UE20CS312 - Data Analytics Worksheet 2b : Multiple Linear Regression

GAURAV MAHAJAN

2022-09-15

###Importing libraries and uploading the dataset

```
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.1

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✓ ggplot2 3.3.6     ✓ purrr   0.3.4
## ✓ tibble  3.1.8     ✓ dplyr   1.0.9
## ✓ tidyr   1.2.0     ✓ stringr 1.4.0
## ✓ readr   2.1.2     ✓ forcats 0.5.2

## Warning: package 'ggplot2' was built under R version 4.2.1

## Warning: package 'tibble' was built under R version 4.2.1

## Warning: package 'tidyr' was built under R version 4.2.1

## Warning: package 'readr' was built under R version 4.2.1

## Warning: package 'purrr' was built under R version 4.2.1

## Warning: package 'dplyr' was built under R version 4.2.1

## Warning: package 'stringr' was built under R version 4.2.1

## Warning: package 'forcats' was built under R version 4.2.1

## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.2.1

## corrplot 0.92 loaded

library(olsrr)

## Warning: package 'olsrr' was built under R version 4.2.1
```

```
##
## Attaching package: 'olsrr'
##
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
df <- read_csv('spotify.csv')
```

```
## Rows: 195 Columns: 13
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (13): danceability, energy, key, loudness, mode, speechiness,
acousticne...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
head(df)
```

```
## # A tibble: 6 × 13
##   danceabil…¹ energy   key loudn…²  mode speec…³ acous…⁴ instr…⁵ liven…⁶
valence
##         <dbl>  <dbl> <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
<dbl>
## 1       0.803 0.624      7   -6.76     0  0.0477  0.451  7.34e-4  0.1
0.628
## 2       0.762 0.703     10   -7.95     0  0.306   0.206  0        0.0912
0.519
## 3       0.261 0.0149     1  -27.5      1  0.0419  0.992  8.97e-1  0.102
0.0382
## 4       0.722 0.736      3   -6.99     0  0.0585  0.431  1.18e-6  0.123
0.582
## 5       0.787 0.572      1   -7.52     1  0.222   0.145  0        0.0753
0.647
## 6       0.778 0.632      8   -6.42     1  0.125   0.0404 0        0.0912
0.827
## # … with 3 more variables: tempo <dbl>, duration_ms <dbl>,
## #   time_signature <dbl>, and abbreviated variable names ¹danceability,
## #   ²loudness, ³speechiness, ⁴acousticness, ⁵instrumentalness, ⁶liveness
## # ℹ Use `colnames()` to see all variable names
```

###Problem-1 (0.5 Points) Check for missing values in the dataset and normalize the dataset.

```
#checking for missing values
sum(is.na(df))
```

```
## [1] 0
```

```
#Normalisation
min_max_norm <- function(x) {
  (x - min(x)) / (max(x) - min(x))
}
df_norm <- as.data.frame(lapply(df, min_max_norm))
```

This implies there is no missing data in the dataset

```
#for scaling :
#for z score scaling to be done centering is done
df<-as.data.frame(scale(df))
```

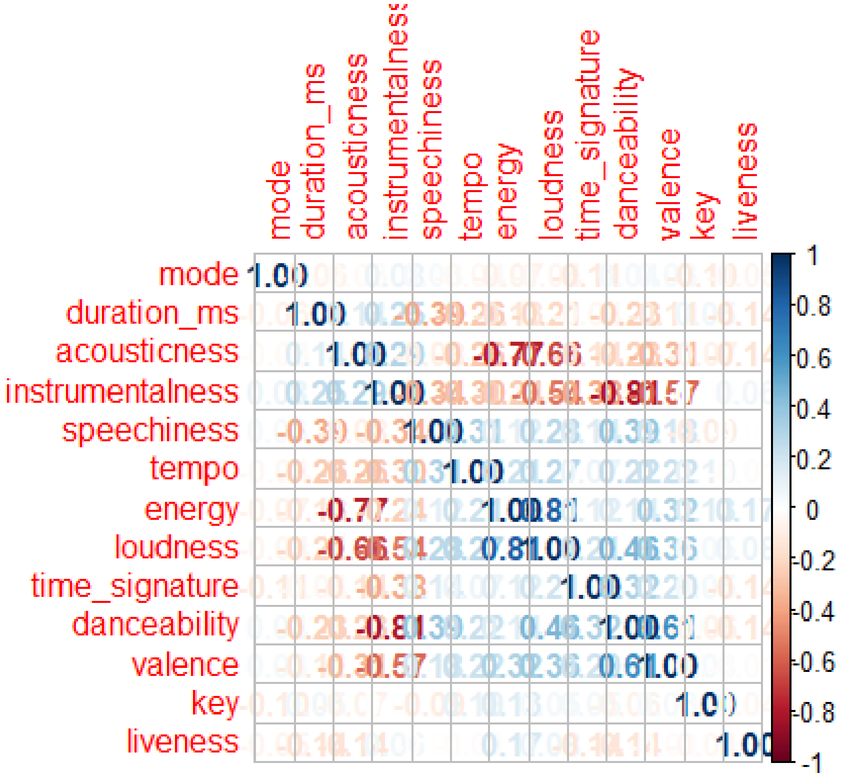
###Problem-2 (2 Points) Fit a linear model to predict the energy rating using all other attributes.Get the summary of the model and explain the results in detail.[Hint : Use the lm() function. Click here To get the documentation of the same.]

```
#For all the attributes fitting a linear model
full_model<-lm(energy~.,data = df)
summary(full_model)

##
## Call:
## lm(formula = energy ~ ., data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00232 -0.22889 -0.00973  0.27796  1.24597
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       9.156e-17  2.920e-02   0.000  1.00000
## danceability     -2.751e-01  5.555e-02  -4.952 1.67e-06 ***
## key               4.970e-02  3.009e-02   1.652  0.10030
## loudness          7.015e-01  4.561e-02  15.381  < 2e-16 ***
## mode             -4.794e-02  3.034e-02  -1.580  0.11582
## speechiness       2.359e-02  3.519e-02   0.670  0.50343
## acousticness     -3.435e-01  4.136e-02  -8.306 2.21e-14 ***
## instrumentalness  1.493e-01  5.577e-02   2.677  0.00811 **
## liveness          2.004e-02  3.100e-02   0.646  0.51880
## valence           2.046e-01  3.884e-02   5.269 3.85e-07 ***
## tempo            -2.395e-02  3.295e-02  -0.727  0.46817
## duration_ms      -1.865e-02  3.303e-02  -0.565  0.57298
## time_signature    2.409e-02  3.220e-02   0.748  0.45535
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4077 on 182 degrees of freedom
## Multiple R-squared:  0.844,  Adjusted R-squared:  0.8338
## F-statistic: 82.08 on 12 and 182 DF,  p-value: < 2.2e-16
```

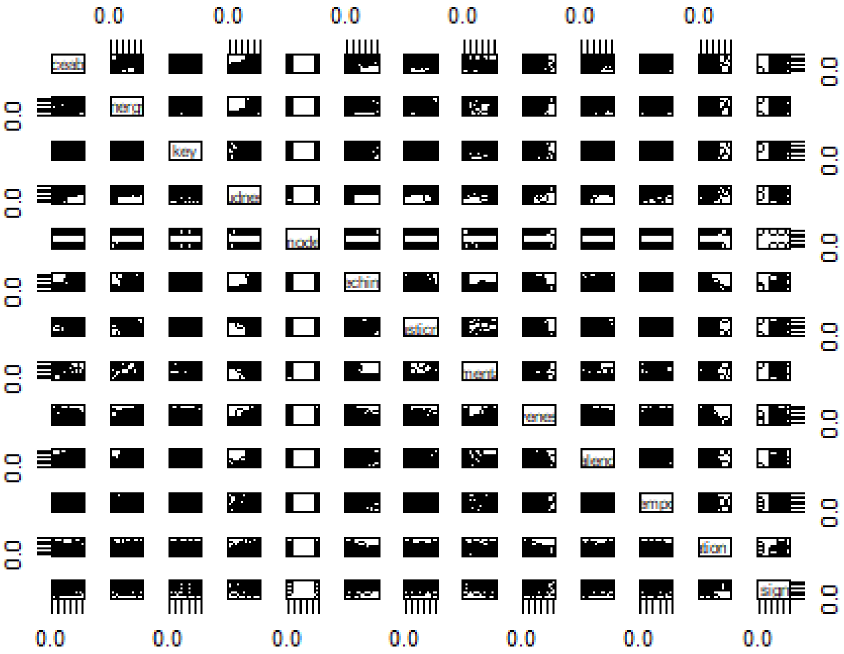
The asterisks tell us the significance .If alpha is 0.05 :: we select : danceability , loudness , acousticness , instrumentalness , valence according to this model.Betas are not all zero seeing he F statistic

###Problem-3 (2 points) With the help of a correlogram and scatter plots, choose the features you think are important and model an MLR. Justify your choice and explain the new findings.

```
df_cor <- cor(df_norm)
corrplot(df_cor, order = "hclust", method = "number")
```



```
plot(df_norm)
```



```
#energy has significant positive correlation with loudness, valence and tempo
#energy has significant negative correlation with acousticness,
instrumentalness
#these variables are significant in the model

model_cor <- lm(energy~loudness+acousticness+valence+tempo+instrumentalness,
data=df_norm)
summary(model_cor)

##
## Call:
## lm(formula = energy ~ loudness + acousticness + valence + tempo +
##     instrumentalness, data = df_norm)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29492 -0.07908  0.00527  0.08305  0.33311
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.282319   0.080630  -3.501 0.000577 ***
## loudness          1.111635   0.077476  14.348  < 2e-16 ***
## acousticness     -0.294210   0.035350  -8.323 1.69e-14 ***
## valence           0.123858   0.036325   3.410 0.000795 ***
## tempo            -0.003382   0.037722  -0.090 0.928645
## instrumentalness  0.230480   0.032243   7.148 1.86e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1154 on 189 degrees of freedom
## Multiple R-squared:  0.8108, Adjusted R-squared:  0.8058
## F-statistic:   162 on 5 and 189 DF,  p-value: < 2.2e-16
```

###Problem-4 (1 Point) Conduct a partial F-test to determine if the attributes not chosen by you in Problem-3 are significant to predict the energy.What are the null and alternate hypotheses? [Hint : Use the anova function between models created in Problem-2 and Problem-3]

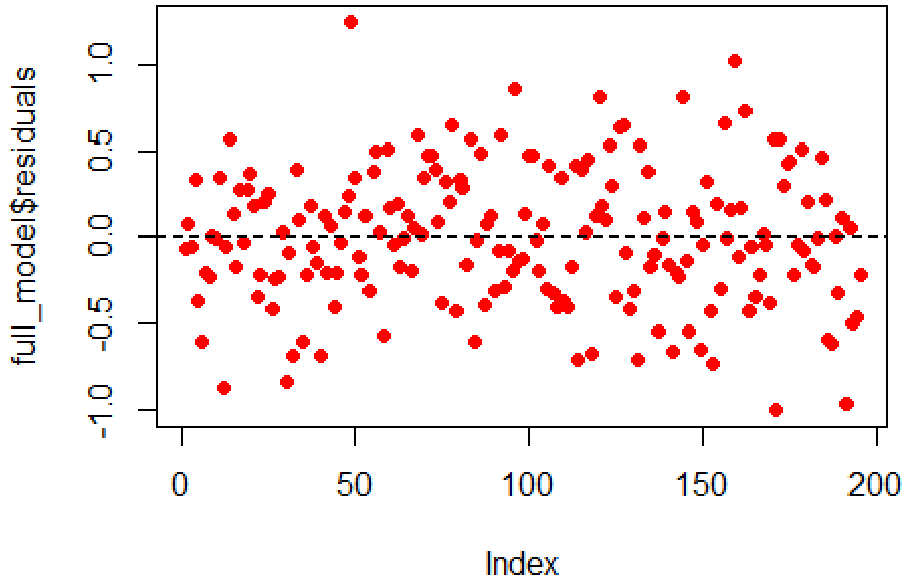
```
anova(model_cor,full_model)

## Analysis of Variance Table
##
## Model 1: energy ~ loudness + acousticness + valence + tempo +
instrumentalness
## Model 2: energy ~ danceability + key + loudness + mode + speechiness +
##     acousticness + instrumentalness + liveness + valence + tempo +
##     duration_ms + time_signature
##   Res.Df     RSS Df Sum of Sq F Pr(>F)
## 1    189  2.5151
## 2    182 30.2566  7   -27.741
```

###Problem-5 (1.5 Points) AIC - Akaike Information Criterion is used to compare different models and determine the best fit for the data. The best-fit model according to AIC is the one that explains greatest amount of variation using the fewest number of attributes. Check this resource to learn more about AIC. Build a model based on AIC using Stepwise AIC regression.Elucidate your observations from the new model. (Hint : Use an appropriate function in olsrr package.)

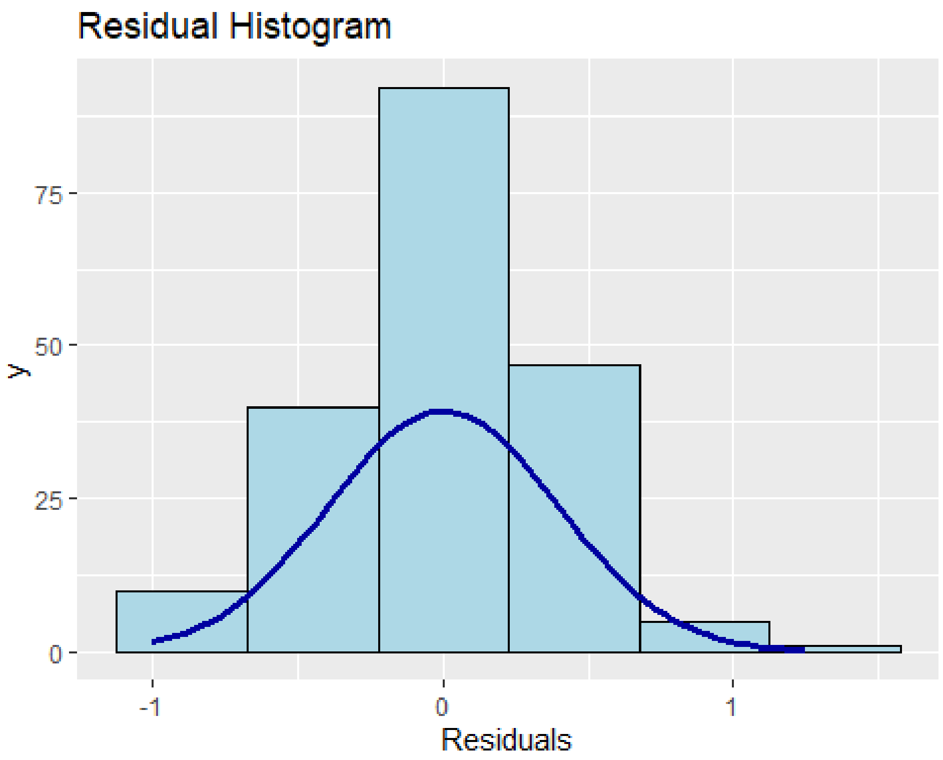
```
stepwise_model<-lm(energy ~ loudness + acousticness + danceability + valence
+ instrumentalness + mode + key , data=df)
summary(stepwise_model)

##
## Call:
## lm(formula = energy ~ loudness + acousticness + danceability +
##     valence + instrumentalness + mode + key, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.05662 -0.24874 -0.01126  0.27930  1.25974
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       9.999e-17  2.900e-02   0.000  1.00000
## loudness          7.075e-01  4.462e-02  15.856  < 2e-16 ***
## acousticness     -3.420e-01  4.005e-02  -8.539 4.63e-15 ***
## danceability     -2.681e-01  5.308e-02  -5.051 1.04e-06 ***
## valence            2.003e-01  3.825e-02   5.238 4.35e-07 ***
## instrumentalness   1.418e-01  5.351e-02   2.650  0.00873 **
## mode              -4.863e-02  2.985e-02  -1.629  0.10491
## key                4.488e-02  2.950e-02   1.521  0.12988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4049 on 187 degrees of freedom
## Multiple R-squared:  0.842,  Adjusted R-squared:  0.8361
## F-statistic: 142.3 on 7 and 187 DF,  p-value: < 2.2e-16
```

###Problem-6 (1 Point) Plot the residuals of the models built till now and comment on it satisfying the assumptions of MLR.

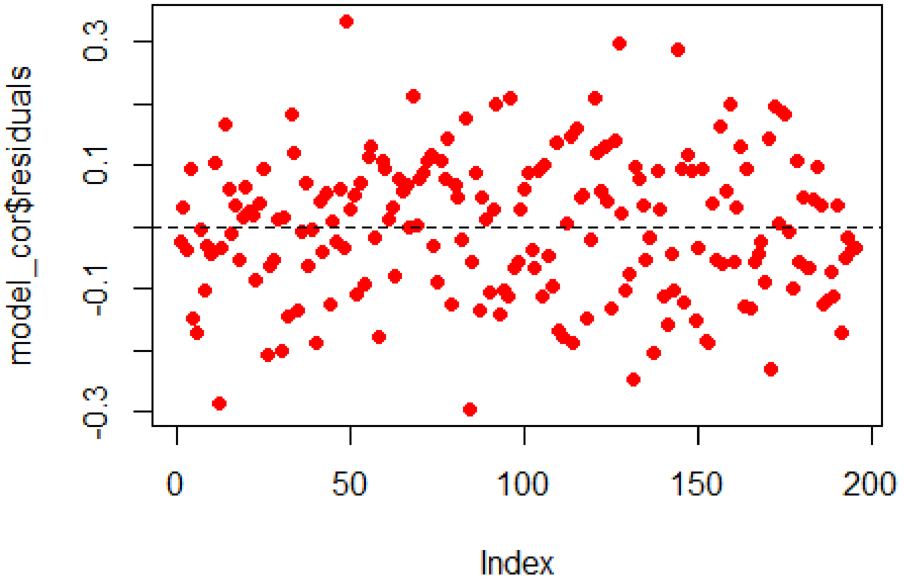
```
plot(full_model$residuals , pch = 16, col="red")
abline(h=0,lty=2)
```



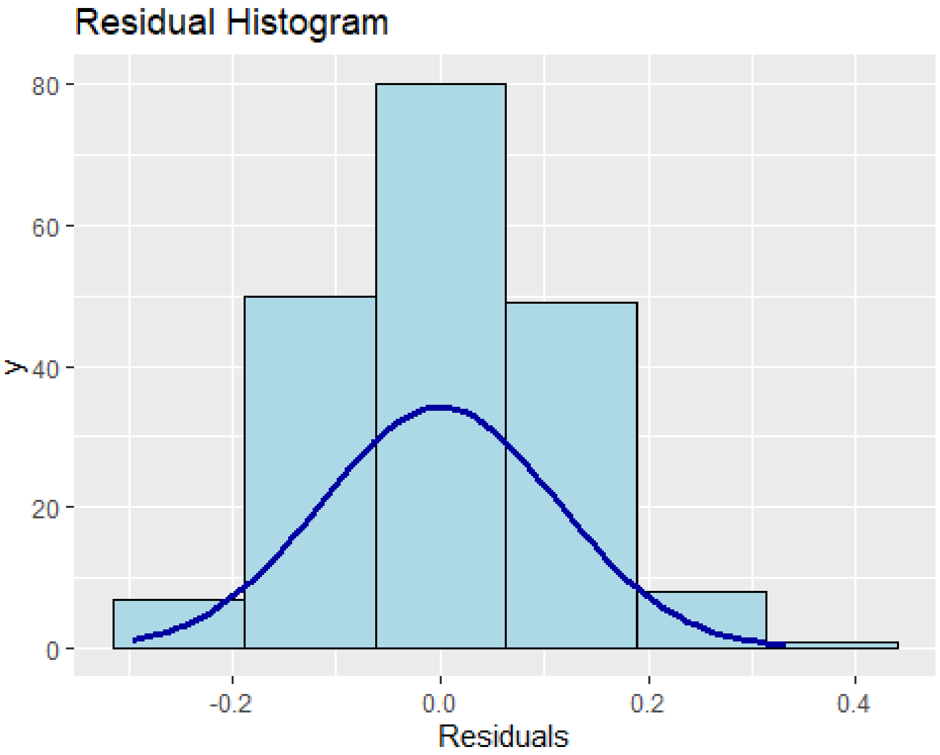
```
ols_plot_resid_hist(full_model)
```



```
plot(model_cor$residuals , pch = 16, col="red")
abline(h=0,lty=2)
```

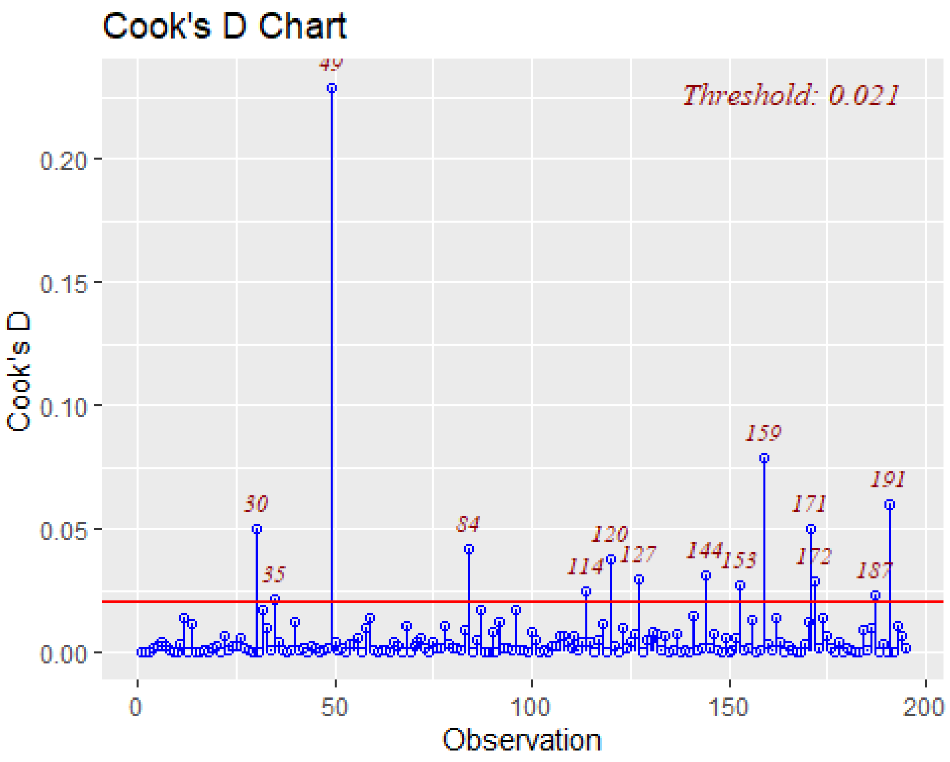


```
ols_plot_resid_hist(model_cor)
```



Problem-7 (2 Points) For the model built in Problem-2 , determine the presence of multicollinearity using VIF. Determine if there are outliers in the data using Cook's Distance. If you find any , remove the outliers and fit the model for Problem-2 and see if the fit improves. [Hint : All the relevant functions can be found in olsrr package. An observation can be termed as an outlier if it has a Cook's distance of more than 4/n where n is the number of records.]

```
ols_vif_tol(full_model)

##           Variables Tolerance      VIF
## 1      danceability 0.2776703 3.601393
## 2               key 0.9467671 1.056226
## 3          loudness 0.4119898 2.427245
## 4              mode 0.9308390 1.074300
## 5       speechiness 0.6921660 1.444740
## 6      acousticness 0.5009458 1.996224
## 7  instrumentalness 0.2755568 3.629016
## 8          liveness 0.8914397 1.121781
## 9           valence 0.5680642 1.760364
## 10            tempo 0.7892957 1.266952
## 11      duration_ms 0.7855373 1.273014
## 12   time_signature 0.8262918 1.210226

cooks <- ols_plot_cooksd_chart(full_model)
```



```
new_df<-df[-c(30,35,49,84,114,120,127,144,153,159,171,172,187,191),]
new_full_model<-lm(energy~.,data=new_df)
summary(new_full_model)

##
## Call:
## lm(formula = energy ~ ., data = new_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.76364 -0.20836  0.01581  0.23506  0.95145
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.001128   0.025283  -0.045 0.964458
## danceability     -0.258483   0.052291  -4.943 1.85e-06 ***
## key               0.088181   0.026094   3.379 0.000903 ***
## loudness          0.838411   0.045399  18.468  < 2e-16 ***
## mode             -0.012666   0.026559  -0.477 0.634036
## speechiness      -0.004528   0.032087  -0.141 0.887947
## acousticness     -0.280188   0.037293  -7.513 3.26e-12 ***
## instrumentalness  0.199483   0.051442   3.878 0.000151 ***
## liveness          0.028416   0.027232   1.043 0.298230
## valence           0.187216   0.033329   5.617 7.90e-08 ***
## tempo            -0.018193   0.029627  -0.614 0.540008
## duration_ms      -0.059788   0.028685  -2.084 0.038647 *
## time_signature    0.036680   0.028430   1.290 0.198761
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.337 on 168 degrees of freedom
## Multiple R-squared:  0.8778, Adjusted R-squared:  0.8691
## F-statistic: 100.6 on 12 and 168 DF,  p-value: < 2.2e-16
```