CS555 Final Project

Tommy Lee

12/9/2021

Data from Kaggle that originated from Spotify: https://www.kaggle.com/vicsuperman/prediction-of-music-genre

Background: This Kaggle dataset was originally extracted from Spotify's API and originally contained about 50,000 rows. For my research project, I wanted to analyze three specific music genres: anime(japanese animation, normally sung by j-pop stars), electronic(short for electronic dance music) and hip-hop. Overall, I am interested to see the differences between these music genres (specifically how anime music genre is doing) and can Spotify songs be modeled based on their audio features or characteristics.

Below are the specific research questions we are interested in:

1. Do any of the audio features have a relationship with the popularity of a song? (I will be using linear regression to check if there is a linear relationship.)
2. Is there a popularity difference between music genres? (I will be using ANOVA to check the mean popularity difference for the three groups.)
3. Can we correctly predict whether a Spotify song is an anime (music genre) song? (I will be using logistic regression to build a classifier.)

Below are the different variables we will be working with:

- music_genre: "Anime", "Hip-Hop" or "Electronic"
- popularity: how popular a song is on a scale of 0 to 100
- danceability: how likely you would dance to this song on scale of 0 to 1
- duration_ms: song duration in milliseconds
- tempo: beats per minute
- valence: The higher the value, the more positive mood for the song on a scale of 0 to 1
- energy: The energy of a song - the higher the value, the more energetic song on a scale of 0 to 1
- speechiness: the higher the value, the more lyrics the song has on a scale of 0 to 1

Clean Dataset

```r
library(dplyr)
set.seed(1)
# Read data
df <- read.csv('C:/Users/Tommy Lee/Desktop/CS 555/CS555_Final_Project/music_genre.csv')

# Only interested in 3 genre
df_interest <- subset(df, subset = music_genre %in% c('Anime','Hip-Hop','Electronic'))

# Sample only 1000
```

```r
df_interest_small <- df_interest %>%  sample_n(size = 1000)


# Select only the columns I am interested in
df_interest_cleaned <- df_interest_small[c('music_genre','popularity','danceability',
                        'duration_ms','tempo','valence','energy','speechiness')]

# Convert tempo column to double
df_interest_cleaned['tempo'] <- as.double(df_interest_cleaned$tempo)
```

```
## Warning: NAs introduced by coercion
```

```r
# Remove -1 in duration_ms column (invalid values)
df_interest_cleaned <- subset(df_interest_cleaned, subset = duration_ms != -1)

# Remove NA values
df_interest_cleaned <- na.omit(df_interest_cleaned)


# Sample of cleaned dataset
head(df_interest_cleaned,3)
```

```
##   music_genre popularity danceability duration_ms   tempo valence energy
## 1  Electronic         29        0.901      233144 140.057   0.669  0.602
## 2       Anime         28        0.572      309627 151.490   0.635  0.800
## 3  Electronic         61        0.677      247200 123.534   0.697  0.484
##   speechiness
## 1      0.3430
## 2      0.0540
## 3      0.0324
```

After reading in the csv file, I filtered only the three genres I was interested in (Anime,Hip-Hop,Electronic). I then took a sample of 1000 from that filtered dataframe. Then, I subsetted certain columns that would be useful in my analysis. I converted the tempo column to a double data type so I can use it in modeling versus as a factor.Lastly, I removed invalid values such as null and -1 from the duration_ms column (duration should be positive).

Linear Regression Model

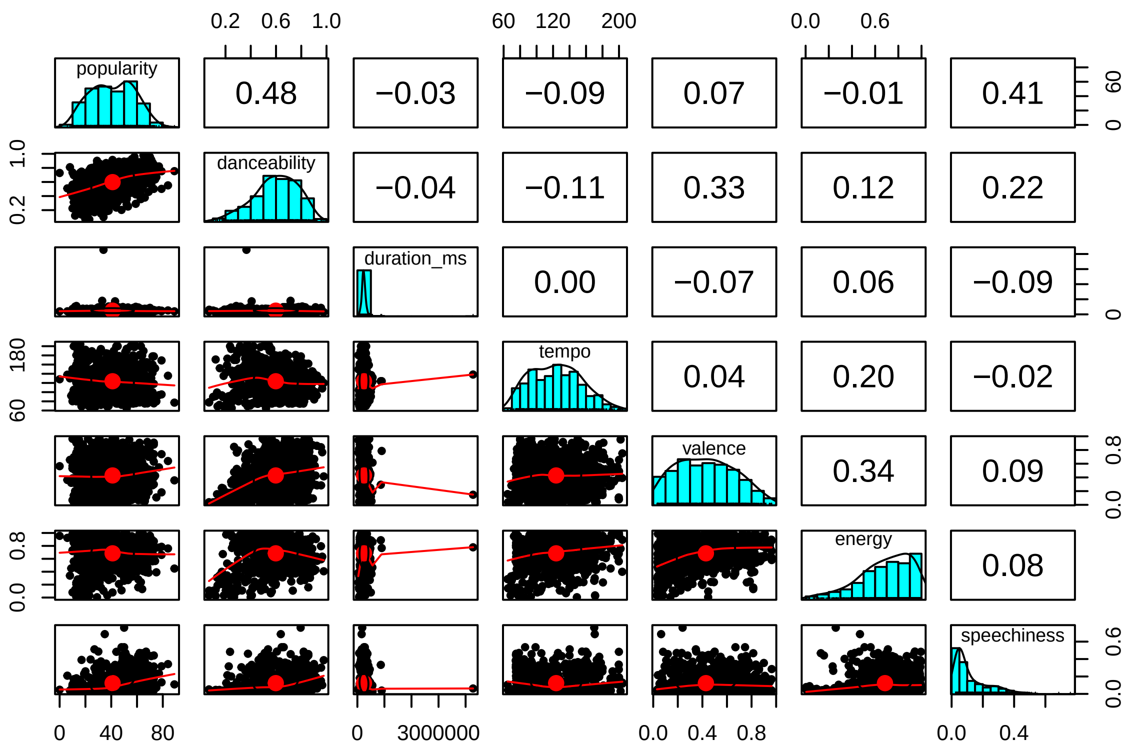
(1) Do any of the audio features have a relationship with the popularity of a song?

```
library(psych)
library(dplyr)
# Count per music genre
knitr::kable(df_interest_cleaned %>% count(music_genre) , digits = 2,
             caption = "Counts Per Music Genre")
```

Table 1: Counts Per Music Genre

music_genre	n
Anime	268
Electronic	247
Hip-Hop	289

```
# Correlation matrix and scatter plot combo
pairs.panels(df_interest_cleaned[c('popularity','danceability','duration_ms',
                                   'tempo','valence','energy','speechiness')])
```



Based on the pairwise plot above, a multiple linear regression model would not be ideal considering that almost all the variables have a very weak association with popularity.Speechiness variable has moderatly positive association, but it violates the linearity assumption. However, the danceability variable has a moderate positive association with popularity (0.48) and shows somewhat a linear relationship meaning we can probably create a simple linear regression model using danceability as the explanatory variable and popularity as the response variable.

Let us go ahead and create the simple linear regression model for popularity and danceability.

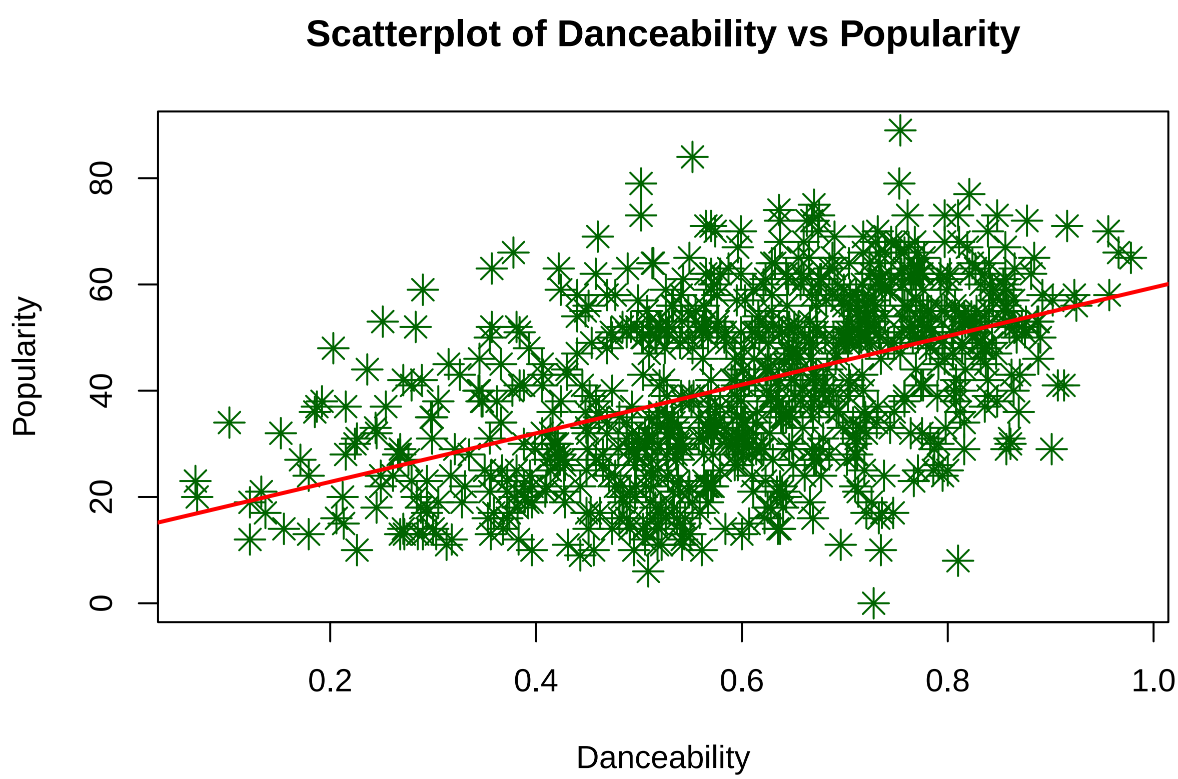
```
# Linear regression model
m <- lm(df_interest_cleaned$popularity~df_interest_cleaned$danceability)
cat('The linear regression model is y = ',round(m$coefficients[1],2),' + ',
    round(m$coefficients[2],2),'\U03B2\u2081\n',sep = '')
```

```
## The linear regression model is y = 13.7 + 45.72ß1
```

```
cat('\U03B2\u2080 =',round(m$coefficients[2],2),'is the slope and it means for every
0.1 unit increase in danceability,there is a 4.572 increase popularity for a Spotifty song.')
```

```
## ß0 = 45.72 is the slope and it means for every
## 0.1 unit increase in danceability,there is a 4.572 increase popularity for a Spotifty song.
```

```
# Scatterplot w/ regression line
plot(df_interest_cleaned$popularity~df_interest_cleaned$danceability,
     pch = 8,cex = 1.5, col = 'dark green',
     xlab = 'Danceability',ylab = 'Popularity',
     main = 'Scatterplot of Danceability vs Popularity')
abline(m,col = 'red',lwd = '2')
```



Now that we have our simple linear regression model, lets make sure that this relationship was not by chance and do some hypothesis testing.

```r
# ANOVA table
anova(m)
```

```
## Analysis of Variance Table
##
## Response: df_interest_cleaned$popularity
##                                  Df Sum Sq Mean Sq F value
## df_interest_cleaned$danceability   1  51074   51074  242.96
## Residuals                        802 168594     210
##                                                 Pr(>F)
## df_interest_cleaned$danceability < 0.00000000000000022 ***
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Let the danceability variable be known as β_1)

Step 1:

- H0: β_1=0 (there is no linear association between danceability and popularity);
- H1: $\beta_1 \neq 0$ (there is a linear association between danceability and popularity);
- $\alpha = 0.05$;

Step 2: Use F test with df of 1 and 802 & p-value

Step 3: Decision Rule: We reject the null hypothesis if F $\geq$ 3.853 or p $<$ 0.05.

```r
paste('This is the F-value we are comparing to',round(qf(0.05,1,802,lower.tail = FALSE),3))
```

```
## [1] "This is the F-value we are comparing to 3.853"
```

Step 4: Calculating F-stats and p-value (values grabbed from ANOVA table)

```r
sum_m <- summary(m)
sum_m
```

```
##
## Call:
## lm(formula = df_interest_cleaned$popularity ~ df_interest_cleaned$danceability)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.984 -10.243   0.013  10.734  45.063
##
## Coefficients:
##                                  Estimate Std. Error t value
## (Intercept)                        13.699      1.828   7.496
## df_interest_cleaned$danceability   45.721      2.933  15.587
##                                              Pr(>|t|)
```

```
## (Intercept)                          0.000000000000174 ***
## df_interest_cleaned$danceability < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.5 on 802 degrees of freedom
## Multiple R-squared:  0.2325, Adjusted R-squared:  0.2315
## F-statistic:   243 on 1 and 802 DF,  p-value: < 0.00000000000000022
```

```r
f_stat <- sum_m$fstatistic[1]
paste('The F-statistic is',f_stat)
```

```
## [1] "The F-statistic is 242.957146877643"
```

```r
anova_m <- anova(m)
paste('The p-value for \U03B2\u2081 is',anova_m$`Pr(>F)`[1])
```

```
## [1] "The p-value for ß1 is 0.0000000000000000000000000000000000000000000000047881489238435"
```

```r
# R squared
paste('This is the R squared:',round(sum_m$r.squared,3))
```

```
## [1] "This is the R squared: 0.233"
```

```r
# 95% confidence interval for beta1
confid <- confint(m, level = 0.95)
paste('The 95% confidence interval for \U03B2\u2081 is [',round(confid[2,1],3),
      ',',round(confid[2,2],3),']')
```

```
## [1] "The 95% confidence interval for ß1 is [ 39.964 , 51.479 ]"
```

Step 5: Conclusion We reject the null hypothesis since our F-stat $242.96 \geq 3.853$. Also, the p-value is almost zero backing up our conclusion of rejecting the null hypothesis. Therefore, we have significant evidence at α level 0.05 that $\beta_1 \neq 0$. This proves that there is a linear relationship between danceability and popularity for Spotify songs.

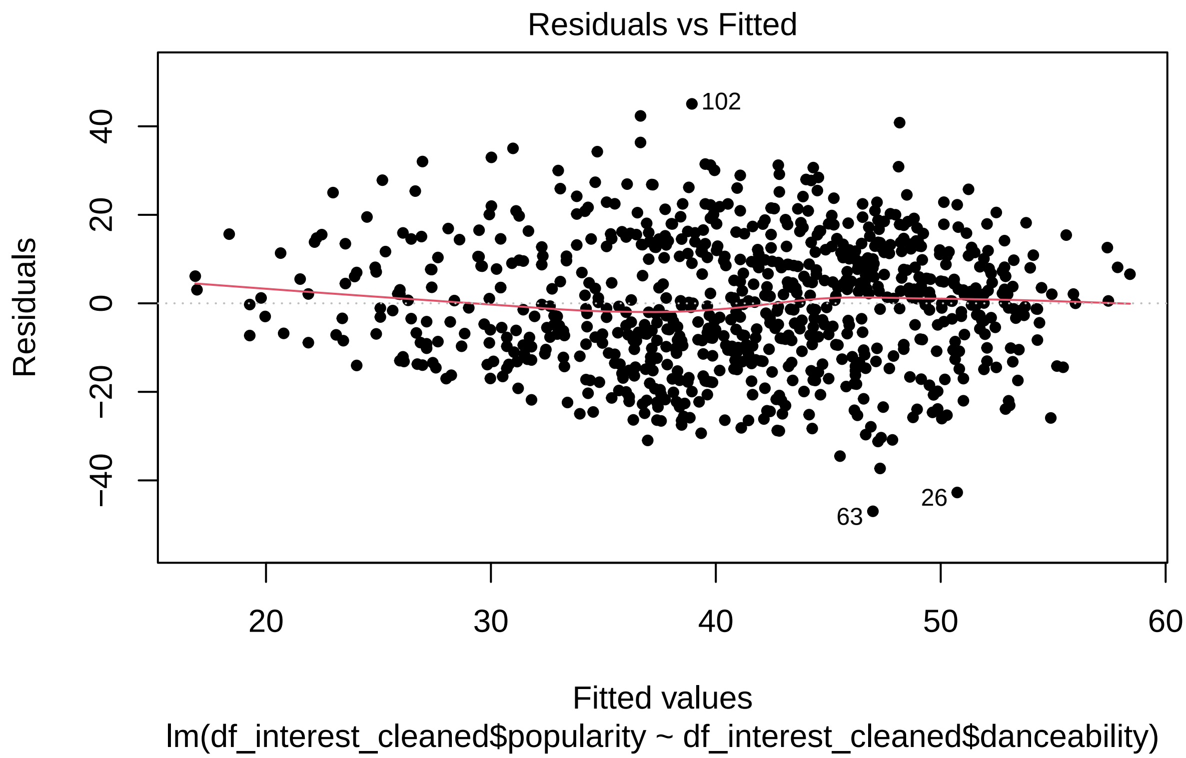
β_1 is interpretted as for every one unit increase in danceability, there is a 45.721 increase in popularity. Let us interpret that with smaller units since popularity is on a scale of 100: for every 0.1 unit in danceability, there is a 4.5721 increase in popularity.

Calculating R squared equals 0.233 shows that the proportion of the variance for the popularity of a Spotify is weakly explained by its danceability.

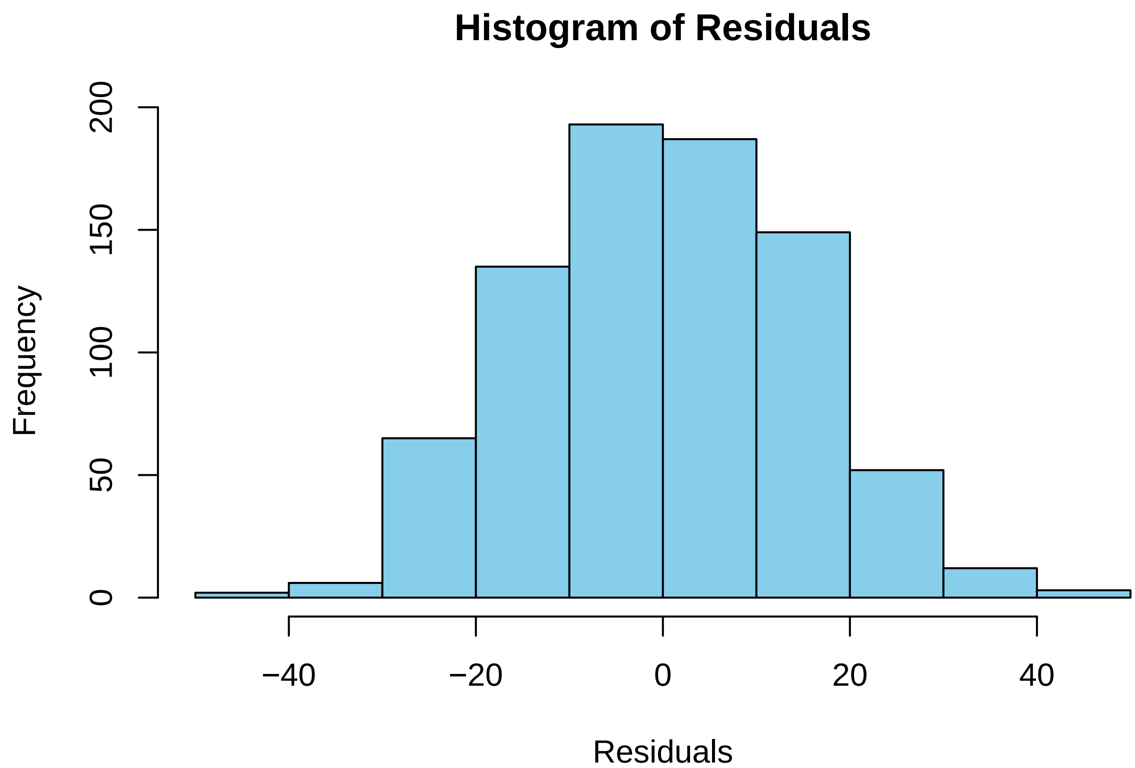
Calculating a 95% confidence interval, We are 95% confident that the true β_1 (danceability) is between [39.964,51.479].

Lets check if our residuals hold up on the assumptions for a linear regression model.

```r
# Residual plot
plot(m,which = 1,pch = 20)
```



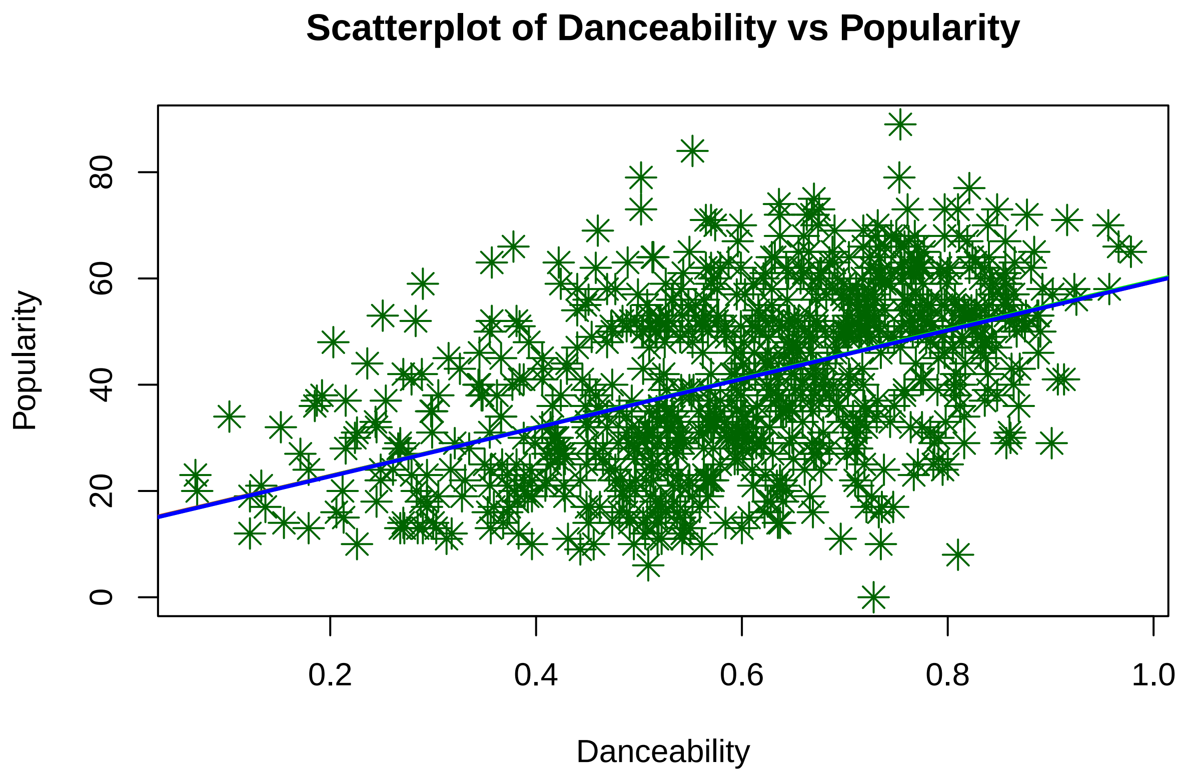

```r
hist(resid(m),col = 'sky blue',main = 'Histogram of Residuals',xlab = 'Residuals' )
```



We also need to test whether removing any outliers will help increase R squared.

```
# Removal of each outlier for later to plot
remove_df1 <- df_interest_cleaned[c(-26),]
remove_df2 <- df_interest_cleaned[c(-63),]
remove_df3 <- df_interest_cleaned[c(-102),]
m1 <- lm(remove_df1$popularity~remove_df1$danceability)
m2 <- lm(remove_df2$popularity~remove_df2$danceability)
m3 <- lm(remove_df3$popularity~remove_df3$danceability)

plot(df_interest_cleaned$popularity~df_interest_cleaned$danceability,
     pch = 8,cex = 1.5, col = 'dark green',
     xlab = 'Danceability',ylab = 'Popularity',
     main = 'Scatterplot of Danceability vs Popularity')
abline(m,col = 'red',lwd = '2')
abline(m2,col = 'green',lwd = '2')
abline(m3,col = 'blue',lwd = '2')
```




```
cat('This is the original R-squared:',round(summary(m)$r.squared,5),'.These are
the R-squared after removing data point ID 41,46,53 and all of them at once:\n',
round(summary(m1)$r.squared,5),',',
round(summary(m2)$r.squared,5),',',
round(summary(m3)$r.squared,5))
```

```
## This is the original R-squared: 0.2325 .These are
## the R-squared after removing data point ID 41,46,53 and all of them at once:
##  0.23705 , 0.23671 , 0.23533
```

For the residual plot, there is linearity since the points are dispersed randomly. The variance is slightly constant, but there are some points that show the data existing more in the middle rather being evenly dispersed. Based on the histogram of the residuals, there is normality as we can see a bell-shaped curve. As for independence, each song should have an unique danceability value since one song cannot affect another song's danceability.

There are 3 outliers based on the residual plot IDs 26,63,102. After removing these outliers one-by-one and all together at once, we notice almost no change to R squared. Also based on the scatter plot, the regression lines are overlapping and barely different from each other. Therefore, there are outliers, but no influence points.

ANOVA

(2) Are the mean popularity levels different between music genres?

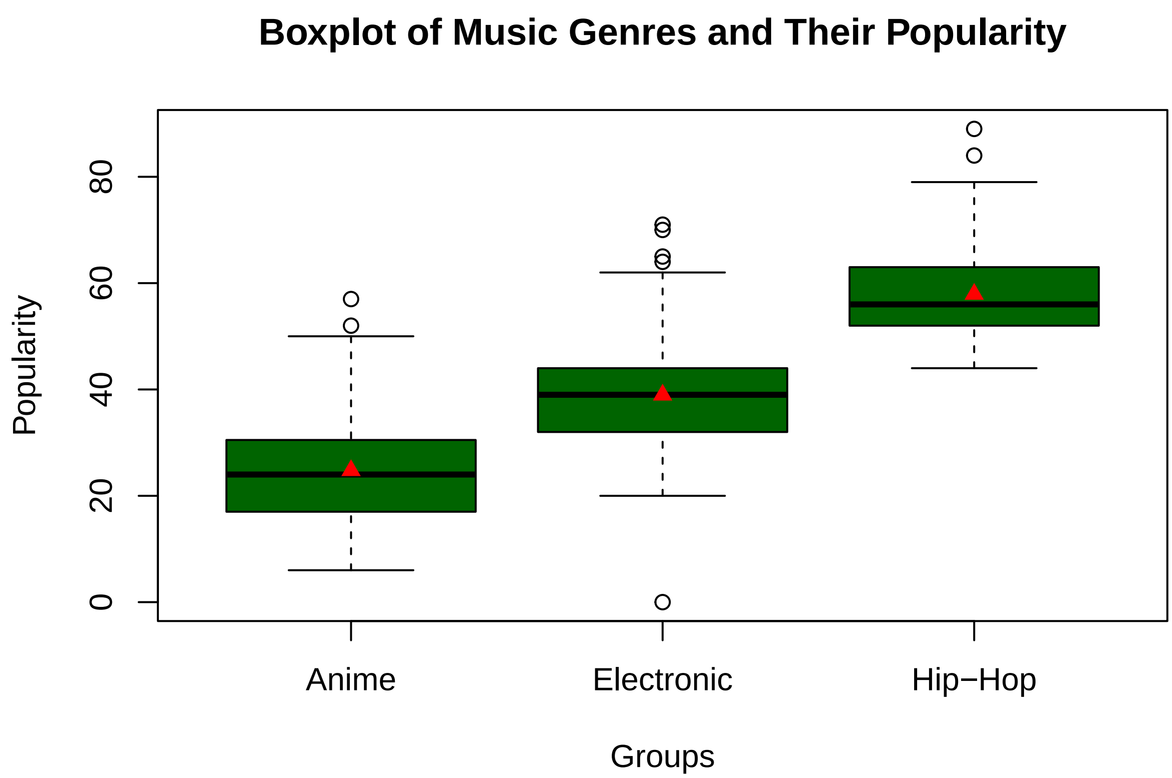
```
# Group by Music Genre
tt<-data.frame(Means=tapply(X=df_interest_cleaned$popularity,
              INDEX=df_interest_cleaned$music_genre, FUN=mean),
           SDS=tapply(X=df_interest_cleaned$popularity,
               INDEX=df_interest_cleaned$music_genre, FUN=sd),
           n=as.numeric(table(df_interest_cleaned$music_genre)))
knitr::kable(tt, digits = 2,caption = "Summary of Popularity per Music Genre")
```

Table 2: Summary of Popularity per Music Genre

	Means	SDS	n
Anime	24.76	9.70	268
Electronic	38.97	9.79	247
Hip-Hop	57.92	7.58	289

```
# Boxplot of popularity
boxplot(df_interest_cleaned$popularity~df_interest_cleaned$music_genre,col = 'dark green'
        ,xlab = 'Groups',ylab = 'Popularity',
        main = 'Boxplot of Music Genres and Their Popularity' )

points(x=1:3, y=tt$Means, col="red", pch=17)
```



One-way ANOVA test

Step 1:

- H0: $\mu_1 = \mu_2 = \mu_3$ (All music genre's popularity means are equal);
- H1: $\mu_1 \neq \mu_2 \neq \mu_3$ (Not all music genre's popularity means are equal);
- $\alpha = 0.05$;

Step 2 & 4: Use F test with df of 2 and 801 & p-value

```
# Convert student group to a factor
df_interest_cleaned$music_genre <- as.factor(df_interest_cleaned$music_genre)
am <- aov(df_interest_cleaned$popularity~df_interest_cleaned$music_genre)

# F critical value
summary(am)
```

```
##                                  Df Sum Sq Mean Sq F value              Pr(>F)
## df_interest_cleaned$music_genre   2 154432   77216   948.1 <0.0000000000000002
## Residuals                       801  65236      81
##
## df_interest_cleaned$music_genre ***
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 3: Decision Rule: We reject the null hypothesis if F $\geq$ 3.007 or p $<$ 0.05.

```
paste('This is the F-value we are comparing to:',round(qf(0.05,2,801,lower.tail = FALSE),4))
```

```
## [1] "This is the F-value we are comparing to: 3.007"
```

Step 5: We reject the null hypothesis since our F-stat 948.1 $\geq$ 3.007. Also, the p-value is less than 0.05 backing up our conclusion of rejecting the null hypothesis. Therefore, we have significant evidence at α level 0.05 that there is a mean difference in popularity between music genres.

Tukey's method

```
TukeyHSD(am)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = df_interest_cleaned$popularity ~ df_interest_cleaned$music_genre)
##
## $`df_interest_cleaned$music_genre`
##                         diff      lwr      upr p adj
## Electronic-Anime   14.20269 12.33361 16.07176     0
## Hip-Hop-Anime      33.15895 31.36194 34.95596     0
## Hip-Hop-Electronic 18.95626 17.12004 20.79248     0
```

After adjusting the p-value using Tukey's method, we can see that there is significant evidence at α level 0.05 that the popularity mean difference is differnt between all music genres.

We are 95% confident that Electronic music genre is 12.33 to 16.07 points more popular than Anime music genre with an average of 14.20 points more popular.

We are 95% confident that Hip-Hop music genre is 31.36 to 34.96 points more popular than Anime music genre with an average of 33.16 points more popular.

We are 95% confident that Hip-Hop music genre is 17.12 to 20.79 points more popular than Electronic music genre with an average of 18.96 points more popular.

Logistic Regression Model

(3) Can we correctly predict whether a Spotify song is an anime song?

```
library(aod)
```

```
## Warning: package 'aod' was built under R version 4.1.2
```

```
#Create dummy variable for anime song classification
df_interest_cleaned$g_anime <- ifelse(df_interest_cleaned$music_genre== 'Anime', 1, 0)
# Create multiple logistic regression model
log_m<- glm(g_anime~popularity + danceability + duration_ms+tempo + valence + energy + speechiness,
         data = df_interest_cleaned,
         family=binomial)
summary(log_m)
```

```
##
## Call:
## glm(formula = g_anime ~ popularity + danceability + duration_ms +
##     tempo + valence + energy + speechiness, family = binomial,
##     data = df_interest_cleaned)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.00568  -0.23089  -0.05297   0.21857   2.80605
##
## Coefficients:
##                  Estimate   Std. Error z value             Pr(>|z|)
## (Intercept)   10.268411949   1.167824383   8.793 < 0.0000000000000002 ***
## popularity    -0.151668296   0.014425080 -10.514 < 0.0000000000000002 ***
## danceability -10.093168692   1.187482328  -8.500 < 0.0000000000000002 ***
## duration_ms   -0.000004388   0.000001836  -2.391               0.0168 *
## tempo          0.003830112   0.004873531   0.786               0.4319
## valence        3.688695916   0.698664065   5.280          0.000000129 ***
## energy        -0.118251051   0.649175772  -0.182               0.8555
## speechiness   -9.363583419   2.025056379  -4.624          0.000003767 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1023.51  on 803  degrees of freedom
## Residual deviance:  363.87  on 796  degrees of freedom
## AIC: 379.87
##
## Number of Fisher Scoring iterations: 7
```

```
# Wald test
wald.test(b=coef(log_m), Sigma = vcov(log_m), Terms = 2:8)
```

```
## Wald test:
```

```
## ----------
##
## Chi-squared test:
## X2 = 164.7, df = 7, P(> X2) = 0.0
```

The Wald test was the global test to make sure at least one variable that has an association with anime music genre. Since the p-value is 0 which is less than α =0.05 level, there is statistically significant evidence that there is at least one variable that has an association with anime music genre.

Lets use only statistically significant variables for our multiple logistic regression model.

```
# ROC curve # install.package("pROC")
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.1.2

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
log_m2<- glm(g_anime~popularity + danceability + duration_ms + valence  + speechiness,
         data = df_interest_cleaned,
         family=binomial)
summary(log_m2)
```

```
##
## Call:
## glm(formula = g_anime ~ popularity + danceability + duration_ms +
##     valence + speechiness, family = binomial, data = df_interest_cleaned)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.95999  -0.22625  -0.05313   0.21844   2.83220
##
## Coefficients:
##                   Estimate    Std. Error z value             Pr(>|z|)
## (Intercept)   10.715801705   0.993713919  10.784 < 0.0000000000000002 ***
## popularity    -0.151523004   0.014403824 -10.520 < 0.0000000000000002 ***
## danceability -10.219254248   1.180385174  -8.658 < 0.0000000000000002 ***
## duration_ms   -0.000004376   0.000001835  -2.385               0.0171 *
## valence        3.693148402   0.670765019   5.506         0.0000000367 ***
## speechiness   -9.306571776   1.955541029  -4.759         0.0000019448 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 1023.51  on 803  degrees of freedom
## Residual deviance:  364.49  on 798  degrees of freedom
## AIC: 376.49
##
## Number of Fisher Scoring iterations: 7
```

```r
odds_df <- data.frame(rbind(exp(cbind(OR = coef(log_m2), confint.default(log_m2)))[2,],
           exp(cbind(OR = coef(log_m2), confint.default(log_m2))/100)[3,],
           exp(cbind(OR = coef(log_m2), confint.default(log_m2)))[4,],
           exp(cbind(OR = coef(log_m2), confint.default(log_m2))/100)[5,],
           exp(cbind(OR = coef(log_m2), confint.default(log_m2))/100)[6,]),
          row.names = c('Popularity','Danceability(100th of a unit)','Duration_MS',
                       'Valence(100th of a unit)','Speechiness(100th of a unit)'))
colnames(odds_df) <- c('Odds Ratio','Lower Bound (95% CI)','Upper Bound (95% CI)')
odds_df
```

```
##                               Odds Ratio Lower Bound (95% CI)
## Popularity                     0.8593981            0.8354757
## Danceability(100th of a unit)  0.9028557            0.8822078
## Duration_MS                    0.9999956            0.9999920
## Valence(100th of a unit)       1.0376219            1.0240698
## Speechiness(100th of a unit)   0.9111336            0.8768726
##                               Upper Bound (95% CI)
## Popularity                               0.8840055
## Danceability(100th of a unit)            0.9239869
## Duration_MS                              0.9999992
## Valence(100th of a unit)                 1.0513533
## Speechiness(100th of a unit)             0.9467333
```

Reject H0:βpopularity=0 or Odd Ratio popularity=1 after adjusting for danceability,duration_ms,valence and speechiness. We have significant evidence at the α =0.05 level thatβpopularity$\neq$ 0 since p-value is almost zero. That is, there is evidence of an association between anime music genre and popularity after adjusting for danceability,duration_ms,valence and speechiness. This means that for every 1 unit increase in popularity is associated with about a 14% decrease of being an anime song. We are 95% confident that the true odds ratio between anime music genre and popularity is between 0.835 and 0.884 after adjusting for danceability,duration_ms,valence and speechiness.

Reject H0:βdanceability=0 or Odd Ratio danceability=1 after adjusting for popularity,duration_ms,valence and speechiness. We have significant evidence at the α =0.05 level thatβdanceability$\neq$ 0 since p-value $0.0171 \leq 0.05$. That is, there is evidence of an association between anime music genre and danceability after adjusting for popularity,duration_ms,valence and speechiness. This means that for every 0.01 unit increase in danceability is associated with about a 10% decrease of being an anime song. We are 95% confident that the true odds ratio between anime music genre and danceability in 100th of a unit is between 0.88 and 0.92 after adjusting for popularity,duration_ms,valence and speechiness.

Reject H0:βduration_ms=0 or Odd Ratio duration_ms=1 after adjusting for danceability,popularity,valence and speechiness. We have significant evidence at the α =0.05 level thatβduration_ms$\neq$ 0 since p-value is almost zero. That is, there is evidence of an association between anime music genre and duration_ms after adjusting for danceability,popularity,valence and speechiness. This means that for every 1 millisecond increase in the song is associated with less than a 0.0001% decrease of being an anime song. We are 95% confident that the true odds ratio between anime music genre and duration_ms is between 0.9999920 and 0.9999992 after adjusting for danceability,popularity,valence and speechiness.

Reject H0:βvalence=0 or Odd Ratio valence=1 after adjusting for danceability,duration_ms,popularity and speechiness. We have significant evidence at the α =0.05 level thatβvalence$\neq$ 0 since p-value is almost zero. That is, there is evidence of an association between anime music genre and valence after adjusting for danceability,duration_ms,popularity and speechiness. This means that for every 0.01 unit increase in valence is associated with about a 3.8% increase of being an anime song. We are 95% confident that the true odds ratio between anime music genre and valence in 100th of a unit is between 1.269 and 1.65 after adjusting for danceability,duration_ms,popularity and speechiness.

Reject H0:βspeechiness=0 or Odd Ratio speechiness=1 after adjusting for danceability,duration_ms,valence and popularity. We have significant evidence at the α =0.05 level thatβspeechiness$\neq$ 0 since p-value is almost zero. That is, there is evidence of an association between anime music genre and speechiness after adjusting for danceability,duration_ms,valence and popularity. This means that for every 0.01 unit increase in speechiness is associated with about a 9% decrease of being an anime song. We are 95% confident that the true odds ratio between anime music genre and speechiness in 100th of a unit is between 0.877 and 0.947 after adjusting for danceability,duration_ms,valence and popularity.

```r
# Predicted values
df_interest_cleaned$prob2 <-predict(log_m2, type=c("response"))


# Build a ROC curve
g2 <- roc(df_interest_cleaned$g_anime ~ df_interest_cleaned$prob2)
```

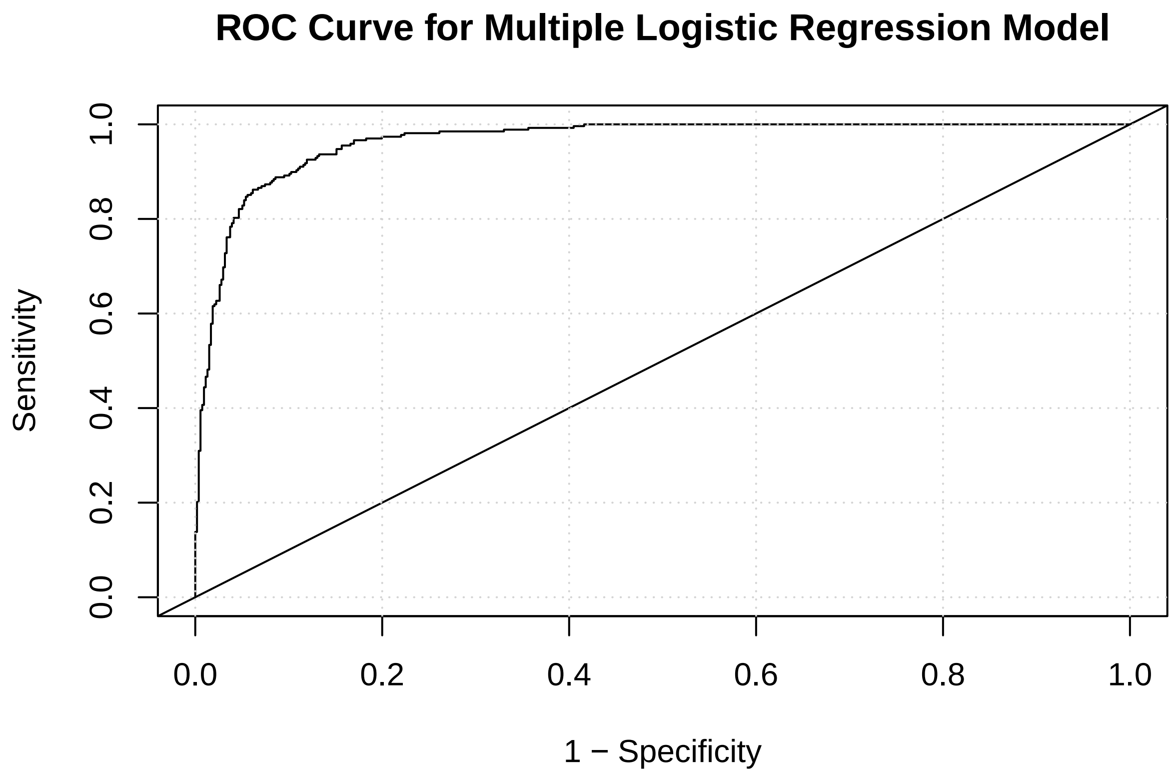
```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```r
# see the results - c-statistics value
print(g2)
```

```
##
## Call:
## roc.formula(formula = df_interest_cleaned$g_anime ~ df_interest_cleaned$prob2)
##
## Data: df_interest_cleaned$prob2 in 536 controls (df_interest_cleaned$g_anime 0) < 268 cases (df_inte
## Area under the curve: 0.9649
```

```r
# plot the ROC Curve
plot(1-g2$specificities, g2$sensitivities, type = "l",
xlab = "1 - Specificity", ylab = "Sensitivity", main = "ROC Curve for Multiple Logistic Regression Mode
abline(a=0,b=1)
grid()
```



Our c-statistics (aka area under the ROC curve) equals 0.9649 which is very high meaning this multiple logistic model is a good fit to determine whether a song is an anime song.

Conclusion & Results

The dataset I am working with for this analysis is a sample of 1000 songs and their audio features or characteristics originally extracted from Spotify's API. The three music genres I am interested in analyzing are Anime, Electronic, and Hip-Hop. My project goals are to see if the music genre anime has any differences between the other two genres and if we can create predictive models.

My first question was to see if any of the audio features or characteristics have a linear relationship with popularity. Multiple linear regression was not a good fit for this question because as seen in the pairwise plot, we see that almost all variables have very little to no correlation or show no signs of linearity with popularity. The danceability variable does seem to have some signs of linearity and after running through a simple linear regression model with popularity as response variable and danceability as explanatory variable, we see that for every 0.1 unit in danceability, there is a 4.5721 increase in popularity "points". Also, based on the residual plots, the four conditions of least-squares regression (linearity,normality,independence, and constant variance) hold true. Constant variance is debatable, but overall its pretty constant. This proves that danceability is correlated with popularity.

My second question was to see if the average popularity is different between the three music genres. Using an one-way ANOVA and the boxplot, we can see that the average popularity between music genres are different. Using pairwise tests, we concluded that in order of lowest to highest popularity: anime, electronic and hip-hop. Unfortunately, that means anime is not as popular as electronic and hip-hop.

My third question was see if we can build a classifier to identify whether a song is anime music genre or not based on audio features. Using multiple logistic regression, we built a predictive model that works splendidly. The model can classify an anime song based on a song's danceability, duration in milliseconds, valence and speechiness.

In terms of limitations and concerns, the simple linear regression model may not be the best model for determining popularity even though the math says otherwise. Popularity should be considered by other variables other than just how danceable the song is. Another concern is how the dataset originally classified the music genres. Songs can have multiple genres, but maybe Spotify has limitations on outputting multiple genres in the dataset.

Back to the project goal and above results, we can say that the anime music genre (unfortunately) is not very popular as opposed to electronic and hip-hop music. The simple linear regression model helps us conclude that the more we want to dance to a song, the more likely the song is popular. This holds true as popular songs (electronic and hip-hop) are played in public places such as bars,clubs and parties versus anime songs which are usually played in conventions that occur only a few times a year and sometimes Japanese malls/stores. As for the classification of whether a song is anime or not, danceability and valence I can see those variables as good predictors, but duration_ms and speechiness were unexpectedly part of the model as I would have not expected those variables to have any association with classifying an anime song.