

# Projet Calcul Financier & Statistique : En avant la musique !

MAHI Riad & Jérôme GAMBIEZ

Mise à jour, le 25 Mai 2023

## Contents

1. Introduction . . . . .	1
2. Analyse univariée . . . . .	2
<b>Générer le pie chart</b>	<b>3</b>
<b>Afficher le pie chart</b>	<b>3</b>

## 1. Introduction

La source de notre jeu de donnée provient du site Kaggle.com:  
<https://www.kaggle.com/datasets/salvatorerastelli/spotify-and-youtube>

Dernière mise à jour du jeu de donnée: Mars 2023  
Création: Février 2023

Nous avons sélectionné, dans le cadre du projet Calcul Financier & Statistique, un jeu de données provenant des plateformes de streaming musical Spotify et YouTube. Nous avons spécifiquement choisi ce jeu de données en raison de l'impact qu'ont ces applications sur la vie quotidienne de millions d'utilisateurs à travers le monde. Étant donné que ces applications sont utilisées par une population provenant de divers pays, notre échantillon ne représente pas une zone géographique spécifique.

Question à mettre en place: - Est-ce l'énergie et la dansabilité d'une musique ont une relation ? - Est-ce l'énergie d'une musique à un impact sur la sensibilité d'un utilisateur à aimer la musique. - Même question pour la dansabilité?

Mise en place du jeu de donnée:

```
dataPath <- "./Spotify_Youtube.csv"
data <- read.csv(dataPath, header=TRUE, stringsAsFactors=FALSE, nrow=50)
```

Les variables qualitatives sont les variables: - Les noms des artistes - Si la musique à été publiée comme un album ou un single

Les variables quantitatives sont: - La dansabilité d'une musique, qui est un indicateur sur la probabilité de danser sur une musique - l'énergie que la musique à définir!

## 2. Analyse univariée

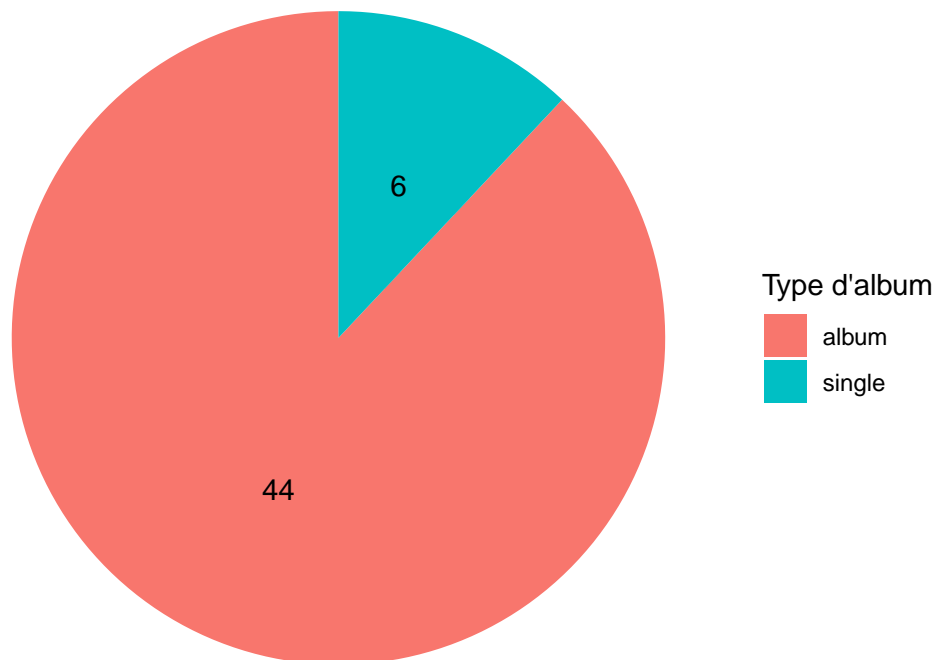
### Répartitions de la variable qualitative “type album”

```
library(ggplot2)

album_type <- table(data$Album_type)
df <- as.data.frame(album_type)
df <- df[order(df$Freq, decreasing = TRUE), ]
# Générer le pie chart
pie_chart <- ggplot(df, aes(x = "", y = Freq, fill = Var1)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  labs(title = "Répartition des types d'albums", fill = "Type d'album") +
  theme_void() +
  geom_text(aes(label = Freq), position = position_stack(vjust = 0.5), size = 4)

# Afficher le pie chart
print(pie_chart)
```

Répartition des types d'albums



### Répartitions de la variable qualitative noms des artistes

```
library(ggplot2)

# Filtrer les vues supérieures à 100 millions
```

```
views_above_100m <- data[data$Views > 100000000, "views"]

# Compter le nombre de chansons pour chaque catégorie de vues
view_counts <- table(views_above_100m)

view_counts
```

```
## < table of extent 0 >
```

## Générer le pie chart

```
pie_chart <- ggplot(data.frame(values = view_counts), aes(x = "", y = values, fill = names(view_counts)))
+ geom_bar(stat = "identity", width = 1) + coord_polar("y", start = 0) + labs(title = "Répartition des
vues > 100 millions") + theme_minimal() + geom_text(aes(label = values), position = position_stack(vjust
= 0.5), size = 4)
```

## Afficher le pie chart

```
print(pie_chart)
```