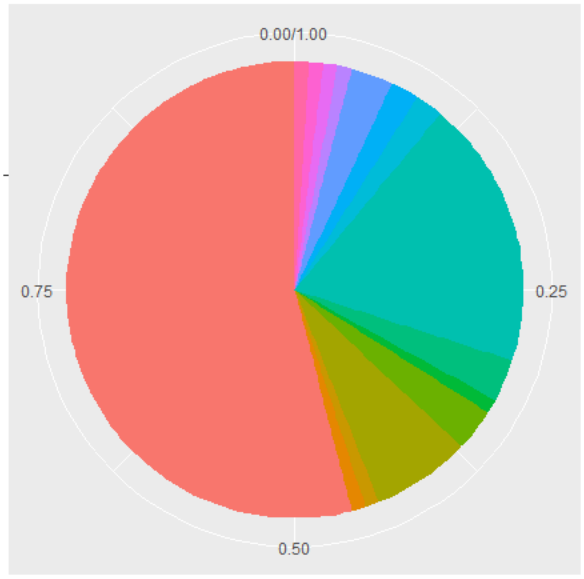
Bivariate analysis

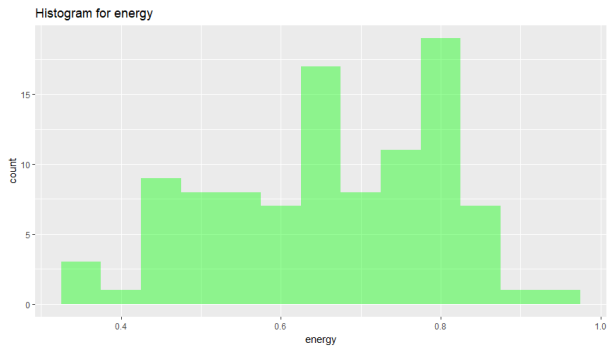
Spotify can share with you informations on audio features of your favourites songs. We use data of 100 samples to determine how is speechiness affected by other factors.

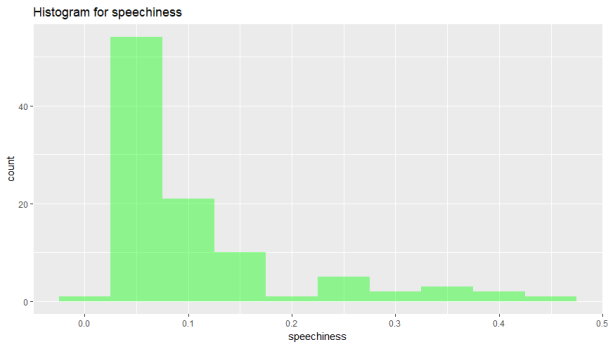
Key	Value Type	Value Description
acousticness	float	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
danceability	float	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
duration_ms	int	The duration of the track in milliseconds.
energy	float	Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.
instrumentalness	float	Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.
liveness	float	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
loudness	float	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typical range between -60 and 0 db.
speechiness	float	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.
tempo	float	The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.
valence	float	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

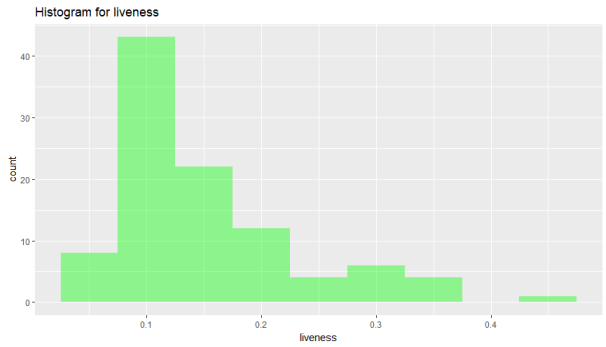


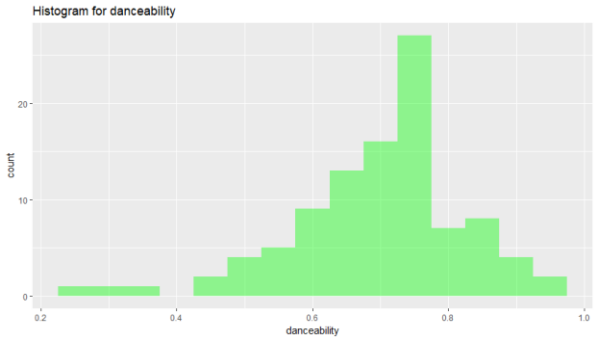
Artists were mainly of American nationality.

There were also quite a few tracks from United Kingdom.









Just from the histograms we can deduct, that high speechiness corresponds to low danceability and high liveness. Also songs from sample are not on average very 'speechy' .

variable	mean[%]	median[%]	1st quartile[%]	3rd quartile[%]	range[%]
speechiness	10.39	6.26	4.31	12.3	2.32-43.1
danceability	69.7	71.4	63.5	77	25.8-92.7
liveness	15	12.5	9.8	17.9	4.2 - 44
energy	66	66.7	55.6	78.7	34.6-93.2

Songs were on average quite energetic, but not that lively. Average danceability was very high, but lower than median, so it suggest the data was shifted to the right, more 'dancable' side. Speechiness was overall very low, despite the huge range.

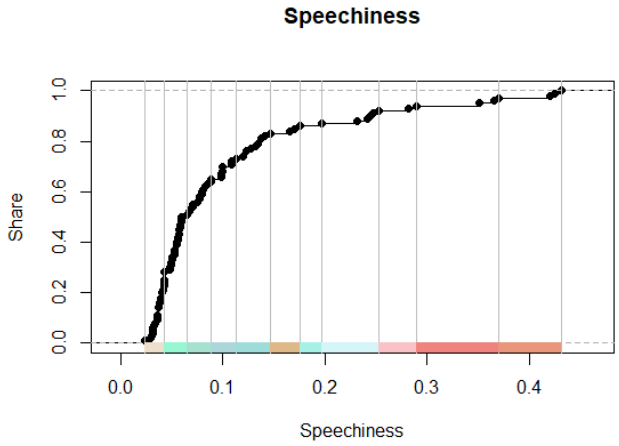
variable	skewness	kurtosis	Pearson coefficient	interquartile skewness	interquartile kurtosis
speechiness	1.94	3.15	0.71	0.51	0.19
danceability	-0.86	1.29	-0.37	-0.17	0.22
liveness	1.35	1.47	0.58	0.34	0.2
energy	-0.32	-0.89	0.25	0.04	0.31

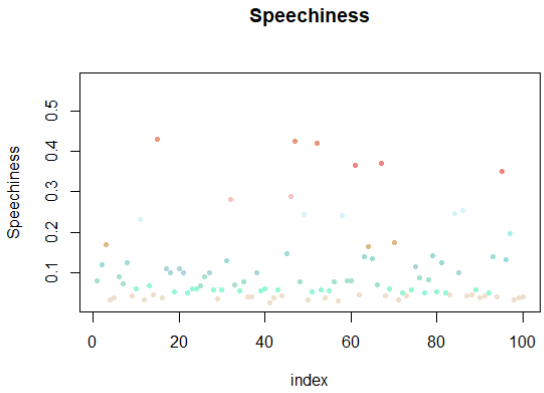
Skewness of variables differ significantly. It is very high for speechiness and liveness – a sign that the data is more on the left side – less 'speechy' and less lively.

Kurtosis for the danceability and liveness is around 1.5 – data is more spiky than the normal distribution. It is very high for speechiness – hence a great aggregation of data on the left side. Value of kurtosis differ for energy – it is below 0, so the shape of the distribution is more flat than a normal distribution.

Why did we choose speechiness? It is an indicator between 0 to 1 showing us how much words is being used in a particular track (most songs are in between 0,33 and 0,66 – average). We thought that this is the only indicator that would suit and won't be obvious to any type of music (as for instrumentalness or tempo for that matter – the first one would be high in a folk song, low in a latin song – and the other one vice versa).

So lets see how speechness compare to other variables, but firstly lets tabulate it to enable analysis.

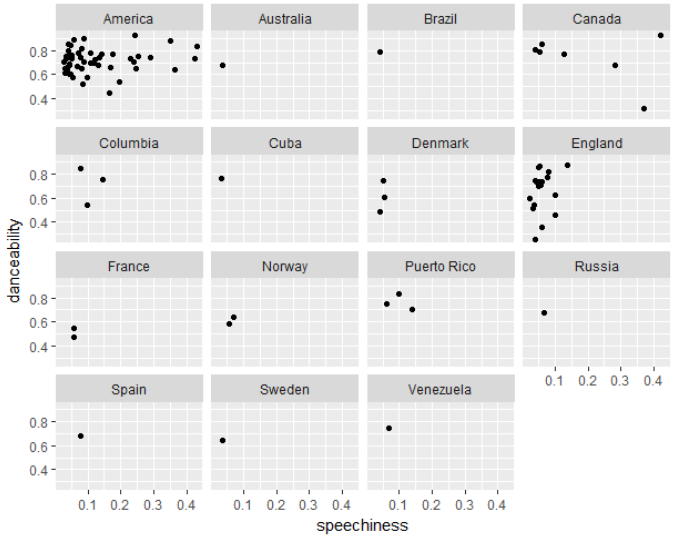


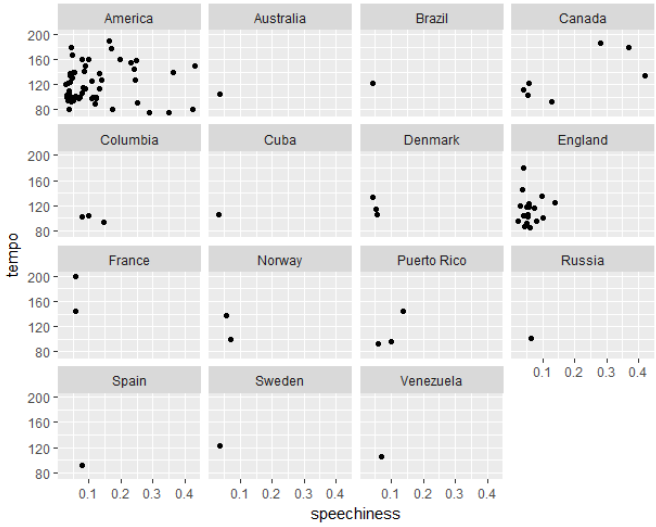


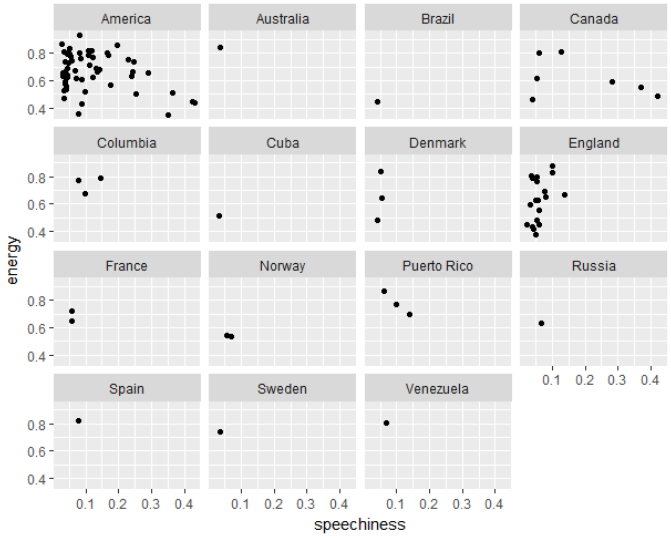
Jenks test:

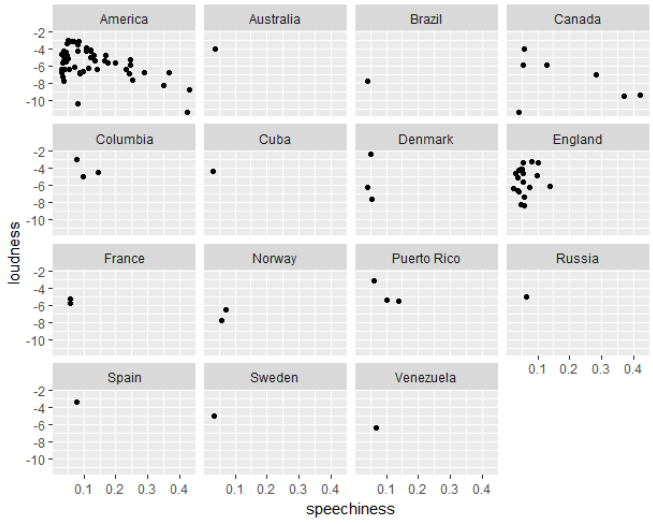
# classes	Goodness of fit	Tabular accuracy
11.0000000	0.9963582	0.9312906

Our variable – speechiness is quite interesting if we look at it in different countries. In England speechness is always low, despite changes in danceability, energy and tempo. We can also observe that in America danceability is very high and speechness differ. If we look at the loudness of American songs – it drops as the speechiness rises. Overall speechness is high only in American and Canadian songs. In Colombia tempo is very low, while the loudness is high.





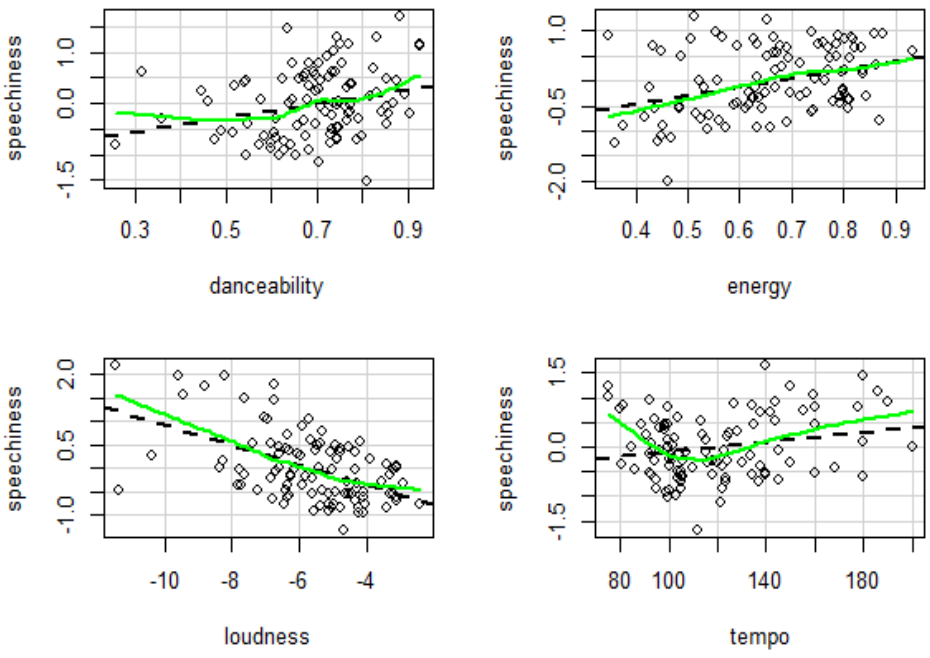




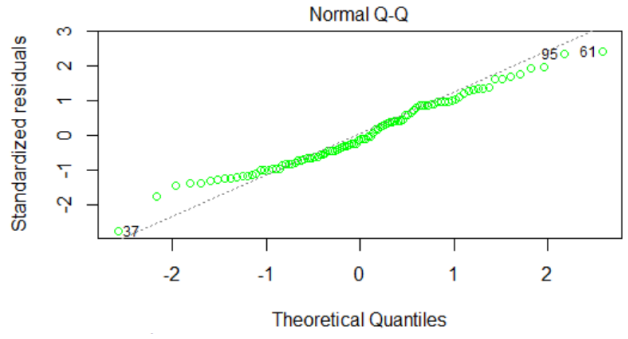
Regression analysis – linear model

While creating this analysis we decided to show you how speechiness (an estimated amount of spoken word in a particular track) is influenced by other music indicators (danceability, energy, loudness and tempo). First we tried using raw (linear) indicator of speechness and it went particularly well (two of our variables turned out to be meaningful – danceability and loudness) and seeing this we decided to take one more shoot and use an logarithmic indicator to hopefully make our results even better. It was a good decision, since one more indicator became significant – energy. Then we had a few more or less successful attempts, but lastly decided to stay with this one. We picked four the most suitable as well as the most interesting indicators (those ones with the smallest p-value) and created a model for you.We performed an regression analysis and, by transforming our main data into logarithmic on and using loops, fitted the best linear model to it:
(formula = log(speechiness) ~ danceability + energy + loudness + tempo)

And those are our results:

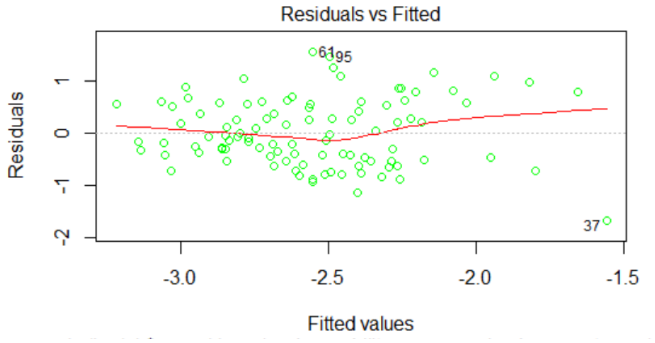


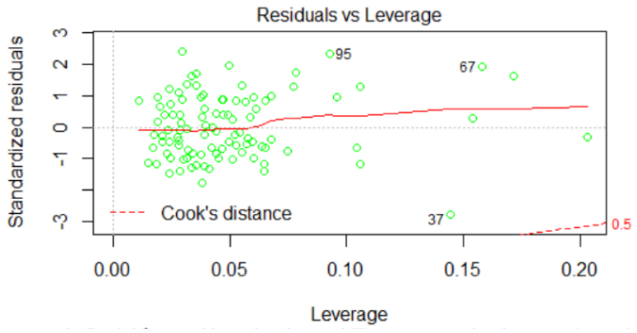
Before we decided to use that model, we firstly checked if it is reliable by plotting it.



As we can see, residual values are normally distributed, so we can say that our dependent variable (speechiness) is normally distributed as well.

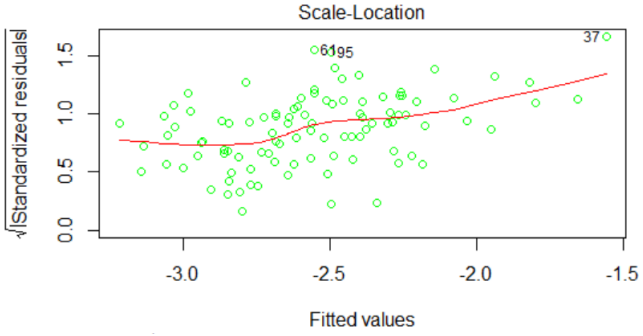
Speechiness is linearly correlated with independent variables because there is no systematic relationship between the residuals and the predicted (fitted) value.



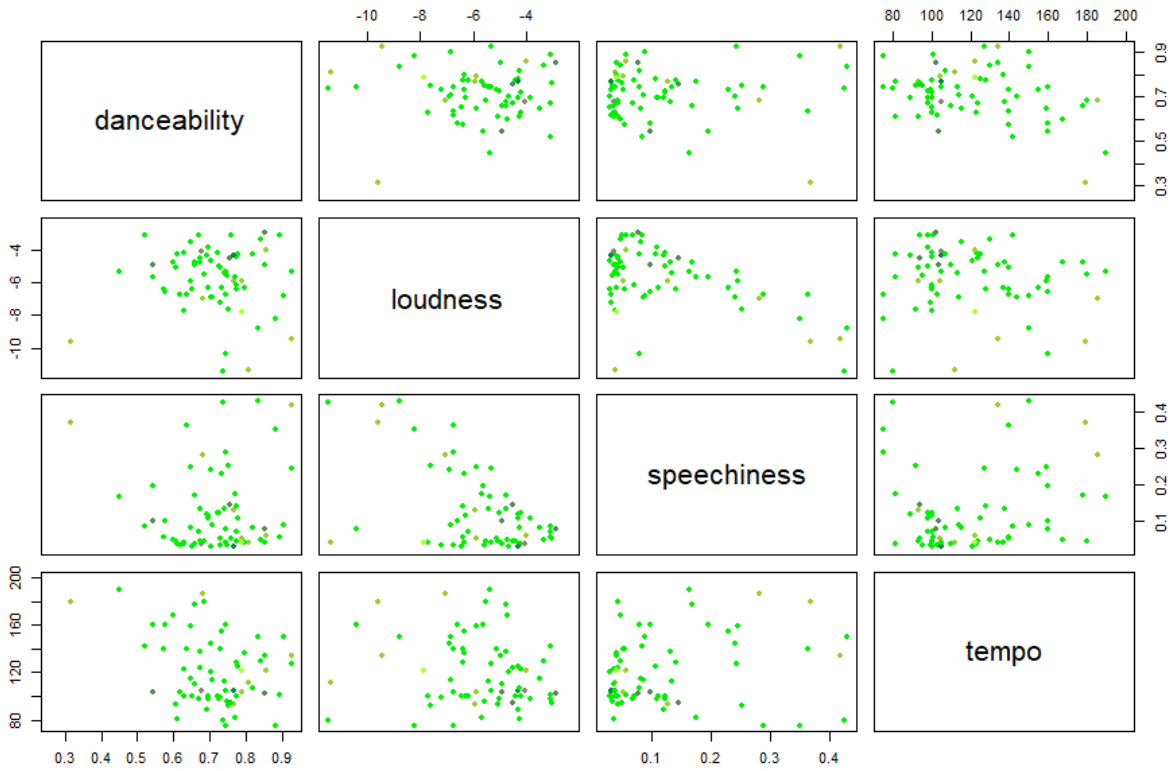


From this plot we can identify that there are three outliers (observation number 67, 95 and 37), that have quite big relative absolute residuals. There is also a few observations with high leverage value, so they have unusual combination of predictor values. All variables have similar Cook' s distance so they are all quite influential (have impact on model parameters).

Assumption of homoscedasticity is satisfied, because we can see from the plot that the variance is constant.



We also checked correlation:



Here we can see how every variable used in our model is correlated with another. Because of how sparse the data is, we can conclude that they are not intercorrelated.

	danceability	energy	loudness	tempo
correlation	0.11	-0.2	-0.44	0.17

As we predicted, the highest correlation is with loudness. For the other variables correlation is not that high.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-6.417785	0.887323	-7.233	1.20E-10
danceability	1.395326	0.564824	2.47	0.01528
energy	1.690702	0.682121	2.479	0.014954
loudness	-0.211701	0.053101	-3.987	0.000131
tempo	0.004822	0.002569	1.877	0.06362

As we can see, our formula would look like this:

Log(speechiness) = -6.417785 + 1.395326*danceability + 1.690702*energy – 0.211701*loudness + 0.004822*tempo

The output shows that F = 6.358 (p =0.000141), indicating that we should clearly reject the null hypothesis that the variables danceability, energy, loudness and tempo collectively have no effect on speechiness. The results also show that the variable loudness is significant controlling for the variable danceability
(p = 0.000131), as is danceability controlling (not as much bus still) for the variable loudness (p=0.015280).
Also the energy variable (since we used logarithm on the speechiness variable) look significant (p=0.014954).
In addition, the output also shows that R 2 = 0.2669 and R 2 adjusted = 0.236.The conclusion is that loudness has the biggest effect on speechiness, although the whole variable it is with the "-" sign, so we can expect that the louder the song would get, the more words will be used by the artists (0,21 times). Also the energy (1.6) and danceability(1.3) seems to increase along with the speechiness which is a little surprising for us, but we assume people just like to dance and sing along ☺.

Authors: Agnieszka Szlendak, Anna Przybycien

Source: https://www.kaggle.com/geomack/spotifyclassification/data

Code:

```r
setwd("C:/Users/aniap/Documents/super-duper-octo-potato")
data<-read.csv("songs with countries.csv", sep=";", dec=".",header = TRUE)
data<-data[,1:15]
attach(data)
names(data)
coutries<-as.data.frame(country)

speechiness<-as.data.frame(speechiness)

countries_tab<-data.frame(table(coutries))
countries_tab[2]

combained<-data.frame(coutries,speechiness)
pcol <- c('antiquewhite2', 'aquamarine', 'azure3', 'cadetblue2', 'burlywood',
          'darkslategray1', 'lavenderblush', 'lightcoral','darksalmon')
library(ggplot2)

ggplot(transform(transform(countries_tab, Freq=Freq/sum(Freq)), labPos=cumsum(Freq)-Freq/2),
       aes(x="", y = Freq, fill = coutries)) +
  geom_bar(width = 1, stat = "identity") +
 # scale_fill_manual(values=pcol) +
  coord_polar(theta = "y") +
  labs(title = "",x="",y="",fill="Countries")

ggplot(as.data.frame(countries_tab), aes(coutries,fill=factor(coutries))) +
  geom_bar(width = 1) +
  coord_polar(theta = "y") +
  labs(x="Opinie",y="Liczba odpowiedzi",fill="Opinie",title="Zadowolenie ze swojego małżeństwa")+
  theme(axis.title.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.x=element_blank())# +
 # scale_fill_manual(values=pcol)+ scale_x_discrete(limits = kolejnosc)
scatterplot(speechiness ~ , data=data,
            xlab="Weight of Car", ylab="Miles Per Gallon",
            main="Enhanced Scatter Plot",
            labels=row.names(data))
```

```r
qplot(data=data,x=data$speechiness, y=data$danceability,
      log = "xy", color=data$danceability)
qplot(data=data,x=data$danceability,y=data$speechiness, geom = "boxplot")
attach(data)
qplot(data=data,x=speechiness,y=loudness,
      facets =~data$country)

ggplot(data, aes(x=data$speechiness, y=data$danceability)) +
  geom_point(shape=1) +    # Use hollow circles
  geom_smooth(method=lm)   # Add linear regression line
#  (by default includes 95% confidence region)

ggplot(data, aes(x=data$danceability, y=data$speechiness, fill=factor(data$key))) +
  geom_point(shape=1) +
  scale_colour_hue(l=50) + # Use a slightly darker palette than normal
  geom_smooth(method=lm,   # Add linear regression lines
              se=FALSE)    # Don't add shaded confidence region


tab<-classIntervals(data$speechiness, style = 'jenks', n=11, intervalClosure = 'right')
tab
jenks.tests(tab)
color_code<-findColours(tab,pcol)
plot(tab, pal=pcol , main='Speechiness', xlab='Speechiness', ylab='Share')
plot(data$speechiness, pch=20, col=color_code, xlab='index', ylab='Speechiness', main='Speechiness',
     ylim=c(min(data$speechines), max(data$speechines)*4/3 ))
plot(density(data$speechiness),col="darksalmon",main="Dochody[PLN]",xlab="Dochody[PLN]")

library(psych)

mod<-function(x){
  ux<-unique(x)
  ux[which.max(tabulate(match(x,ux)))]
}
```

```r
IQR<-function(x){
  return(quantile(x,0.75)-quantile(x,0.25))
}
cv<-function(v){return(sd(v)/mean(v))}
variability.fun <- function(x) {
  c(Qx=Qx(x),Vx=Vx(x),IQR = IQR(x), mode = mod(x), var = var(x), cv = cv(x))
}

Qx<-function(v){return((quantile(v,0.75)-quantile(v,0.25))/2)}
Vx<-function(v){return(100*(Qx(v)/quantile(v,0.5)))}

pearson<-function(v){return((mean(v)-mod(v))/sd(v))}
skewnessIQR<-function(v){
  return(((quantile(v,0.75)-median(v,na.rm = FALSE))-(median(v,na-rm=FALSE)-quantile(v,0.25)))/
           + ((quantile(v,0.75)-median(v,na.rm = FALSE))+(median(v,na.rm=FALSE)-quantile(v,0.25))))
}
skewnessIQR<-function(v){
  return(((quantile(v,0.75)-median(v))-(median(v)-quantile(v,0.25)))/
           + ((quantile(v,0.75)-median(v))+(median(v)-quantile(v,0.25))))
}
kurtosisIQR<-function(v){
  return( (quantile(v,0.75)-quantile(v,0.25))/
            + (2*(quantile(v,0.9)-quantile(v,0.1))) )
}
skewkurtosi.fun<-function(x){
  c(skewness=skew(x),kurtosis=kurtosi(x),pearson=pearson(x),skewnessIQR=skewnessIQR(x),
    kurtosisIQR=kurtosisIQR(x))
}
robust.fun<-function(x){
  c(meanAD=mad(x,center=mean(x),na.rm=FALSE),medianAD=mad(x,center=median(x),na.rm=FALSE),
    trimmean=mean(x,trim=0.05),
    winsormean=winsor.mean(x,trim=0.1),winsorsd=winsor.sd(x,trim=0.1))
}

summary(data$speechiness)
variability.fun(data$speechiness)
skewkurtosi.fun(data$speechiness)
robust.fun(data$speechiness)
```

```r
IQR<-function(x){
  return(quantile(x,0.75)-quantile(x,0.25))
}
cv<-function(v){return(sd(v)/mean(v))}
variability.fun <- function(x) {
  c(Qx=Qx(x),Vx=Vx(x),IQR = IQR(x), mode = mod(x), var = var(x), cv = cv(x))
}

Qx<-function(v){return((quantile(v,0.75)-quantile(v,0.25))/2)}
Vx<-function(v){return(100*(Qx(v)/quantile(v,0.5)))}

pearson<-function(v){return((mean(v)-mod(v))/sd(v))}
skewnessIQR<-function(v){
  return(((quantile(v,0.75)-median(v,na.rm = FALSE))-(median(v,na-rm=FALSE)-quantile(v,0.25)))/
           + ((quantile(v,0.75)-median(v,na.rm = FALSE))+(median(v,na.rm=FALSE)-quantile(v,0.25))))
}
skewnessIQR<-function(v){
  return(((quantile(v,0.75)-median(v))-(median(v)-quantile(v,0.25)))/
           + ((quantile(v,0.75)-median(v))+(median(v)-quantile(v,0.25))))
}
kurtosisIQR<-function(v){
  return( (quantile(v,0.75)-quantile(v,0.25))/
            + (2*(quantile(v,0.9)-quantile(v,0.1))) )
}
skewkurtosi.fun<-function(x){
  c(skewness=skew(x),kurtosis=kurtosi(x),pearson=pearson(x),skewnessIQR=skewnessIQR(x),
    kurtosisIQR=kurtosisIQR(x))
}
robust.fun<-function(x){
  c(meanAD=mad(x,center=mean(x),na.rm=FALSE),medianAD=mad(x,center=median(x),na.rm=FALSE),
    trimmean=mean(x,trim=0.05),
    winsormean=winsor.mean(x,trim=0.1),winsorsd=winsor.sd(x,trim=0.1))
}

summary(data$speechiness)
variability.fun(data$speechiness)
skewkurtosi.fun(data$speechiness)
robust.fun(data$speechiness)
```

```r
summary(data$danceability)
variability.fun(data$danceability)
skewkurtosi.fun(data$danceability)
robust.fun(data$danceability)

summary(data$liveness)
variability.fun(data$liveness)
skewkurtosi.fun(data$liveness)
robust.fun(data$liveness)

summary(data$energy)
variability.fun(data$energy)
skewkurtosi.fun(data$energy)
robust.fun(data$energy)

summary(data$loudness)
variability.fun(data$loudness)
skewkurtosi.fun(data$loudness)
robust.fun(data$loudness)

par(mar=c(7,4,5,2))
qplot(data$speechiness, geom="histogram",binwidth = 0.05,
      main = "Histogram for speechiness",
      xlab = "speechiness",
      fill=I("green"),
      alpha=I(.4))
qplot(data$energy, geom="histogram",binwidth = 0.05,
      main = "Histogram for energy",
      xlab = "energy",
      fill=I("green"),
      alpha=I(.4))
qplot(data$liveness, geom="histogram",binwidth = 0.05,
      main = "Histogram for liveness",
      xlab = "liveness",
      fill=I("green"),
      alpha=I(.4))
```

```r
library(classInt)
tab <- classIntervals(spfree, n=11, style='jenks', intervalClosure = 'right');
jenks.tests(tab)
boxplot(speechiness, col='lightblue')

library(np)

qplot(log(data$speechiness), geom="histogram",binwidth = 0.5,
      main = "Histogram for speechiness",
      xlab = "speechiness",
      fill=I("green"),
      alpha=I(.4))

skim<-data[,4:7]
skim<-data.frame(skim,data[,10:15])
skim2<-data.frame(skim[,1:5],skim[,7:10])
full<-lm(log(data$speechiness)~.,data=skim2)
summary(full)
step(full,direction = "both")
full<-lm(formula = log(data$speechiness) ~ danceability + energy +
           loudness + tempo, data = skim2)
#DIAGNOSTICS (we check assumptions)
library(car)
#linearity?
crPlots(full)
#services is boolean (you work - 1, you don't - 0, dumming variable)
qqPlot(full$residuals)
#homoscedasticity?
ncvTest(full)
plot(full)

#not significant p-value,
#p-value = 0.14 - area under right tale of function

#are data correlated (is one residual influencing another?)
#265. - not an outlier, but slighly changing coefficients
#HOMEWORK variance inflation factor VIR (multicolinearinty problem)
#VIF(1/(1-Rj^2))
cor(x=data$loudness, y = data$danceability, use = "everything",#danceability*tempo -0.3
    method = "pearson")
```