Final presentation

Jaejin Kim

5/6/2019

Research Question

Is there a causal relation or correlation between the economic state of a country and the popularity of musics? Would it be possible to detect a latent factor, possible optimism, that affects both economic growth rate and the kinds of music that pepole listen to?

Motivation

We've all heard of the Jazz Age, when Jazz music gained nationwide popularity in the US. Music historians often say that this reflects the unprecedented economic boom and political development of the time. There is also the hippie movement in the 60s and 70s, when psychedelic rock became hugely popular following the outbreak of the Vietnam war, civil rights movement, and the spread of psychedelic drugs. However, we do not hear too much about what happened in the musical trends during the time of, for example, economic depression. Are musical historians making unwarranted claims? If it is true that there is such relation, in a time when we hear more and more about faltering economic conditions and increasing conflicts both internationally and internationally, would angrier or gloomier music gain more popularity? I wanted to study if the commonly made claim that the socio-economic state of a country affects the musical trends.

Process

This was the first individual data analysis research project that I worked on, and also it was the first time I worked with APIs and a large dataset. Initially I spent bulk of time trying to learn how to use APIs. Even after I figured it out, I was faced with another challenge which was trying to figure out how to join two different big dataset using song titles and artists that were often times registered differently in the two datasets. After working on it for few weeks, I took Yoav's advice to move on to analysis. This left me with limited time to explore and analyze the data, but I was able to observe some interesting results.

Attach Pacakges

```
suppressPackageStartupMessages(require(Rspotify))
suppressPackageStartupMessages(require(plyr))
suppressPackageStartupMessages(require(dplyr))
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

```
suppressPackageStartupMessages(require(psych))
suppressPackageStartupMessages(require(devtools))
suppressPackageStartupMessages(require(ggplot2))
suppressPackageStartupMessages(require(openair))
```

```
## Warning: package 'openair' was built under R version 3.5.2
```

```r
suppressPackageStartupMessages(require(data.table))
```

```
## Warning: package 'data.table' was built under R version 3.5.2
```

```r
suppressPackageStartupMessages(require(knitr))
suppressPackageStartupMessages(require(lmtest))
suppressPackageStartupMessages(require(psych))
```

Import Data

```r
billboard <- read.csv("../Data/Raw/Hot100.csv")
```

Data Cleaning

I had to

```r
billboard$WeekID <- as.Date(billboard$WeekID, format="%m/%d/%Y")
billboard$Performer <- sub(pattern="Featuring", replacement="feat\\.", x=billboard$Performer)

#To match Rspotify syntax
colnames(billboard)[colnames(billboard)=="Song"] <- "title"
colnames(billboard)[colnames(billboard)=="Performer"] <- "artist"

#To match Spotify format
##Remove "feat." in the artist column
billboard$artist_spotify <- stringr::str_replace_all(billboard$artist, " feat.*", "")
##Remove paranthesis in the front and back
billboard[billboard$id==0,]$title <- stringr::str_replace_all(billboard[billboard$id==0,]$title, " \\((
billboard[billboard$id==0,]$title <- stringr::str_replace_all(billboard[billboard$id==0,]$title, "\\((.

#To use Rspotify
billboard$title <- as.character(billboard$title)
billboard$artist <- as.character(billboard$artist)
billboard$artist_spotify <- as.character(billboard$artist_spotify)
```

Requesting Spotify music features and joining it to Billboard data

Below is the helper function that I created, and ran. This went through each and every rows of the Billboard data, which took hours to run. Looking back, I think I could have made it more efficient by making it skip rows that are identical to ones before and copy the music features.

```r
#get_spotify_features= function(track){
#  songs <- try(searchTrack(track, keys),silent = T)
#  if (class(songs)=="try-error"){
#    return(rep(0,length(features_name)))
#  }
#  else{
#  songs <- arrange(songs, desc(popularity))[1,]
#  song_id=songs[,"id"][1]
#  features=getFeatures(song_id, keys)
#  return(features)
```

```
#  }
#}

#for(i in 1:nrow(billboard)){
#   billboard[i,11:26] <-  get_spotify_features(track = billboard$title[i])
#}


### Update the data for the failed IDs
#failed_ids = which(billboard$id =="0")

#for(i in failed_ids){
#  billboard[i,17:32]=  get_spotify_features1(track = billboard$track[i],billboard$artist[i])
#  print(i)
#}
```

Data Exploration

Load dataset and format date

```
billboard <- read.csv("../Data/Clean/Billboard_spotify.csv")
billboard$time <- as.Date(billboard$time)
```

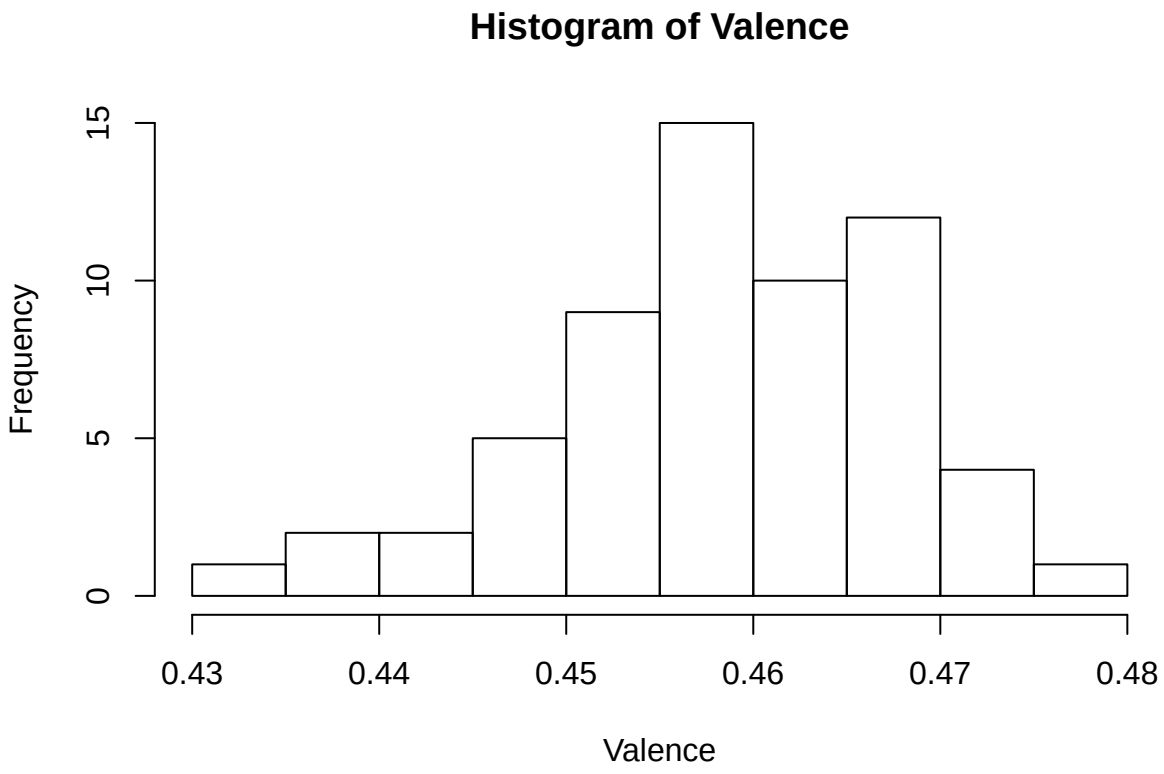
Extract data to visualize the trend in music features

```
#total danceability
total.dance<-aggregate(list(danceability = billboard$danceability),
                           list(time = cut(billboard$time, "1 year")),
                           mean)
total.dance$time <- as.Date(total.dance$time)

#total valence
total.val<-aggregate(list(valence = billboard$valence),
                           list(time = cut(billboard$time, "1 year")),
                           mean)
total.val$time <- as.Date(total.val$time)

#total energy
total.energy<-aggregate(list(energy = billboard$energy),
                           list(time = cut(billboard$time, "1 year")),
                           mean)
total.energy$time <- as.Date(total.energy$time)

#total loudness
total.loud<-aggregate(list(loudness = billboard$loudness),
                           list(time = cut(billboard$time, "1 year")),
                           mean)
total.loud$time <- as.Date(total.loud$time)

#total tempo
total.tempo<-aggregate(list(tempo = billboard$tempo),
```

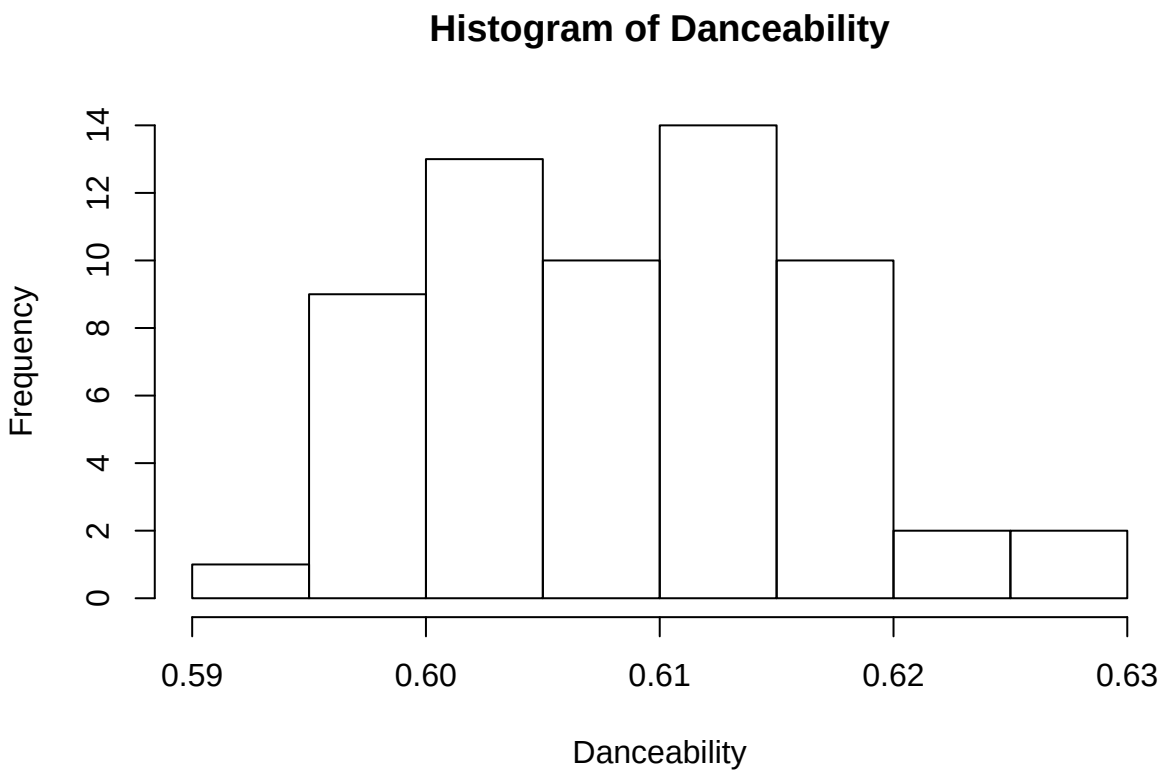
```
                            list(time = cut(billboard$time, "1 year")),
                            mean)
total.tempo$time <- as.Date(total.tempo$time)
```

Distribution of music features

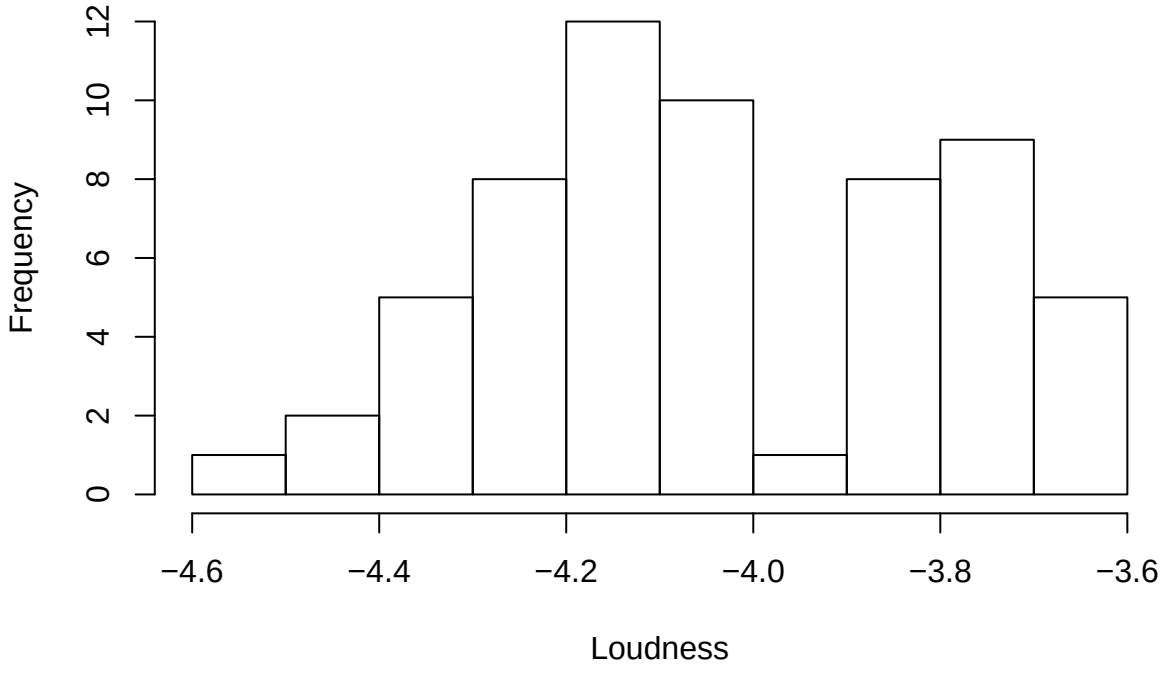
```
hist(total.val$valence, main = "Histogram of Valence", xlab = "Valence")
```



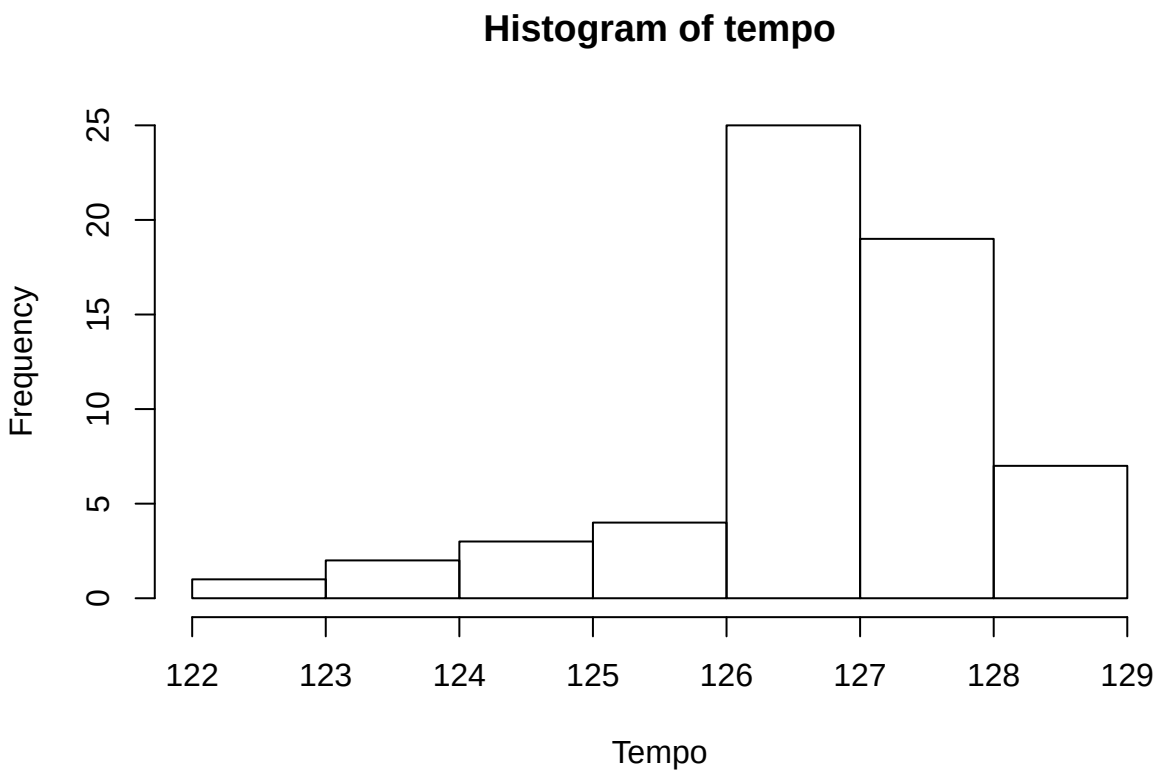

```
hist(total.dance$danceability, main = "Histogram of Danceability", xlab = "Danceability")
```



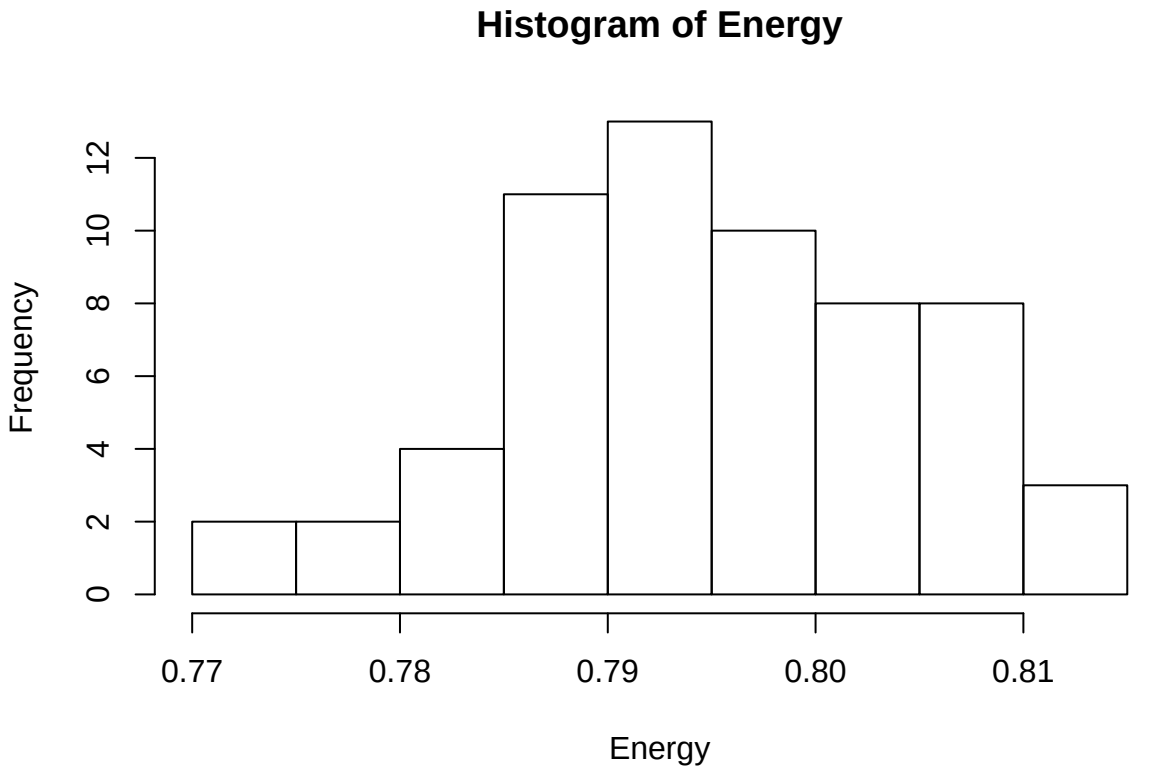

```
hist(total.loud$loudness, main = "Histogram of Loudness", xlab = "Loudness")
```




```
hist(total.tempo$tempo, main = "Histogram of tempo", xlab = "Tempo")
```

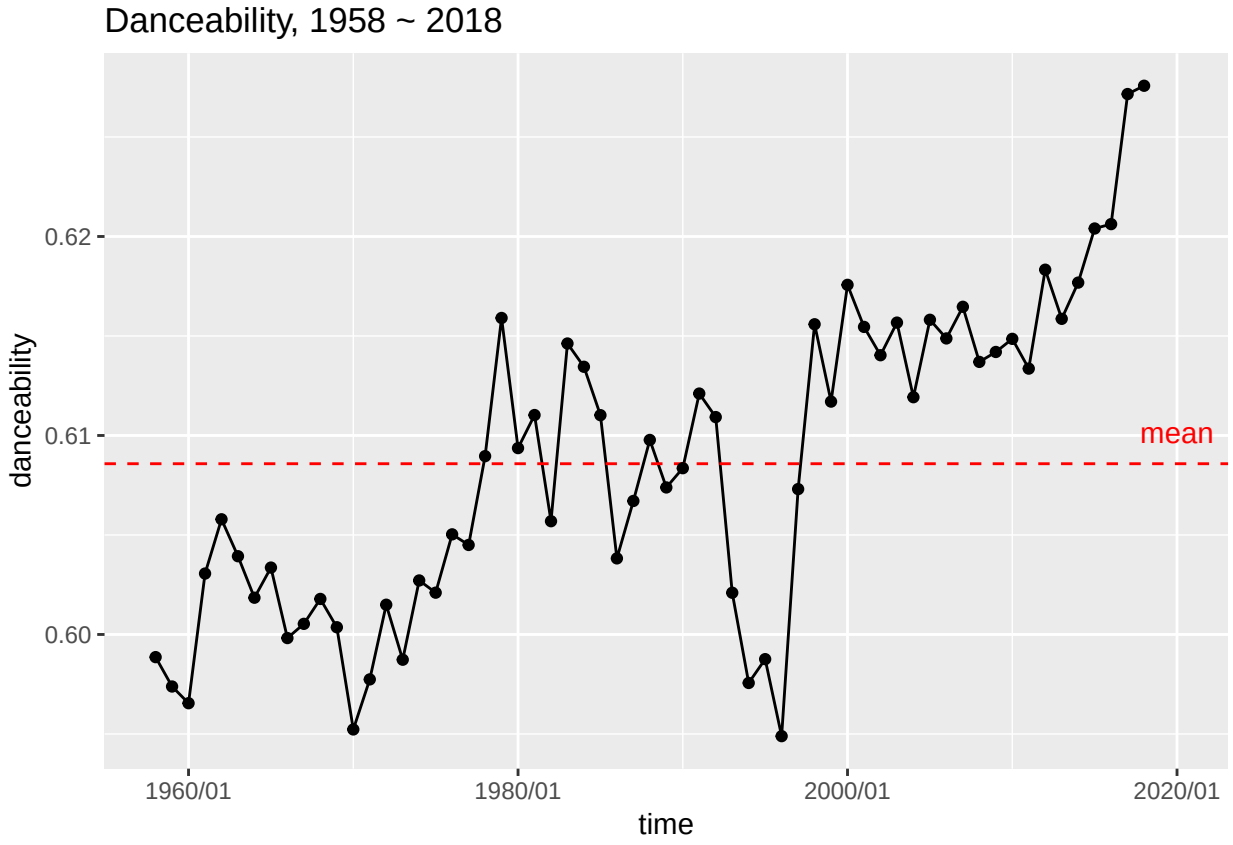



```
hist(total.energy$energy, main = "Histogram of Energy", xlab = "Energy")
```

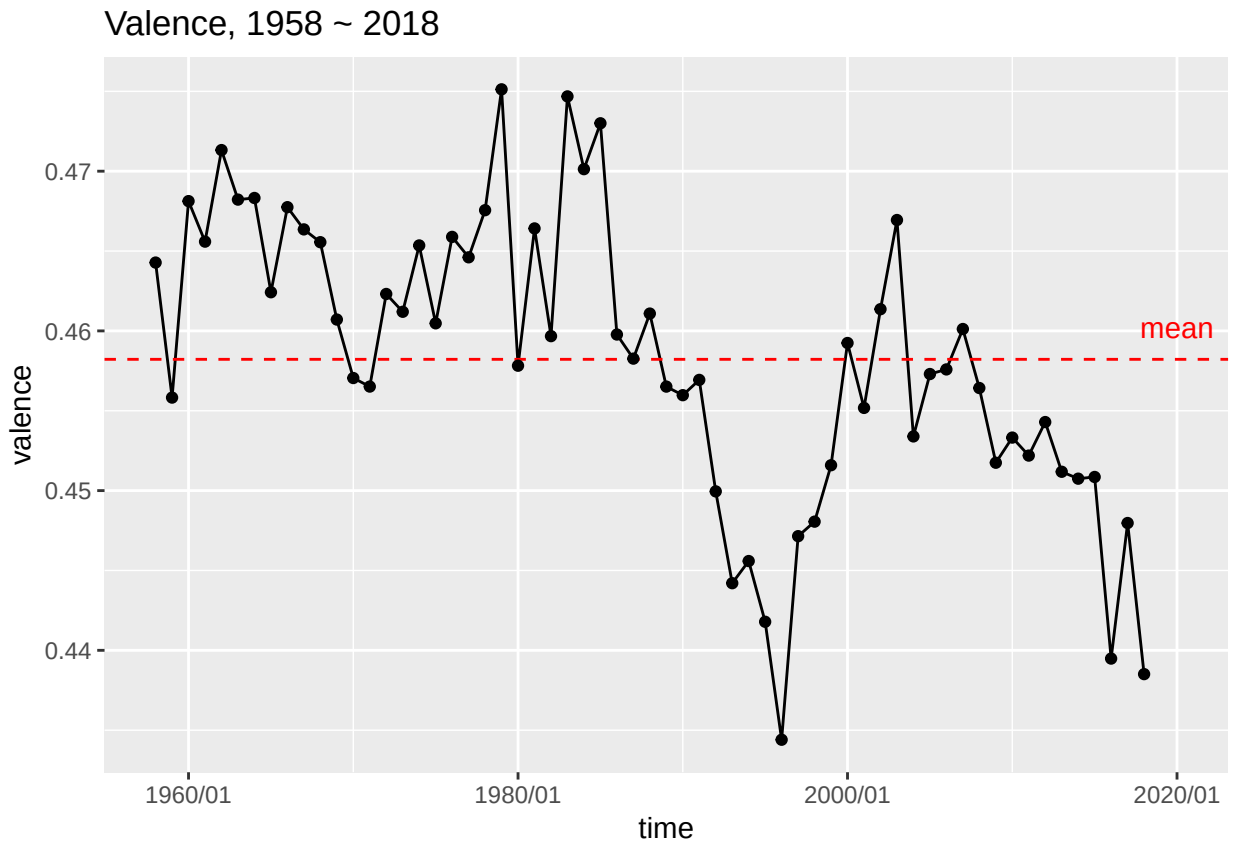


Trends in music features

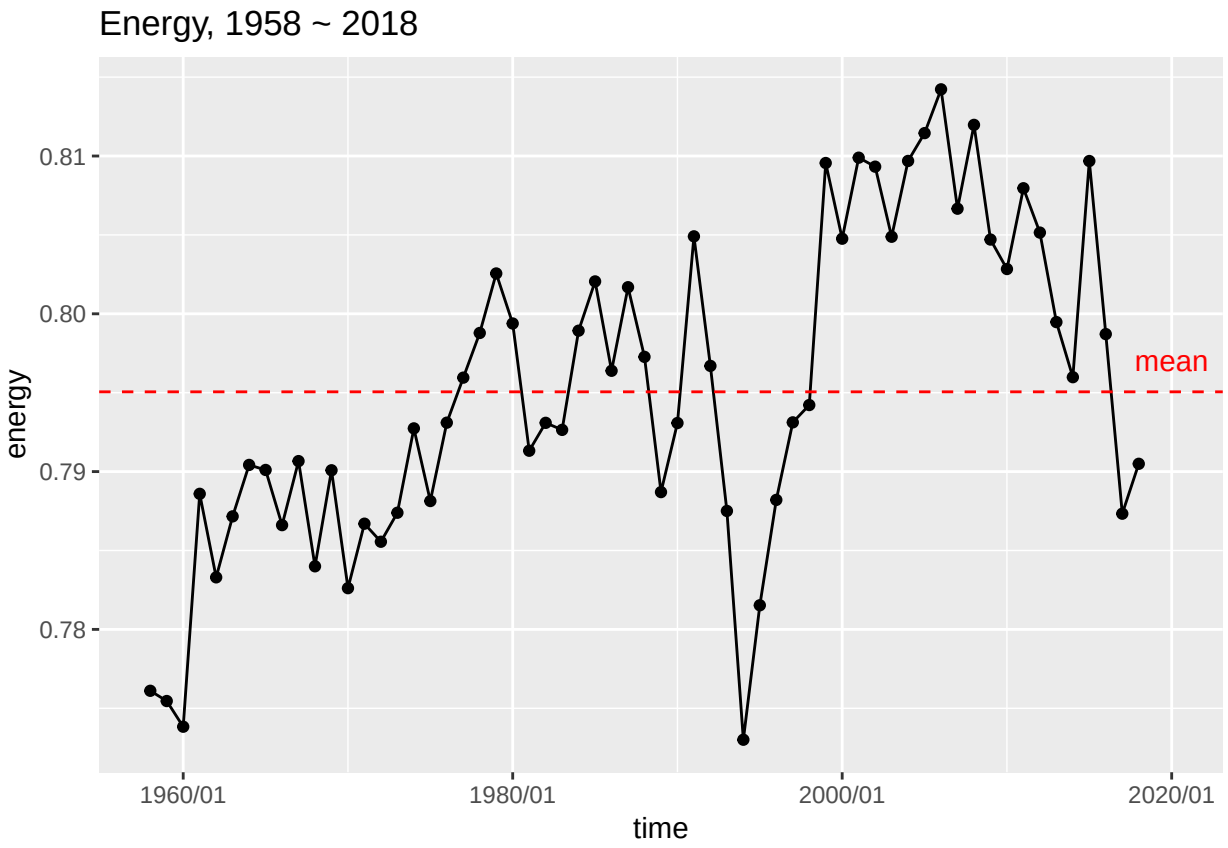
```
#plot danceability
ggplot(data=total.dance, aes(x=time, y=danceability)) +
  geom_line() +
  geom_hline(yintercept = mean(total.dance$danceability), linetype = "dashed", color ="red") +
  annotate(geom="text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(total.dance$danceability),
           vjust = -1, color = "red") +
  geom_point() +
  ggtitle("Danceability, 1958 ~ 2018") +
  scale_x_date(date_labels = '%Y/%m')
```



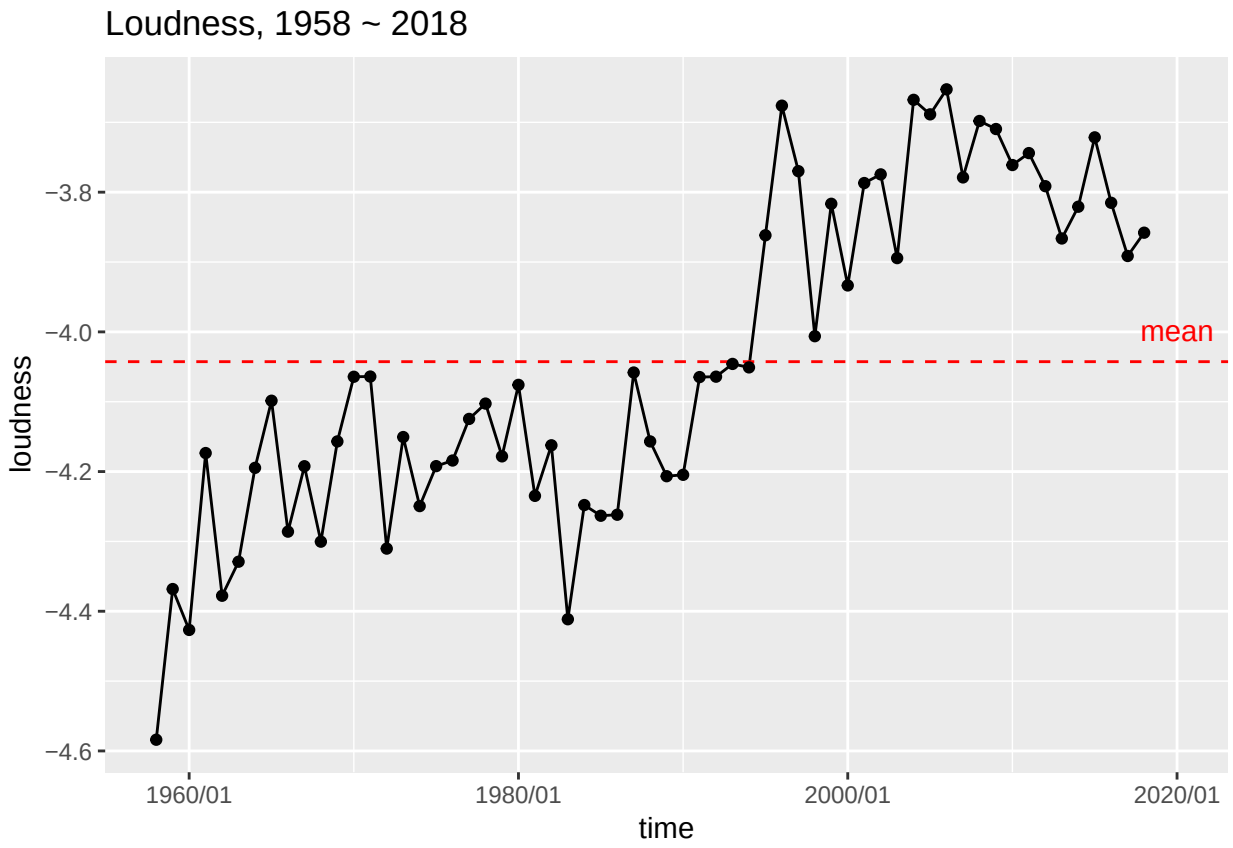

```
#plot valence
ggplot(data=total.val, aes(x=time, y=valence)) +
  geom_line() +
  geom_hline(yintercept = mean(total.val$valence), linetype = "dashed", color ="red") +
  annotate(geom="text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(total.val$valence),
           vjust = -1, color = "red") +
  geom_point() +
  ggtitle("Valence, 1958 ~ 2018") +
  scale_x_date(date_labels = '%Y/%m')
```



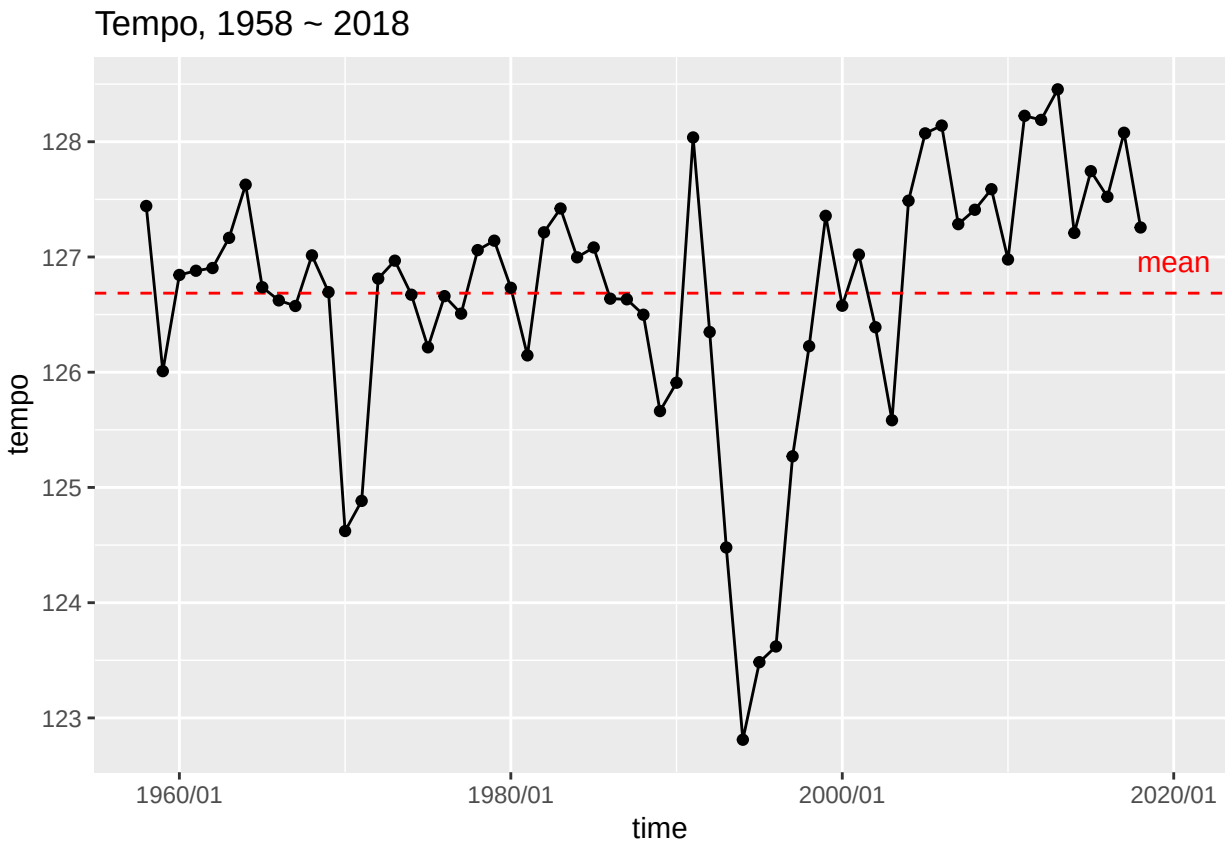

```
#plot energy
ggplot(data=total.energy, aes(x=time, y=energy)) +
  geom_line() +
  geom_hline(yintercept = mean(total.energy$energy), linetype = "dashed", color ="red") +
  annotate(geom="text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(total.energy$energy),
           vjust = -1, color = "red") +
  geom_point() +
  ggtitle("Energy, 1958 ~ 2018") +
  scale_x_date(date_labels = '%Y/%m')
```




```
#plot loudness
ggplot(data=total.loud, aes(x=time, y=loudness)) +
  geom_line() +
  geom_hline(yintercept = mean(total.loud$loudness), linetype = "dashed", color ="red") +
  annotate(geom = "text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(total.loud$loudness),
           vjust = -1, color = "red") +
  geom_point() +
  ggtitle("Loudness, 1958 ~ 2018") +
  scale_x_date(date_labels = '%Y/%m')
```

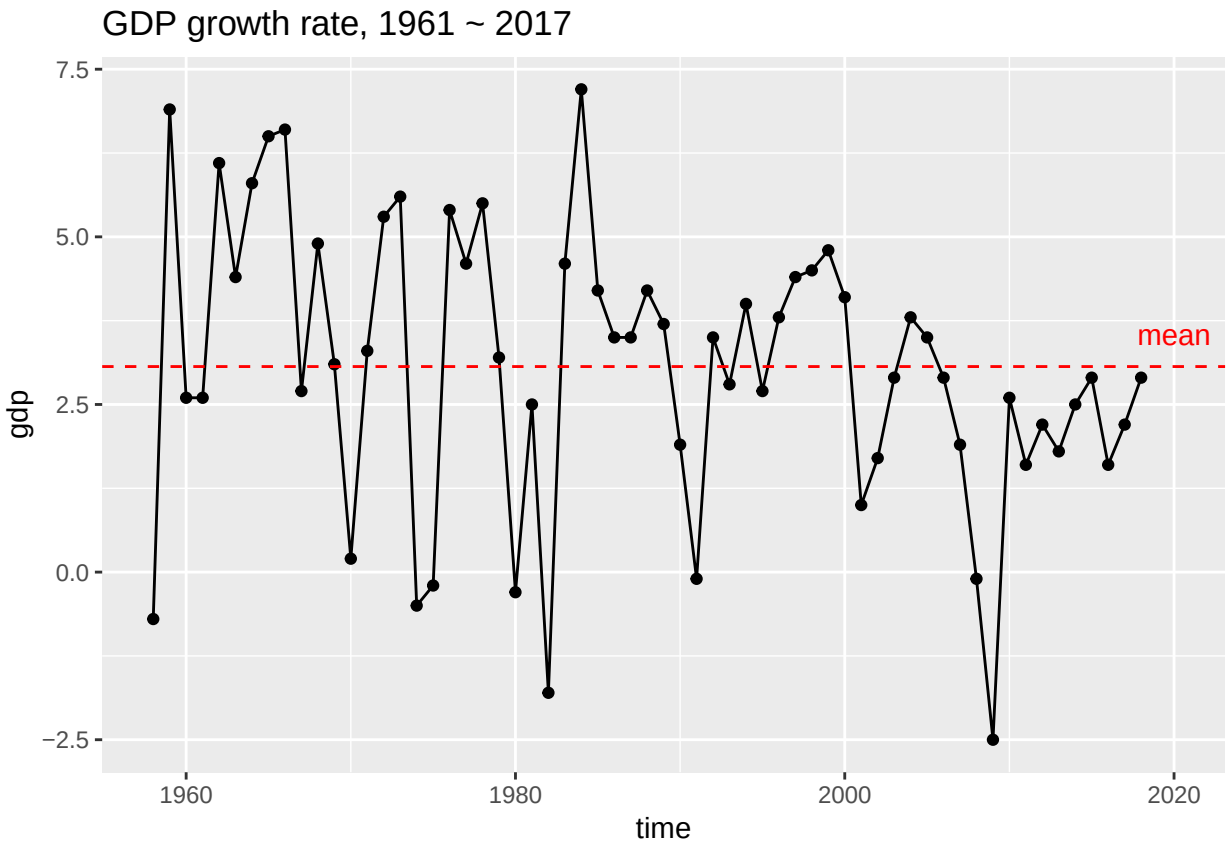



```
#plot tempo
ggplot(data=total.tempo, aes(x=time, y=tempo)) +
  geom_line() +
  geom_hline(yintercept = mean(total.tempo$tempo), linetype = "dashed", color ="red") +
  annotate(geom="text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(total.tempo$tempo),
           vjust = -1, color = "red") +
  geom_point() +
  ggtitle("Tempo, 1958 ~ 2018") +
  scale_x_date(date_labels = '%Y/%m')
```



GDP growth rate

```
GDP <- read.csv("../Data/Clean/GDP_cleaned.csv")
GDP$time <- as.Date(GDP$time)

ggplot(data=GDP, aes(x=time, y=gdp)) +
  geom_line() +
  geom_point() +
  ggtitle("GDP growth rate, 1961 ~ 2017") +
  scale_x_date(date_labels = '%Y') +
  annotate(geom="text", label = "mean",
           x = as.Date("2020-01-01"), y = mean(GDP$gdp),
           vjust = -1, color = "red") +
  geom_hline(yintercept = mean(GDP$gdp), linetype = "dashed", color ="red")
```

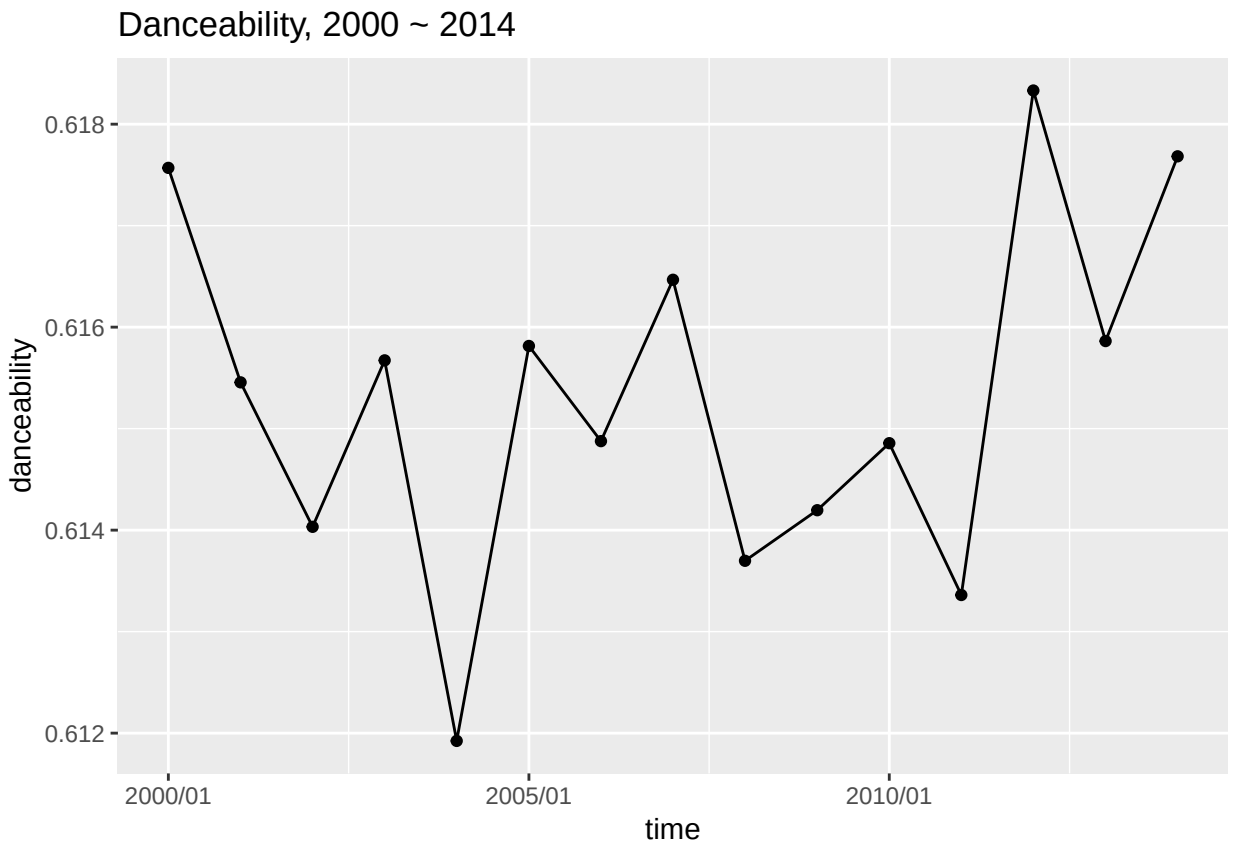


Exploratory Analysis - Financial Crisis

```
Fin.crisis <- subset(billboard,
                     format.Date(time, "%Y")=="2000" |
                     format.Date(time, "%Y")=="2001" |
                     format.Date(time, "%Y")=="2002" |
                     format.Date(time, "%Y")=="2003" |
                     format.Date(time, "%Y")=="2004" |
                     format.Date(time, "%Y")=="2005" |
                     format.Date(time, "%Y")=="2006" |
                     format.Date(time, "%Y")=="2007" |
                     format.Date(time, "%Y")=="2008" |
                     format.Date(time, "%Y")=="2009" |
                     format.Date(time, "%Y")=="2010" |
                     format.Date(time, "%Y")=="2011" |
                     format.Date(time, "%Y")=="2012" |
                     format.Date(time, "%Y")=="2013" |
                     format.Date(time, "%Y")=="2014")

Fin.crisis$time <- as.Date(Fin.crisis$time)
```

Danceability

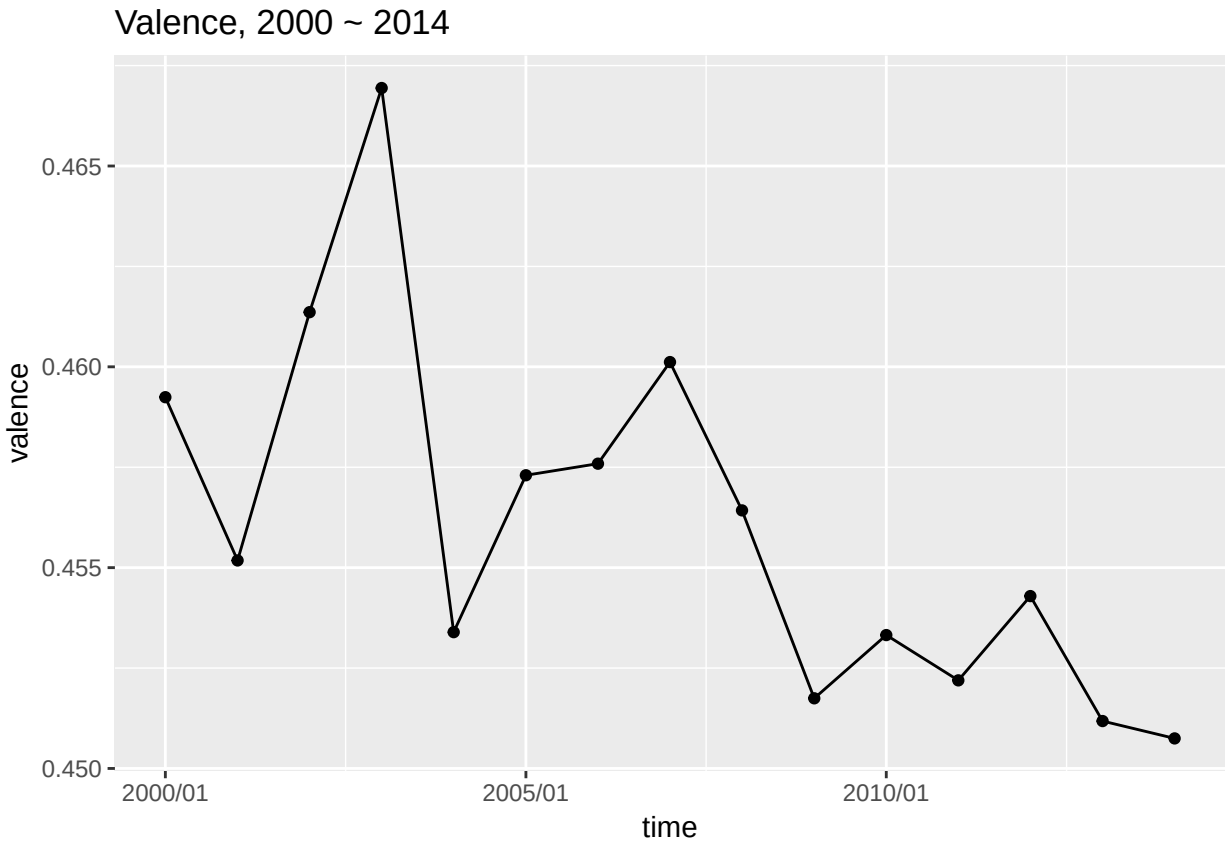
```
#create yearly average danceability
fin.dance.av <- aggregate(list(danceability = Fin.crisis$danceability),
                          list(time = cut(Fin.crisis$time, "1 year")),
                          mean)
fin.dance.av$time <- as.Date(fin.dance.av$time)

#plot
ggplot(data=fin.dance.av, aes(x=time, y=danceability)) +
  geom_line() +
  geom_point() +
  ggtitle("Danceability, 2000 ~ 2014") +
  scale_x_date(date_labels = '%Y/%m')
```



Valence

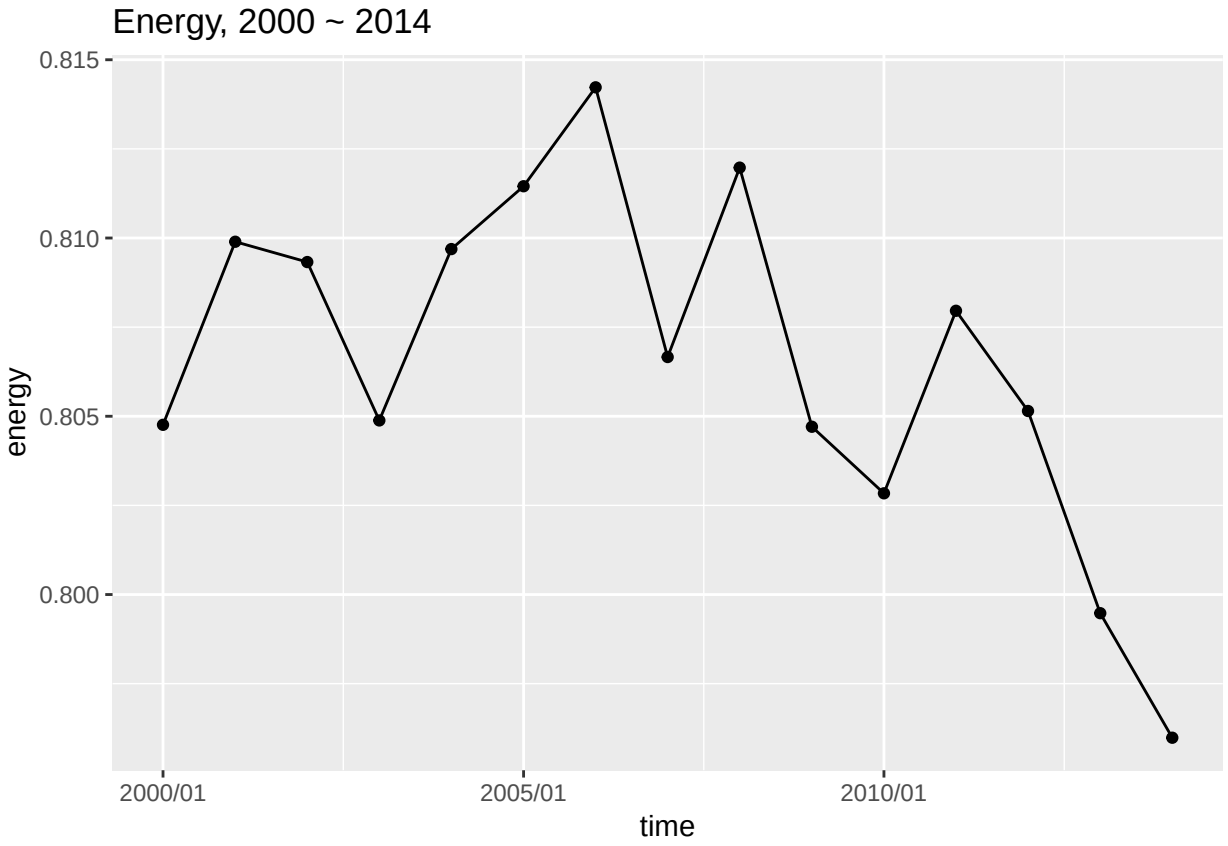
```
#create yearly average valence
fin.valence.av <- aggregate(list(valence = Fin.crisis$valence),
                          list(time = cut(Fin.crisis$time, "1 year")),
                          mean)
fin.valence.av$time <- as.Date(fin.valence.av$time)

#plot
ggplot(data=fin.valence.av, aes(x=time, y=valence)) +
```

```
  geom_line() +
  geom_point() +
  ggtitle("Valence, 2000 ~ 2014") +
  scale_x_date(date_labels = '%Y/%m')
```



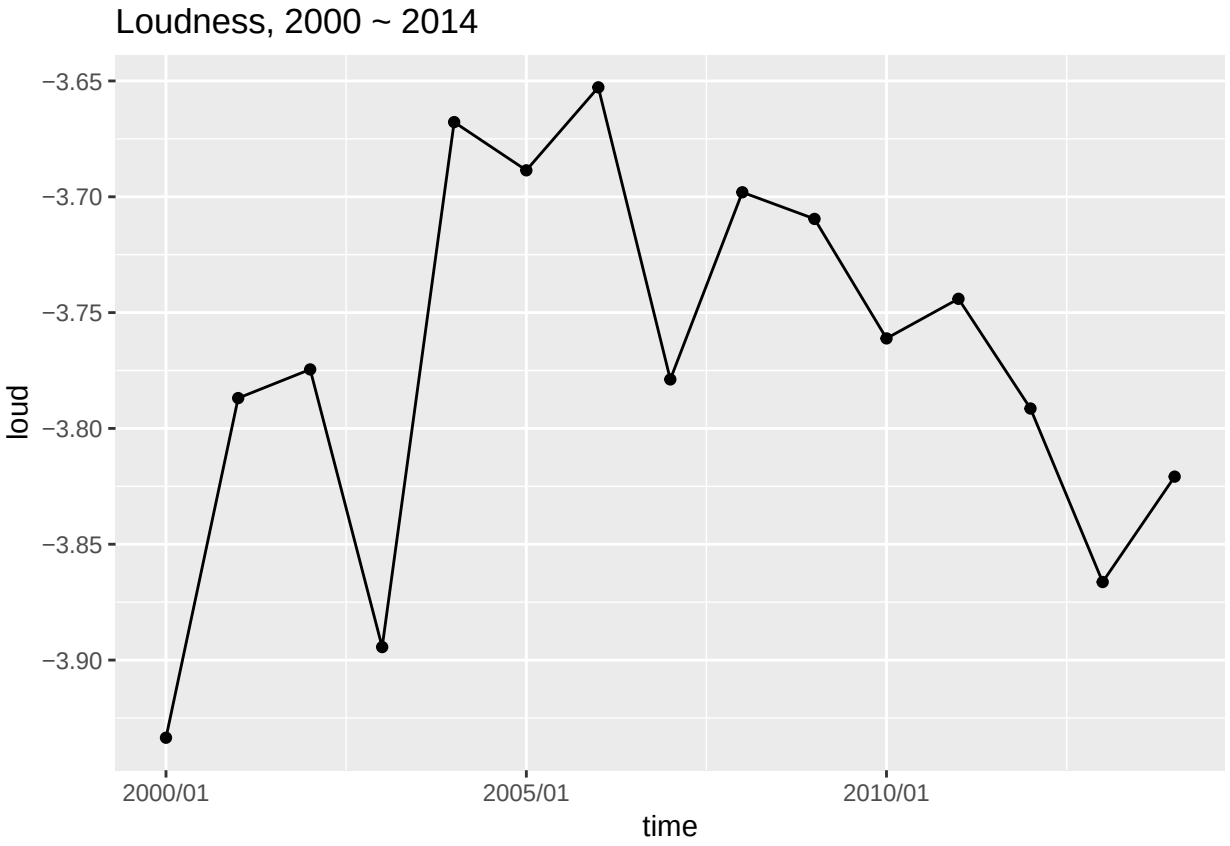
Energy

```
#create yearly average energy
fin.energy.av <- aggregate(list(energy = Fin.crisis$energy),
                           list(time = cut(Fin.crisis$time, "1 year")),
                           mean)
fin.energy.av$time <- as.Date(fin.energy.av$time)

#plot
ggplot(data=fin.energy.av, aes(x=time, y=energy)) +
  geom_line() +
  geom_point() +
  ggtitle("Energy, 2000 ~ 2014") +
  scale_x_date(date_labels = '%Y/%m')
```



Loudness

```
#create yearly average loudness
fin.loud.av <- aggregate(list(loud = Fin.crisis$loud),
                         list(time = cut(Fin.crisis$time, "1 year")),
                         mean)
fin.loud.av$time <- as.Date(fin.loud.av$time)

#plot
ggplot(data=fin.loud.av, aes(x=time, y=loud)) +
  geom_line() +
  geom_point() +
  ggtitle("Loudness, 2000 ~ 2014") +
  scale_x_date(date_labels = '%Y/%m')
```



Time Series Analysis

Danceability

```
grangertest(GDP$gdp ~ total.dance$danceability, order=1)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:1) + Lags(total.dance$danceability, 1:1)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:1)
##   Res.Df Df      F  Pr(>F)
## 1     57
## 2     58 -1 4.6521 0.03525 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(GDP$gdp ~ total.dance$danceability, order=2)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:2) + Lags(total.dance$danceability, 1:2)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 2.3939 0.1009
```

```r
grangertest(GDP$gdp ~ total.dance$danceability, order=3)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:3) + Lags(total.dance$danceability, 1:3)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:3)
##   Res.Df Df      F  Pr(>F)
## 1     51
## 2     54 -3 2.2001 0.09933 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(GDP$gdp ~ total.dance$danceability, order=4)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:4) + Lags(total.dance$danceability, 1:4)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 2.0644 0.1002
```

```r
grangertest(GDP$gdp ~ total.dance$danceability, order=5)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:5) + Lags(total.dance$danceability, 1:5)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 1.1236 0.3616
```

```r
grangertest(total.dance$danceability ~ GDP$gdp, order=1)
```

```
## Granger causality test
##
## Model 1: total.dance$danceability ~ Lags(total.dance$danceability, 1:1) + Lags(GDP$gdp, 1:1)
## Model 2: total.dance$danceability ~ Lags(total.dance$danceability, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 1.7658 0.1892
```

```r
grangertest(total.dance$danceability ~ GDP$gdp, order=2)
```

```
## Granger causality test
##
## Model 1: total.dance$danceability ~ Lags(total.dance$danceability, 1:2) + Lags(GDP$gdp, 1:2)
## Model 2: total.dance$danceability ~ Lags(total.dance$danceability, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 1.0803 0.3467
```

```r
grangertest(total.dance$danceability ~ GDP$gdp, order=3)
```

```
## Granger causality test
##
## Model 1: total.dance$danceability ~ Lags(total.dance$danceability, 1:3) + Lags(GDP$gdp, 1:3)
```

```
## Model 2: total.dance$danceability ~ Lags(total.dance$danceability, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.6042 0.6153
```

```
grangertest(total.dance$danceability ~ GDP$gdp, order=4)
```

```
## Granger causality test
##
## Model 1: total.dance$danceability ~ Lags(total.dance$danceability, 1:4) + Lags(GDP$gdp, 1:4)
## Model 2: total.dance$danceability ~ Lags(total.dance$danceability, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 1.0198 0.4067
```

```
grangertest(total.dance$danceability ~ GDP$gdp, order=5)
```

```
## Granger causality test
##
## Model 1: total.dance$danceability ~ Lags(total.dance$danceability, 1:5) + Lags(GDP$gdp, 1:5)
## Model 2: total.dance$danceability ~ Lags(total.dance$danceability, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.7372 0.5995
```

Valence

```
grangertest(GDP$gdp ~ total.val$valence, order=1)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:1) + Lags(total.val$valence, 1:1)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 0.2024 0.6545
```

```
grangertest(GDP$gdp ~ total.val$valence, order=2)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:2) + Lags(total.val$valence, 1:2)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 0.4693  0.628
```

```
grangertest(GDP$gdp ~ total.val$valence, order=3)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:3) + Lags(total.val$valence, 1:3)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.8011  0.499
```

```r
grangertest(GDP$gdp ~ total.val$valence, order=4)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:4) + Lags(total.val$valence, 1:4)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 1.1452 0.3468
```

```r
grangertest(GDP$gdp ~ total.val$valence, order=5)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:5) + Lags(total.val$valence, 1:5)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.5946 0.7042
```

```r
grangertest(total.val$valence ~ GDP$gdp, order=1)
```

```
## Granger causality test
##
## Model 1: total.val$valence ~ Lags(total.val$valence, 1:1) + Lags(GDP$gdp, 1:1)
## Model 2: total.val$valence ~ Lags(total.val$valence, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 0.4968 0.4838
```

```r
grangertest(total.val$valence ~ GDP$gdp, order=2)
```

```
## Granger causality test
##
## Model 1: total.val$valence ~ Lags(total.val$valence, 1:2) + Lags(GDP$gdp, 1:2)
## Model 2: total.val$valence ~ Lags(total.val$valence, 1:2)
##   Res.Df Df     F Pr(>F)
## 1     54
## 2     56 -2 0.616 0.5438
```

```r
grangertest(total.val$valence ~ GDP$gdp, order=3)
```

```
## Granger causality test
##
## Model 1: total.val$valence ~ Lags(total.val$valence, 1:3) + Lags(GDP$gdp, 1:3)
## Model 2: total.val$valence ~ Lags(total.val$valence, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.0983 0.9606
```

```r
grangertest(total.val$valence ~ GDP$gdp, order=4)
```

```
## Granger causality test
##
## Model 1: total.val$valence ~ Lags(total.val$valence, 1:4) + Lags(GDP$gdp, 1:4)
## Model 2: total.val$valence ~ Lags(total.val$valence, 1:4)
##   Res.Df Df      F Pr(>F)
```

```
## 1     48
## 2     52 -4 0.1737 0.9508
```

```
grangertest(total.val$valence ~ GDP$gdp, order=5)
```

```
## Granger causality test
##
## Model 1: total.val$valence ~ Lags(total.val$valence, 1:5) + Lags(GDP$gdp, 1:5)
## Model 2: total.val$valence ~ Lags(total.val$valence, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.3845 0.8568
```

Energy

```
grangertest(GDP$gdp ~ total.energy$energy, order=1)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:1) + Lags(total.energy$energy, 1:1)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:1)
##   Res.Df Df      F  Pr(>F)
## 1     57
## 2     58 -1 5.1906 0.02648 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(GDP$gdp ~ total.energy$energy, order=2)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:2) + Lags(total.energy$energy, 1:2)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:2)
##   Res.Df Df      F  Pr(>F)
## 1     54
## 2     56 -2 2.9125 0.06292 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(GDP$gdp ~ total.energy$energy, order=3)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:3) + Lags(total.energy$energy, 1:3)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:3)
##   Res.Df Df      F  Pr(>F)
## 1     51
## 2     54 -3 3.9839 0.01262 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(GDP$gdp ~ total.energy$energy, order=4)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:4) + Lags(total.energy$energy, 1:4)
```

```
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:4)
##   Res.Df Df      F  Pr(>F)
## 1     48
## 2     52 -4 3.1764 0.02148 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(GDP$gdp ~ total.energy$energy, order=5)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:5) + Lags(total.energy$energy, 1:5)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:5)
##   Res.Df Df      F  Pr(>F)
## 1     45
## 2     50 -5 2.1202 0.08029 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(total.energy$energy ~ GDP$gdp, order=1)
```

```
## Granger causality test
##
## Model 1: total.energy$energy ~ Lags(total.energy$energy, 1:1) + Lags(GDP$gdp, 1:1)
## Model 2: total.energy$energy ~ Lags(total.energy$energy, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 1.4976 0.2261
```

```
grangertest(total.energy$energy ~ GDP$gdp, order=2)
```

```
## Granger causality test
##
## Model 1: total.energy$energy ~ Lags(total.energy$energy, 1:2) + Lags(GDP$gdp, 1:2)
## Model 2: total.energy$energy ~ Lags(total.energy$energy, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 0.6256 0.5388
```

```
grangertest(total.energy$energy ~ GDP$gdp, order=3)
```

```
## Granger causality test
##
## Model 1: total.energy$energy ~ Lags(total.energy$energy, 1:3) + Lags(GDP$gdp, 1:3)
## Model 2: total.energy$energy ~ Lags(total.energy$energy, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.8219 0.4878
```

```
grangertest(total.energy$energy ~ GDP$gdp, order=4)
```

```
## Granger causality test
##
## Model 1: total.energy$energy ~ Lags(total.energy$energy, 1:4) + Lags(GDP$gdp, 1:4)
## Model 2: total.energy$energy ~ Lags(total.energy$energy, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 1.3278 0.2732
```

```r
grangertest(total.energy$energy ~ GDP$gdp, order=5)
```

```
## Granger causality test
##
## Model 1: total.energy$energy ~ Lags(total.energy$energy, 1:5) + Lags(GDP$gdp, 1:5)
## Model 2: total.energy$energy ~ Lags(total.energy$energy, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 1.2815 0.2886
```

Loudness

```r
grangertest(GDP$gdp ~ total.loud$loudness, order=1)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:1) + Lags(total.loud$loudness, 1:1)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:1)
##   Res.Df Df      F  Pr(>F)
## 1     57
## 2     58 -1 4.0592 0.04866 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(GDP$gdp ~ total.loud$loudness, order=2)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:2) + Lags(total.loud$loudness, 1:2)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:2)
##   Res.Df Df     F Pr(>F)
## 1     54
## 2     56 -2 1.611 0.2091
```

```r
grangertest(GDP$gdp ~ total.loud$loudness, order=3)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:3) + Lags(total.loud$loudness, 1:3)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 2.1848 0.1011
```

```r
grangertest(GDP$gdp ~ total.loud$loudness, order=4)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:4) + Lags(total.loud$loudness, 1:4)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:4)
##   Res.Df Df      F  Pr(>F)
## 1     48
## 2     52 -4 3.2997 0.01814 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(GDP$gdp ~ total.loud$loudness, order=5)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:5) + Lags(total.loud$loudness, 1:5)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:5)
##   Res.Df Df      F  Pr(>F)
## 1     45
## 2     50 -5 2.1033 0.08243 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
grangertest(total.loud$loudness ~ GDP$gdp, order=1)
```

```
## Granger causality test
##
## Model 1: total.loud$loudness ~ Lags(total.loud$loudness, 1:1) + Lags(GDP$gdp, 1:1)
## Model 2: total.loud$loudness ~ Lags(total.loud$loudness, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 0.0011 0.9735
```

```r
grangertest(total.loud$loudness ~ GDP$gdp, order=2)
```

```
## Granger causality test
##
## Model 1: total.loud$loudness ~ Lags(total.loud$loudness, 1:2) + Lags(GDP$gdp, 1:2)
## Model 2: total.loud$loudness ~ Lags(total.loud$loudness, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 0.3435 0.7108
```

```r
grangertest(total.loud$loudness ~ GDP$gdp, order=3)
```

```
## Granger causality test
##
## Model 1: total.loud$loudness ~ Lags(total.loud$loudness, 1:3) + Lags(GDP$gdp, 1:3)
## Model 2: total.loud$loudness ~ Lags(total.loud$loudness, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.2771 0.8417
```

```r
grangertest(total.loud$loudness ~ GDP$gdp, order=4)
```

```
## Granger causality test
##
## Model 1: total.loud$loudness ~ Lags(total.loud$loudness, 1:4) + Lags(GDP$gdp, 1:4)
## Model 2: total.loud$loudness ~ Lags(total.loud$loudness, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 0.3417 0.8484
```

```r
grangertest(total.loud$loudness ~ GDP$gdp, order=5)
```

```
## Granger causality test
##
## Model 1: total.loud$loudness ~ Lags(total.loud$loudness, 1:5) + Lags(GDP$gdp, 1:5)
```

```
## Model 2: total.loud$loudness ~ Lags(total.loud$loudness, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.3241 0.8958
```

Tempo

```
grangertest(GDP$gdp ~ total.tempo$tempo, order=1)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:1) + Lags(total.tempo$tempo, 1:1)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 0.2832 0.5967
```

```
grangertest(GDP$gdp ~ total.tempo$tempo, order=2)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:2) + Lags(total.tempo$tempo, 1:2)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 2.3466 0.1054
```

```
grangertest(GDP$gdp ~ total.tempo$tempo, order=3)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:3) + Lags(total.tempo$tempo, 1:3)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 1.3485 0.2691
```

```
grangertest(GDP$gdp ~ total.tempo$tempo, order=4)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:4) + Lags(total.tempo$tempo, 1:4)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 1.2246 0.3128
```

```
grangertest(GDP$gdp ~ total.tempo$tempo, order=5)
```

```
## Granger causality test
##
## Model 1: GDP$gdp ~ Lags(GDP$gdp, 1:5) + Lags(total.tempo$tempo, 1:5)
## Model 2: GDP$gdp ~ Lags(GDP$gdp, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.7985 0.5566
```

```
grangertest(total.tempo$tempo ~ GDP$gdp, order=1)
```

```
## Granger causality test
##
## Model 1: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:1) + Lags(GDP$gdp, 1:1)
## Model 2: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:1)
##   Res.Df Df      F Pr(>F)
## 1     57
## 2     58 -1 0.7542 0.3888
```

```
grangertest(total.tempo$tempo ~ GDP$gdp, order=2)
```

```
## Granger causality test
##
## Model 1: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:2) + Lags(GDP$gdp, 1:2)
## Model 2: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:2)
##   Res.Df Df      F Pr(>F)
## 1     54
## 2     56 -2 1.1474 0.3251
```

```
grangertest(total.tempo$tempo ~ GDP$gdp, order=3)
```

```
## Granger causality test
##
## Model 1: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:3) + Lags(GDP$gdp, 1:3)
## Model 2: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:3)
##   Res.Df Df      F Pr(>F)
## 1     51
## 2     54 -3 0.5207   0.67
```

```
grangertest(total.tempo$tempo ~ GDP$gdp, order=4)
```

```
## Granger causality test
##
## Model 1: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:4) + Lags(GDP$gdp, 1:4)
## Model 2: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:4)
##   Res.Df Df      F Pr(>F)
## 1     48
## 2     52 -4 0.4373  0.781
```

```
grangertest(total.tempo$tempo ~ GDP$gdp, order=5)
```

```
## Granger causality test
##
## Model 1: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:5) + Lags(GDP$gdp, 1:5)
## Model 2: total.tempo$tempo ~ Lags(total.tempo$tempo, 1:5)
##   Res.Df Df      F Pr(>F)
## 1     45
## 2     50 -5 0.4366 0.8206
```

Additional exploration

What happened during the 1990s? We see a huge drop in all of the music features except for loudness. My theory is that it is because of the 'golden era' of Hip-Hop, which could explain why there music features such as valence and danceability decreased significantly. However, when looking at the artists who stayed the longest amount of time on the chart, there are not that many Hip-Hop artists. It could be that the top

artists on the charts have always been pop artists regardless of what era we look at, but the majority of artists below the the usual pop artists were those rappers who are known for gritty and dark boom-baps.

```
nineties <- subset(billboard,
                   format.Date(WeekID, "%Y")=="1990" |
                   format.Date(WeekID, "%Y")=="1991" |
                   format.Date(WeekID, "%Y")=="1992" |
                   format.Date(WeekID, "%Y")=="1993" |
                   format.Date(WeekID, "%Y")=="1994" |
                   format.Date(WeekID, "%Y")=="1995" |
                   format.Date(WeekID, "%Y")=="1996" |
                   format.Date(WeekID, "%Y")=="1997" |
                   format.Date(WeekID, "%Y")=="1998" |
                   format.Date(WeekID, "%Y")=="1999")
```

Top Artists who stayed the longest on the Billboard chart in the 1990s

```
artist90s <- table(nineties$artist)
kable(head(sort(artist90s[artist90s!=0], decreasing = T), n = 50), format = "markdown",
      caption = "Top Artists who stayed the longest on the Billboard chart in the 1990s")
```

Var1	Freq
Madonna	407
Mariah Carey	386
Boyz II Men	327
TLC	286
Whitney Houston	285
Elton John	279
Celine Dion	265
Janet Jackson	258
Mary J. Blige	255
Gloria Estefan	240
Toni Braxton	236
Michael Bolton	228
En Vogue	221
R. Kelly	218
Shania Twain	210
Aerosmith	207
Jodeci	207
SWV	201
Color Me Badd	200
Ace Of Base	190
Rod Stewart	190
Jon Secada	187
Bryan Adams	184
Brandy	183
LL Cool J	180
Michael Jackson	178
Hootie & The Blowfish	177
Tevin Campbell	177
Phil Collins	172
Richard Marx	171
U2	166

Var1	Freq
Salt-N-Pepa	163
Monica	162
R.E.M.	162
Sarah McLachlan	160
Keith Sweat	159
M.C. Hammer	159
Sheryl Crow	157
Vanessa Williams	157
Babyface	155
Backstreet Boys	154
Melissa Etheridge	154
Collective Soul	150
LeAnn Rimes	144
Tony Toni Tone	142
Metallica	140
After 7	139
Tim McGraw	139
Savage Garden	138
Spice Girls	138

Top Songs which stayed the longest on the Billboard chart in the 1990s

```
songs90s <- table(nineties$title)
kable(head(sort(songs90s[songs90s!=0], decreasing = T), n = 50), format = "markdown",
      caption = "Top Songss which stayed the longest on the Billboard chart in the 1990s")
```

Var1	Freq
Hold On	77
Please Don't Go	77
I Want You	72
How Do I Live	69
Believe	68
Crazy	68
Just Another Day	68
Foolish Games/You Were Meant For Me	65
Lately	64
You	64
Macarena (Bayside Boys Mix)	60
Real Love	58
Stay	58
I Don't Want To Wait	56
I Miss You	56
Always	55
Barely Breathing	55
Missing	55
Too Close	53
Creep	52
How's It Going To Be	52
I'll Be There	52
Show Me Love	52

Var1	Freq
Truly Madly Deeply	52
All My Life	51
Come With Me	51
Angel	50
Close To You	50
Hero	50
I Swear	50
For You	49
One More Try	49
Run-Around	49
Counting Blue Cars	48
Dreams	48
Everything's Gonna Be Alright	48
This Kiss	48
Nobody Knows	47
You Make Me Wanna...	47
Follow You Down/Til I Hear It From You	46
Together Again	46
What's Up	46
100% Pure Love	45
Another Night	45
Someday	45
Whoomp! (There It Is)	45
As I Lay Me Down	44
Come To My Window	44
Give It Up	44
Heaven	44

Takeaways

1. From the Granger-causality test, we see that danceability Granger-causes GDP growth rate with lag 1 and 3, energy Granger-causes GDP growth rate with lag 1, 2, 3, 4, and 5, and loudness Granger-causes GDP growth rate with lag1, 4, and 5.
2. Valence and Tempo do not Granger-causes GDP growth rate.
3. GDP growth rate does not Granger-causes any of the music features, meaning if there is a latent factor, i.e. optimism, it is reflected in the music trend before it is reflected in the market.

Limitations

1. The dataset was not comprehensive, as I was not able to request and join Spotify music features for some of the songs on the chart, although it was a small proportion of the full dataset.
2. Was not able to identify the big drop in music feautres except for loudness in the 90s, which might have affected the accuracy of the analyses.
3. I felt like the analyses were not comprehensive enough. Also, I was not entirely confident with choosing the number of lags (order), as it felt like p-hacking.

Next step

1. Update the helper function and fix missing data issue
2. Supervised machine learning approach for causal inference (See Supervised machine learning appraoch under Required Skills and Analysis Plan of Readme.md)
3. Latent factor analysis