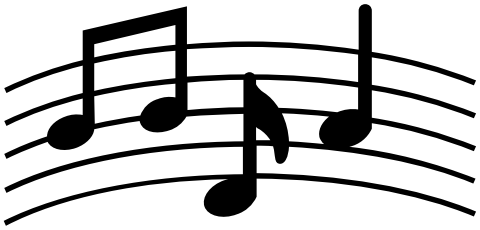
UNIVERSITÉ
DE REIMS
CHAMPAGNE-ARDENNE

L'IMPACT DE L'ÉNERGIE, LA DANCEABILITY ET LE TEMPO D'UNE MUSIQUE SUR SA POPULARITÉ



ERTAS Elif

Master de Statistique pour l'évaluation et la prévision

TABLES DES MATIERES

PARTIE A : Compte rendu de l'analyse

I. Introduction

Depuis des décennies, la musique se trouve partout dans notre quotidien, encore plus aujourd'hui avec toutes les nouvelles technologies et en particulier les réseaux sociaux. Il existe une multitude de plateformes de streaming musical dans lesquelles on peut découvrir des musiques du monde entier. Cependant, il y a toujours des musiques plus populaires que d'autres, d'ailleurs les classements peuvent être très différents d'un pays à l'autre. La culture musicale peut varier d'une zone à l'autre, c'est donc pour cela que notre étude se fera uniquement sur le classement mondial des musiques. Spotify est l'un des services de streaming musical les plus connues, cette plateforme permet une écoute instantanée de diverses musiques. De plus, c'est une plateforme gratuite, ce qui permet d'avoir encore plus d'utilisateurs. La plateforme compte aujourd'hui plus de 500 millions d'utilisateurs mensuels, répartis dans le monde entier et de tout âge.

Cette étude est possible grâce à une base de données regroupant les 100 meilleures musiques de chaque année entre 2010 et 2019 ce qui nous fait un total de 1000 musiques.

Cette base de données provient d'un site appartenant à Spotify (Annexe 1) qui permet de récolter les informations de n'importe quelles musiques sur leur plateforme.

À partir de ces données officielles, nous voulons étudier l'impact de l'énergie, de la danceability et le tempo d'une musique sur sa popularité.

II. Description des données

Notre base de données est composée de 1000 titres de musiques définis par 17 variables :

Variables	Description	Type de variable
Title	Le titre	Chaîne de charactère
Artist	Le(les) artiste(s)	Chaîne de charactère
Genre	Le genre	Chaîne de charactère
Year released	L'année de sortie	Numérique
Added	La date d'ajout au top100	Date
Bpm	Battement par minute (=tempo)	Numérique
Nrgy	L'énergie (0 à 100). Plus c'est élevé et plus la musique est énergétique	Numérique
Dnce	Danceability = Chanson propice à la danse (0 à 100). Plus c'est élevé et plus c'est facile de danser dessus	Numérique
dB	Décibel	Numérique
Live	Liveliness Plus la valeur est grande et plus il est probable que la musique a été enregistré en direct	Numérique
Val	Valence (0 à 100) L'ambiance de la musique, plus la valeur est grande et plus la musique est 'joyeuse'	Numérique
Dur	Durée La durée en secondes de la musique	Numérique
Acous	L'acoustique de la musique (0 à 100)	Numérique
Spch	Speechiness Plus la valeur est grande et plus il y a de mots prononcés dans la musique	Numérique
Pop	Popularité (0 à 100)	Numérique
Top Year	L'année où la musique a été à son top	Date
Artist Type	Le type d'artiste, c'est-à-dire solo, duo ou groupe.	Chaîne de caractère

III. Importation et préparation des données

Nos données sont stockées dans un fichier csv avec la virgule comme délimiteur. Le problème est que certains titres de musiques sont beaucoup trop longs, cela représente 7 titres sur 1000, donc pas très compliqué à corriger à la main. En effet, certains titres sont complétés par le film auquel ils sont reliés donc on a trop de chaînes de caractères et cela nous empêche d'importer correctement toutes les données. Après les modifications sur les titres, on peut donc importer correctement nos données. Notre base de données est maintenant bien structurée et il n'y a aucune valeur manquante.

Après l'importation des données dans notre logiciel, on décide de prendre uniquement les variables qui nous intéressent pour l'étude, c'est-à-dire l'énergie, la danceability et le tempo (bpm) pour chaque titre de musique. On s'assure que chacune de ces variables soient de format numérique et que les titres des musiques soient bien des chaînes de caractères. (Annexe 2)

IV. Analyse de données

On observe qu'en moyenne le tempo des musiques de notre base de données tourne autour de 121 bpm, l'énergie de 70, la danceability de 67 et la popularité de 75. Ici, on peut même dire que 50 % des musiques sont au-delà de 76 en popularité. (Annexe 3)

Après avoir observé la corrélation entre chaque variable deux à deux, on en conclut qu'il n'y aucune forte corrélation donc on n'a pas de variables qui veulent « dire la même chose ». Chaque information est donc bonne à prendre. (Annexe 4)

Nous avons fait une régression multiple avec l'ensemble de nos variables. D'après les résultats, notre modèle n'est pas significatif donc la prédiction ne sera pas très fiable. Cela veut dire que la popularité de chaque musique ne dépend donc pas des variables que l'on a prise en compte. (Annexe 5)

On pourrait alors prendre d'autres variables que nous avions au départ, malgré cela, on a toujours un modèle pas fiable et donc non-utilisable pour prédire la popularité. (Annexe 6)

Cela veut dire que la popularité des musiques ne dépend pas significativement des variables que l'on a dans notre base de données. C'est-à-dire que deux musiques avec des caractéristiques complétement différentes peuvent avoir la même popularité.

Après avoir tout de même continuer notre analyse, on observe qu'aucun individu n'a une valeur influente. (Annexe 7)

De plus, les individus ne suivent pas une loi gaussienne. (Annexe 8)

Il n'y a ni problème d'orthogonalité, ni de problème d'homoscédasticité. (Annexe 9 et Annexe 10)

V. Conclusion

Pour conclure, nous n'avons pas trouvé de variables qui peuvent vraiment expliquer la popularité d'une musique. En effet, des musiques avec des caractéristiques complétement différentes peuvent se trouver au même seuil de popularité. Une musique n'est donc pas vraiment populaire par son tempo, l'énergie qu'elle apporte ou sa capacité à danser dessus. On peut alors toujours se demander comment une musique peut être populaire ? Il y a un effet de mode et le partage exerce également une forte influence sur la popularité d'une musique. On a par exemple le réseau social TikTok qui a permis à des milliers de titres de musiques de revenir à la mode ou même de nouvelles musiques à devenir connues. On pourrait alors poursuivre nos recherches avec d'autres variables.

PARTIE B : Annexes

➢ Annexe 1

À partir de ce lien : http://organizeyourmusic.playlistmachinery.com/ on peut extraire les données concernant les playlists que l'on choisit. Personnellement, j'ai choisi de prendre le TOP100 mondial de chaque année entre 2010 et 2019 que l'on retrouve directement sur la plateforme Spotify par playlist.

Voici par exemple celle de 2010 : https://open.spotify.com/playlist/37i9dQZF1DXc6IFF23C9jj?si=bb4cb1c4e9e94cd7

➢ Annexe 2

On observe que le titre est une chaîne de caractères et que les variables bpm, energy, danceabiliy et popularite sont bien numériques.

```
'data.frame':   1000 obs. of  5 variables:
 $ titre       : chr  "STARSTRUKK (feat. Katy Perry)" "My First Kiss (feat. Ke$ha)" "I Need A Dollar"
"Airplanes (feat. Hayley Williams of Paramore)" ...
 $ bpm         : num  140 138 95 93 104 82 128 92 146 109 ...
 $ energy      : num  81 89 48 87 85 93 81 52 59 84 ...
 $ danceability: num  61 68 84 66 69 55 82 60 50 64 ...
 $ popularite  : num  70 68 72 80 79 71 75 71 87 86 ...
```

➢ Annexe 3

On observe plusieurs informations concernant chacune de nos variables.

```
> summary(don)
    titre                bpm             energy       danceability     popularite
 Length:1000        Min.   : 65.0   Min.   : 6.00   Min.   :19.00   Min.   :35.00
 Class :character   1st Qu.:100.0   1st Qu.:59.00   1st Qu.:59.00   1st Qu.:70.00
 Mode  :character   Median :122.0   Median :71.00   Median :68.00   Median :76.00
                    Mean   :121.3   Mean   :69.50   Mean   :66.88   Mean   :74.84
                    3rd Qu.:134.0   3rd Qu.:81.25   3rd Qu.:75.00   3rd Qu.:81.00
                    Max.   :206.0   Max.   :98.00   Max.   :96.00   Max.   :95.00
>
```

➢ Annexe 4

-Première corrélation entre bpm et energy :

```
data:  bpm and energy
t = 3.7719, df = 998, p-value = 0.0001715
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.05698257 0.17923262
sample estimates:
      cor
0.1185568
```

-Deuxième corrélation entre bpm et danceability :

```
data:  bpm and danceability
t = -3.5497, df = 998, p-value = 0.0004037
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.17245978 -0.05001362
sample estimates:
       cor
-0.1116605
```

-Troisième corrélation entre energy et danceability :

```
data:  energy and danceability
t = -4.1186, df = 998, p-value = 4.127e-05
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.18975163 -0.06782984
sample estimates:
       cor
-0.1292793
```

➔ Chacune des valeurs est faible donc on a peu de risque de colinéarité entre les variables.

➢ Annexe 5

```
> summary(reg1)

Call:
lm(formula = popularite ~ bpm + energy + danceability, data = don)

Residuals:
    Min      1Q  Median      3Q     Max
-39.052  -4.808   0.883   5.866  21.846

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  80.253061   2.341142  34.279  < 2e-16 ***
bpm           0.003042   0.010442   0.291   0.7708
energy       -0.125097   0.017202  -7.272 7.16e-13 ***
danceability  0.043550   0.020908   2.083   0.0375 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.558 on 996 degrees of freedom
Multiple R-squared:  0.05887,   Adjusted R-squared:  0.05603
F-statistic: 20.77 on 3 and 996 DF,  p-value: 4.689e-13
```

Le R^2 est très faible (=0.05) donc le modèle pas fiable.

Par exemple : popularite = 80.25 + 0.003*bpm – 0.12*energy + 0.04*danceability

➢ Annexe 6

```
> summary(reg_tot)

Call:
lm(formula = pop ~ bpm + energy + danceability + dB + val + acous +
    spch, data = Spotify)

Residuals:
    Min      1Q  Median      3Q     Max
-39.599  -4.914   1.037   5.831  22.242

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 82.447908   3.608881  22.846  < 2e-16 ***
bpm          0.001272   0.010647   0.119   0.9049
energy      -0.150744   0.027460  -5.490 5.12e-08 ***
danceability 0.023601   0.022932   1.029   0.3037
dB           0.173514   0.193897   0.895   0.3711
val          0.032405   0.014457   2.242   0.0252 *
acous        0.010255   0.016484   0.622   0.5340
spch         0.032024   0.030505   1.050   0.2941
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.542 on 992 degrees of freedom
Multiple R-squared:  0.06598,   Adjusted R-squared:  0.05939
F-statistic: 10.01 on 7 and 992 DF,  p-value: 3.947e-12
```

Le R^2 est très faible (=0.05) donc le modèle toujours pas fiable malgré l’ajout de toutes les variables.

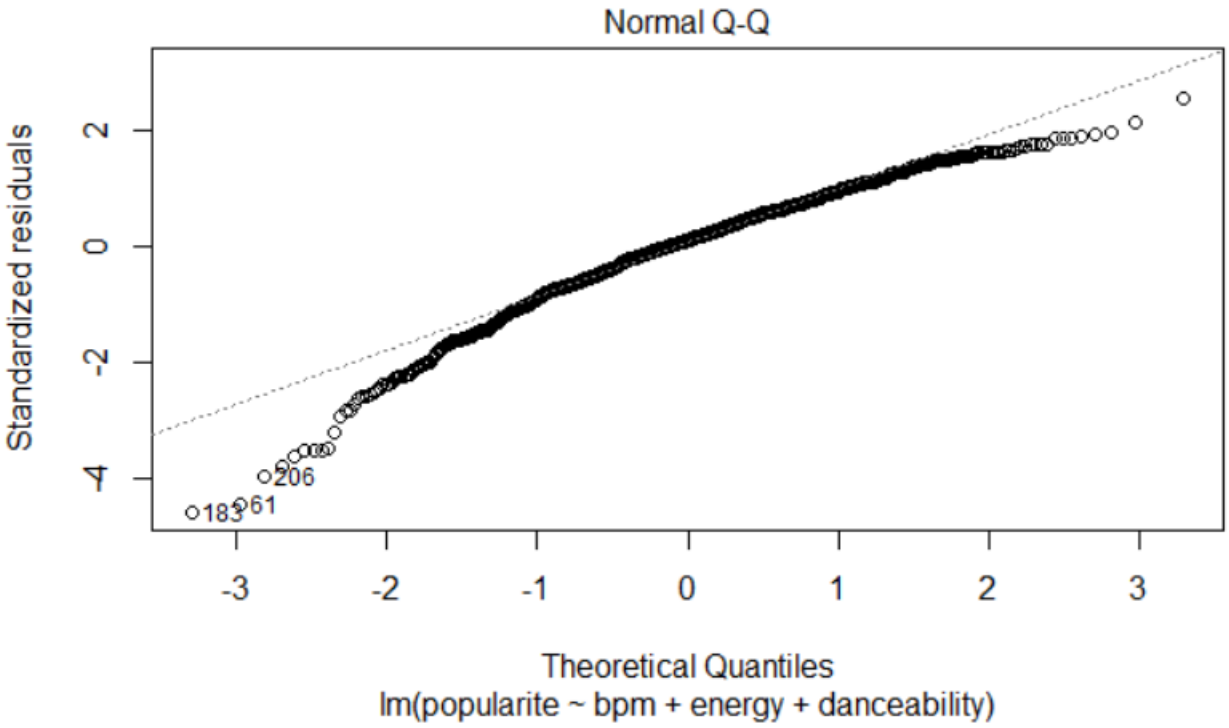
Annexe 7

```
> cooks.distance(reg1)[cooks.distance(reg1) > 1]
named numeric(0)
```

Il n'y a aucune distance de Cook supérieur à 1, donc cela signifie qu'il n'y a aucune valeur influente. On garde tous les individus.

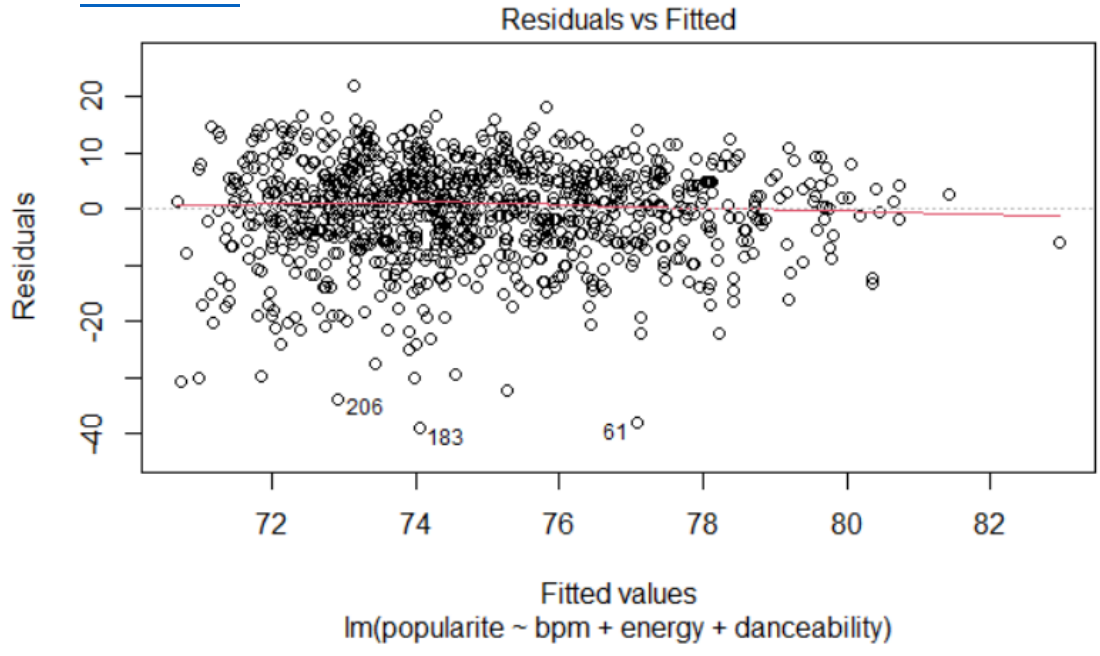
Annexe 8

Graphique Quantile-Quantile



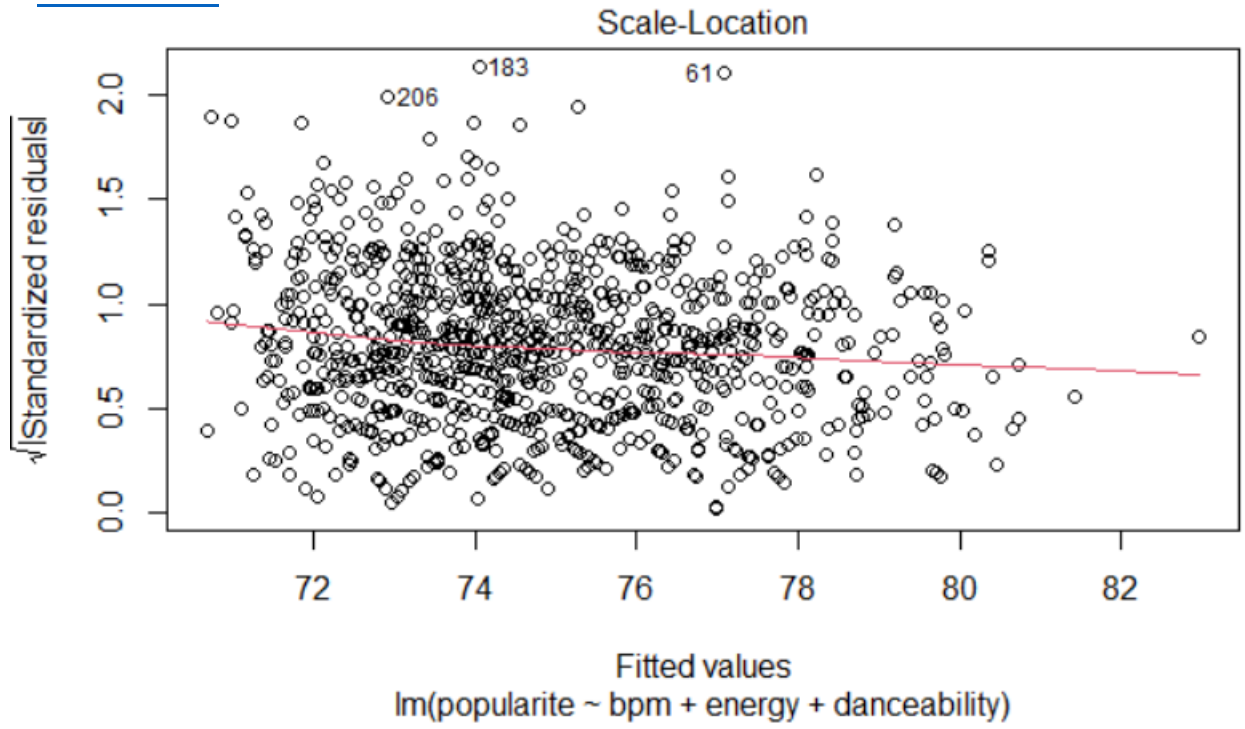
Nos individus ne suivent pas vraiment la diagonale, donc ils ne suivent pas une loi gaussienne.

Annexe 9



Les individus ne suivent pas la ligne rouge donc pas de problème d'orthogonalité. On a un nuage de point.

➢ Annexe 10



On a de nouveau un nuage de point, sans forme particulière. Les individus ne suivent toujours pas la ligne rouge donc pas de problème d'homoscédasticité.

PARTIE C : Codes

➔ R Studio

```
####### Biblioteque ######

library(readr)

######## Lecture et affectation ########

Spot <- read_csv("Spotify2010-2019_Top100.csv", col_types = cols(bpm = col_double(), nrgy = col_double(), dnce = col_double(), dB = col_double()))

Spotify <- as.data.frame(Spot) #convertis en data frame

str(Spotify)  #pour vérifier la nature des variables importées

names(Spotify)   #liste des noms de variables

don <- Spotify[,c(1,6:8,15)]  #on prend uniquement les colonnes que l'on va étudier

colnames(don) < -c("titre","bpm","energy","danceability","popularite") #on nomme le nom des colonnes

str(don) #format

attach(don)

View(don)

###### Analyse de donnees ############

summary(don)

#correlation entre chaque variables

cor.test(bpm, energy, method = "pearson")

cor.test(bpm, danceability, method = "pearson")

cor.test(energy, danceability, method = "pearson")

#correlation avec la variable explicative

cor.test(popularite, bpm, method = "pearson")

cor.test(popularite, energy, method = "pearson")

cor.test(popularite, danceability, method = "pearson")

reg_tot<-lm(pop~bpm+energy+danceability+dB+val+acous+spch, data=Spotify) #regression totale

summary(reg_tot)

reg1<-lm(popularite~bpm+energy+danceability, data=don) #regression pour notre étude

summary(reg1)

confint(reg1) #intervalle de confiance

hatvalues(reg1) #effet levier

cooks.distance(reg1)[cooks.distance(reg1) > 1]  #distance cook

plot(reg1) #graphiques
```

➔ SAS

```
/* BIBLIOTHEQUE (library) */
LIBNAME Spotify "/home/u62346911/Spotify";
%let accueil = /home/u62346911 ;
%let Spotify = &accueil./Spotify ;
options validvarname = v7 ;

/* appel de la macro variable accueil : &accueil. */
FILENAME BDD "&Spotify./Spotify2010-2019_Top100.csv";

/* importer un fichier csv */
PROC IMPORT datafile=BDD out=spotify.don dbms=csv replace;
        delimiter=',' getnames=yes;
run;

/*Presentation du contenu de la table */
PROC CONTENTS data=spotify.don;
run;

/* Description des variables */
PROC PRINT data=spotify.don;
run;

/* test corrélation des variables */
PROC CORR data=spotify.don;
        var pop bpm nrgy dnce;
run;

/* Regression multiple */
PROC REG data=spotify.don;
        model pop=bpm nrgy dnce;
        run;
```