Statistique inférentielle et analyse de données

Projet R - Le Top 50 Spotify

SOMMAIRE

Introduction

Écouter de la musique est une tâche que l'on peut qualifier de quotidienne pour une grande partie de la population mondiale. En effet, cela est devenu très répandu à tel point que les sites de streaming de musique comme Spotify, Deezer ou encore Apple Music établissent des playlists regroupant les morceaux préférés des auditeurs à travers le monde. Sur spotify, qui est l'exemple que nous traiterons dans ce projet, l'une de ces playlist porte le nom de "Top 50". Mais quels sont alors les critères qui font que ces morceaux sont autant appréciés ? Doivent-ils nécessairement être dansant pour être dans le top ? Ou encore la popularité du chanteur est-elle indispensable pour faire connaître un morceau ? C'est ce que nous allons tenter de déterminer ici !

Chaque morceau de musique à des caractéristiques qui lui sont propres et certaines sont méconnues du grand public alors qu'elles sont pourtant riches en informations. On peut citer l'énergie d'un morceau ou encore le nombre de battements par seconde/minute.

Le jeu de données trouvé provient du site internet Kaggle (https://www.kaggle.com/leonardopena/top50spotify2019). Il contient 50 échantillons (pour les 50 chansons de la playlist) . Les variables de la base de données caractérisant les morceaux sont :

- **Nom du morceau** (Track.Name) (Variable qualitative) : C'est tout simplement le nom de la chanson.
- **Nom de l'artiste** (Artist.Name) (Variable qualitative) : Cette fois c'est le nom de l'artiste.
- **Genre** (Genre) (Variable qualitative) : Une chanson peut être un morceau de rap, de pop, classique, etc… C'est cet aspect que donne le genre.
- **Battements par minutes** (Beats.Per.Minute) : C'est une unité de mesure utilisée pour exprimer le tempo de la musique et comme son nom l'indique donne le nombre de battements d'une chanson se produisant en une minute. Il prend ici ses valeurs entre 85 et 190.
- **Energie** (Energy) : Plus la valeur de cette variable est élevée, plus le son peut être qualifié d'énergique. Les valeurs qu'elle prend vont de 32 à 88.
- **Caractère dansant** (Danceability) : Comme pour l'énergie, sauf que cette fois cela traite de la facilité à danser sur le morceau. Les valeurs qu'elle prend vont de 29 à 90.
- **Intensité** (Loudness) : C'est, comme son nom l'indique, l'intensité du morceau. Plus sa valeur est élevée, plus le morceau est "bruyant". Les valeurs qu'elle prend vont de -11 à -2.
- **Vivacité** (Liveness) : C'est un paramètre quantifiant la qualité d'un morceau à être joué en live, en concert. Comme les précédentes, plus la valeur est élevée, plus la chanson en live sera agréable. Les valeurs qu'elle prend vont de 5 à 58.
- **Caractère motivant** (Valence) : Cette variable reflète à quel point le morceau influence positivement sur notre humeur.De même, plus la valeurs est élevée, plus c'est positif. Les valeurs qu'elle prend vont de 10 à 95.

- **Durée** (Length) : C'est, comme son nom l'indique, la durée du morceau. Elle est exprimée en secondes ici. Les morceaux durent, pour le plus court 115 secondes et 309 secondes pour le plus long.

- **Acoustique** (Acousticness..) : C'est la partie acoustique du morceau, qui correspond à son étude physique facile à effectuer. Plus la valeur est élevée, plus le morceau est "acoustique". Les valeurs prises par cette variable vont de 1 à 75.

- **Nombre de paroles** (Speechiness.) : Cela correspond au nombre de paroles présentes dans le morceau. Comme précédemment, plus la valeur est élevée, plus il y a de paroles. Les valeurs prises par cette variable vont de 3 à 46. (⚠ différent du nombre de mots dans la chanson)

- **Popularité** (Popularity) : C'est tout simplement relatif au nombre d'écoutes et au lieu où le morceau est écouté à travers le monde. Plus la valeur est élevée, plus le morceau peut être qualifié de populaire. Les valeurs prises par cette variable vont de 70 à 95.(Avoir un minimum aussi haut montre déjà qu'un morceau doit nécessairement être populaire pour être dans le top 50).

Statistiques descriptives:

Première visualisation des données

Afin de procéder à l'analyse des données, il faut ouvrir le jeu de données, à l'aide de la procédure suivante:

```
spotify <- read.table(file="C:/Users/PC/Desktop/R-ENSC/spotify.txt",
                      sep = ',',
                      header=TRUE
                      )
head(spotify)
str(spotify)

# On transforme le type de la variable "Table" en valeur numérique
spotify <- transform(spotify, Table = as.numeric(Table))
summary(spotify)
attach(spotify)
```

Figure 1: Lecture des données.

La fonction summary permet d'accéder à un aperçu rapide des différentes caractéristiques de la bdd comme l'illustre la capture suivante. De plus, la fonction head permet de s'assurer que les données ont été traitées correctement. On remarque alors 2 types de résultats différents, certaines caractéristiques sont remplies de valeurs numériques, summary les a alors analysées; d'autres sont composées de caractères, utilisables pour l'analyse.

	V8	V9	V10	V11	V12	V13
1	Loudness..dB..	Liveness	Valence.	Length.	Acousticness..	Speechiness.	P
2	-6	8	75	191	4	3
3	-4	8	61	302	8	9
4	-4	16	70	186	12	46
5	-8	8	55	198	12	19
6	-4	11	18	175	45	7

Figure 2: Premier aperçu avec la fonction head.

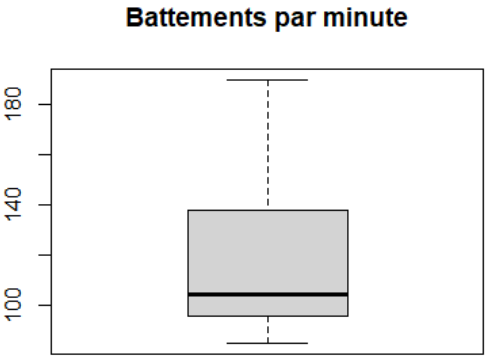
```
> summary(spotify)
       id         Track.Name        Artist.Name           Genre          Beats.Per.Minute
 Min.   : 1.00   Length:50          Length:50          Length:50          Min.   : 85.0
 1st Qu.:13.25   Class :character   Class :character   Class :character   1st Qu.: 96.0
 Median :25.50   Mode  :character   Mode  :character   Mode  :character   Median :104.5
 Mean   :25.50                                                           Mean   :120.1
 3rd Qu.:37.75                                                           3rd Qu.:137.5
 Max.   :50.00                                                           Max.   :190.0
     Energy       Danceability   Loudness..dB..      Liveness        Valence.        Length.
 Min.   :32.00   Min.   :29.00   Min.   :-11.00   Min.   : 5.00   Min.   :10.00   Min.   :115.0
 1st Qu.:55.25   1st Qu.:67.00   1st Qu.: -6.75   1st Qu.: 8.00   1st Qu.:38.25   1st Qu.:176.8
 Median :66.50   Median :73.50   Median : -6.00   Median :11.00   Median :55.50   Median :198.0
 Mean   :64.06   Mean   :71.38   Mean   : -5.66   Mean   :14.66   Mean   :54.60   Mean   :201.0
 3rd Qu.:74.75   3rd Qu.:79.75   3rd Qu.: -4.00   3rd Qu.:15.75   3rd Qu.:69.50   3rd Qu.:217.5
 Max.   :88.00   Max.   :90.00   Max.   : -2.00   Max.   :58.00   Max.   :95.00   Max.   :309.0
 Acousticness..   Speechiness.     Popularity
 Min.   : 1.00   Min.   : 3.00   Min.   :70.00
 1st Qu.: 8.25   1st Qu.: 5.00   1st Qu.:86.00
 Median :15.00   Median : 7.00   Median :88.00
 Mean   :22.16   Mean   :12.48   Mean   :87.50
 3rd Qu.:33.75   3rd Qu.:15.00   3rd Qu.:90.75
 Max.   :75.00   Max.   :46.00   Max.   :95.00
```

Figure 3: Aperçu des propriétés des données avec la fonction summary.
(notons que les valeurs pour id sont inutiles.)

Afin d'avoir un premier graphique synthétisant les valeurs obtenues sur la popularité du morceau, nous pouvons tracer un boxplot (boite à moustache) avec la fonction suivante:

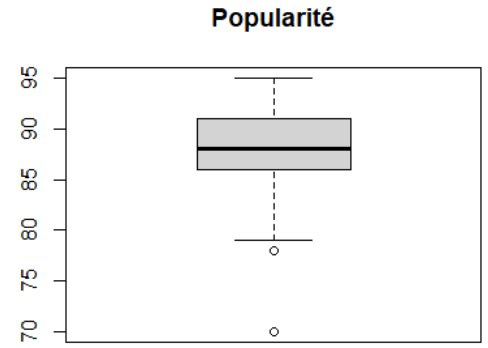
```
13  boxplot(Popularity)
14  title("Popularité")

16  boxplot(Beats.Per.Minute)
17  title("Battements par minute")
```



Dans ce cas-ci, nous pouvons observer que la médiane est plutôt proche de la valeur minimale alors que la valeur maximale est nettement supérieure au troisième quartile. On en déduit qu'il y a peu de morceaux ayant un grand nombre de BPM mais qu'ils ont plutôt un nombre de BPM de l'ordre de 115.

Figure 4 : Boxplot de la variable battements par minute



Nous pouvons ainsi observer que la médiane est plutôt centrée par rapport au premier et troisième quartile alors que la valeur minimale est nettement inférieure au premier quartile. Nous en déduisons qu'il y a peu de morceaux qui ne sont pas populaires et qu'ils ont en moyenne un niveau de popularité de l'ordre de 87%.

Figure 5 : Boxplot de la variable popularité

Tracer les histogrammes des différents critères nous permet d'observer rapidement la répartition des différentes valeurs.

```
HistoPerso <- function(x,from,to,title,xlab,ylab){
  #affichage de l'histogramme de x entre "from" et "to" par pas de "by"
  hist(x,prob=TRUE,breaks=seq(from,to,by=(to-from)/7),main=title,xlab=xlab,ylab=ylab)
  #affichage de l'estimation à noyau de la densité avec largeur de fenêtre width
  lines(density(x,width=2*(summary(x)[5]-summary(x)[2])), xlim=c(min(x)-sd(x),max(x)+sd(x)))
}
# afficher 4 graphiques sur une même fenêtre
par(mfrow=c(2,2))

HistoPerso(Popularity,70,95,"Histogramme de la popularité ","Popularité","Densité")
HistoPerso(Beats.Per.Minute,85,190,"Histogramme  des BPM","BPM","Densité")
HistoPerso(Danceability,29,90,"Histogramme du caractère dansant ","Caractère dansant","Densité")
HistoPerso(Acousticness..,1,75,"Histogramme de l'acoustique","Acoustique","Densité")
```

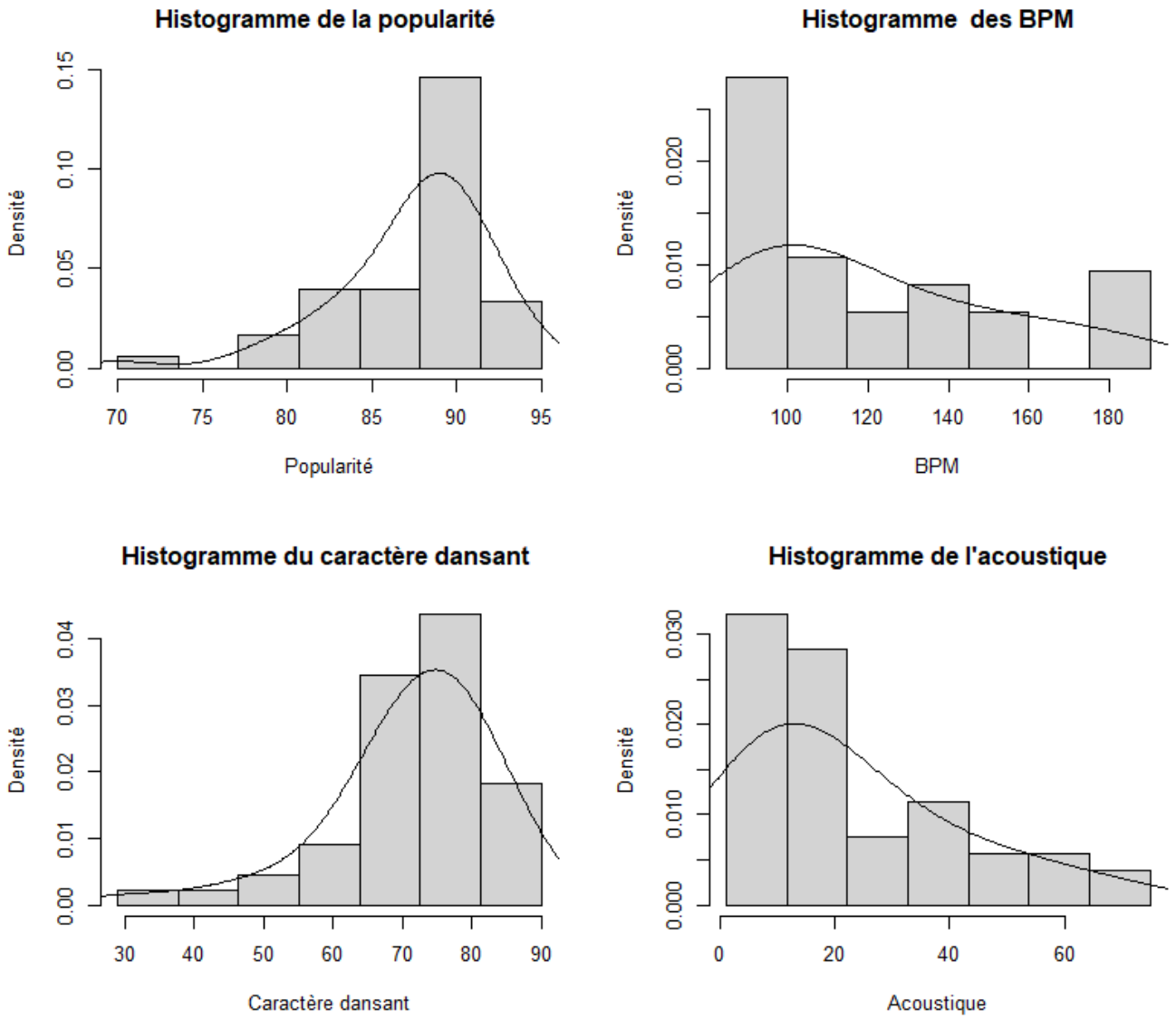


Figure 6 : Histogramme des 4 variables: popularité, BPM, caractère dansant et de l'acoustique

Aucune des histogrammes choisis ne semble montrer qu'une de nos variables suit une loi normale. On testera néanmoins ces hypothèses dans la partie "Analyse inférentielle" pour la variable popularité qui semble être celle s'en rapprochant le plus(avec un fort pic au niveau de la partie centrée).

On remarque un fort pic sur l'histogramme de la popularité au niveau de la valeur 90 renforçant l'idée que les morceaux présents dans ce top 50 sont à priori très populaires. Au contraire, pour celui des BPM le fort pic se situe pour des valeurs basses, appuyant l'idée que les morceaux présents dans ce top ont un tempo plutôt lent.

De même que pour la popularité, le caractère dansant semble être nécessaire pour être dans le top avec deux forts pics pour des valeurs élevées: 70 et 80 environ. Alors que pour l'acoustique qui semble suivre le comportement de la variable BPM, les pics sont vers des valeurs basses (entre 1 et 20).

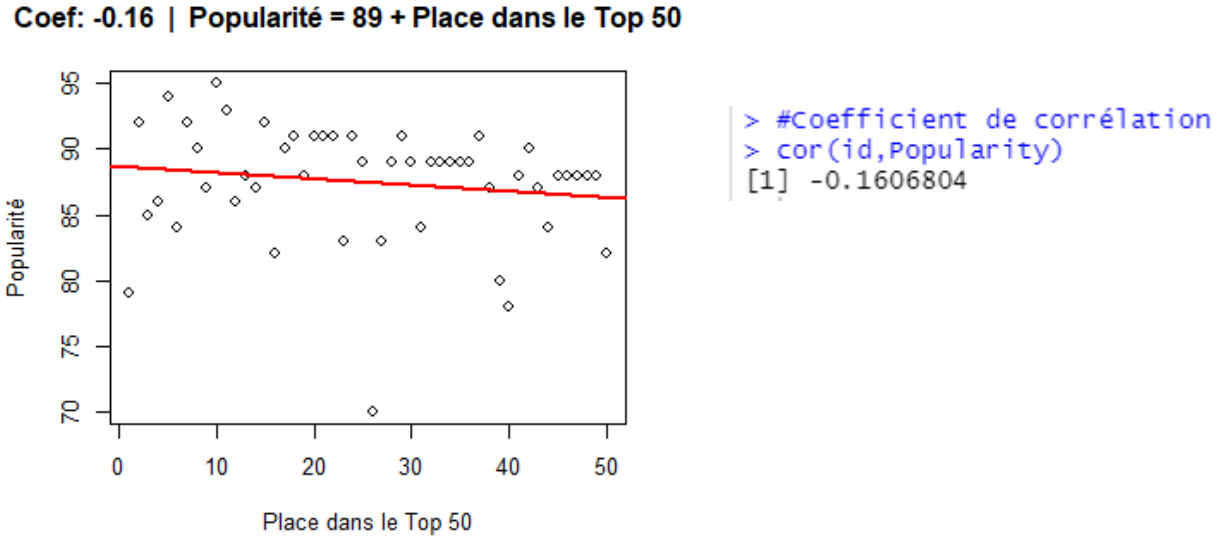
Dans la suite de notre étude il sera alors utile de vérifier ces premières observations. A savoir que la place d'un morceau dans le top 50 est relative à sa popularité, son nombre de BPM, son côté dansant et son acousticité. On pourra également vérifier dans une partie d'étude en composantes

principales (ACP) si les comportements des variables sont liés entre eux, comme par exemple plus un morceau est populaire plus il est dansant et plus son nombre de BPM est bas.

→ Lien de la popularité avec la place dans le classement

```
#On va chercher à établir qu'elle sont les liens entre la position (plus ou moins élevée )
#d'un morceau dansle top 50 et les autres parmètres. Pour cela on définit cette fonction permettant
#de représenter un paramètre en fonction de l'autre et ainsi trouver une potentielle corrélation.
LinReg <- function(x,y,xlab,ylab){
  cor = cor.test(x,y, method="pearson")
  reg <- lm(y ~ x)
  coef = coefficients(reg)

  plot(x,y, ylab=ylab, xlab=xlab)
  abline(reg, col='red', lwd=2)
  title(paste("Coef:",round(cor(x,y),2)," | ",
              ylab,"=", round(coef[1],0),
              "+",xlab,"*",round(coef[2],0)))
} # Calcul du coef de corrélation, trace un
# graphique avec une approximation linéaire.
LinReg(id,Popularity,"Place dans le Top 50","Popularité")
```

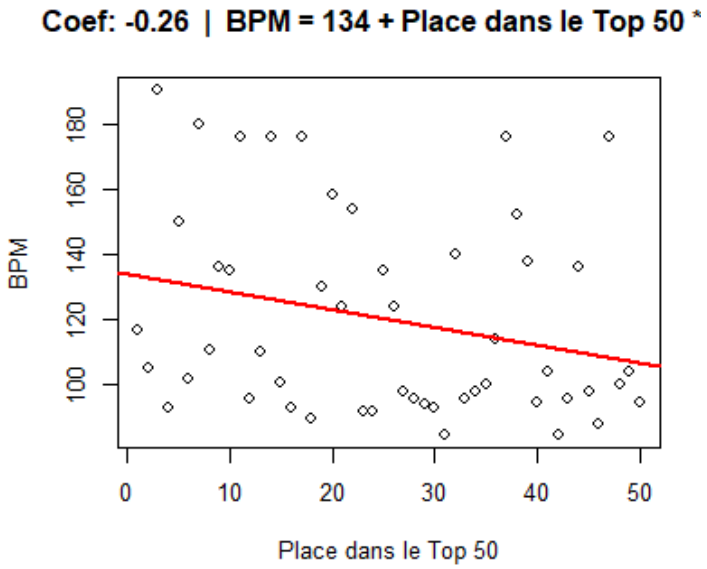



```
> #Coefficient de corrélation
> cor(id,Popularity)
[1] -0.1606804
```

Figure 7 : coefficient de corrélation de la popularité en fonction de la place dans le Top 50

Après avoir tracé la place dans le classement en fonction de la popularité, nous obtenons le résultat suivant. Il illustre un coefficient négatif proche de 0 (=-0,16). On peut donc en déduire que, contrairement à notre hypothèse de dépendance , un morceau n'a pas nécessairement besoin d'être populaire pour être en haut du top 50(il faut néanmoins qu'il dépasse 70 pour y être).

→ Lien du nombre de BPM avec la place dans le classement




```
> #Coefficient de corrélation
> cor(id,Beats.Per.Minute)
[1] -0.2591926
```

Figure 8 : coefficient de corrélation des BPM en fonction de la place dans le Top 50

Après avoir tracé la place dans le classement en fonction du nombre de BPM, nous obtenons le résultat suivant. Il illustre un coefficient négatif proche de 0 (=-0,25). Ainsi, comme précédemment, on peut donc en déduire que, contrairement à notre hypothèse de dépendance , un morceau n'a pas nécessairement besoin d'avoir un nombre bas de BPM pour être en haut du top 50.

De même pour le caractère dansant et pour la partie acoustique, aucune des variables ne semble corrélés avec la place dans le top 50.

→ Lien des autres caractéristiques entre elles.

Même si la place dans le top 50 semble être indépendante des différentes variables, ces dernières peuvent présenter des comportements similaires entre elles.

Nous allons donc regarder s'il existe des corrélations entre des variables.Pour cela nous réaliserons des test du Khi-deux pour mettre en évidence une potentiel dépendance entre les variables, notamment entre le caractère dansant du morceau et son nombre de BPM, ce qui semble tout à fait pertinent, mais aussi entre sa popularité et son caractère dansant.

```
> chisq.test(Danceability,Beats.Per.Minute)

        Pearson's Chi-squared test

data:  Danceability and Beats.Per.Minute
X-squared = 916.11, df = 840, p-value = 0.0344
```

```
> chisq.test(Danceability,Popularity)

        Pearson's Chi-squared test

data:  Danceability and Popularity
X-squared = 501.24, df = 476, p-value = 0.2045
```

On obtient une p-value= 0.0344 << 0.05 pour le caractère dansant et le nombre de battements par minutes , on peut ainsi conclure qu'il existe un lien entre ces deux variables. Au contraire pour le caractère dansant et la popularité on a une p-value= 0.2045 >>0.05. Cela montre que les deux variables n'ont vraisemblablement aucun lien.

Analyse inférentiel

D'après les estimations à noyau de la densité sur l'histogramme de la popularité, on peut émettre comme hypothèse que l' échantillon "Popularity"suit une loi normale.
Nous avons donc fait un test Shapiro-Wilk avec pour hypothèse nulle est H0 : l'échantillon "Popularity" suit une loi normale. L'hypothèse alternative est H1 : non H0.

On se donne comme risque de première espèce 5%.

```
> shapiro.test(Popularity)

        Shapiro-Wilk normality test

data:  Popularity
W = 0.89305, p-value = 0.0002855
```

On obtient pour notre échantillon une p-value < 2.2*e-16 << 0.05 donc H0 est rejeté. Cet échantillon ne suit pas une loi normale.

Analyse en Composantes Principales

L'ACP est une méthode d'analyse de données multivariées permettant de décrire graphiquement et numériquement les données. Cette méthode permet de réduire le nombre de variables quantitatives à étudier afin de faciliter la visualisation.

Nous supposons que les variables quantitatives ne sont pas toutes linéairement dépendantes entre elles.

Dans notre cas, nos données comportent 11 variables quantitatives et 3 qualitatives. L'ACP permettra de porter l'étude sur un nombre de variables quantitatives inférieur, et d'essayer d'observer des tendances selon les variables qualitatives.

```
# On charge le plugin PCAmixdata, qui permet de réaliser une ACP
require(PCAmixdata)

# On stocke le résultats de l'ACP dans la variable res.
# X.quanti : Variables Quantitatives  X.quali : Variables Qualitatives
res<-PCAmix(X.quanti=spotify[c(1,5,6,7,8,9,10,11,12,13,14 )],X.quali = spotify[, c(2,3,4 )],graph=FALSE)

# On affiche toutes les valeurs propres de la diagonalisation de la matrice de nos variables.
round(res$eig,digit=2)
```

Pour déterminer le nombre de dimensions on va utiliser le **critère du coude**. Les λi sont les valeurs propres de la dimension i, les Ɛi sont les différences premières et δi les différences secondes.

Ɛ1=λ1-λ2=4,97-4,36=0,61 Ɛ2= λ2-λ3=4,36-4,02=0,34

δ1= Ɛ1- Ɛ2=0,61-0,34

Et ainsi de suite…

Finalement on obtient i=5, on gardera donc 5 dimensions pour notre ACP.

```
> round(res$eig,digit=2)
       Eigenvalue Proportion Cumulative
dim 1        4.97       4.25       4.25
dim 2        4.36       3.73       7.98
dim 3        4.02       3.43      11.41
dim 4        3.78       3.23      14.64
dim 5        3.61       3.08      17.72
dim 6        3.60       3.08      20.80
dim 7        3.46       2.96      23.76
dim 8        3.31       2.83      26.58
dim 9        3.17       2.71      29.29
dim 10       3.11       2.66      31.95
dim 11       3.04       2.60      34.54
dim 12       3.00       2.56      37.11
dim 13       3.00       2.56      39.67
dim 14       3.00       2.56      42.23
dim 15       3.00       2.56      44.80
```

Maintenant que l'on a déterminé le nombre de dimensions que l'on souhaite conserver, on va regarder la qualité de représentation de nos variables quantitatives sur ces dernières.

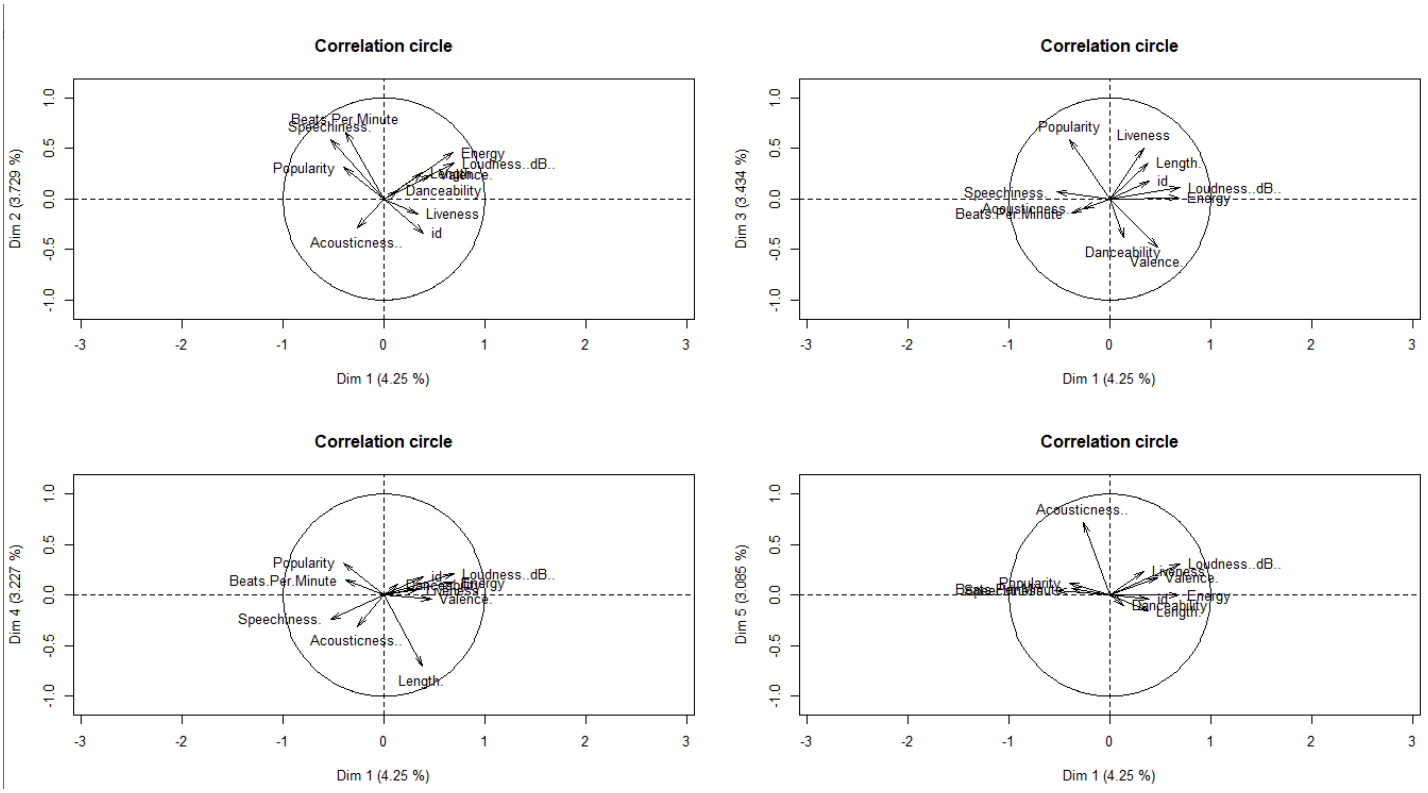
```
> # Affiche la qualité de représentation
> # des variables sur les dimensions.
> round(res$quanti$cos2,digit=3)
                dim 1 dim 2 dim 3 dim 4 dim 5
id              0.152 0.115 0.031 0.034 0.001
Beats.Per.Minute 0.144 0.441 0.020 0.021 0.004
Energy          0.471 0.211 0.000 0.015 0.000
Danceability    0.020 0.007 0.147 0.011 0.010
Loudness..dB..  0.479 0.124 0.012 0.044 0.098
Liveness        0.111 0.021 0.250 0.003 0.057
Valence.        0.220 0.058 0.230 0.001 0.031
Length.         0.145 0.066 0.126 0.500 0.025
Acousticness..  0.069 0.081 0.009 0.095 0.517
Speechiness.    0.278 0.343 0.005 0.058 0.002
Popularity      0.162 0.097 0.344 0.103 0.015
```

Sur ce tableau, on voit les qualités de représentation des différentes variables sur les différentes dimensions. On voit que la variable *Energy* est plutôt bien représentée (47%) sur la dimension 1, *Valence* moyennement sur les dimensions 1 et 3 (22% et 23 %), *Beats.Per.Minute* bien représentée (41%) sur la dimension 2 et *Acousticness.. (51%)* sur la dimension 1.Cette dernière est d'ailleurs la seule à être bien représentée sur la dimension 5.

→ On voit dès lors que les couples de dimensions (1,2), (1,3), (1,4), (1,5), seront les plus représentatifs des variables. Cela se vérifie en traçant les cercles de corrélations des 5 dimensions entre elles.

```
par(mfrow=c(2,2))
plot(res,axes=c(1,2),choice="cor")
plot(res,axes=c(1,3),choice="cor")
plot(res,axes=c(1,4),choice="cor")
plot(res,axes=c(1,5),choice="cor")
```

Figure 9 : Cercles de corrélations selon plusieurs dimensions



On voit que dans la première figure,en haut à gauche, les variables id et Speechless ont un angle de quasiment 180° entre eux. On peut en déduire une forte corrélation (non parfaite car ils ne sont pas projetés à 100%, et n'ont pas un angle exactement égal à 180°). Ainsi plus le morceaux est haut dans le classement moins il contient de paroles et inversement. On voit également que la variable Energy est relativement bien représentée, et perpendiculaire aux flèches de Beats.Per.Minute et id. On peut déjà conclure à une corrélation plutôt faible entre Energy et Id, et Energy & Beats.Per.Minutes. Intuitivement, ce résultat semble étrange et montre qu'un morceau n'a pas besoin d'avoir un fort tempo pour être énergique ou encore un morceau n'a pas besoin d'être énergique pour être dans le haut du top 50. Notons que la plupart des données sont assez mal représentées sur les différents cercles et ce type d'analyse reste approximatif, cela permet de se faire une idée et on aurait tout aussi bien pu faire le même type de raisonnement pour chacun des cercles(difficile avec le

dernier où seul une variable est bien représentée).

De fait de la grande quantité de variable qualitative dans nos données, il est intéressant de représenter les différents individus de l'étude sur les plans (1,2) et (1,3) (les plus intéressants). De fait de la restriction de pages du rapport, nous ne présenterons que les graphiques les plus représentatifs et cela pour la variable qualitative genre qui reste la plus pertinente.

```
plot(res,axes=c(1,2),choice="ind",
     coloring.ind = spotify$Genre, label=FALSE)
plot(res,axes=c(1,3),choice="ind",
     coloring.ind = spotify$Genre, label=FALSE)
```

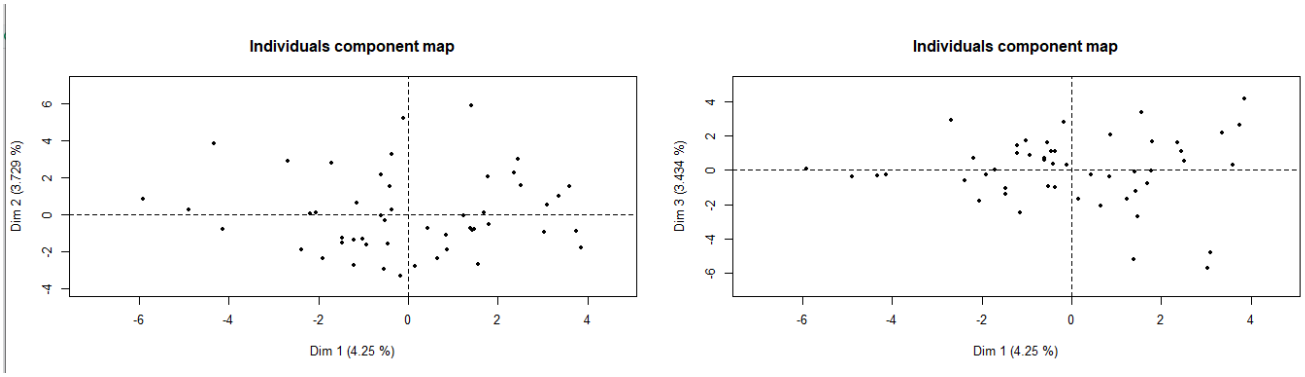


Figure 10 : Nuage de points de la variable qualitative Genre selon les plans 1-2 et 1-3

Globalement, aucun genre spécifique ne semble se détacher des autres, autrement dit peu importe le type de musique, le top 50 est accessible à tous.

Conclusion

Cette étude statistique nous a permis de confronter nos hypothèses aux données réelles. Nous avons pu affirmer la pensée générale ou réfuter des idées reçues. Par exemple, plus un morceau est populaire plus il a de chance d'être dans le haut du classement; cette idée a pu être évincée dans cette étude. On peut aussi citer le fait que plus un morceau à de paroles moins il a de chances d'être à une place élevée dans le Top 50

Certaines découvertes semblent assez surprenantes et ont tendance à beaucoup trop diverger de la réalité comme le lien entre le caractère dansant et la place dans le classement. Ces données sont basées sur le Top 50 spotify de l'année 2019 qui est peut être différent de ce que l'on a l'habitude de remarquer maintenant d'où ces différentes choses que l'on a constaté.

Pour conclure, en effectuant des recherches pour comprendre les valeurs des variables, nous avons renforcé nos connaissances sur les critères qui permettent de qualifier un morceau de musique. Comme c'est un secteur qui nous intéresse tout particulièrement tous les deux et qui est souvent source de débat nous espérions répondre à la question : Que faut -il à un morceau pour être dans ce genre de classement ? Nous y avons partiellement répondu mais cela à été très enrichissant.