sqrt{\frac{s_{X_1}^2}{n_1} + \frac{s_{X_2}^2}{n_2}}}$$

koja ima aproksimativnu t-distribuciju sa stupnjevima slobode

$$v = \frac{(s_{X_1}^2/n_1 + s_{X_2}^2/n_2)^2}{(s_{X_1}^2/n_1)^2/(n_1 - 1) + (s_{X_2}^2/n_2)^2/(n_2 - 1)}$$

gdje je

$$s_{X_i}^2 = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (X_i^j - \bar{X}_i)^2$$

za $i = 1, 2$.

Provedimo sada t-test uz pretpostavku nejednakih varijanci.

```
t.test(major_songs$danceability, minor_songs$danceability, alt = "greater", var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  major_songs$danceability and minor_songs$danceability
## t = -29.588, df = 160275, p-value = 1
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
```

```
##  -0.02545718         Inf
## sample estimates:
## mean of x mean of y
## 0.5459727 0.5700892
```

Zbog dobivene p-vrijednosti nećemo odbaciti hipotezu H_0 o jednakosti prosječnih vrijednosti te možemo reći da pjesme u duru nisu "plesnije" od onih pisanih u molu, kao što nam je ranije indicirao "boxplot" dijagram vrijednosti naših dvaju promatranih uzoraka.

Mozemo li temeljem danih varijabli predvidjeti popularnost neke pjesme?

Pogledajmo korelacije popularnosti sa svakom mogućom varijablom. Kao potencijalne kandidate uzmimo 3 varijable s najvećom korelacijom.

```
cor(dataset[c(5,6,7,8,9,10,12,13,15,16,18)])[1,]
```

```
##       popularity     acousticness     danceability      duration_ms
##       1.00000000      -0.38129531       0.25656447       0.00234802
##           energy instrumentalness         liveness         loudness
##       0.24892177      -0.21098311      -0.16799519       0.36301074
##      speechiness            tempo          valence
##      -0.15107582       0.08103891       0.06007629
```

Kandidati su nam acousticness, loudness, danceability.

Kakve karakteristike bi vaša pjesma trebala imati ako želite da ona bude što popularnija?

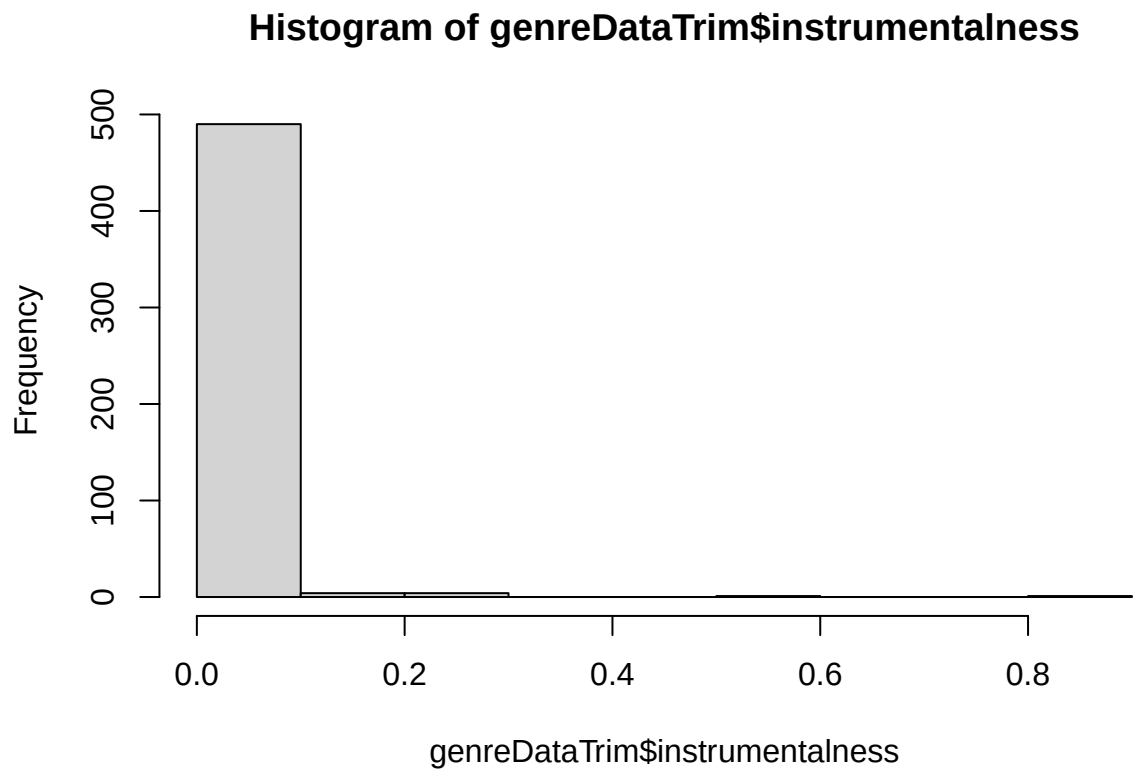
Odaberimo žanr Pop. Za početak pogledajmo kako korelira sa svim varijablama tako da dobijemo par kandidata.

```
genreData <- subset(dataset, ï..genre == "R&B")

genreData = genreData[order(-genreData$popularity),]
genreDataTrim = genreData[0:500,]

cor(genreDataTrim[c(5,6,7,8,9,10,12,13,15,16,18)])[1,]
```

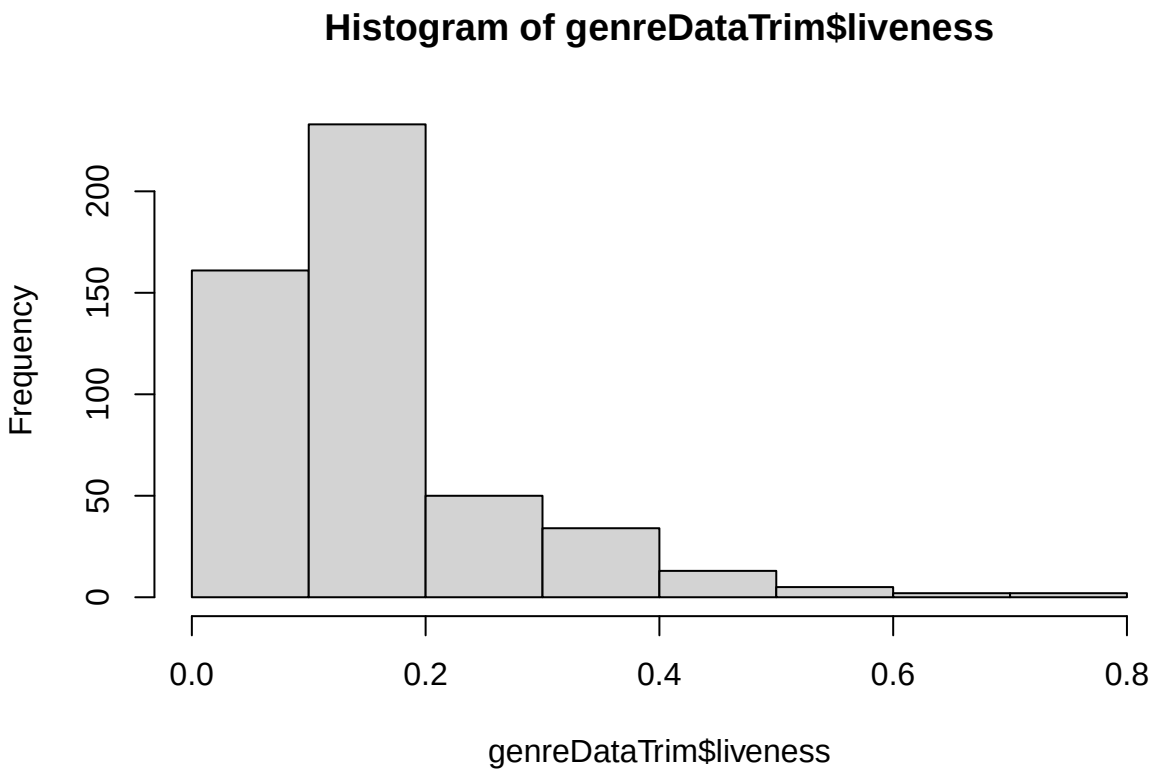
```
##       popularity     acousticness     danceability      duration_ms
##      1.000000000     -0.017206612     -0.005770168     -0.013689281
##           energy instrumentalness         liveness         loudness
##      0.119855496     -0.081299940     -0.079719876      0.145843546
##      speechiness            tempo          valence
##     -0.077066013      0.035628455      0.014793741
```

Vidimo da su 3 najkoreliranije varijable loudness, energy, speechiness. Nećemo koristiti instrumentalness i liveness zbog nepravilne distribucije, što ćemo i pokazati histogramom.

```
hist(genreDataTrim$instrumentalness)
```

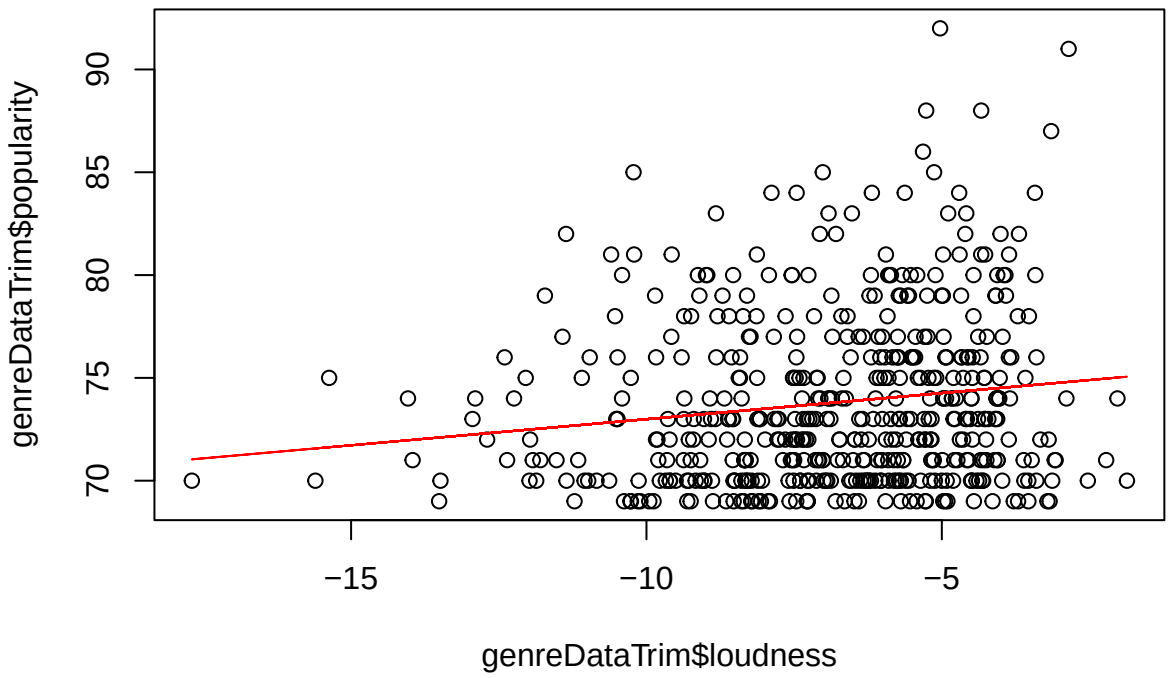



```
hist(genreDataTrim$liveness)
```

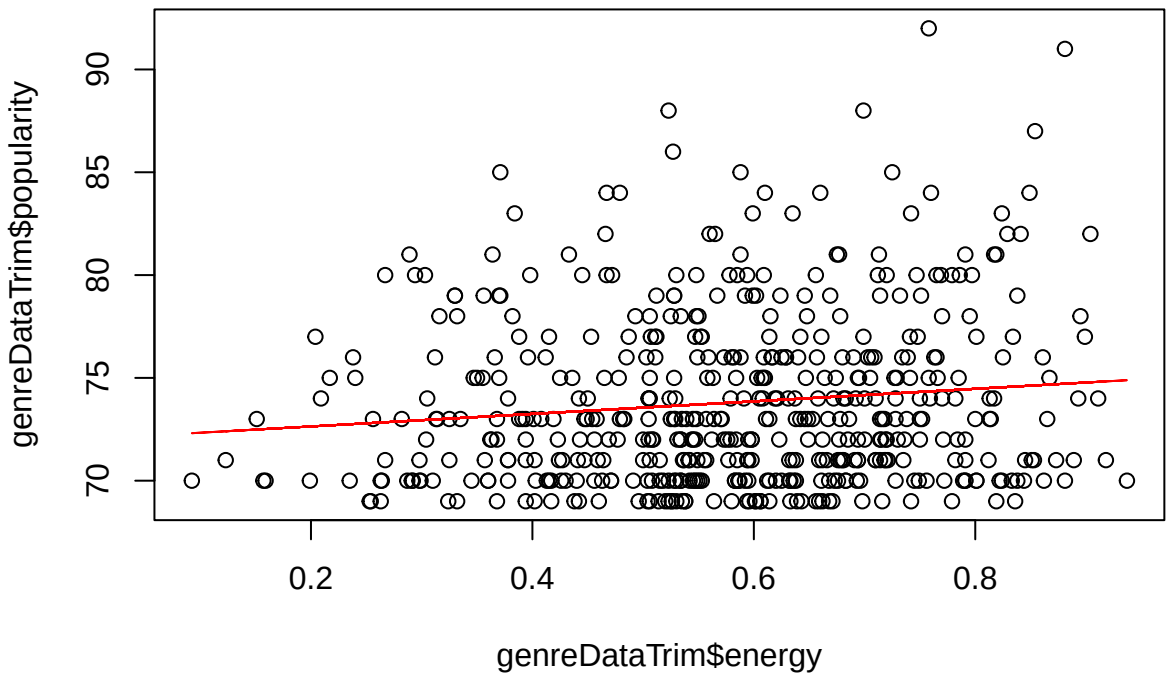


Pokazat ćemo jednostavnu regresiju sa našim kandidatima.

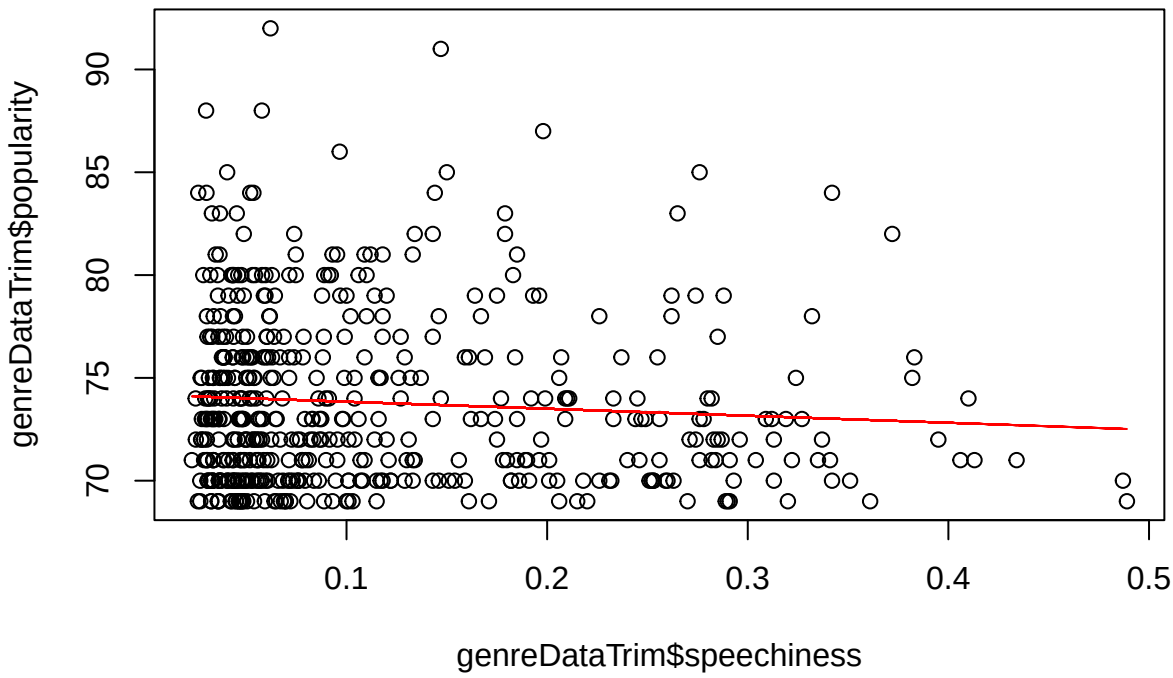
```
fit.loudness <- lm(popularity~loudness, data=genreDataTrim)

plot(genreDataTrim$loudness, genreDataTrim$popularity)
lines(genreDataTrim$loudness, fit.loudness$fitted.values,col='red')
```




```
fit.energy <- lm(popularity~energy, data=genreDataTrim)

plot(genreDataTrim$energy, genreDataTrim$popularity)
lines(genreDataTrim$energy, fit.energy$fitted.values,col='red')
```




```
fit.speechiness <- lm(popularity~speechiness, data=genreDataTrim)

plot(genreDataTrim$speechiness, genreDataTrim$popularity)
lines(genreDataTrim$speechiness, fit.speechiness$fitted.values,col='red')
```



Grafovi pokazuju da će pjesma što je glasnija i energičnija biti i popularnija. Također, što je više "govorljiva" bit će manje popularna. Uočavamo da su svi nagibi pravaca mali što upućuje na blage efekte određenih varijabli na popularnost.

Nakon što smo dobili modele treba provjeriti jesu li narušene pretpostavke modela.

Počnimo sa normalnosti reziduala.

```
require(nortest)
```

```
## Loading required package: nortest
```

```
lillie.test(rstandard(fit.loudness))
```

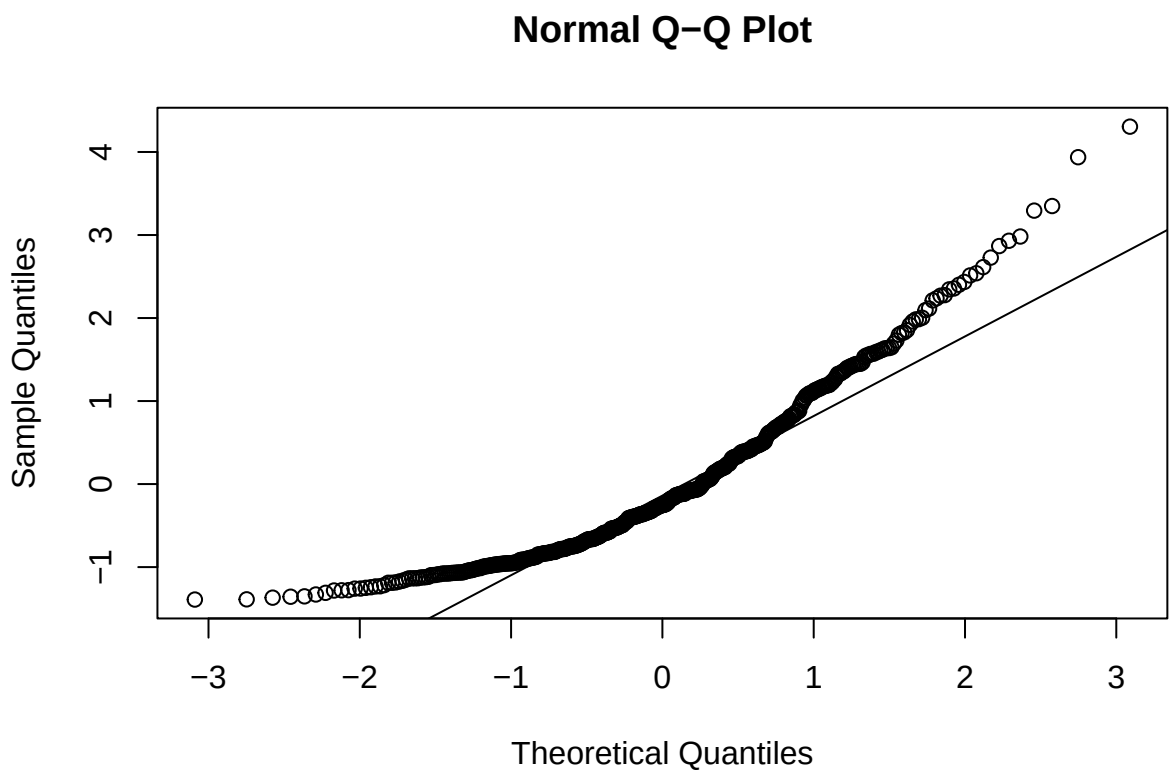
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  rstandard(fit.loudness)
## D = 0.11768, p-value < 2.2e-16
```

```
selected.model = fit.loudness
summary(fit.loudness)
```

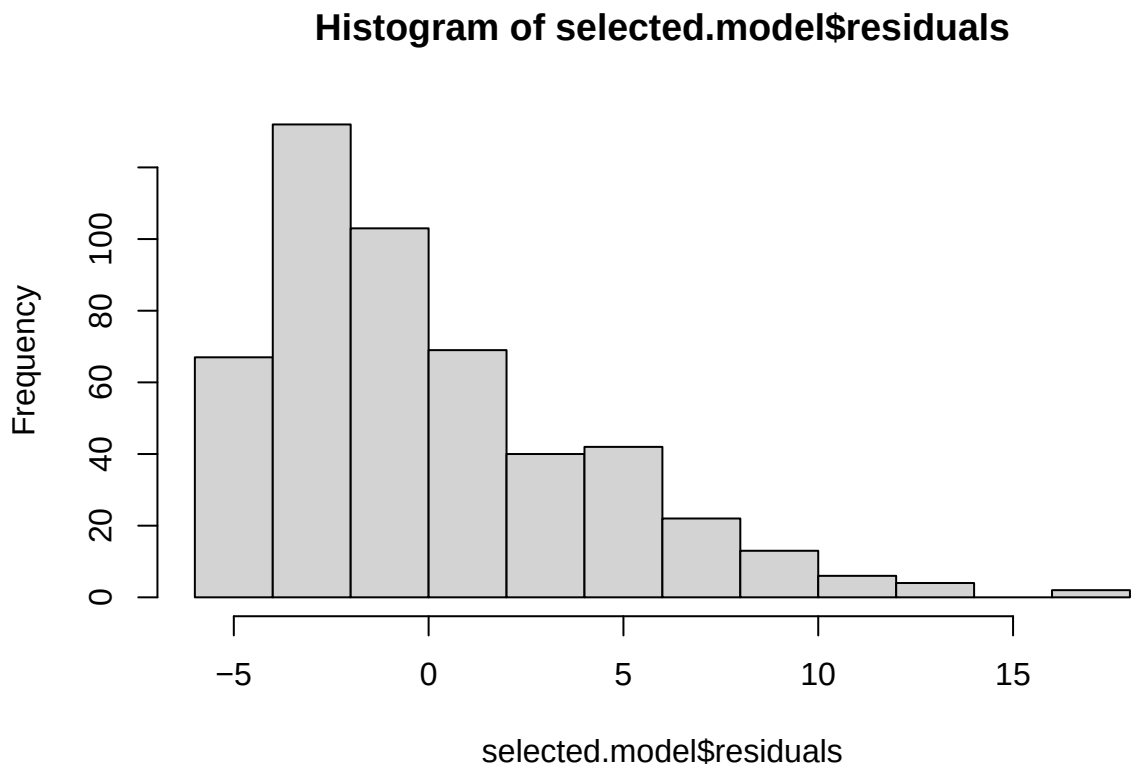
```
##
## Call:
```

```
## lm(formula = popularity ~ loudness, data = genreDataTrim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.723 -3.238 -1.038  2.090 17.747
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 75.52938    0.55888  135.14  < 2e-16 ***
## loudness     0.25385    0.07716    3.29  0.00107 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.129 on 498 degrees of freedom
## Multiple R-squared:  0.02127,    Adjusted R-squared:  0.01931
## F-statistic: 10.82 on 1 and 498 DF,  p-value: 0.001074
```

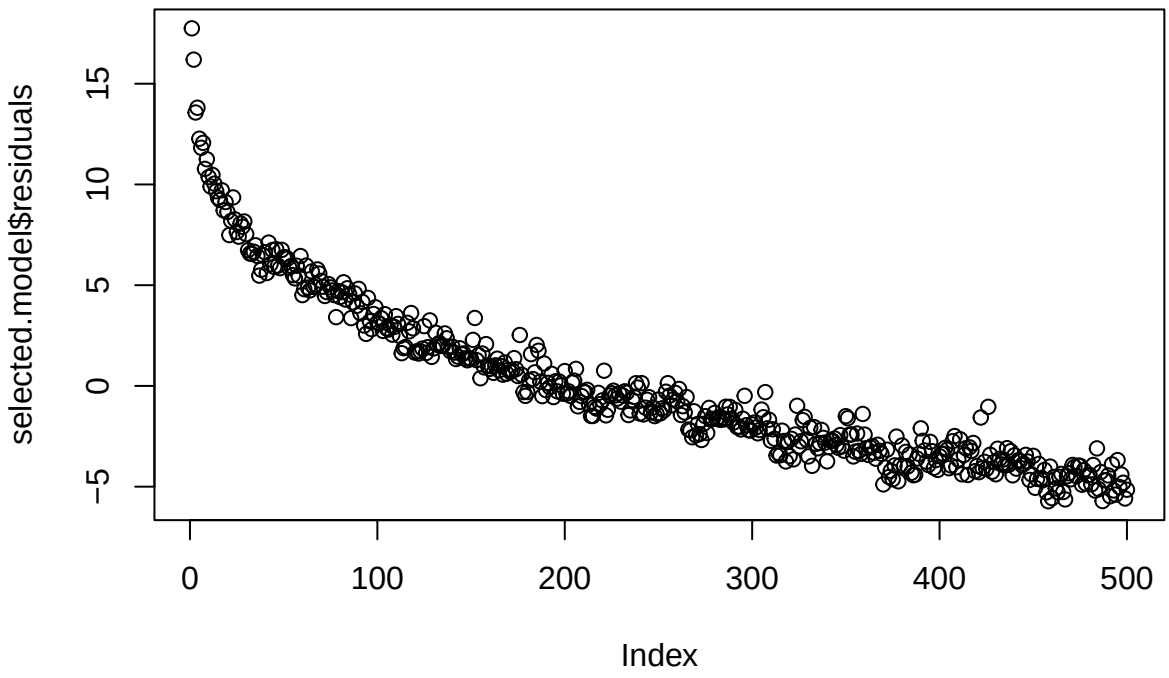
```
qqnorm(rstandard(selected.model))
qqline(rstandard(selected.model))
```



```
hist(selected.model$residuals)
```




```
plot(selected.model$residuals)
```




```
lillie.test(rstandard(fit.energy))
```

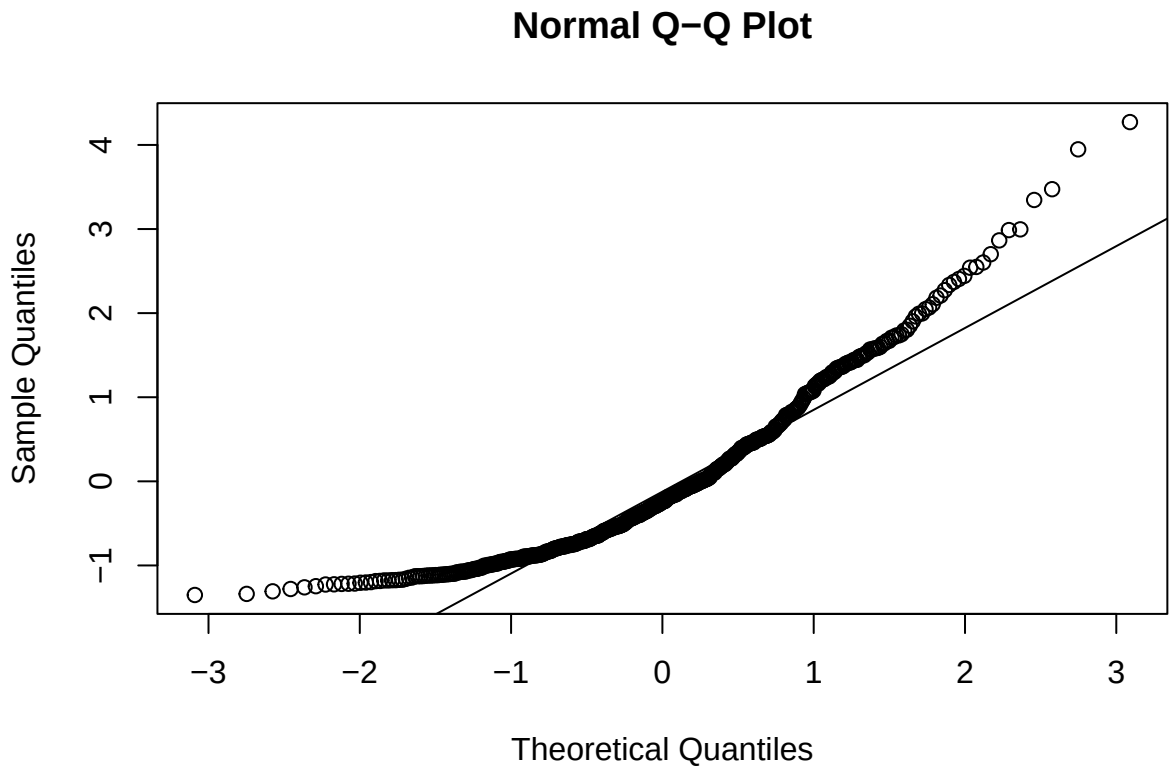
```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  rstandard(fit.energy)
## D = 0.10587, p-value = 1.206e-14
```

```
selected.model = fit.energy
summary(fit.energy)
```

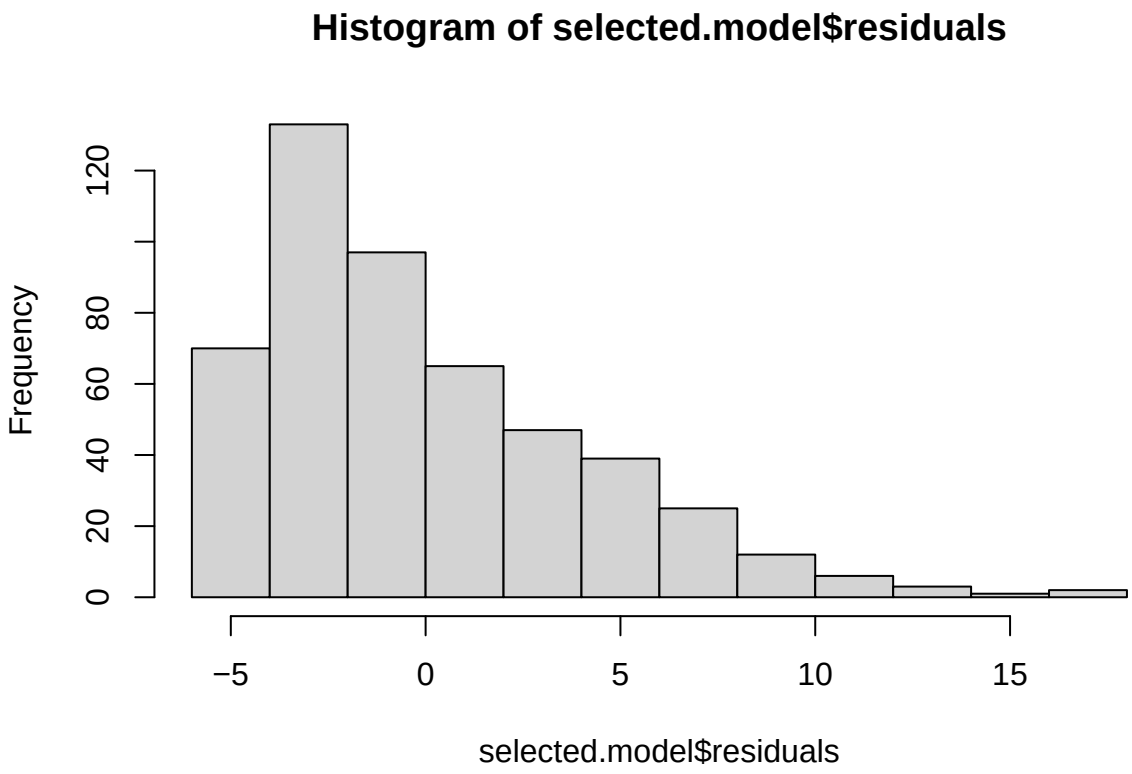
```
##
## Call:
## lm(formula = popularity ~ energy, data = genreDataTrim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.579 -3.223 -1.004  2.208 17.659
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  72.0276     0.6813 105.715   <2e-16 ***
## energy        3.0518     1.1328   2.694   0.0073 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 4.143 on 498 degrees of freedom
## Multiple R-squared:  0.01437,    Adjusted R-squared:  0.01239
## F-statistic: 7.258 on 1 and 498 DF,  p-value: 0.007296
```

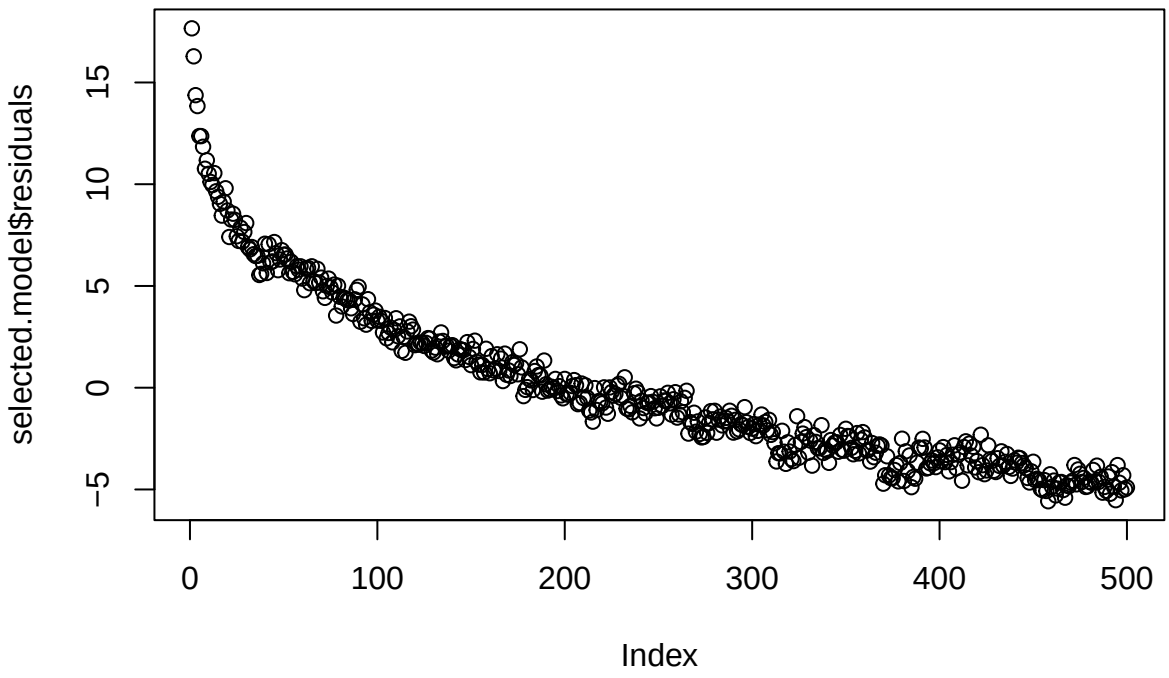
```
qqnorm(rstandard(selected.model))
qqline(rstandard(selected.model))
```




```
hist(selected.model$residuals)
```



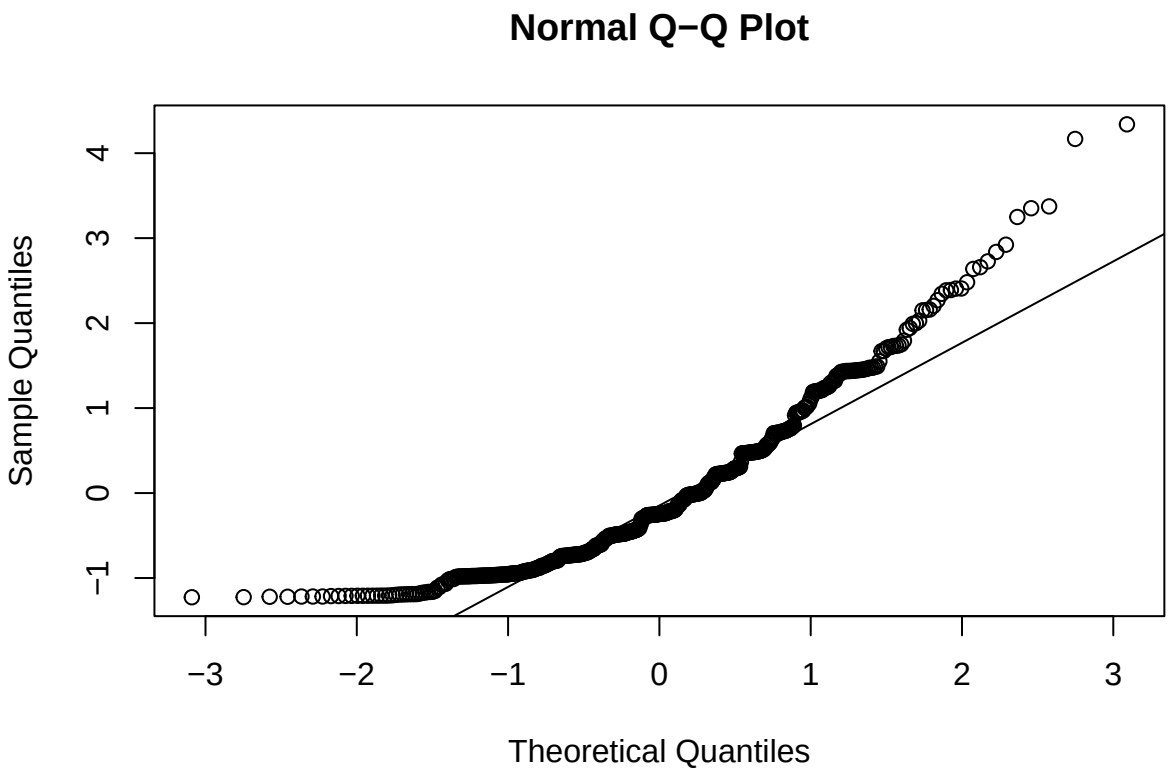

```
plot(selected.model$residuals)
```



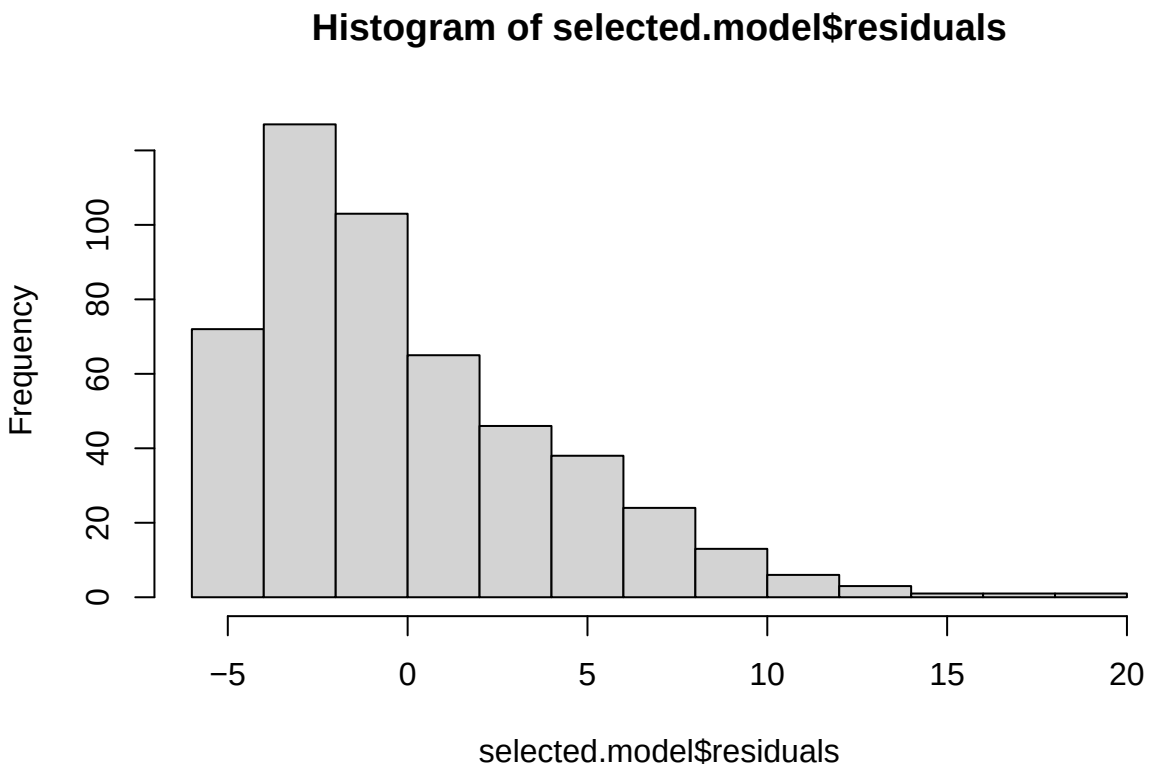

```
selected.model = fit.speechiness
summary(fit.speechiness)
```

```
##
## Call:
## lm(formula = popularity ~ speechiness, data = genreDataTrim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.092 -3.287 -1.021  2.075 18.031
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  74.1804     0.2912 254.706   <2e-16 ***
## speechiness  -3.4086     1.9761  -1.725   0.0852 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.161 on 498 degrees of freedom
## Multiple R-squared:  0.005939,   Adjusted R-squared:  0.003943
## F-statistic: 2.975 on 1 and 498 DF,  p-value: 0.08516
```

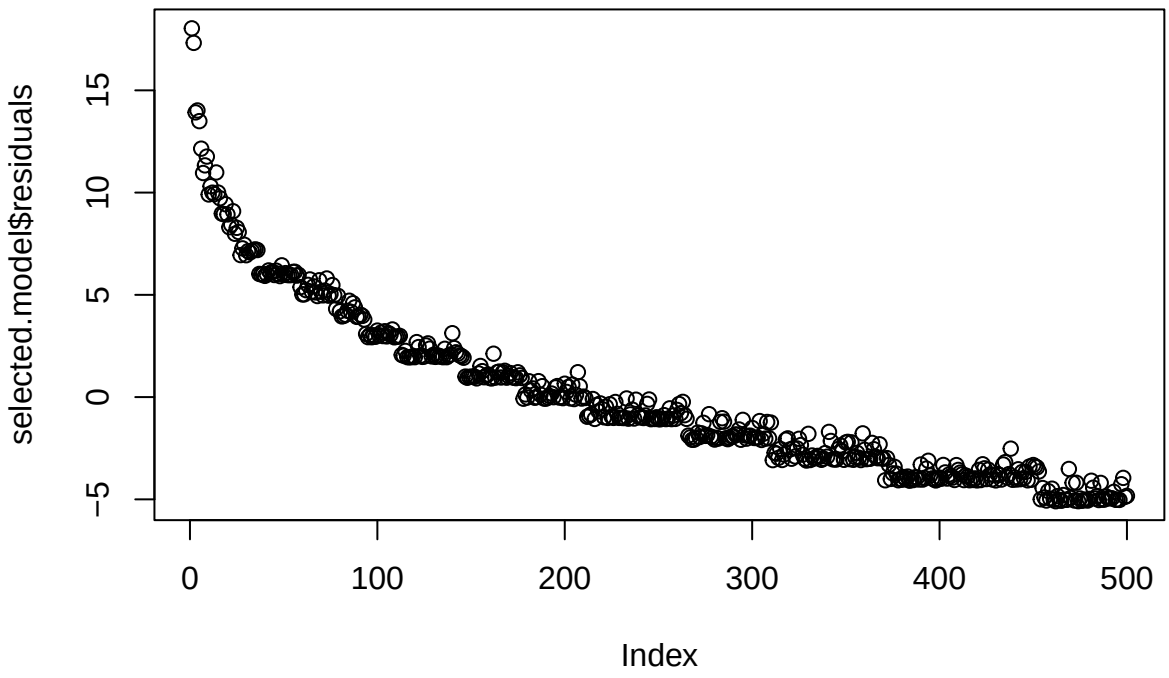
```
qqnorm(rstandard(selected.model))
qqline(rstandard(selected.model))
```




```
hist(selected.model$residuals)
```




```
plot(selected.model$residuals)
```




```
lillie.test(rstandard(fit.speechiness))
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  rstandard(fit.speechiness)
## D = 0.12013, p-value < 2.2e-16
```

Histogrami i qq-plotova nam ukazuju na "slabe" lijeve repove. P-vrijednost je jako mala za svaki test, čak i uz korištenje preporučene Lillieforsove inačice. Stoga, ne možemo donositi zaključke iz ovih regresijskih modela.

Višestruka regresija

Prije nego procjenimo model višestruke regresije moramo provjeriti da pojedini parovi varijabli nisu (previše) korelirani.

```
cor(cbind(genreDataTrim$loudness, genreDataTrim$energy, genreDataTrim$speechiness)) # korelacijski koef
```

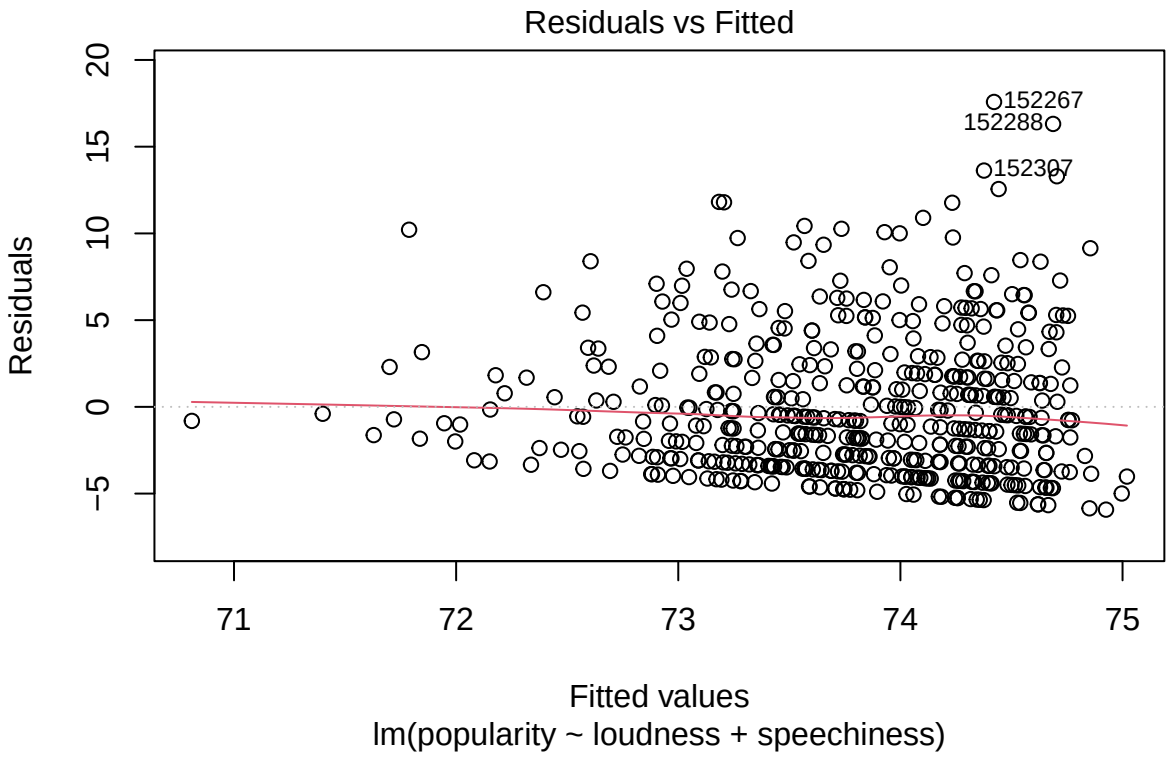
```
##             [,1]      [,2]        [,3]
## [1,]  1.00000000 0.7351149 -0.01080719
## [2,]  0.73511486 1.0000000  0.11998491
## [3,] -0.01080719 0.1199849  1.00000000
```

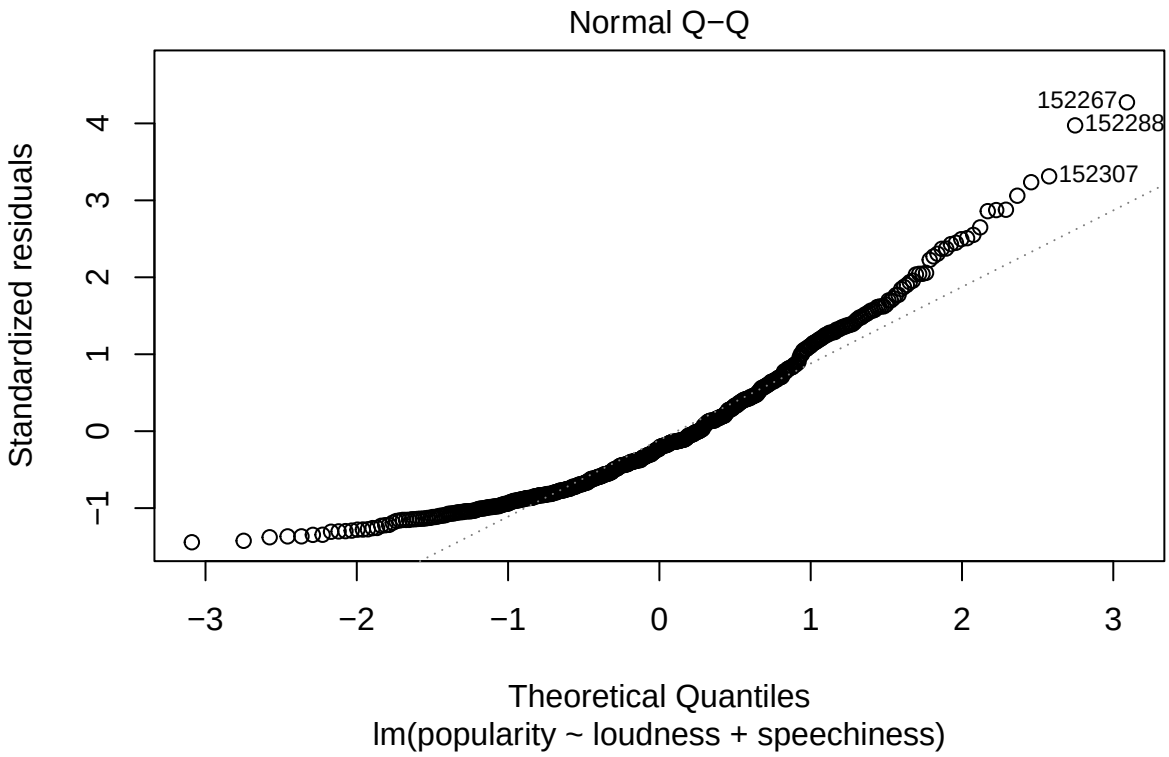
Pošto su loudness i energy jako korelirane regresija s njima će nam dati neke rezultate, ali na temelju njih ne bi smjeli donositi zaključke. Stoga ćemo probati oba slučaja, odnosno kombinacije loudness + speechiness i energy + speechiness.

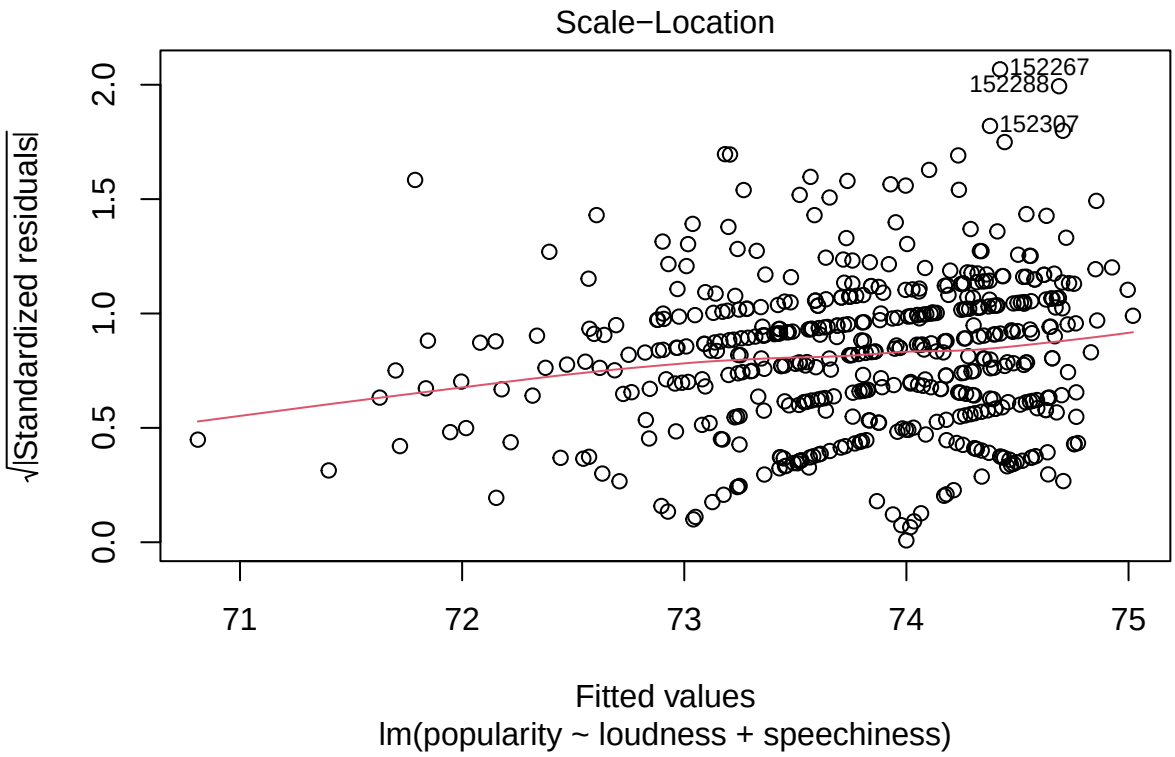
```
fit.multi = lm(popularity ~ loudness + speechiness, genreDataTrim)
summary(fit.multi)
```

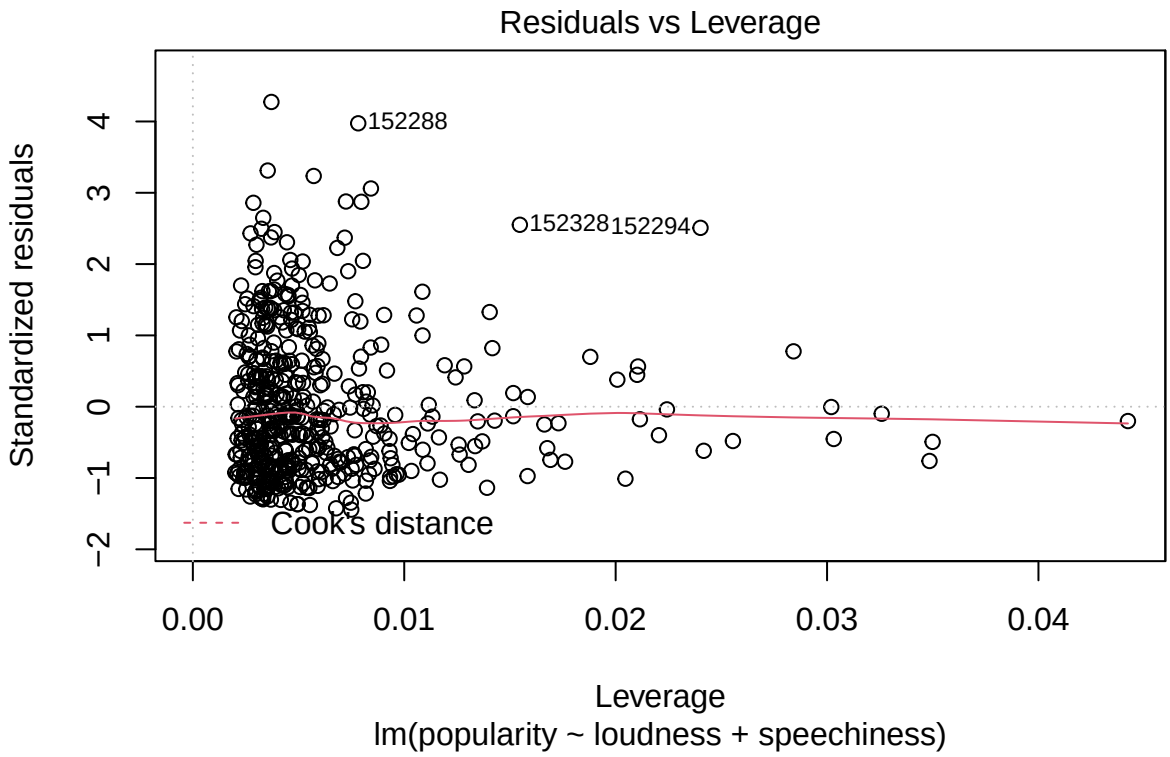
```
##
## Call:
## lm(formula = popularity ~ loudness + speechiness, data = genreDataTrim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.926 -3.231 -0.894  2.279 17.579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 75.89826    0.59825 126.868  < 2e-16 ***
## loudness     0.25243    0.07702   3.277  0.00112 **
## speechiness -3.33927    1.95715  -1.706  0.08860 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.121 on 497 degrees of freedom
## Multiple R-squared:  0.02697,    Adjusted R-squared:  0.02305
## F-statistic: 6.888 on 2 and 497 DF,  p-value: 0.00112
```

```
plot(fit.multi)
```


Residuals vs Fitted
Residuals
20
15
10
5
0
−5
152267
152288
152307
71
72
73
74
75
Fitted values
lm(popularity ~ loudness + speechiness)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(popularity ~ loudness + speechiness)
152267
152288
152307
−3
−2
−1
0
1
2
3
−1
0
1
2
3
4

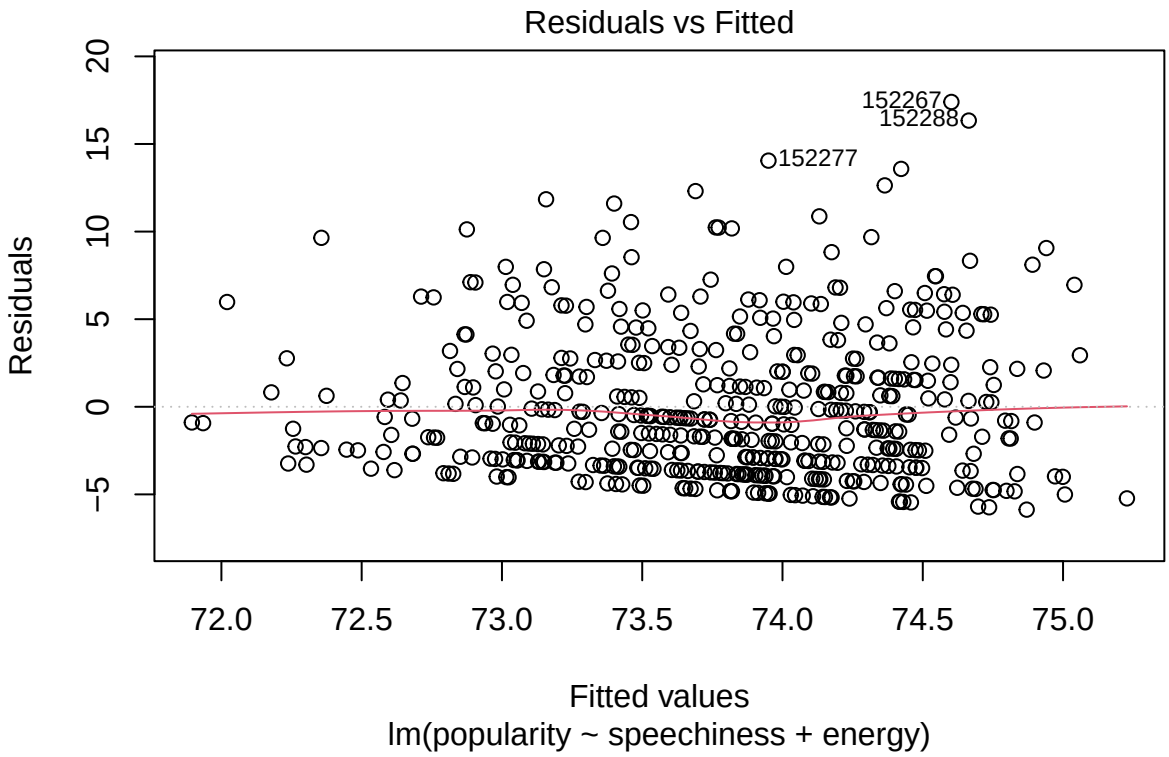

Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
152267
152288
152307
71
72
73
74
75
Fitted values
lm(popularity ~ loudness + speechiness)

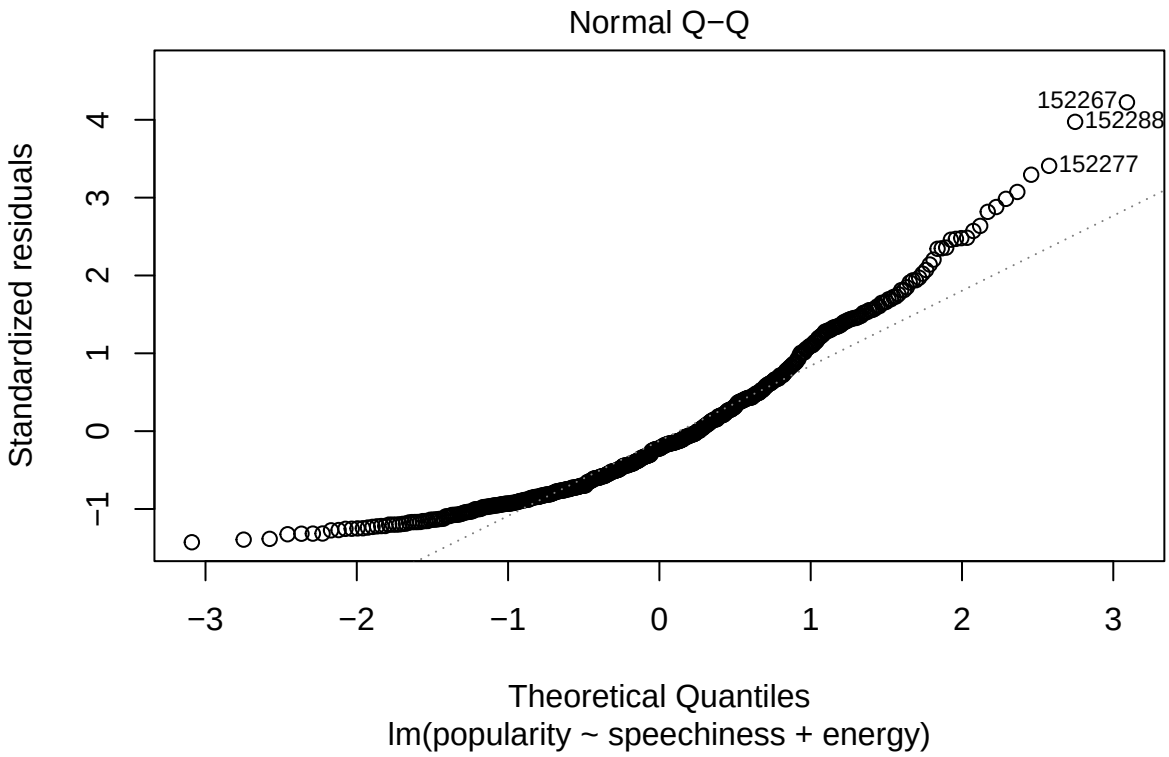


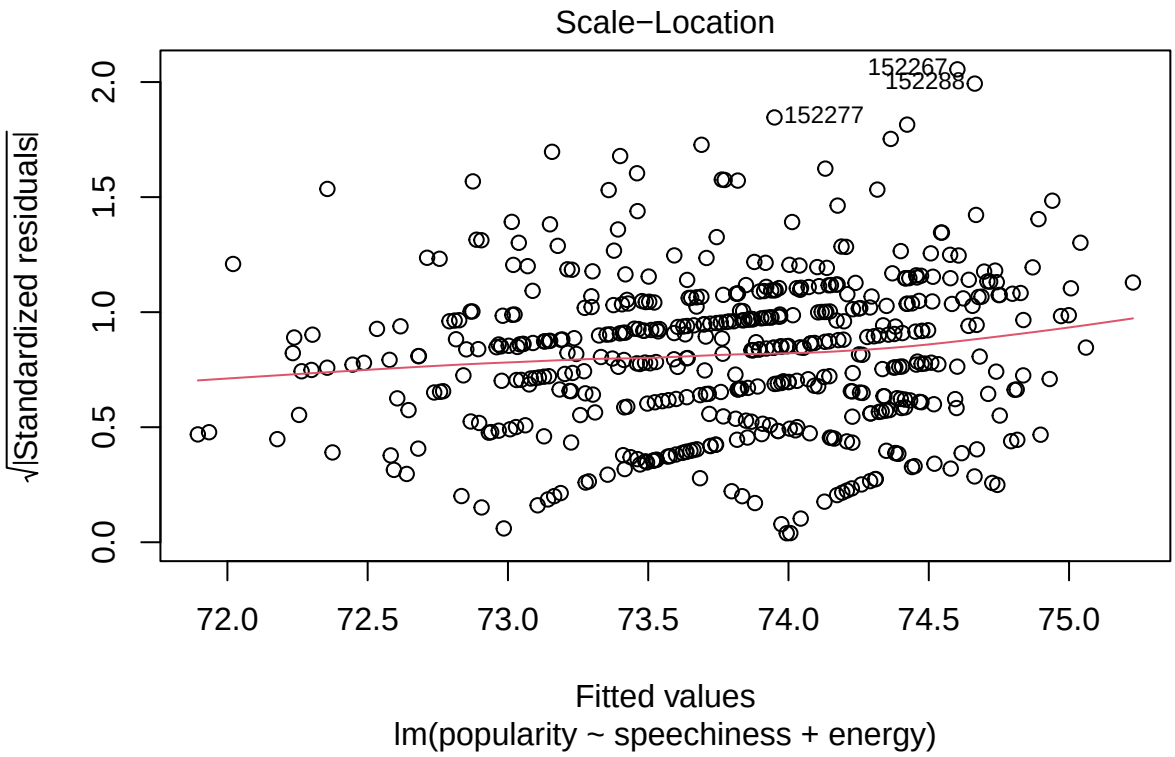

```
fit.multi = lm(popularity ~ speechiness + energy, genreDataTrim)
summary(fit.multi)
```

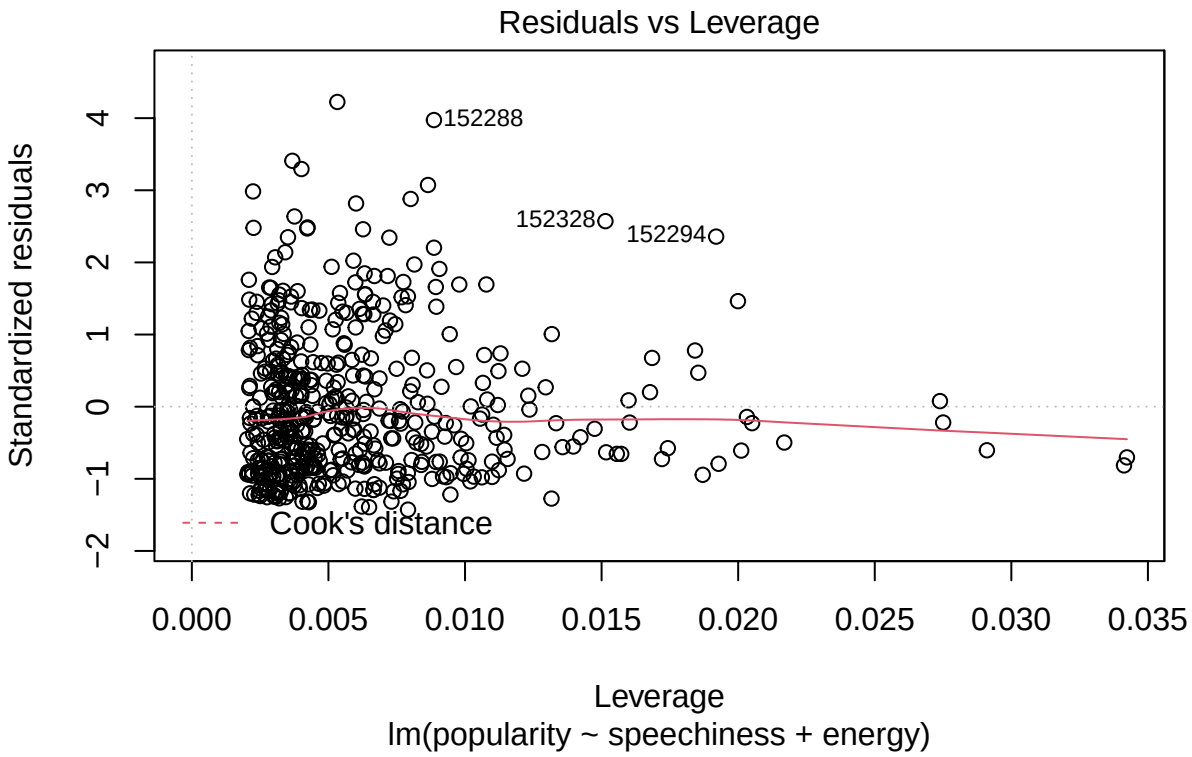
```
##
## Call:
## lm(formula = popularity ~ speechiness + energy, data = genreDataTrim)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8703 -3.1898 -0.9016  2.1600 17.3980
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  72.3288     0.6944 104.162  < 2e-16 ***
## speechiness  -4.1037     1.9754  -2.077  0.03828 *
## energy        3.3352     1.1372   2.933  0.00351 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.13 on 497 degrees of freedom
## Multiple R-squared:  0.02285,    Adjusted R-squared:  0.01892
## F-statistic: 5.811 on 2 and 497 DF,  p-value: 0.003202
```

```
plot(fit.multi)
```


Residuals vs Fitted
Residuals
Fitted values
lm(popularity ~ speechiness + energy)
152267
152288
152277
20
15
10
5
0
−5
72.0
72.5
73.0
73.5
74.0
74.5
75.0


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(popularity ~ speechiness + energy)
152267
152288
152277
−3
−2
−1
0
1
2
3
−1
0
1
2
3
4


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
152267
152288
152277
72.0
72.5
73.0
73.5
74.0
74.5
75.0
Fitted values
lm(popularity ~ speechiness + energy)



Vidimo da model kombinacije loudness + speechiness bolji jer dobijemo R^2 0.02305, što je ujedno i najbolji model jer ima najveći R^2.

Pošto smo vidjeli da nijedna varijabla niti kombinacija varijabli ne utječe pretjerano na popularnost, pogledat ćemo kako utječu imena izvođača.

Iz dataseta smo stvorili podskup po umjetnicima koji sadrži prosječnu popularnost njihovih pjesama. Iz njega smo eliminirali umjetnike sa manje od 20 pjesama.

```
library(tidyverse)

options(warn = -1)

genreData %>% group_by(name = artist_name) %>% summarise(
    avgPopularity = mean(popularity),
    songCount = nrow(genreData[which((genreData$artist_name) == name),])
) -> pop

pop = pop[which(pop$songCount > 20),]
pop = pop[order(-pop$avgPopularity),]

top10Artist = pop[1:10,]
top10Artist
```

```
## # A tibble: 10 x 3
##    name              avgPopularity songCount
##    <chr>                     <dbl>     <int>
```

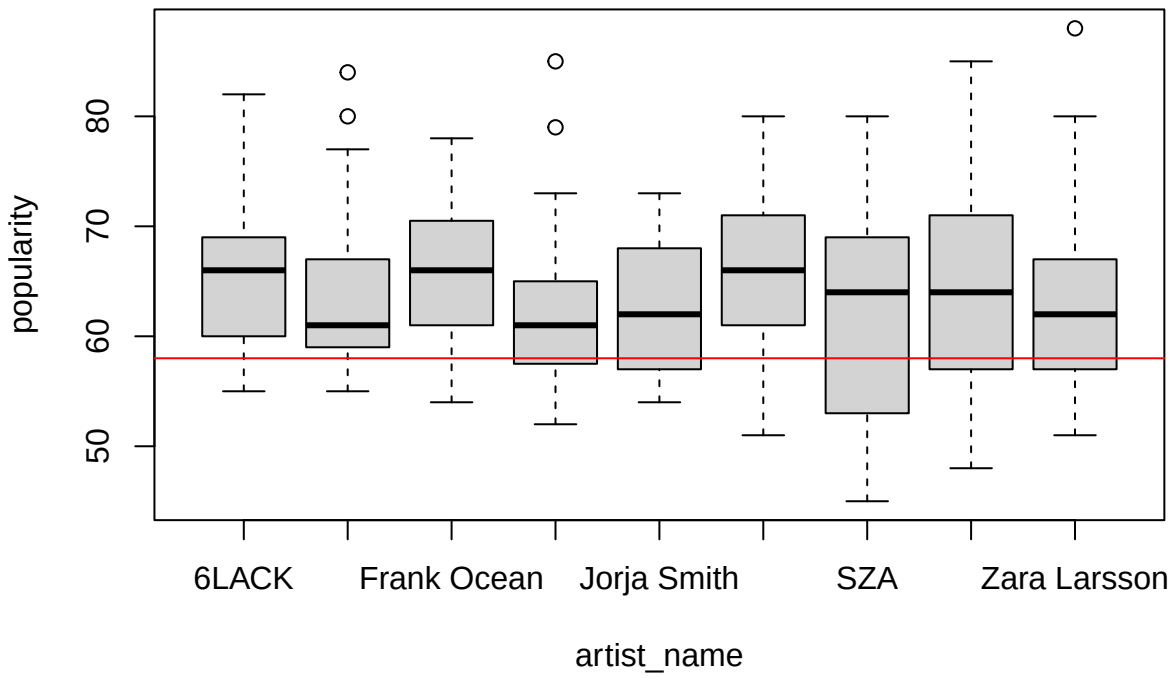
```
##  1 Joji                     66.3        24
##  2 Frank Ocean              66.0        35
##  3 Rex Orange County        65.9        24
##  4 6LACK                    65.1        34
##  5 The Weeknd               64.4        87
##  6 Zara Larsson             64.3        21
##  7 Ella Mai                 64.2        33
##  8 Jorja Smith              62.9        29
##  9 SZA                      62.3        25
## 10 H.E.R.                   62.1        39
```

Pogledajmo barplot 10 najpopularnijih umjetnika gdje ćemo ujedno prikazati granicu trećeg kvantila.

```
thirdQuantile = quantile(genreData$popularity, c(0.75))
thirdQuantile
```

```
## 75%
##  58
```

```
newData = genreData[which(genreData$artist_name == top10Artist[i,]$name),]

for (i in 2:10) {
  newData <- rbind(newData, (genreData[which(genreData$artist_name == top10Artist[i,]$name),]))
}

boxplot(popularity ~ artist_name, data=newData)
abline(h=58, col="red")
```



Uočavamo da su svi umjetnici daleko iznad granice te zaključujemo da popularnost pjesme pak najviše ovisi o samom umjetniku.