Spotify_Songs

Thibault Eudes

28/10/2021

La base données 'Spotify Songs' rescence les informations de 28356 titres, provenant de l'API de l'entreprise. Les chansons possèdent un ensemble de informations sur leur provenance, artiste etc. mais également des colonnes de scoring sur différentes charactéristiques comme l'energie, l'accoustiquee et autres. Nous nous interesserons dans cette étude aux différentes caractéristiques qui permettent à une chanson d'être populaire.

Importation et nettoyage des données

```
library(tidyverse)
```

```
## Warning: le package 'tidyverse' a été compilé avec la version R 4.1.1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

```
## Warning: le package 'ggplot2' a été compilé avec la version R 4.1.1

## Warning: le package 'tibble' a été compilé avec la version R 4.1.1

## Warning: le package 'tidyr' a été compilé avec la version R 4.1.1

## Warning: le package 'readr' a été compilé avec la version R 4.1.1

## Warning: le package 'purrr' a été compilé avec la version R 4.1.1

## Warning: le package 'dplyr' a été compilé avec la version R 4.1.1

## Warning: le package 'stringr' a été compilé avec la version R 4.1.1

## Warning: le package 'forcats' a été compilé avec la version R 4.1.1

## -- Conflicts --------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(kableExtra)
```

```
## Warning: le package 'kableExtra' a été compilé avec la version R 4.1.1

##
## Attachement du package : 'kableExtra'

## L'objet suivant est masqué depuis 'package:dplyr':
##
##     group_rows
```

```
library(skimr)
```

```
## Warning: le package 'skimr' a été compilé avec la version R 4.1.1
```

```
library(chron)
```

```
## Warning: le package 'chron' a été compilé avec la version R 4.1.1
```

```
library(ggthemes)
```

```
## Warning: le package 'ggthemes' a été compilé avec la version R 4.1.1
```

```
ssini <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/202
```

```
## Rows: 32833 Columns: 23
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (10): track_id, track_name, track_artist, track_album_id, track_album_na...
## dbl (13): track_popularity, danceability, energy, key, loudness, mode, speec...

##
## 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.
```

```
#Certaines chansons sont en double: on ne garde qu'une seule version
ssini<-ssini %>%
  distinct(track_id,.keep_all = T)

ss<-ssini %>%
  mutate(.,duree=as.times(format(as.POSIXct(Sys.Date())+duration_ms/1000-3600, "%H:%M:%S"))) %>%
  mutate(.,dureem=duration_ms/60000) %>%
  as.data.frame()
#Duree sera utilisée pour visualiser des moyennes etc.
#R représente mal les heures, on utilisera un pourcentage de minutes (dureem) pour les graphiques
```

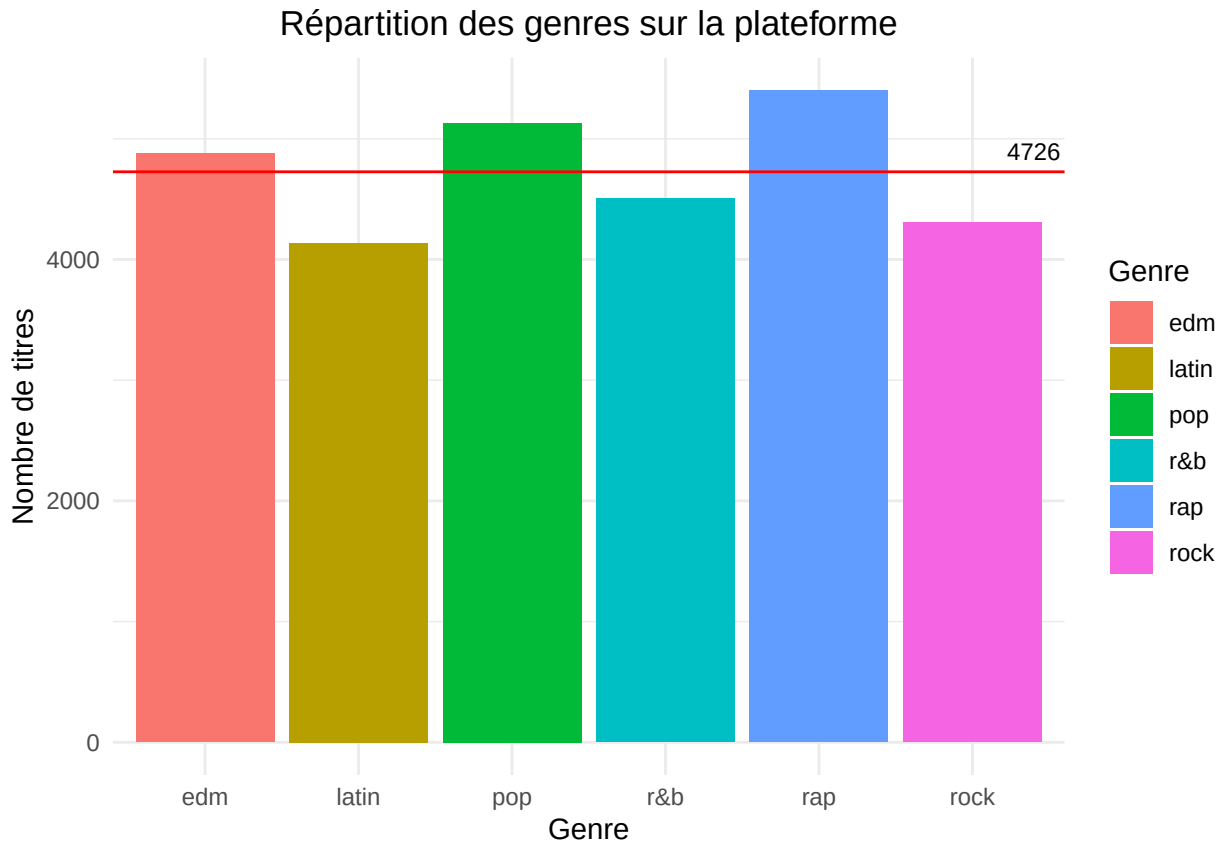
###1 Quelle est la durée moyenne des titres sur Spotify ?

```
mean(ss$duree)
```

```
## [1] 00:03:46
```

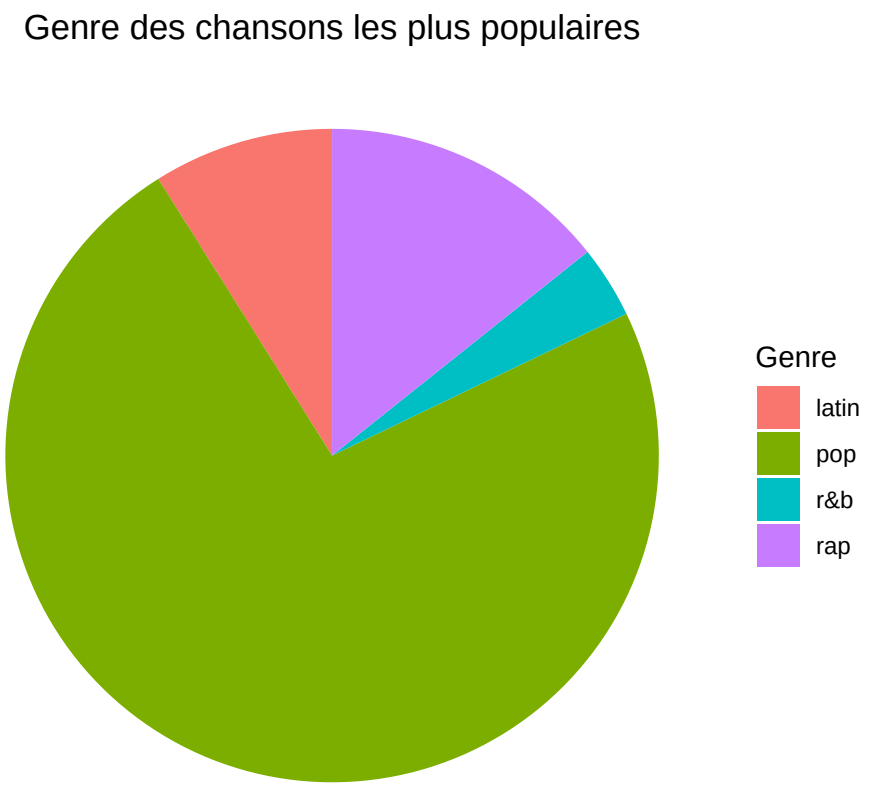
###2 Spotify a t-il une repartition équitable des genres de musique ?

```
m<-mean(table(ss$playlist_genre))

ss %>%
  ggplot(.,aes(playlist_genre),table(playlist_genre)) +
  geom_bar(aes(fill = playlist_genre  )) +
  labs(title = "Répartition des genres sur la plateforme",y="Nombre de titres",x="Genre") +
  geom_hline(yintercept = m, color = "red")+
  annotate(geom="text",label=m
,x=6.4,y=m+170,color="black",size=3)+
  scale_fill_discrete(name="Genre")+
  theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))
```



###3 Y a t-il un genre predominant parmis les chansons les plus populaires ?

```
N=50

t<-ss %>%
  select(track_name,track_artist,playlist_genre,duree,track_popularity,track_album_release_date) %>%
  top_n(.,N,wt=track_popularity) %>%
  arrange(desc(track_popularity))

pie<-t %>%
  ggplot(aes(x="",fill=playlist_genre,label=))+
  geom_bar(width = 1)+
    labs(title = "Genre des chansons les plus populaires",x=NULL,y=NULL) +
    scale_fill_discrete(name="Genre")+
  theme_void()+
  theme(plot.title = element_text(hjust = 0.5))

pie + coord_polar(theta = "y", start=0)
```

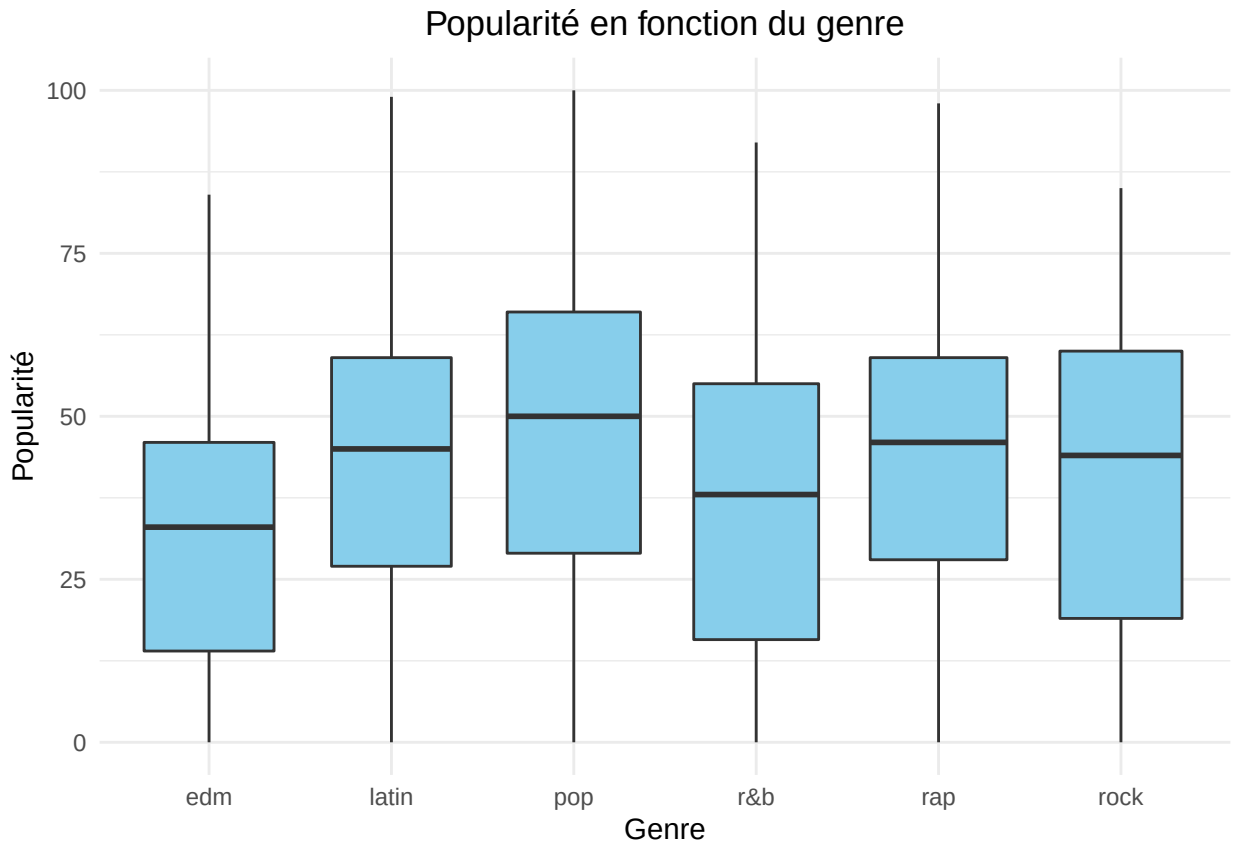



```
G<-table(t$playlist_genre)/N
G
```

```
##
## latin   pop   r&b   rap
##  0.10  0.82  0.04  0.16
```

###4 Cela se verifie t il sur l'echantillon global?

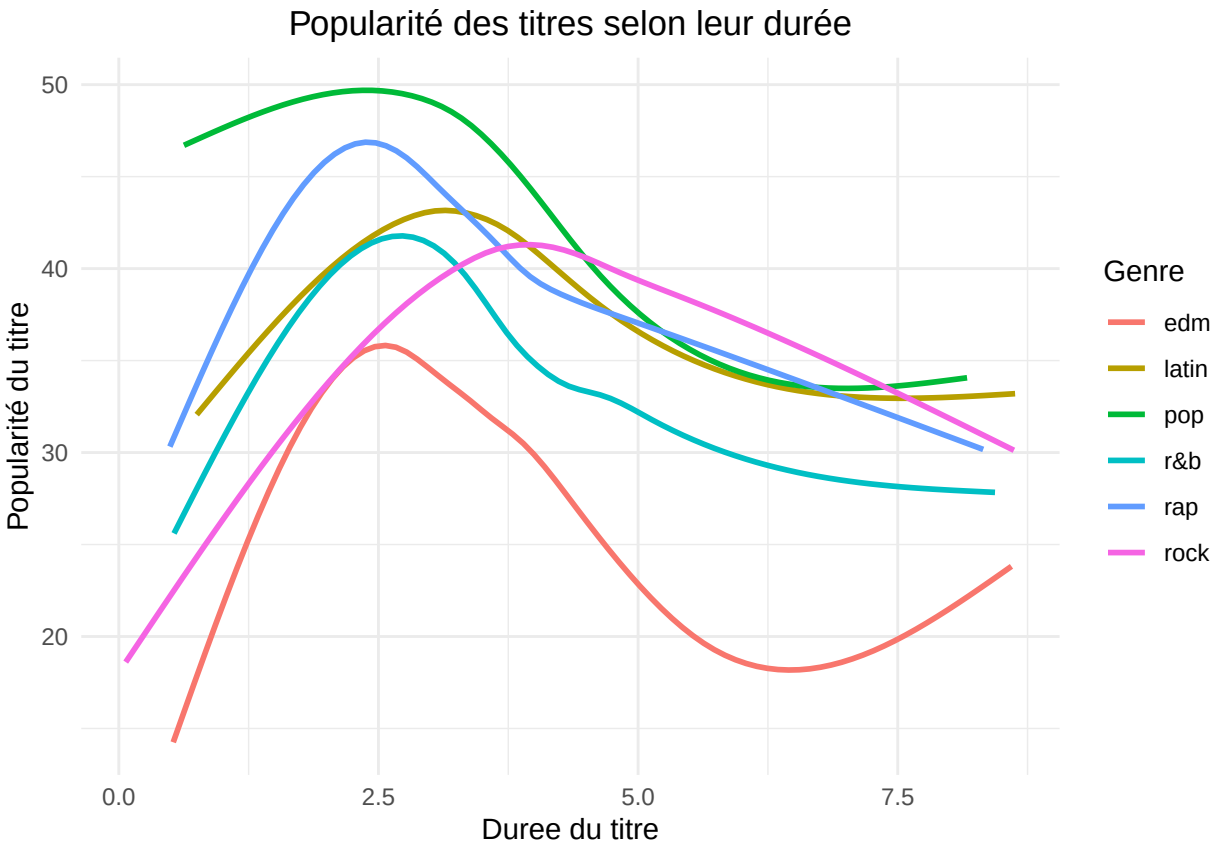
```
ss %>%
  ggplot(aes(playlist_genre,track_popularity))+
  geom_boxplot(varwidth = T,fill="skyblue")+
    labs(title = "Popularité en fonction du genre",y="Popularité",x="Genre") +
  theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))
```



###5 Les styles de musique sont-ils soumis aux mêmes contraintes de durée de chanson?

```
ggplot(ss,aes(x = dureem,y = track_popularity,color=playlist_genre))+
    geom_smooth(se=F) +
    labs(title = "Popularité des titres selon leur durée",x="Duree du titre",y="Popularité du titre",co
  theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))
```

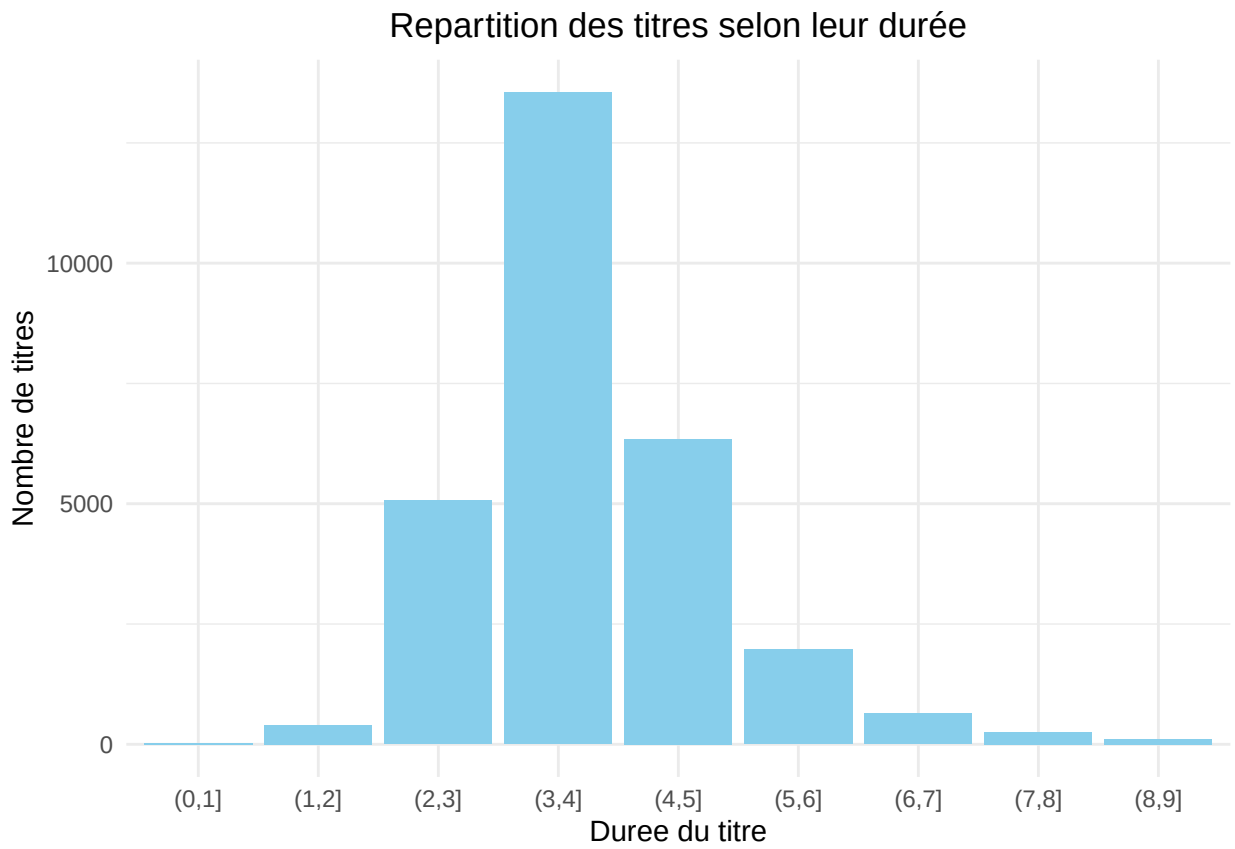
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



L'EDM semble dépendre plus de la durée que les autres genres

###6 Cela se verifie t-il dans la durée des chansons (tous genres confondus)?

```
ss5<-ss %>%
  mutate(catd=cut(ss$dureem,breaks=c(0:2,3,4,5:10)))


ss5 %>%
  ggplot(aes(catd))+
  geom_bar(fill="skyblue")+
  labs(title = "Repartition des titres selon leur durée",x="Duree du titre",y="Nombre de titres",)+
  theme_minimal()+
  theme(plot.title = element_text(hjust = 0.5))
```

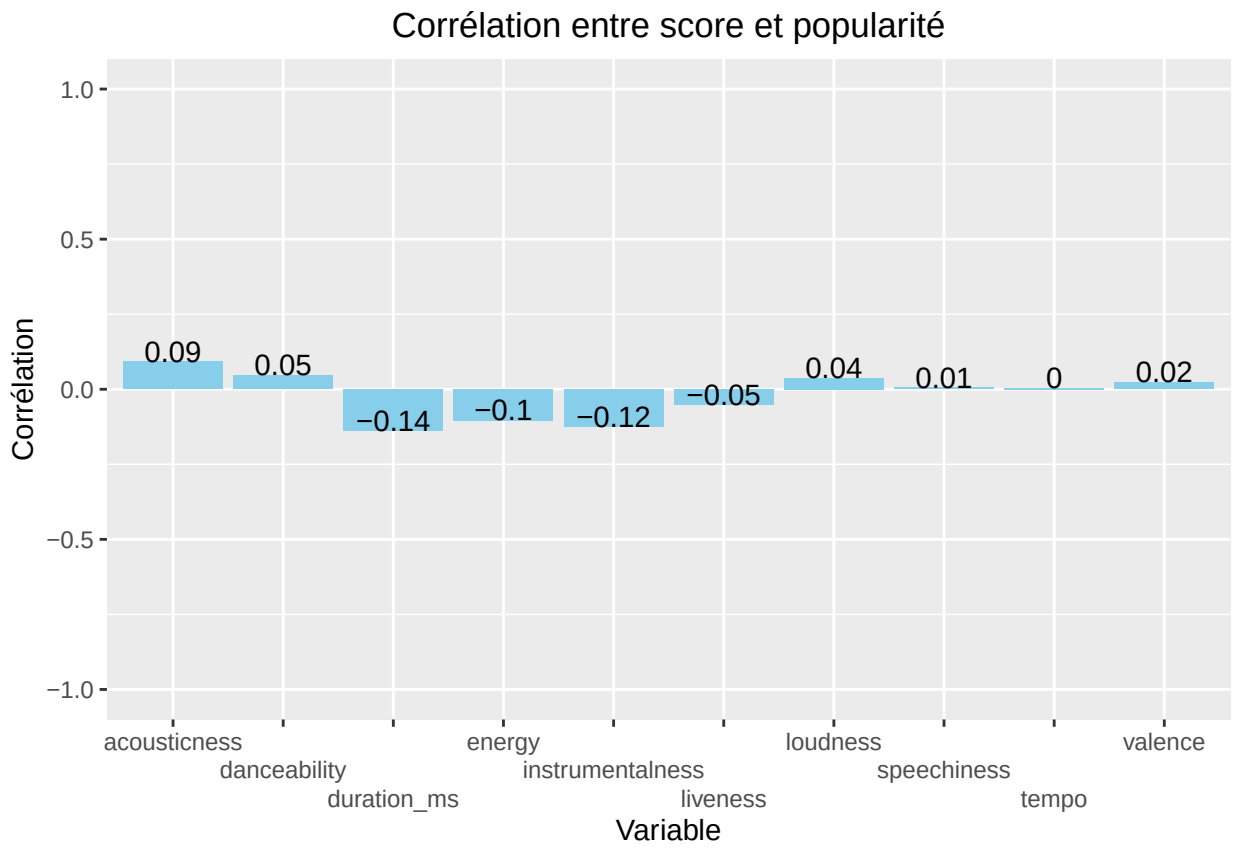


###7 Y a t-il une formule pour faire une chanson populaire ?

```
corr_list<-ss[c(12,13,15,17:23)]

B<-corr_list %>%
  summarize_all(funs(cor(.,ss$track_popularity))) %>%
  t() %>%
  as.data.frame() %>%
  rownames_to_column("var") %>%
  rename("correlation avec la popularité" = V1)
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

```
mincor<-min(B$`correlation avec la popularité`)
maxcor<-max(B$`correlation avec la popularité`)



B %>%
  ggplot(aes(var,`correlation avec la popularité`),ylim=c(-1,1))+
  geom_col(fill="skyblue")+
  ylim(-1,1)+
  geom_text(aes(label=round(`correlation avec la popularité`,2)), vjust =0)+
    scale_x_discrete(guide = guide_axis(n.dodge=3))+
  labs(title = "Corrélation entre score et popularité",x="Variable",y="Corrélation",size=12)+
  theme(plot.title = element_text(hjust = 0.5))
```

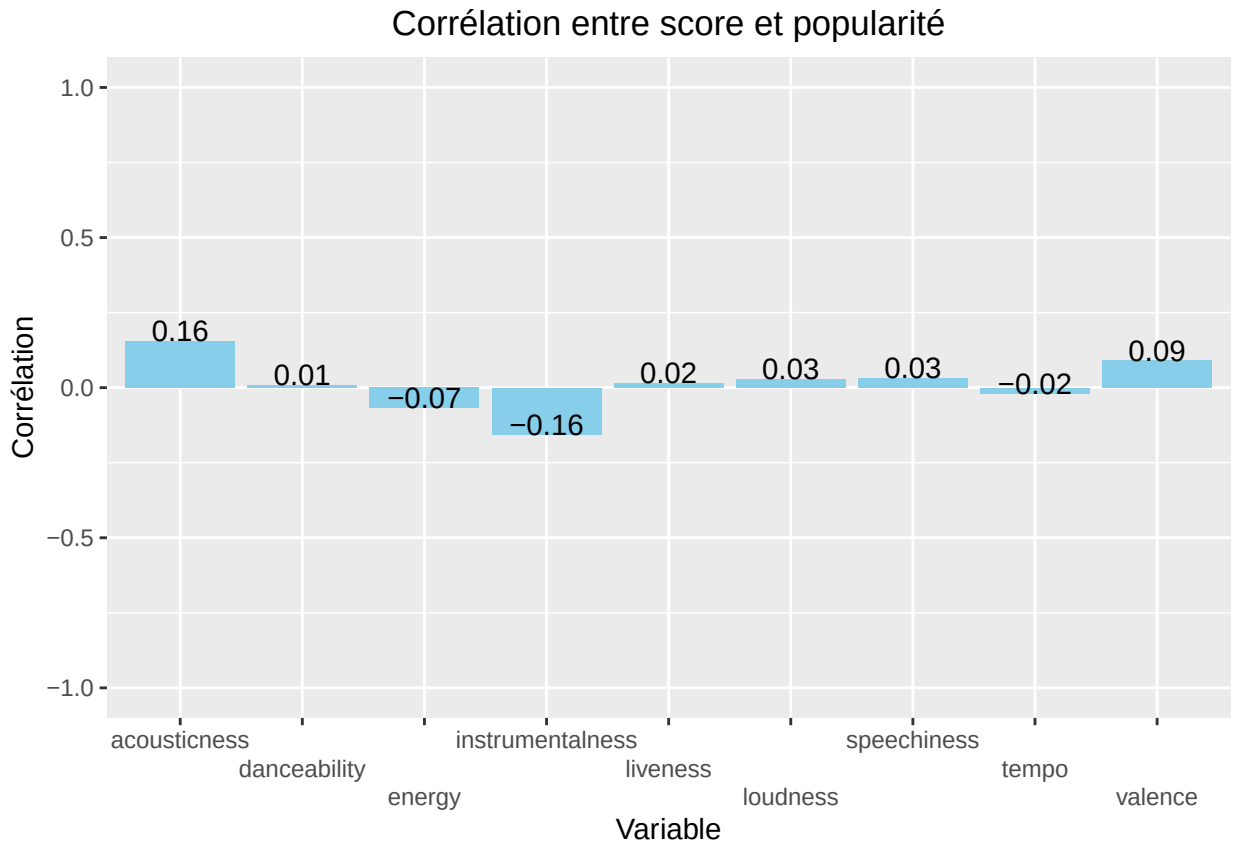


Les corrélations sont toutes plutot basses on peut donc supposer qu'il n'y a à priori pas de préférence nette. On peut retenir une très légère préférence pour les musiques avec des paroles (cor= -0.12)

###8 La musique électronique est elle plus dépendante de certaines caractéristiques pour être populaire ?

```
edm_corr <-ss %>%
  filter(playlist_genre=="edm") %>%
    .[c(12,13,15,17:22)]
edm_pop<-ss %>%
  filter(playlist_genre=="edm") %>%
    .[4]

EDM<-edm_corr %>%
```

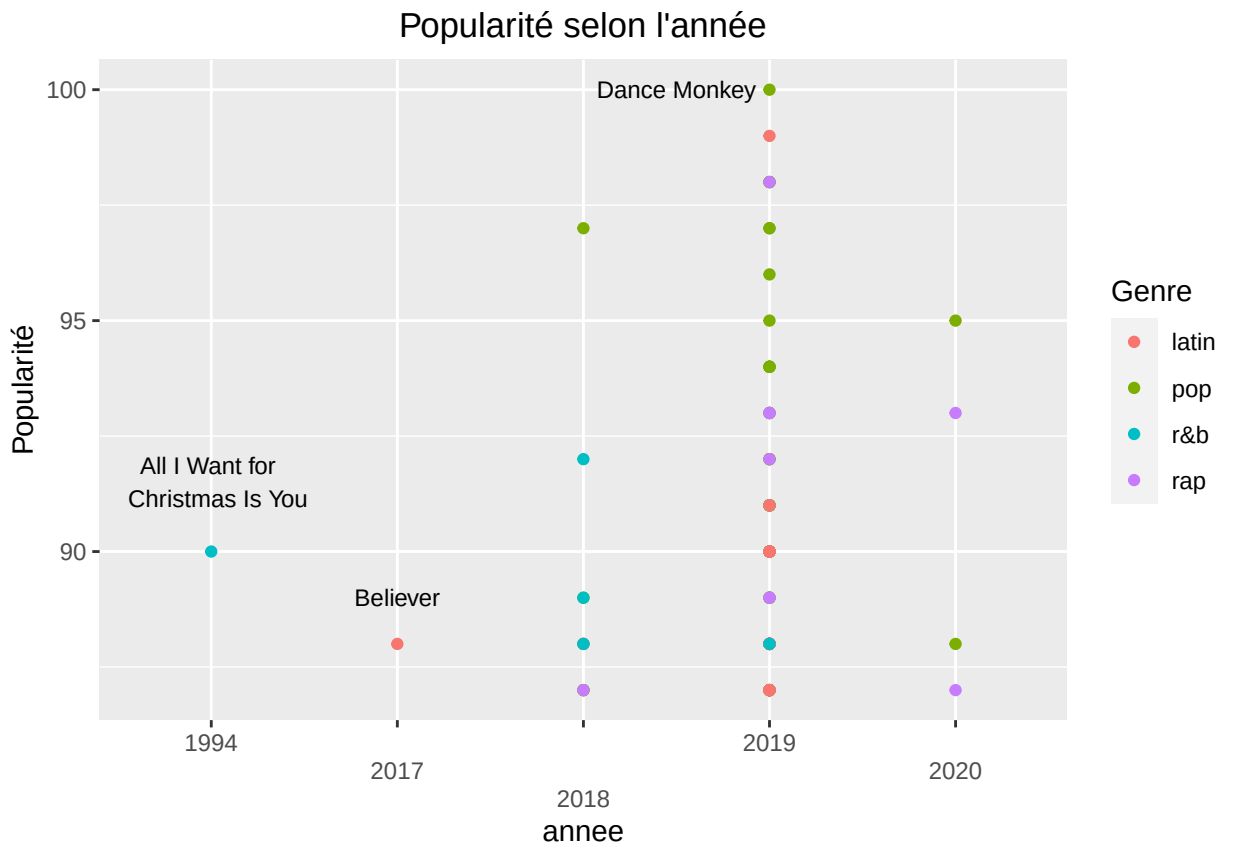
```
  summarize_all(funs(cor(.,edm_pop$track_popularity))) %>%
  t() %>%
  as.data.frame() %>%
  rownames_to_column("var") %>%
  rename("correlation avec la popularité" = V1)

EDM %>%
  ggplot(aes(var,`correlation avec la popularité`),ylim=c(-1,1))+
  geom_col(fill="skyblue")+
  ylim(-1,1)+
  geom_text(aes(label=round(`correlation avec la popularité`,2)), vjust =0)+
    scale_x_discrete(guide = guide_axis(n.dodge=3))+
  labs(title = "Corrélation entre score et popularité",x="Variable",y="Corrélation",size=12)+
  theme(plot.title = element_text(hjust = 0.5))
```



###9 Les chansons les plus populaires sont elles determinées par leur ancienneté?

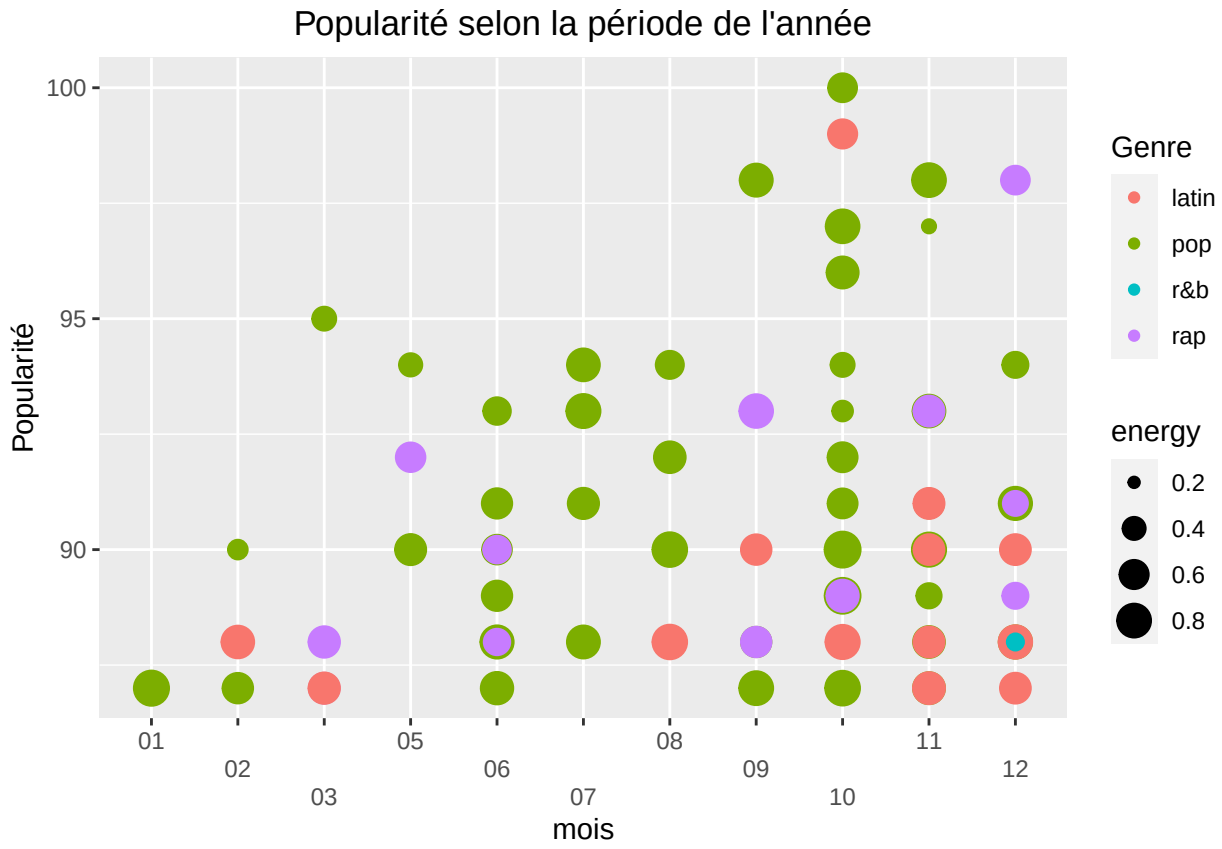
```
Top100 <-
ss %>%
  top_n(.,100,wt=track_popularity) %>%
  mutate(annee=format(as.Date(.$track_album_release_date), "%Y")) %>%
  mutate(mois=format(as.Date(.$track_album_release_date), "%m"))


Top100 %>%
  ggplot(aes(x=annee,y=track_popularity))+
```

```
  geom_point(aes(col=playlist_genre))+
      scale_x_discrete(guide = guide_axis(n.dodge=3))+
  labs(title = "Popularité selon l'année",y="Popularité",color="Genre")+
annotate(geom="text",label="All I Want for
  Christmas Is You",x=1,y=91.5,color="black",size=3)+
  annotate(geom="text",label="Believer",x=2,y=89,color="black",size=3)+
  annotate(geom="text",label="Dance Monkey",x=3.5,y=100,color="black",size=3)+
  theme(plot.title = element_text(hjust = 0.5))
```



###10 Y a t-il des tendances de periodes de sorties de tubes ? On prendra l'année 2019, où l'ensemble des mois hors avril sont représentés

```
Top100 %>%
  filter(annee==2019) %>%
  ggplot(aes(x=mois,y=track_popularity))+
  geom_point(aes(col=playlist_genre, size=energy))+
      scale_x_discrete(guide = guide_axis(n.dodge=3))+
    labs(title = "Popularité selon la période de l'année",y="Popularité",color="Genre")+
  theme(plot.title = element_text(hjust = 0.5))
```


Popularité selon la période de l'année
Popularité
100
95
90
01
02
03
05
06
07
08
09
10
11
12
mois
Genre
latin
pop
r&b
rap
energy
0.2
0.4
0.6
0.8