Linear Regression

2022-12-06

```
data <-  read.csv("spotify_cleaned.csv")
```

Data Science Questions: Are the variables contributing for predicting “popularity” of the songs is same for different genres?

Create a new variable named “Valence_C”.

```
data$Valence_C <- rep(0,nrow(data))

data1 <- within(data, {
  Valence_C[valence>=0.8 & valence<=1] <- "more positive"
  Valence_C[valence>=0.5 & valence<0.8] <- "moderate"
  Valence_C[valence<=0.499] <- "more negative"
})
head(data1)
```

```
##   X                                                  name           artist
## 1 1 I Don't Care (with Justin Bieber) - Loud Luxury Remix       Ed Sheeran
## 2 2                       Memories - Dillon Francis Remix         Maroon 5
## 3 3                       All the Time - Don Diablo Remix     Zara Larsson
## 4 4                     Call You Mine - Keanu Silva Remix The Chainsmokers
## 5 5               Someone You Loved - Future Humans Remix    Lewis Capaldi
## 6 6     Beautiful People (feat. Khalid) - Jack Wins Remix       Ed Sheeran
##   popularity year genre  subgenre danceability energy key loudness mode
## 1         66 2019   pop dance pop        0.748  0.916   6   -2.634    1
## 2         67 2019   pop dance pop        0.726  0.815  11   -4.969    1
## 3         70 2019   pop dance pop        0.675  0.931   1   -3.432    0
## 4         60 2019   pop dance pop        0.718  0.930   7   -3.778    1
## 5         69 2019   pop dance pop        0.650  0.833   1   -4.672    1
## 6         67 2019   pop dance pop        0.675  0.919   8   -5.385    1
##   speechiness acousticness instrumantalness liveness valence   tempo duration
## 1      0.0583       0.1020         0.00e+00   0.0653   0.518 122.036   194754
## 2      0.0373       0.0724         4.21e-03   0.3570   0.693  99.972   162600
## 3      0.0742       0.0794         2.33e-05   0.1100   0.613 124.008   176616
## 4      0.1020       0.0287         9.43e-06   0.2040   0.277 121.956   169093
## 5      0.0359       0.0803         0.00e+00   0.0833   0.725 123.976   189052
## 6      0.1270       0.0799         0.00e+00   0.1430   0.585 124.982   163049
##       Valence_C
## 1      moderate
## 2      moderate
## 3      moderate
## 4 more negative
## 5      moderate
## 6      moderate
```

Fit multiple linear regression models separately for different genres.

```
set.seed(12)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
library(tidyverse)
```

```
## ── Attaching packages
## ───────────────────────────────────────
## tidyverse 1.3.2 ──
```

```
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.5.0
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ✓ purrr   0.3.5
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
## ✕ purrr::lift()   masks caret::lift()
```

```
pop <- data1[data1$genre=="pop",]
edm <- data1[data1$genre=="edm",]
```

```
names(pop)
```

```
##  [1] "X"                 "name"              "artist"            "popularity"
##  [5] "year"              "genre"             "subgenre"          "danceability"
##  [9] "energy"            "key"               "loudness"          "mode"
## [13] "speechiness"       "acousticness"      "instrumantalness" "liveness"
## [17] "valence"           "tempo"             "duration"          "Valence_C"
```

```
training_samples <- pop$popularity %>%
  createDataPartition(p=0.8, list=FALSE)

train <- pop[training_samples, ]
test <- pop[-training_samples, ]
dim(train)
```

```
## [1] 4407   20
```

Fit the FULL linear regression model.

```
fit1 <- lm(popularity ~ danceability + energy + loudness + speechiness + acousticness + instruma
ntalness + liveness + valence + tempo +Valence_C, data = train)
summary(fit1)
```

```
##
## Call:
## lm(formula = popularity ~ danceability + energy + loudness +
##     speechiness + acousticness + instrumantalness + liveness +
##     valence + tempo + Valence_C, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.194 -17.056   4.545  19.186  58.333
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           74.966782   4.959693  15.115  < 2e-16 ***
## danceability          14.311075   3.227621   4.434 9.48e-06 ***
## energy               -33.654892   3.451728  -9.750  < 2e-16 ***
## loudness               2.548691   0.198625  12.832  < 2e-16 ***
## speechiness           22.378111   5.546528   4.035 5.56e-05 ***
## acousticness           1.132331   2.091348   0.541   0.5882
## instrumantalness     -10.358720   2.168184  -4.778 1.83e-06 ***
## liveness               0.372393   2.730148   0.136   0.8915
## valence                4.192508   3.821088   1.097   0.2726
## tempo                  0.002425   0.015471   0.157   0.8754
## Valence_Cmore negative   0.246728   1.379914   0.179   0.8581
## Valence_Cmore positive  -4.052253   1.548440  -2.617   0.0089 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.18 on 4395 degrees of freedom
## Multiple R-squared:  0.07447,    Adjusted R-squared:  0.07215
## F-statistic: 32.15 on 11 and 4395 DF,  p-value: < 2.2e-16
```

Remove insignificant variables.

```
fit2 <- lm(popularity ~ danceability + energy + loudness + speechiness + instrumantalness + Vale
nce_C, data = train)
summary(fit2)
```

```
##
## Call:
## lm(formula = popularity ~ danceability + energy + loudness +
##     speechiness + instrumantalness + Valence_C, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.044 -17.094   4.524  19.181  58.725
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             77.8923     3.9773  19.584  < 2e-16 ***
## danceability            14.7807     3.0557   4.837 1.36e-06 ***
## energy                 -33.8176     3.0221 -11.190  < 2e-16 ***
## loudness                 2.5462     0.1982  12.844  < 2e-16 ***
## speechiness             23.0326     5.4536   4.223 2.46e-05 ***
## instrumantalness       -10.5435     2.1567  -4.889 1.05e-06 ***
## Valence_Cmore negative  -0.9738     0.8297  -1.174   0.2406
## Valence_Cmore positive  -3.0531     1.2655  -2.413   0.0159 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.18 on 4399 degrees of freedom
## Multiple R-squared:  0.07414,    Adjusted R-squared:  0.07267
## F-statistic: 50.32 on 7 and 4399 DF,  p-value: < 2.2e-16
```

Check interactions.

```
fit12 <- lm(popularity ~ (danceability+energy+loudness+speechiness+instrumantalness)^2, data=tra
in)
summary(fit12)
```

```
##
## Call:
## lm(formula = popularity ~ (danceability + energy + loudness +
##     speechiness + instrumantalness)^2, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -62.687 -16.864   4.522  19.050  53.836
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    103.1230    14.9440   6.901 5.92e-12 ***
## danceability                    -5.9375    22.2770  -0.267 0.789844
## energy                         -72.3426    14.8384  -4.875 1.12e-06 ***
## loudness                         3.8407     0.9789   3.924 8.85e-05 ***
## speechiness                     28.1324    60.8885   0.462 0.644082
## instrumantalness               -70.0682    18.7063  -3.746 0.000182 ***
## danceability:energy             41.5464    21.7212   1.913 0.055850 .
## danceability:loudness            0.3138     1.4425   0.218 0.827802
## danceability:speechiness       -74.1727    42.3132  -1.753 0.079681 .
## danceability:instrumantalness  -1.8488    14.8859  -0.124 0.901163
## energy:loudness                 -1.2025     0.6406  -1.877 0.060553 .
## energy:speechiness              21.8262    47.0904   0.463 0.643032
## energy:instrumantalness         29.7824    13.9461   2.136 0.032772 *
## loudness:speechiness            -4.4678     3.3171  -1.347 0.178077
## loudness:instrumantalness       -4.5224     0.8919  -5.071 4.12e-07 ***
## speechiness:instrumantalness    57.2961    62.7741   0.913 0.361432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.09 on 4391 degrees of freedom
## Multiple R-squared:  0.08224,    Adjusted R-squared:  0.0791
## F-statistic: 26.23 on 15 and 4391 DF,  p-value: < 2.2e-16
```

```
fit3 <- lm(popularity~danceability+energy+loudness+speechiness+instrumantalness+energy*loudness+
loudness*instrumantalness,data=train)
summary(fit3)
```

```
## 
## Call:
## lm(formula = popularity ~ danceability + energy + loudness + 
##     speechiness + instrumantalness + energy * loudness + loudness * 
##     instrumantalness, data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -62.679 -16.902   4.628  19.004  51.384 
## 
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                85.3323     4.8193  17.706  < 2e-16 ***
## danceability               13.6163     2.8745   4.737 2.24e-06 ***
## energy                    -42.2346     5.2307  -8.074 8.68e-16 ***
## loudness                    3.5558     0.4316   8.239 2.26e-16 ***
## speechiness                23.9101     5.4301   4.403 1.09e-05 ***
## instrumantalness          -36.5057     5.5841  -6.537 6.97e-11 ***
## energy:loudness            -1.1351     0.6160  -1.843   0.0654 .  
## loudness:instrumantalness  -3.1077     0.6226  -4.991 6.22e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 24.12 on 4399 degrees of freedom
## Multiple R-squared:  0.07821,    Adjusted R-squared:  0.07674 
## F-statistic: 53.32 on 7 and 4399 DF,  p-value: < 2.2e-16
```

Make Predictions

```
pred1 <- fit1 %>% predict(test)
p1 = data.frame(
  RMSE=RMSE(pred1,test$popularity),
  R2=R2(pred1,test$popularity)
)

pred2 <- fit2 %>% predict(test)
p2 <- data.frame(
  RMSE=RMSE(pred2,test$popularity),
  R2=R2(pred2,test$popularity)
)

pred3 <- fit3 %>% predict(test)
p3 <- data.frame(
  RMSE=RMSE(pred3,test$popularity),
  R2=R2(pred3,test$popularity)
)
```

```
summary(fit1)$fstatistic[1]
```

```
##   value
## 32.1482
```

```
summary(fit1)$adj.r.squared
```

```
## [1] 0.07215349
```

```
summary(fit1)$sigma #RSE
```

```
## [1] 24.18283
```

```
all=rbind(p1,p2,p3)
all=cbind(all,c(summary(fit1)$fstatistic[1],summary(fit2)$fstatistic[1],summary(fit3)$fstatistic[1]))
all=cbind(all,c(summary(fit1)$adj.r.squared,summary(fit2)$adj.r.squared,summary(fit3)$adj.r.squared))
all=cbind(all,c(summary(fit1)$sigma,summary(fit2)$sigma,summary(fit3)$sigma))

all=cbind(all,c("fit1","fit2","fit3"))
colnames(all)[c(3,4,5,6)]<-c("F stat","Adj R 2","RSE","models")
all
```

```
##       RMSE         R2   F stat    Adj R 2      RSE models
## 1 24.56187 0.06293309 32.14820 0.07215349 24.18283   fit1
## 2 24.56240 0.06287795 50.32470 0.07266956 24.17610   fit2
## 3 24.48017 0.06916223 53.31969 0.07674341 24.12294   fit3
```

It turns out that fit3 is the best model.

Next we check the predictors for genres “EDM” and compared with “Pop”.

```
training_samples <- edm$popularity %>%
  createDataPartition(p=0.8,list = FALSE)

train <- edm[training_samples,]
test <- edm[-training_samples,]
dim(train)
```

```
## [1] 4836   20
```

```
names(train)
```

```
##  [1] "X"                "name"             "artist"           "popularity"      
##  [5] "year"             "genre"            "subgenre"         "danceability"    
##  [9] "energy"           "key"              "loudness"         "mode"            
## [13] "speechiness"      "acousticness"     "instrumantalness" "liveness"        
## [17] "valence"          "tempo"            "duration"         "Valence_C"
```

Fit FULL linear regression model for EDM.

```
fit11 <- lm(popularity ~ danceability+energy+loudness+speechiness+acousticness+instrumantalness+
liveness+valence+tempo+Valence_C, data = train)
summary(fit11)
```

```
## 
## Call:
## lm(formula = popularity ~ danceability + energy + loudness + 
##     speechiness + acousticness + instrumantalness + liveness + 
##     valence + tempo + Valence_C, data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -60.222 -17.139   1.452  16.119  60.599 
## 
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)    
## (Intercept)             69.80804    5.19775  13.430  < 2e-16 ***
## danceability            -1.61857    2.98997  -0.541   0.5883    
## energy                 -22.54855    3.39266  -6.646 3.34e-11 ***
## loudness                 1.05649    0.19034   5.551 3.00e-08 ***
## speechiness             -5.85242    4.54543  -1.288   0.1980    
## acousticness            18.75716    2.42862   7.723 1.37e-14 ***
## instrumantalness       -11.67195    1.11720 -10.448  < 2e-16 ***
## liveness                -0.50596    1.91175  -0.265   0.7913    
## valence                  3.42872    2.88262   1.189   0.2343    
## tempo                   -0.06207    0.02144  -2.895   0.0038 ** 
## Valence_Cmore negative  -3.01397    1.23370  -2.443   0.0146 *  
## Valence_Cmore positive  -0.09883    1.62645  -0.061   0.9515    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 22 on 4824 degrees of freedom
## Multiple R-squared:  0.09797,    Adjusted R-squared:  0.09592 
## F-statistic: 47.63 on 11 and 4824 DF,  p-value: < 2.2e-16
```

Remove insignificant variables.

```
fit22 <- lm(popularity ~ energy+loudness+acousticness+instrumantalness+tempo+Valence_C,data=trai
n)
summary(fit22)
```

```
##
## Call:
## lm(formula = popularity ~ energy + loudness + acousticness +
##     instrumantalness + tempo + Valence_C, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.994 -17.040   1.451  16.197  60.350
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            70.63965    4.14229  17.053  < 2e-16 ***
## energy                -22.92205    3.34518  -6.852 8.18e-12 ***
## loudness                1.08253    0.18969   5.707 1.22e-08 ***
## acousticness           18.71702    2.40287   7.789 8.18e-15 ***
## instrumantalness      -11.83083    1.07050 -11.052  < 2e-16 ***
## tempo                  -0.06221    0.02095  -2.969  0.00301 **
## Valence_Cmore negative  -4.03112    0.72655  -5.548 3.04e-08 ***
## Valence_Cmore positive   0.72013    1.45498   0.495  0.62066
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22 on 4828 degrees of freedom
## Multiple R-squared:  0.09733,    Adjusted R-squared:  0.09602
## F-statistic: 74.37 on 7 and 4828 DF,  p-value: < 2.2e-16
```

Check interactions.

```
fit12 <- lm(popularity~(energy+loudness+acousticness+instrumantalness+tempo)^2,data = train)
summary(fit12)
```

```
## 
## Call:
## lm(formula = popularity ~ (energy + loudness + acousticness + 
##     instrumantalness + tempo)^2, data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -61.181 -17.354   1.618  15.903  61.269 
## 
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                    174.99957   29.15749   6.002 2.09e-09 ***
## energy                        -126.28142   28.83189  -4.380 1.21e-05 ***
## loudness                         5.00799    1.74073   2.877 0.004033 ** 
## acousticness                   -15.48766   22.37814  -0.692 0.488915    
## instrumantalness               -36.98138   14.61259  -2.531 0.011412 *  
## tempo                           -0.89089    0.22725  -3.920 8.97e-05 ***
## energy:loudness                  0.02925    0.87817   0.033 0.973426    
## energy:acousticness             -4.79823   18.10678  -0.265 0.791023    
## energy:instrumantalness         26.52900    9.93878   2.669 0.007628 ** 
## energy:tempo                     0.79113    0.22253   3.555 0.000381 ***
## loudness:acousticness            1.82139    1.16484   1.564 0.117969    
## loudness:instrumantalness       -1.08406    0.51276  -2.114 0.034553 *  
## loudness:tempo                  -0.03137    0.01284  -2.443 0.014613 *  
## acousticness:instrumantalness    5.23843    9.01513   0.581 0.561220    
## acousticness:tempo               0.41017    0.11723   3.499 0.000472 ***
## instrumantalness:tempo          -0.02880    0.08831  -0.326 0.744372    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 22.01 on 4820 degrees of freedom
## Multiple R-squared:  0.09745,    Adjusted R-squared:  0.09464 
## F-statistic:  34.7 on 15 and 4820 DF,  p-value: < 2.2e-16
```

```
fit33 <- lm(popularity~energy+loudness+acousticness+instrumantalness+tempo+energy*instrumantalne
ss+energy*tempo+loudness*acousticness+loudness*instrumantalness+loudness*tempo+acousticness*temp
o,data = train)
summary(fit33)
```

```
##
## Call:
## lm(formula = popularity ~ energy + loudness + acousticness +
##     instrumantalness + tempo + energy * instrumantalness + energy *
##     tempo + loudness * acousticness + loudness * instrumantalness +
##     loudness * tempo + acousticness * tempo, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -60.852 -17.348   1.649  15.852  61.511
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               173.93172   27.04614   6.431 1.39e-10 ***
## energy                   -124.98282   26.31185  -4.750 2.09e-06 ***
## loudness                    4.94943    1.52289   3.250 0.001162 **
## acousticness              -20.87972   15.32116  -1.363 0.173008
## instrumantalness          -39.70270    9.87629  -4.020 5.91e-05 ***
## tempo                      -0.87870    0.21524  -4.082 4.53e-05 ***
## energy:instrumantalness    25.38726    9.21056   2.756 0.005868 **
## energy:tempo                0.77807    0.20934   3.717 0.000204 ***
## loudness:acousticness       1.40103    0.64173   2.183 0.029070 *
## loudness:instrumantalness  -1.13734    0.50221  -2.265 0.023578 *
## loudness:tempo             -0.03033    0.01197  -2.535 0.011286 *
## acousticness:tempo          0.41191    0.11624   3.544 0.000399 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22 on 4824 degrees of freedom
## Multiple R-squared:  0.09737,    Adjusted R-squared:  0.09531
## F-statistic: 47.31 on 11 and 4824 DF,  p-value: < 2.2e-16
```

Make predictions

```
pred11 <- fit11 %>% predict(test)
p11=data.frame(
  RMSE=RMSE(pred11,test$popularity),
  R2=R2(pred11,test$popularity)
)

pred22 <- fit22 %>% predict(test)
p22=data.frame(
  RMSE=RMSE(pred22,test$popularity),
  R2=R2(pred22,test$popularity)
)

pred33 <- fit33 %>% predict(test)
p33=data.frame(
  RMSE=RMSE(pred33,test$popularity),
  R2=R2(pred33,test$popularity)
)
```

```
all2=rbind(p11,p22,p33)
all2=cbind(all2,c(summary(fit11)$fstatistic[1],summary(fit22)$fstatistic[1],summary(fit33)$fstatistic[1]))
all2=cbind(all2,c(summary(fit11)$adj.r.squared,summary(fit22)$adj.r.squared,summary(fit33)$adj.r.squared))
all2=cbind(all2,c(summary(fit11)$sigma,summary(fit22)$sigma,summary(fit33)$sigma))

all2=cbind(all2,c("fit11","fit22","fit33"))
colnames(all2)[c(3,4,5,6)] <- c("F stat","Adj R 2","RSE","models")
all2
```

```
##       RMSE         R2   F stat    Adj R 2      RSE models
## 1 22.32606 0.07767128 47.63290 0.09591728 21.99751  fit11
## 2 22.32488 0.07775583 74.37116 0.09602481 21.99620  fit22
## 3 22.26614 0.08191485 47.30527 0.09530761 22.00492  fit33
```

It turns out that fit33j is the best model.