

2020 /2021 (2º semestre)

Métodos de Aprendizagem Não Supervisionados

Professor José Dias

**Análise De Componentes Principais E De Clustering De  
Músicas No Top Billboard Entre 2010 E 2019, A Partir  
De Características Definidas Pelo Spotify**

Turma LCD-PL

Catarina Castanheira N. 92478

João Martins N. 93259

Joel Paula N. 93392

## Índice

Introdução.....	3
Dados .....	4
Análise de Componentes Principais (PCA) .....	5
Verificação da Adequabilidade de uma Análise de Componentes Principais neste Contexto .....	5
Seleção do Número de Componentes Principais.....	7
4 Componentes Principais .....	8
5 Componentes Principais: .....	9
6 Componentes Principais: .....	9
Decisão sobre número de componentes a usar .....	9
Interpretação e Nomeação das Componentes Principais.....	10
Análise de <i>clustering</i> das observações.....	10
<i>Clustering</i> probabilístico usando GMM .....	10
<i>Clustering</i> probabilístico usando GMM e 4 clusters .....	12
<i>Clustering</i> usando um método hierárquico .....	12
<i>Clustering</i> usando métodos Partitivos .....	14
Cluster com K-médias ( <i>K-means</i> ) .....	14
Cluster com PAM ( <i>Partition Around Medoids</i> ).....	15
Conclusão .....	17
Referências.....	18
Anexo I – KMO .....	19
Anexo II - Análise de Componentes Principais – 10 Componentes .....	20
Anexo III - <i>Scree-Plot</i> : Número de Componentes vs. Valores Próprios .....	21
Anexo IV - Análise de Componentes Principais – 5 Componentes .....	22
Anexo V – Cluster GMM vs Género musical (top.genre) .....	23
Anexo VI – Cluster GMM vs Artistas .....	26

## Introdução

O Spotify é uma plataforma Sueca de *streaming* de áudio e prestadora de serviços de *media*.

Este relatório foca-se nos dados obtidos por esta plataforma relativamente às músicas que estiveram no top 10 nos EUA por ano desde 2010 até 2019. Este *dataset* foi retirado de:

<https://www.kaggle.com/leonardopena/top-spotify-songs-from-20102019-by-year> (Henrique, 2019)

e tem 15 variáveis a serem exploradas relativas a um conjunto de características musicais definidas pelo Spotify (Lamere, 2016):

- **ID (X)** – ID da música
- **Title** – Título da música
- **Artist** – Artista da música
- **Genre (top.genre)** – Género da música
- **Year** – Ano em que a música apareceu no top
- **Beats Per Minute (bpm)** – Batida da música
- **Energy (nrgy)** – Energia da música
- **Danceability (dnce)** – Facilidade com que se dança com a música
- **Loudness (dB)** – Volume da música
- **Liveness (live)** – Probabilidade da música ter sido gravada ao vivo
- **Valence (val)** – Quão positiva é a música
- **Acousticness (acous)** – Quão acústica é a música
- **Speechiness (spch)** – Medida representativa da quantidade de palavras que são ditas na música
- **Popularity (pop)** – Popularidade da música
- **Duration (dur)** – Duração da música

O objetivo deste projeto é fazer uma Análise de Componentes Principais (PCA) e de *Clustering* do dataset em questão.

A Análise de Componentes Principais é relevante para este projeto uma vez que temos um *dataset* com muitas variáveis que poderá ser representado por um conjunto menor de variáveis que contém a maioria da informação do *dataset* original, com o intuito de observar tendências e clusters, ver as relações entre as variáveis e observações, facilitar a interpretação dos resultados e eliminar a multicolinearidade.

Para a Análise de Componentes Principais iremos começar por averiguar se este método de pré-processamento é adequado ao *dataset*, analisando depois o número de Componentes Principais (PC) ideais a serem extraídos. Identificadas as componentes principais, passamos para a análise de *Clustering*, na qual testamos diferentes métodos: método probabilístico com o GMM (*Gaussian Mixture Model*), método hierárquico aglomerativo usando a distância de *Ward* e métodos partitivos, usando *K-Means* e o PAM (*partition around medoids*).

Tanto no caso de componentes principais, como na análise de *clusters* fazemos uma interpretação

da informação latente em cada *cluster*/componente, a partir tanto nas variáveis de *Profile*, como das de *Input*.

## Dados

O *dataset* é constituído por 15 variáveis e 603 observações. Todas as variáveis são do tipo inteiro à exceção das variáveis *title*, *artist* e *top.genre* que são categóricas e que serão as usadas para *Profile*. Os géneros de música deste *dataset* estão reunidos maioritariamente na categoria *Dance Pop* (54% das observações) seguido das categorias *Pop* e *Canadian Pop* (10% e 6% das observações no total).

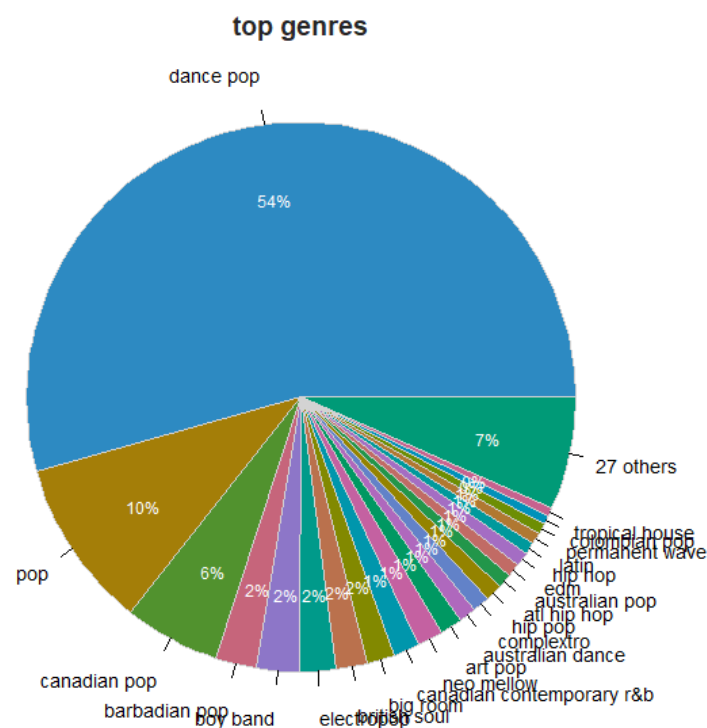


Figura 1 Proportão dos géneros musicais na amostra

A variável *year* tem valores de 2010 até 2019 (que correspondem aos anos em que as músicas apareceram no top), sendo que os anos com menor e maior quantidade de observações são 2019 e 2015 respetivamente. Na tabela abaixo estão algumas das medidas descritivas das variáveis inteiras:

Variável INPUT	Mínimo	Máximo	Média	Desvio Padrão
<b>Bpm</b>	43	206	118.7	24.34
<b>Nrgy</b>	4	98	70.62	16.07
<b>Dnce</b>	23	97	64.49	13.13
<b>DB</b>	-15	-2	-5.488	1.70
<b>Live</b>	2	74	17.8	13.09
<b>Val</b>	4	98	52.31	22.43

<b>dur</b>	134	424	224.7	34.16
<b>acous</b>	0	99	14.35	20.78
<b>spch</b>	3	48	8.372	7.48
<b>pop</b>	0	99	66.63	14.27

Tabela 1 Variáveis de Input

## Análise de Componentes Principais (PCA)

### Verificação da Adequabilidade de uma Análise de Componentes Principais neste

#### Contexto

De entre as vantagens de se utilizar a análise de componentes principais encontramos a redução de dimensionalidade e a eliminação da multicolinearidade, por exemplo. Contudo, para no fim do processo alcançarmos uma redução significativa, isto é, que obtemos um número de componentes “muito inferior” ao número de variáveis originais, temos de garantir que existem relações entre estas. Se não encontrarmos correlações entre as variáveis que compõem o *dataset* original, a PCA perde a sua utilidade.

Neste sentido, torna-se essencial computar uma matriz de correlações entre as nossas variáveis de *input* (*bpm*, *nrgy*, *dnce*, *dB*, *live*, *val*, *dur*, *acous*, *spch*, *pop*) e verificar a existência de correlações moderadas a fortes (i.e., com um valor absoluto igual ou superior a 0.30). Na tabela abaixo conseguimos encontrar esta informação:

	bpm	nrgy	dnce	dB	live	val	dur	acous	spch	pop
bpm	1.00	0.10	-0.18	0.05	0.07	0.00	-0.03	-0.12	0.05	-0.02
nrgy	0.10	1.00	0.14	0.66	0.18	0.40	-0.15	-0.58	0.10	-0.09
dnce	-0.18	0.14	1.00	0.13	-0.04	0.49	-0.18	-0.25	-0.04	0.08
dB	0.05	0.66	0.13	1.00	0.06	0.34	-0.17	-0.35	-0.06	0.01
live	0.07	0.18	-0.04	0.06	1.00	0.02	0.10	-0.10	0.14	-0.09
val	0.00	0.40	0.49	0.34	0.02	1.00	-0.26	-0.25	0.12	0.02
dur	-0.03	-0.15	-0.18	-0.17	0.10	-0.26	1.00	0.09	0.05	-0.11
acous	-0.12	-0.58	-0.25	-0.35	-0.10	-0.25	0.09	1.00	0.00	0.02
spch	0.05	0.10	-0.04	-0.06	0.14	0.12	0.05	0.00	1.00	-0.05
pop	-0.02	-0.09	0.08	0.01	-0.09	0.02	-0.11	0.02	-0.05	1.00

Figura 2 matriz de correlações entre variáveis de Input

Sendo uma matriz simétrica, basta analisar o conjunto de valores acima da diagonal. Na matriz foram destacados os valores absolutos superior a 0.30. É possível verificar que existem algumas correlações moderadas/fortes entre as variáveis, nomeadamente entre: *dB* e *nrgy*, *val* e *nrgy*, *val* e *dnce*, *val* e *dB*, *acous* e *nrgy*, e *acous* e *dB*. As correlações nos últimos dois pares são negativas. Contudo, também conseguimos perceber que existem variáveis independentes entre si, ou muito próximas da independência (veja-se os exemplos dos pares entre *val* e *bpm* ou *pop* e *dB*, com um *r*

igual a 0 e igual a 0.01, respetivamente). Metade destas variáveis *input* – *bpm*, *live*, *dur*, *spch*, *pop* – acabam mesmo por não ter nenhuma correlação moderada ou forte com qualquer outra.

Apesar de não encontramos nesta análise de correlações impeditivos para a análise das componentes principais, deixamos nota que o facto de termos muitas variáveis sem correlações moderadas ou fortes com outras poderá criar impacto na variabilidade total das variáveis originais que conseguirá ser retida pelas componentes principais que sejam determinadas mais à frente. Para a determinação da adequabilidade de se analisar componentes principais, também executámos outros testes, além deste estudo inicial das correlações.

Um deles foi o teste de Bartlett, que testa a hipótese nula de que as variáveis de *input* não se correlacionam entre si na população de onde foram retiradas. Neste especto, os nossos resultados demonstram que a hipótese nula não é rejeitada.

Outro teste que se aplicou foi a determinação do índice KMO (Kaiser-Meyer-Olkin Measure of Sampling Adequacy). Este indicador fornece informação sobre a proporção da variância das variáveis que pode estar a ser alvo de um fator subjacente, sendo que o ideal é obter-se um valor superior a 0.7. No nosso caso, o índice geral obtido foi de 0.62. Os dados abaixo, que detalham este índice por variável, indicam mesmo que só 6 das 10 têm um índice superior a 0.6 (um valor já considerado medíocre), e somente uma destas apresenta um índice superior ao ideal de 0.70 (a variável *dur*):

bpm	nrqy	dnce	dB	live	val	dur	acous	spch	pop
0.47	0.60	0.54	0.65	0.65	0.66	0.76	0.65	0.36	0.50

Figura 3 índice KMO por variável de Input

Comparando estes dados com aquela informação já obtida na análise da matriz de correlações, verificamos que *bpm*, *spch*, *pop*, que aqui obtêm um índice KMO baixo também já revelavam a inexistência de correlações moderadas/fortes com qualquer outra variável. Relativamente a *live* e *dur*, revelam aqui a probabilidade de existência de um fator subjacente que explique parte da sua variância.

Finalizadas estas primeiras análises, e antes de iniciarmos a análise das componentes principais, uma vez que cada uma das variáveis é representada por medidas distintas, com escalas diferenciadas, é necessário proceder à estandardização das mesmas. Daqui em diante, será utilizado então um *dataset* composto pelas variáveis de input estandardizadas, todas elas com média igual a 0 e variância igual a 1.

## Seleção do Número de Componentes Principais

Na determinação do número de componentes principais a serem retidos, foram considerados alguns critérios complementares: o critério de Kaiser e a análise do gráfico com a representação do valor próprio de cada componente criada, e a análise da variância acumulada.

Sendo que o valor próprio de cada componente corresponde à variância explicada por essa mesma componente, e estando os dados estandardizados, significa então que se dado valor próprio de uma componente for superior a 1 ela explicará uma maior variabilidade nos dados que a que conseguimos obter com qualquer variável original de *input* (que têm cada uma uma variância igual a 1). No sentido contrário, se o valor próprio de dada componente for inferior a 1, significa que ela tem pior poder explicativo da variabilidade que uma variável original. Uma vez que o objetivo da PCA passa por encontrar componentes em número inferior ao das variáveis de *input* e que expliquem o máximo da variabilidade dos dados originais, então interessa-nos reter à partida todas aquelas componentes que a ela tenham associados valores próprios iguais ou superiores a 1. No nosso caso, conseguimos identificar 4 componentes principais, através deste critério:

[1] 2.59 1.47 1.13 1.02 0.91 0.81 0.80 0.65 0.39 0.24

Figura 4 (critério de Kaiser) valores próprios associados a cada uma das variáveis de *Input*

Com o segundo critério utilizado, que pressupõe a análise do “*Scree plot*” associado a estes dados dos valores próprios, verificamos que o maior decréscimo nos valores próprios é obtido através das duas primeiras componentes (a terceira componente tem já um valor próprio próximo do das restantes) (ver Anexo III - *Scree-Plot*: Número de Componentes vs. Valores Próprios). Contudo, com a análise da variância explicada através de duas componentes – 0.41 (ver Figura 5 Tabela de Valores próprio e variância explicada) – percebemos que considerar somente este número de componentes não faz sentido, pelo que descartamos já qualquer análise subsequente com 2 componentes.

O terceiro critério utilizado foi o estudo da variância total explicada obtida a partir das componentes que considerássemos reter. Nesta análise considerámos que seria interessante reter pelo menos 70% da informação inicial dos dados, ou seja, escolher um número de componentes principais cuja variância explicada fosse de pelo menos 0.70.

Com base na imagem abaixo, verificamos que a variância acumulada explicada com 4 componentes é de somente 0.62. O objetivo de 0.70 é alcançado se incluirmos uma quinta componente principal. Neste cenário, com uma variância acumulada de 0.71, é necessário ter em mente que mesmo com as 5 componentes principais teremos a perda de 29% da variabilidade inicial dos dados. Contudo, a escolha das componentes a reter ficam sempre subjacentes a um último critério: o do custo-benefício. Será de considerar acrescentar uma 6ª componente para ficarmos com uma variância

total explicada de 0.79 (perda de 21% da variabilidade inicial)? E 7 componentes, que significa uma perda de somente 13%?

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
SS loadings	2.59	1.47	1.13	1.02	0.91	0.81	0.80	0.65	0.39	0.24
Proportion Var	0.26	0.15	0.11	0.10	0.09	0.08	0.08	0.06	0.04	0.02
Cumulative Var	0.26	0.41	0.52	0.62	0.71	0.79	0.87	0.94	0.98	1.00
Proportion Explained	0.26	0.15	0.11	0.10	0.09	0.08	0.08	0.06	0.04	0.02
Cumulative Proportion	0.26	0.41	0.52	0.62	0.71	0.79	0.87	0.94	0.98	1.00

Figura 5 Tabela de Valores próprio e variância explicada

Com base na análise até ao momento, revelou-se interessante “testar” três cenários: extração de 4 componentes principais, 5 componentes principais, e 6 componentes principais. Nestes cenários, cujos resultados são discutidos de seguida, procedemos à rotação das componentes principais pelo método *Varimax*, com o objetivo de interpretar mais facilmente os *loadings* de cada componente, e analisámos também a comunalidade de cada variável *input*. Idealmente, cada variável *input* estará associada a um único componente, através do *loading* mais elevado. No que diz respeito à comunalidade, sendo ela representativa do quão uma dada variável original está representada nas componentes principais, e sendo também o objetivo com a PCA a retenção do máximo de informação, será interessante obter valores o mais próximo de 1 para cada variável de *input*.

#### 4 Componentes Principais

Neste caso, olhando para a tabela dos *loadings*, verifica-se que as variáveis *bpm*, *nrgy*, *dnce*, *dB*, *acous* e *spch* se encontram bem representadas num dos componentes principais. O mesmo não acontece com as variáveis *live*, *val*, *dur* e *pop*. Já olhando para a tabela das comunalidades, verificamos que as variáveis *live*, *dur* e *pop* têm valores muito baixos: na realidade, a variabilidade retida destas variáveis dos dados originais corresponde somente a proporções de 0.42, 0.49 e 0.44, respetivamente.

Loadings:										
	RC1	RC4	RC2	RC3						
bpm	0.22	-0.70	0.30	0.25						
nrgy	0.89			0.14						
dnce	0.20	0.77	0.25		bpm	nrgy	dnce	dB	live	val
dB	0.80			-0.11	0.69	0.81	0.69	0.66	0.42	0.69
live	0.20	-0.10	-0.37	0.49					0.49	0.54
val	0.45	0.54	0.33	0.29						0.77
dur	-0.20	-0.11	-0.66							0.44
acous	-0.73									
spch				0.87						
pop	-0.13		0.65							

Figura 6 loadings dos componentes rodados e comunalidades (4 componentes)



## 5 Componentes Principais:

Com 5 componentes principais, obtemos uma melhoria nos *loadings*. Neste caso, vemos que só as variáveis *live* e *val* é que mantêm piores coeficientes: Na tabela das comunalidades, as variáveis *live* e *acous* são as que estão pior representadas no conjunto dos componentes.

Loadings:

	RC1	RC4	RC2	RC3	RC5										
bpm	0.17	-0.78	-0.24	0.20		bpm	nrgy	dnce	dB	<u>live</u>	val	dur	<u>acous</u>	spch	pop
nrgy	0.89			0.12		0.74	0.81	0.69	0.67	0.59	0.71	0.60	0.55	0.78	0.97
dnce	0.21	0.71	-0.34	0.13											
dB	0.80			-0.12											
<u>live</u>	0.27		0.55	0.45											
<u>val</u>	0.43	0.42	-0.47	0.34											
dur	-0.14		0.75		-0.11										
acous	-0.74														
spch				0.87											
pop					0.98										

Figura 7 loadings dos componentes rodados e comunalidades (5 componentes)

## 6 Componentes Principais:

Com 6 componentes principais voltamos a obter uma melhoria nos diversos indicadores, mas não deixamos de ter variáveis cuja informação original é ainda “mal representada” nas componentes extraídas. Neste cenário, a variável *input dur* não está claramente associada a um só componente. Olhando para as comunalidades, a variável *acous* é aqui a que tem pior representação no total dos componentes.

Loadings:

	RC1	RC6	RC2	RC4	RC3	RC5											
bpm		-0.12		0.94			bpm	0.90	0.86	0.84	0.78	0.73	0.71	<u>dur</u>	<u>acous</u>	0.94	0.99
nrgy	0.90	0.13			0.12												
dnce		0.89		-0.16	-0.12												
dB	0.87		-0.13														
live	0.12		0.81	0.17	0.14												
val	0.34	0.73	-0.14		0.22												
<u>dur</u>	-0.13	-0.36	0.59	-0.29		-0.13											
<u>acous</u>	-0.65	-0.27	-0.20	-0.17	0.15												
spch			0.11		0.96												
pop						0.99											

Figura 8 loadings dos componentes rodados e comunalidades (6 componentes)

## Decisão sobre número de componentes a usar

Perante estas análises, e para o propósito do projeto, foi decidida a criação de 5 componentes principais, cujos detalhes se encontram no Anexo IV - Análise de Componentes Principais – 5 Componentes. Consideramos que com este número de componentes, os dados originais encontram-se razoavelmente representados. Conseguimos reduzir a dimensionalidade dos mesmos para metade (passando de 10 variáveis input para 5 componentes principais), e ainda assim também uma variância explicada que corresponde a 71% daquela obtida com as variáveis originais. Considerámos

que o aumento de complexidade de 5 para 6 componentes, com um incremento de 8 pontos percentuais na variabilidade original explicada não se justificava.

### Interpretação e Nomeação das Componentes Principais

Olhando novamente para a tabela com os componentes rodados relativa a 5 componentes principais, considerando os valores dos loadings mais elevados em cada um, podemos fazer uma breve análise da informação representada:

- Componente 1: os valores mais elevados obtidos são os respeitantes às variáveis de *input Energy*, *Loudness* (dB), e *Acousticness* (esta com correlação negativa com a componente); significa que temos aqui as músicas tendencialmente mais enérgicas, com o volume mais alto, e menos acústicas; com base nisto, podemos designar esta a componente electrónica e de energia;
- Componente 2: a variável *input* que se destaca neste componente é a da duração, estando aqui então representadas as músicas tendencialmente mais longas; podemos designar esta componente a das músicas longas;
- Componente 3: o valor de destaque pertence à variável de *input Speechiness*, o que significa que há maior probabilidade de encontrarmos aqui representadas as músicas com maior quantidade de “palavra cantada”; podemos designar esta a componente vocal;
- Componente 4: destaca-se a correlação negativa com as *Beats Per Minute* (BPM) e a positiva com *Danceability*; nesta componente temos as músicas com um *andamento* mais lento e que são mais facilmente dançáveis;
- Componente 5: destaca-se aqui a elevada associação com a variável *Popularity*; nesta componente temos claramente as músicas que tendencialmente têm a popularidade mais elevada; esta componente será então designada como a da popularidade.

### Análise de *clustering* das observações

Partimos para uma análise de *clustering* probabilístico utilizando GMM (Gaussian mixture models), tendo em conta que este tipo de modelo pode detetar clusters em amostras que se sobrepõem no mesmo espaço.

#### *Clustering* probabilístico usando GMM

O modelo mais interessante pareceu-nos ser o VEE (volume variável, mas mesma forma e orientação para todos os clusters), com 6 componentes, uma vez que apresenta o BIC (Bayesian information criterion) absoluto mais baixo.

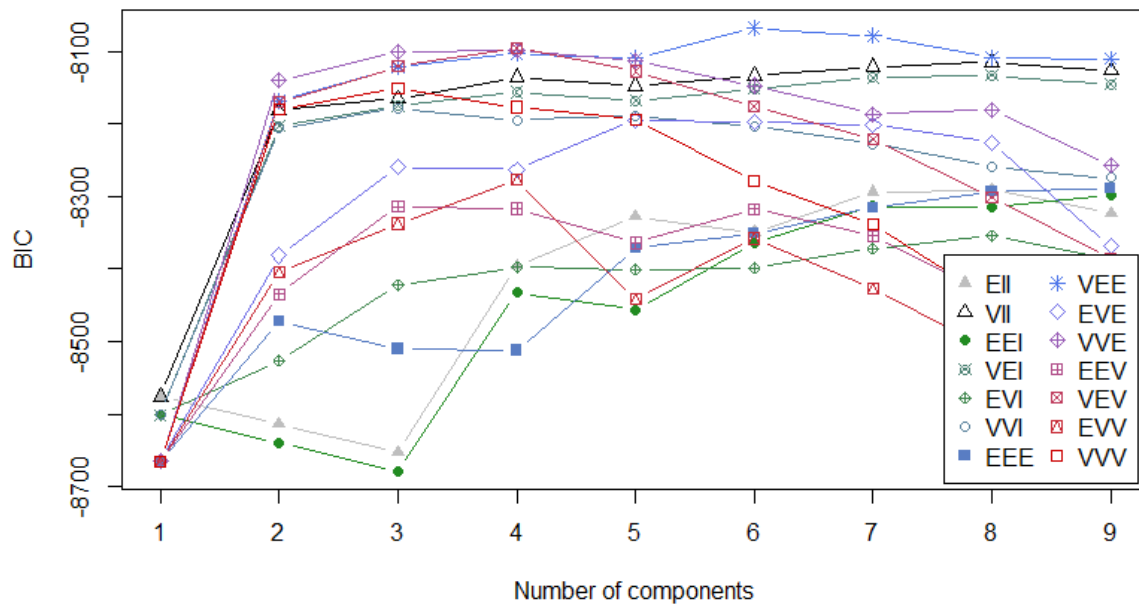


Figura 9 análise de BIC para modelo GMM

O BIC é um bom indicador, já que ele acrescenta uma penalização para modelos com mais clusters, evitando assim o *overfitting*.

Conseguimos perceber, analisando as médias de cada componente principal, em cada *cluster*, quais os focos de cada um desses clusters.

- O *cluster* 1 está principalmente focado em músicas com maior popularidade mais melancólicas e instrumentais.
- O *cluster* 2 em música longa, vocal e dançável, menor popularidade.
- O *cluster* 3 em música mais energética e eletrónica.
- O *cluster* 4 em música mais melancólica e instrumental (menos energética/eletrónica e menos vocal), sem foco na popularidade.
- O *cluster* 5 em música com mais popularidade, mais energética, vocal e dançável.
- O *cluster* 6 em música mais energética/eletrónica, mas menos dançável, não focado na popularidade.

Means:	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
Energetic/electronic	-0.24399377	-0.4216098	-0.6309365	-1.7961949	0.4953767	0.59465874
Live/Long	-0.05345438	1.4823720	-0.6983703	0.09082809	0.5135826	0.08430107
Vocal	-0.38574753	1.5639046	-0.2173371	-0.59957721	0.7177386	0.12979974
Dance	-0.19690334	0.7651407	-0.1095485	0.23174285	0.4599267	-0.86493818
Popularity	0.3890861	-0.5170330	0.0741644	-0.40432537	0.2964807	-1.35742000

Figura 10 Médias dos componentes principais, por cluster (6 clusters GMM)

Analisando os clusters com as variáveis de PROFILE, não conseguimos uma explicação do género musical (*top.genre*). Isto devido ao *dataset* ter uma preponderância de um único género – “dance

pop” – que, ao mesmo tempo, é muito heterogéneo e também por falta de observações para a maioria dos outros géneros (ver Anexo V – *Cluster GMM vs Género musical (top.genre)*).

Fizemos a mesma tentativa de identificar os clusters com artistas (uma outra variável de *Profile*) e verificamos que existem vários artistas que atravessam vários clusters (ver Anexo VI – *Cluster GMM vs Artistas*).

Isto pode ser explicado pelo facto de que os artistas experimentam diferentes estilos e tipos de música, mesmo dentro do seu género.

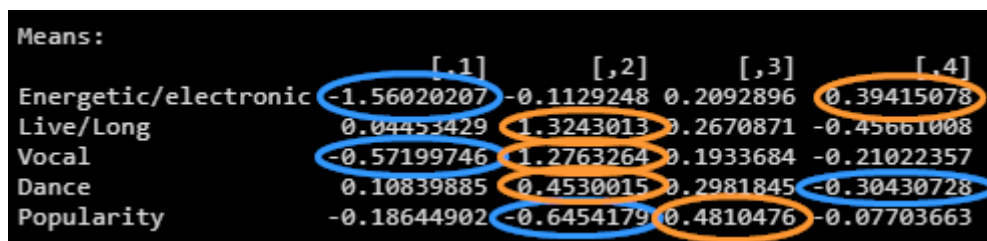
#### *Clustering probabilístico usando GMM e 4 clusters*

Uma outra interpretação do gráfico do BIC poderia levar-nos a seleccionar um *clustering* de 4 componentes (modelo VEV – volume variável, forma igual e direcção variável), por isso decidimos tentar.

Mais uma vez, não conseguimos detetar uma associação dos clusters aos géneros musicais, nem a artistas específicos. Por um lado, pela preponderância dos géneros mais numerosos na amostra – *dance pop*, *pop* e *canadian pop* – e por outro porque os artistas nos tops fazem músicas de diversos estilos que atravessam os nossos clusters.

Olhando para as médias dos componentes principais, aí sim, é possível dar um nome a cada *cluster*:

- *Cluster 1* – músicas mais instrumentais dançáveis
- *Cluster 2* – músicas vocais mais longas e dançáveis
- *Cluster 3* – músicas com mais popularidade
- *Cluster 4* – músicas energéticas menos dançáveis



Means :	[.1]	[.2]	[.3]	[.4]
Energetic/electronic	-1.56020207	-0.1129248	0.2092896	0.39415078
Live/Long	0.04453429	1.3243013	0.2670871	-0.45661008
Vocal	-0.57199746	1.2763264	0.1933684	-0.21022357
Dance	0.10839885	0.4530015	0.2981845	-0.30430728
Popularity	-0.18644902	-0.6454179	0.4810476	-0.07703663

Figura 11 Médias dos componentes principais, por cluster (4 clusters GMM)

É interessante reparar que, usando este critério, as músicas mais populares não parecem estar tão extremadas nas outras componentes.

#### *Clustering usando um método hierárquico*

Usamos para critério de aglomeração a distância de *Ward* – minimização da variância entre os pontos do mesmo *cluster* – seguido de um corte a um nível que crie alguma diferença de informação substancial. Usámos a distância de *Ward* por ser menos suscetível a ruído e *outliers*, embora este método tenha a desvantagem de obter sempre *clusters* globulares, por natureza.

O corte foi selecionado baseado na experiência com o método anterior e uma verificação da adequação com o método *Silhouette*, que mede a distância média dos pontos de cada *cluster* aos outros pontos do *cluster*, obtendo uma distância entre -1 e 1. Quanto mais perto de 1 mais compactos ou homogêneos são os *clusters*

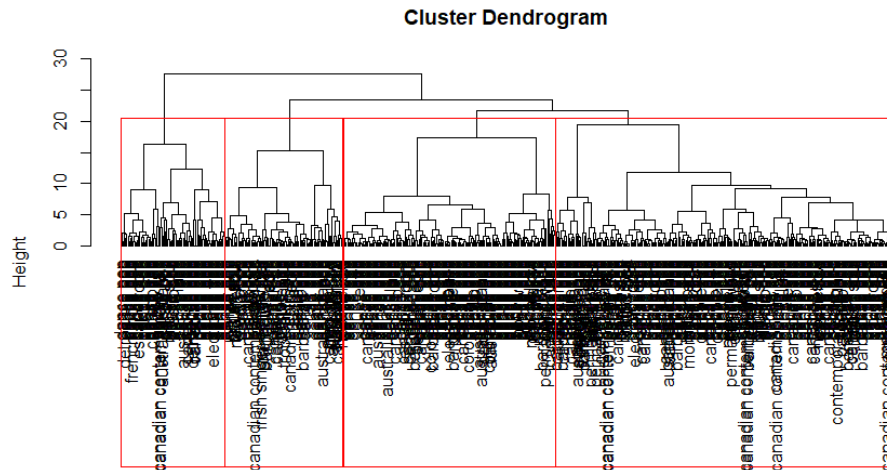


Figura 12 Dendrograma de aglomeração hierárquica com distância de Ward baseado nos PCs, com um corte para 4 clusters

Decidimos começar por experimentar 4 *clusters*, com o que obtivemos a seguinte análise, baseada nas médias dos nossos componentes principais:

- Cluster 1 – músicas com maior popularidade, menos baseadas na voz e menos dançáveis
- Cluster 2 – músicas mais longas, mais focadas na voz e dançáveis
- Cluster 3 – músicas mais eletrônicas e energéticas e com menor popularidade
- Cluster 4 – músicas menos eletrônicas e menos energéticas, mais curtas

	Energetic/electronic	Live/Long	Vocal	Dance	Popularity	cluster
1	-0.02	0.26	-0.47	-0.20	0.42	1
2	0.24	0.87	1.46	1.05	0.15	2
3	0.65	-0.31	-0.01	-0.18	-0.72	3
4	-1.34	-0.94	0.09	-0.01	-0.04	4

Figura 13 4 Clusters hierárquicos vs médias dos componentes principais

Usando o método *Silhouette* obtemos uma média de comprimento de silhueta de 0.13, o que indica não existir estrutura substantiva no nosso modelo.

Mais uma vez, a tentativa de associar artistas ou géneros musicais aos nossos *clusters* revela-se infrutífera (como é visível pelo dendrograma, no caso dos géneros musicais).

Experimentámos o método com 2, 3, 4, 5 e 10 clusters, sendo que a melhor largura de silhueta foi obtida com 2 *clusters* – 0.28 – ainda na zona da estrutura fraca.

	Energetic/electronic	Live/Long	Vocal	Dance	Popularity
1	-0.03804505	-0.1356666	-0.2269417	-0.1626488	-0.02356098
2	0.24470953	0.8726208	1.4597117	1.0461734	0.15154654

Figura 14 2 Clusters hierárquicos vs médias dos componentes principais

Esta estrutura de dois clusters parece sugerir um *cluster* com músicas mais dançáveis e suportadas na voz do artista e outro que seria mais próximo da média em todos os componentes. Pelo que este método não nos parece fornecer grandes dados ou oferecer um bom modelo de *clustering*.

## Clustering usando métodos Partitivos

### Cluster com K-médias (*K-means*)

O método *k-means* parte de um determinado número de clusters e tenta atribuir todos os pontos do nosso universo a esses *clusters* minimizando a distância dos pontos ao centro do *cluster*.

Dois métodos de avaliar a performance deste algoritmo são a análise de *Silhouette* (comprimento médio da silhueta) e o WSS (*Within Sum of Squares*), que soma a distância de cada ponto ao centroide do seu *cluster* – uma medida de dispersão.

Começamos por testar estas medidas em até 20 *clusters* e selecionamos aqueles que nos dão melhores garantias.

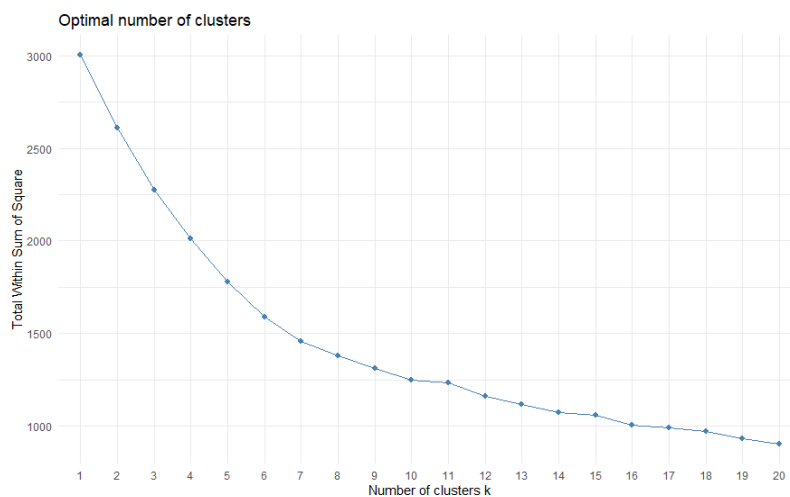


Figura 15 Scree Plot para K-Means, baseado na distância interna (WSS)

A partir do *Scree Plot* decidimos experimentar 6 *clusters*, o que nos dá uma distância média de silhueta de 0.21 – nenhuma estrutura aparente.

Seguindo o gráfico de melhores distâncias médias de silhueta, decidimos experimentar 3 clusters.

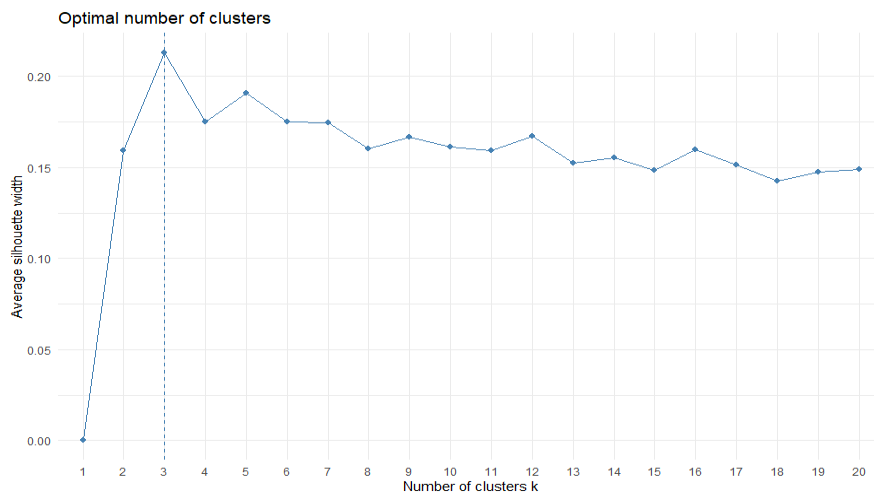


Figura 16 - distância média de silhueta, dependendo do número de cluster, usando k-means

Analisando estes 3 *clusters* a partir dos componentes principais, parecem existir um *cluster* mais focado em músicas suportadas em voz, outro mais suportado em música menos eletrônica e mais instrumental e outro mais focado em música eletrônica, nenhum dos quais se diferenciando muito pela popularidade, ao contrário do que vimos nos métodos anteriores.

```
> kmeans.k3$centers
```

	Energetic/electronic	Live/Long	Vocal	Dance	Popularity
1	0.1794580	0.7968403	1.43489066	0.85039040	0.1897871
2	-0.8776196	0.4295548	-0.71800999	0.05767557	0.1856753
3	0.4119500	-0.4582138	-0.03654911	-0.27713203	-0.1532107

Figura 17 Centro dos 3 clusters de K-médias em relação aos componentes principais

Verificamos que a diferença entre clusters não é muito acentuada e que são bastante dispersos, o que dificulta a sua utilização. Isso é confirmado pelo seu valor médio de silhueta ser de 0.19, estabelecendo a inexistência de uma estrutura.

#### Cluster com PAM (*Partition Around Medoids*)

O método PAM é uma evolução do método *K-médias*, usando em vez da distância ao centroide a distância ao “Medóide” – a distância média entre todos os pontos. Evidentemente este cálculo é bastante pesado, sendo por isso muitas vezes otimizado calculando a distância apenas para um número considerado suficiente de “pontos”. O pacote que usamos usa a distância euclidiana e também a distância de *Manhattan*.

Com base no comprimento médio da silhueta, selecionamos 6 *clusters* para o nosso modelo, o que também parece fazer sentido quando verificamos o *scree plot* baseado na WSS.

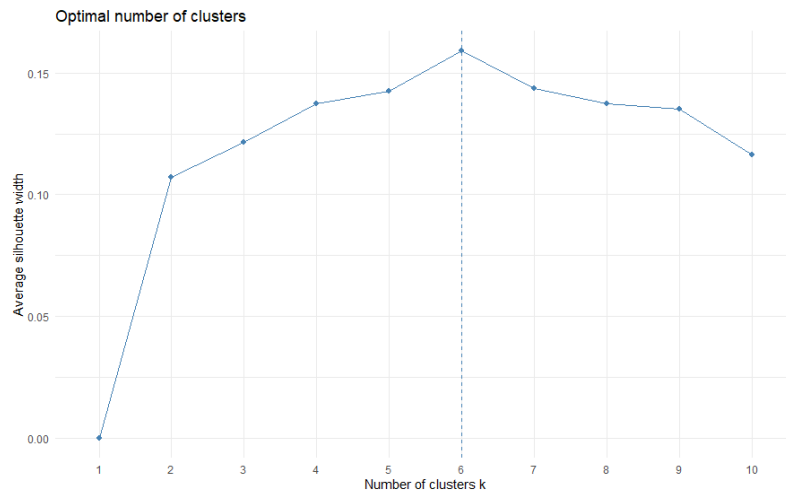


Figura 18 comprimento médio da silhueta para diferente número de clusters PAM

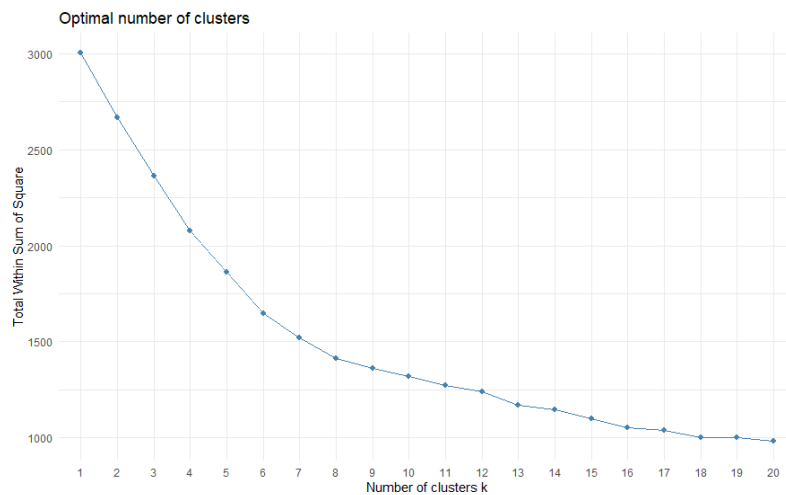


Figura 19 Scree Plot com WSS para diferente número de clusters no modelo PAM

A partir dos centros dos medóides, podemos interpretar os *clusters* obtidos:

- Cluster 1 – músicas mais curtas
- Cluster 2 – músicas com maior popularidade
- Cluster 3 – músicas mais energéticas e eletrónicas, mais instrumentais que vocais
- Cluster 4 – músicas mais acústicas e melancólicas
- Cluster 5 – músicas mais vocais e dançáveis
- Cluster 6 – músicas menos populares e dançáveis

	Energetic/electronic	Live/Long	Vocal	Dance	Popularity
282	0.40987241	-1.14987105	0.02907438	0.22844662	0.22138947
388	-0.04370215	0.02741656	0.37474666	-0.49717906	1.25635313
121	0.42073194	0.06394269	-0.95069600	-0.04013054	0.03496470
474	-2.20609416	0.30300022	-0.41173908	0.34466798	0.07531635
601	0.20577132	1.30258405	1.29363157	1.87742790	-0.22955078
203	0.12251255	0.13526195	0.26613696	-0.97048359	-1.24706936

Figura 20 centro dos medóides, para 6 clusters PAM



Tendo em conta que o comprimento médio da silhueta é de 0.16 (oscilando entre 0.28 e 0.03, para os componentes), podemos concluir que também este modelo apresenta pouca estrutura e não parece ser apropriado para utilização real.

```
> pam.k6$silinfo$clus.avg.widths  
[1] 0.27537846 0.10504710 0.18661416 0.09203235 0.03036814 0.10292181  
> pam.k6$silinfo$avg.width  
[1] 0.1591346
```

Figura 21 comprimento médio da Silhouette por cada cluster e em média geral, para os 6 clusters PAM

## Conclusão

O PCA, apesar de ser um dos mais conhecidos métodos de redução de dimensionalidade, tem alguns pressupostos nos quais se apoia para a sua eficácia. Um dos principais é o nível de correlação entre as variáveis de input que possamos considerar. No nosso caso, é evidenciado através da matriz de correlações e também através do índice KMO a existência de correlações fracas em muitas das variáveis e também um valor fraco para o índice. Ainda assim, obtemos 5 componentes principais, que correspondem a metade do número de variáveis de input originais, e com uma variância explicada total acima de 70% dos dados originais.

O género *pop* (*dance pop*, *pop* e *canadian pop*) é preponderante e isso acaba por dificultar a definição de *clusters*. Mesmo dentro de um género (ex: *dance pop*) existem grandes diferenças de características. Precisaríamos de uma amostra mais expressiva para os restantes géneros, para conseguir estudar melhor a relação de *clusters* com os géneros.

Tendo em conta os dados obtidos, os géneros musicais da moda parecem ditar a sua entrada no top e os artistas parecem ter uma preponderância sobre as características da música, cruzando estilos dentro do mesmo género.

Poderemos colocar a hipótese de não serem estas as variáveis mais interessantes para analisar as características da música de um género. Por outro lado, os géneros que são aqui utilizados podem estar mais ligados a características que o *Spotify* usa para segmentar os seus clientes e não para segmentar as músicas em si. Isso é visível com a separação geográfica (exemplo: “*canadian pop*”, “*canadian latin*”), quando a música, muitas das vezes, não tem fronteiras.

## Referências

- Henrique, L. (2019). *Top Spotify songs from 2010-2019 - BY YEAR*. Retrieved from Kaggle:  
<https://www.kaggle.com/leonardopena/top-spotify-songs-from-20102019-by-year>
- Lamere, P. (2016). *Organize Your Music*. Retrieved from  
<http://organizeyourmusic.playlistmachinery.com/>

## Anexo I – KMO

Kaiser-Meyer-Olkin factor adequacy

Call: KMO(r = corrs2)

Overall MSA = 0.62

MSA for each item =

bpm	nrge	dnce	dB	live	val	dur	acous	spch	pop
0.47	0.60	0.54	0.65	0.65	0.66	0.76	0.65	0.36	0.50

## Anexo II - Análise de Componentes Principais – 10 Componentes

### Principal Components Analysis

Call: principal(r = df\_stdZ, nfactors = length(colnames(df\_stdZ)),  
rotate = "none", scores = TRUE)

Standardized loadings (pattern matrix) based upon correlation matrix

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	h2	u2	com
bpm	0.09	0.44	-0.43	0.56	-0.22	0.40	0.23	0.20	0.10	0.02	1	3.7e-15	5.0
nrgy	0.83	0.31	-0.10	-0.11	0.04	-0.21	0.00	-0.02	-0.05	0.37	1	1.3e-15	1.9
dnce	0.48	-0.55	0.38	-0.07	0.02	0.39	0.17	0.02	0.35	0.08	1	8.9e-16	4.8
dB	0.73	0.12	-0.28	-0.16	0.09	-0.33	-0.11	0.31	0.26	-0.22	1	1.4e-15	3.1
live	0.15	0.54	0.33	0.03	0.41	0.37	-0.52	0.03	0.00	-0.03	1	-2.9e-15	4.6
val	0.70	-0.29	0.30	0.16	-0.15	0.08	0.04	0.35	-0.40	-0.08	1	-4.4e-16	3.5
dur	-0.35	0.42	0.22	-0.38	0.33	0.03	0.56	0.30	-0.04	0.00	1	1.3e-15	5.4
acous	-0.69	-0.18	0.13	0.13	-0.11	-0.18	-0.31	0.53	0.11	0.18	1	2.2e-16	3.2
spch	0.07	0.33	0.62	0.52	-0.08	-0.40	0.16	-0.14	0.12	-0.05	1	-3.3e-15	3.8
pop	-0.01	-0.42	-0.24	0.45	0.73	-0.12	0.10	0.01	-0.04	0.03	1	1.4e-15	2.8

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
SS loadings	2.59	1.47	1.13	1.02	0.91	0.81	0.80	0.65	0.39	0.24
Proportion Var	0.26	0.15	0.11	0.10	0.09	0.08	0.08	0.06	0.04	0.02
Cumulative Var	0.26	0.41	0.52	0.62	0.71	0.79	0.87	0.94	0.98	1.00
Proportion Explained	0.26	0.15	0.11	0.10	0.09	0.08	0.08	0.06	0.04	0.02
Cumulative Proportion	0.26	0.41	0.52	0.62	0.71	0.79	0.87	0.94	0.98	1.00

Mean item complexity = 3.8

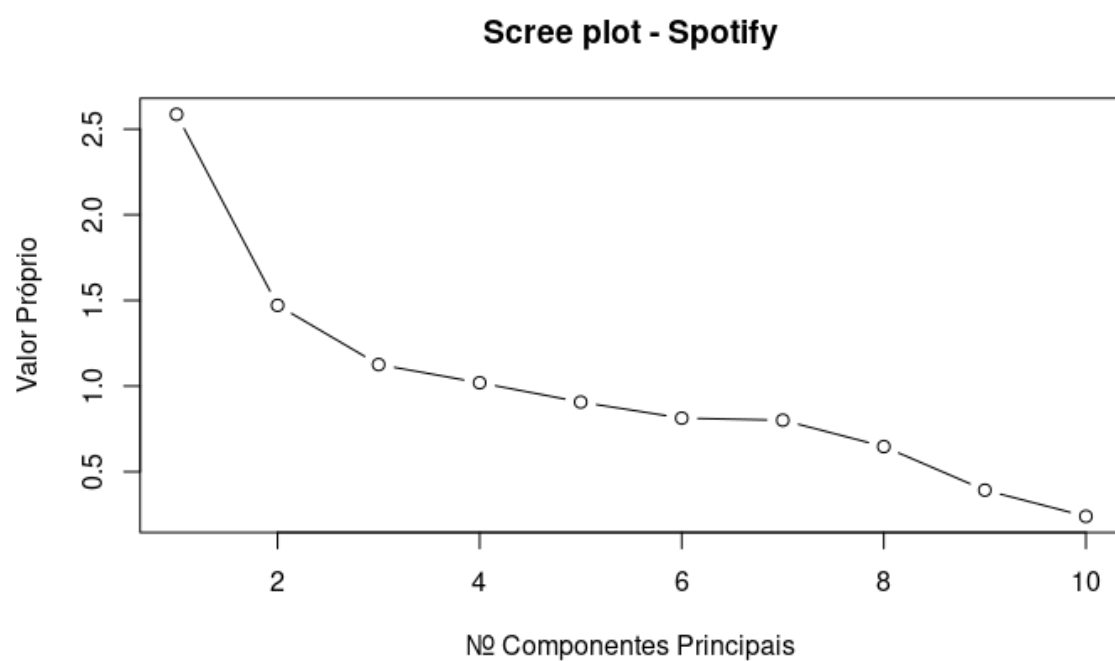
Test of the hypothesis that 10 components are sufficient.

The root mean square of the residuals (RMSR) is 0  
with the empirical chi square 0 with prob < NA

Fit based upon off diagonal values = 1

> |

### Anexo III - *Scree-Plot*: Número de Componentes vs. Valores Próprios



## Anexo IV - Análise de Componentes Principais – 5 Componentes

Principal Components Analysis

Call: principal(r = df\_stdZ, nfactors = 5, rotate = "none", scores = TRUE)

Standardized loadings (pattern matrix) based upon correlation matrix

	PC1	PC2	PC3	PC4	PC5	h2	u2	com
bpm	0.09	0.44	-0.43	0.56	-0.22	0.74	0.261	3.2
nrgy	0.83	0.31	-0.10	-0.11	0.04	0.81	0.186	1.3
dnce	0.48	-0.55	0.38	-0.07	0.02	0.69	0.310	2.8
dB	0.73	0.12	-0.28	-0.16	0.09	0.67	0.335	1.5
live	0.15	0.54	0.33	0.03	0.41	0.59	0.409	2.8
val	0.70	-0.29	0.30	0.16	-0.15	0.71	0.292	2.0
dur	-0.35	0.42	0.22	-0.38	0.33	0.60	0.403	4.5
acous	-0.69	-0.18	0.13	0.13	-0.11	0.55	0.447	1.3
spch	0.07	0.33	0.62	0.52	-0.08	0.78	0.223	2.6
pop	-0.01	-0.42	-0.24	0.45	0.73	0.97	0.026	2.6

	PC1	PC2	PC3	PC4	PC5
SS loadings	2.59	1.47	1.13	1.02	0.91
Proportion Var	0.26	0.15	0.11	0.10	0.09
Cumulative Var	0.26	0.41	0.52	0.62	0.71
Proportion Explained	0.36	0.21	0.16	0.14	0.13
Cumulative Proportion	0.36	0.57	0.73	0.87	1.00

Mean item complexity = 2.5

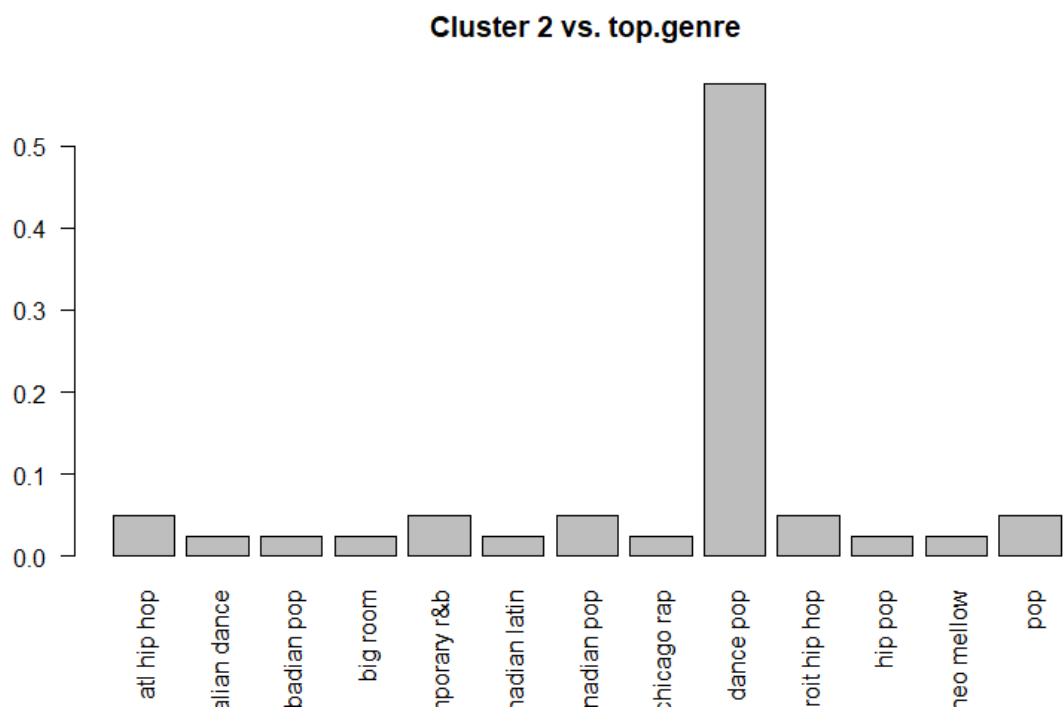
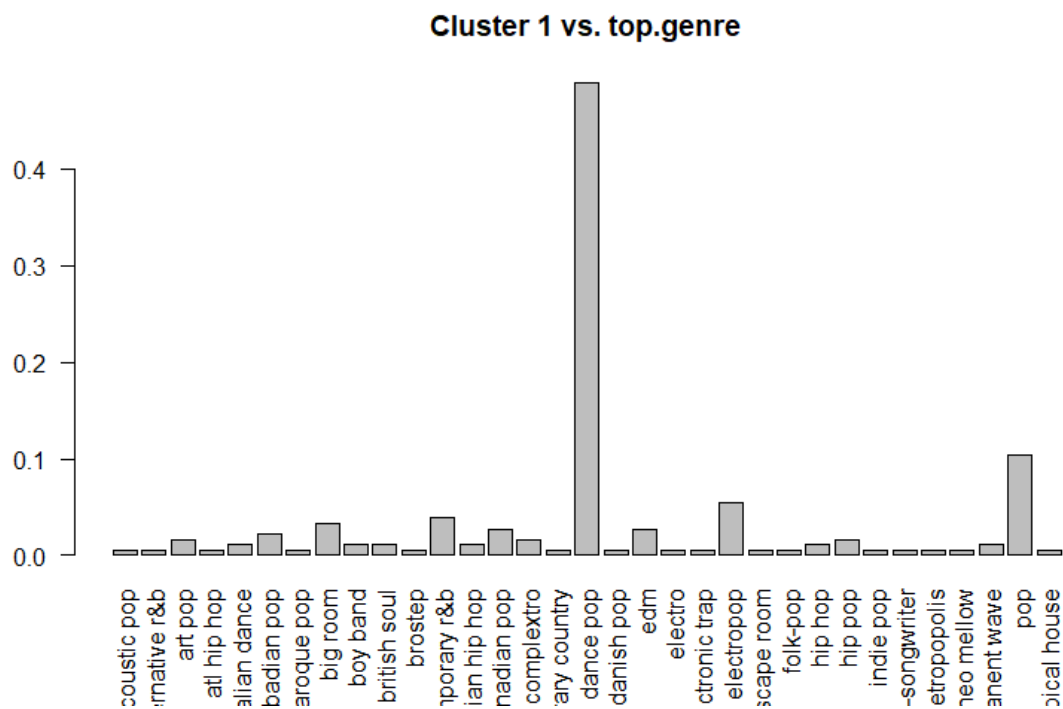
Test of the hypothesis that 5 components are sufficient.

The root mean square of the residuals (RMSR) is 0.1

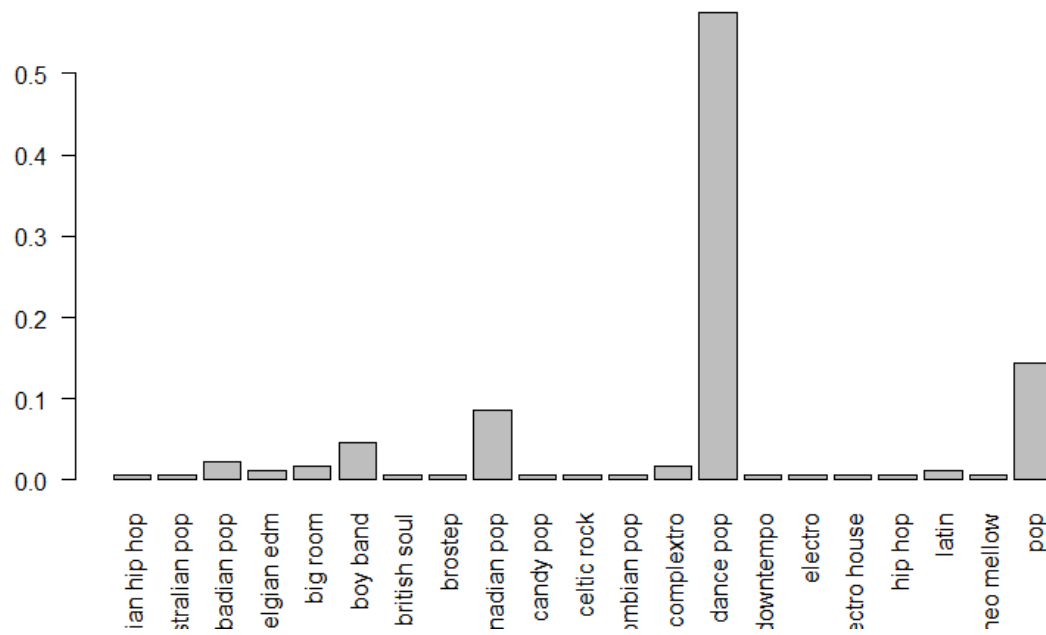
with the empirical chi square 574.62 with prob < 6.1e-122

Fit based upon off diagonal values = 0.76

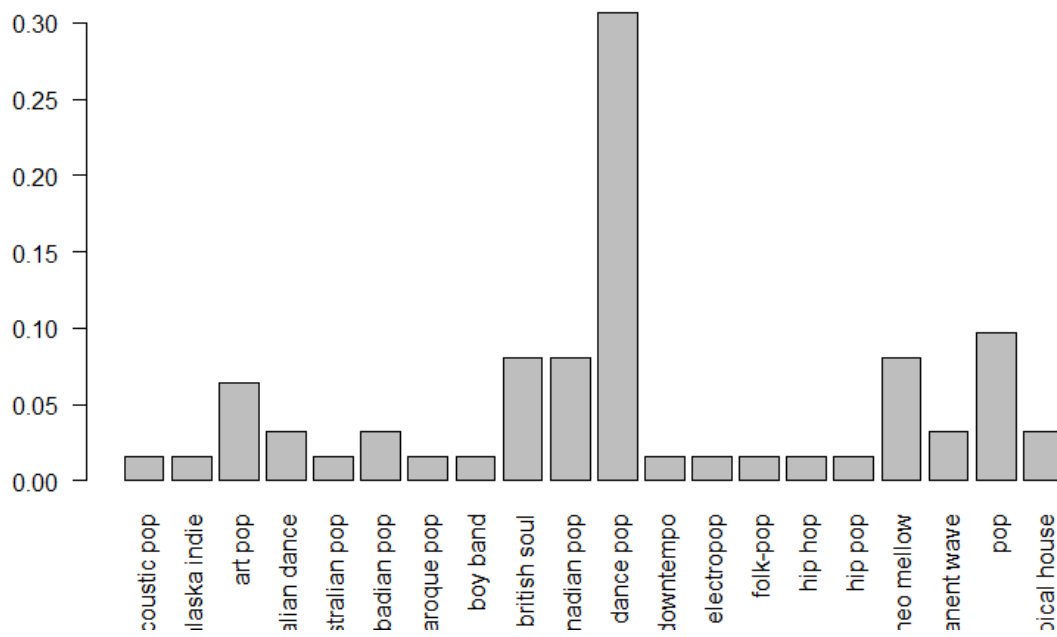
## Anexo V – Cluster GMM vs Género musical (top.genre)



Cluster 3 vs. top.genre

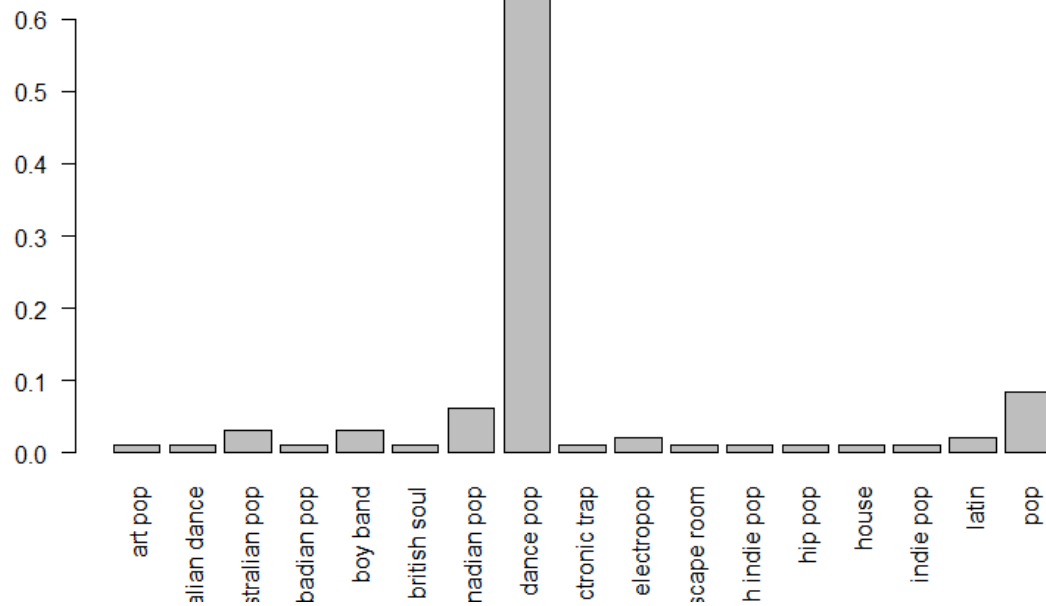


Cluster 4 vs. top.genre

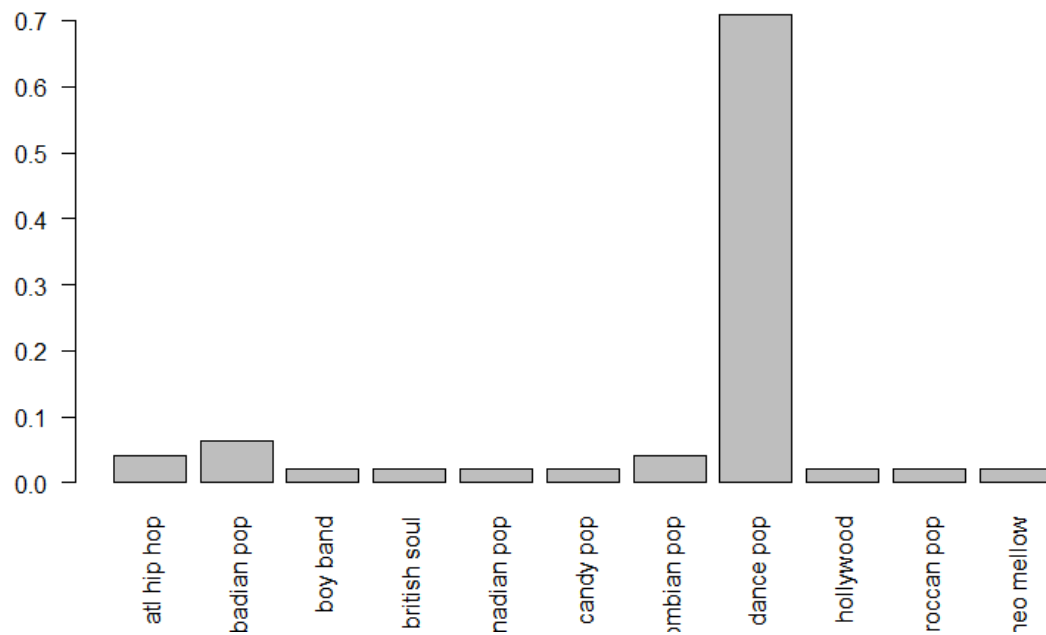




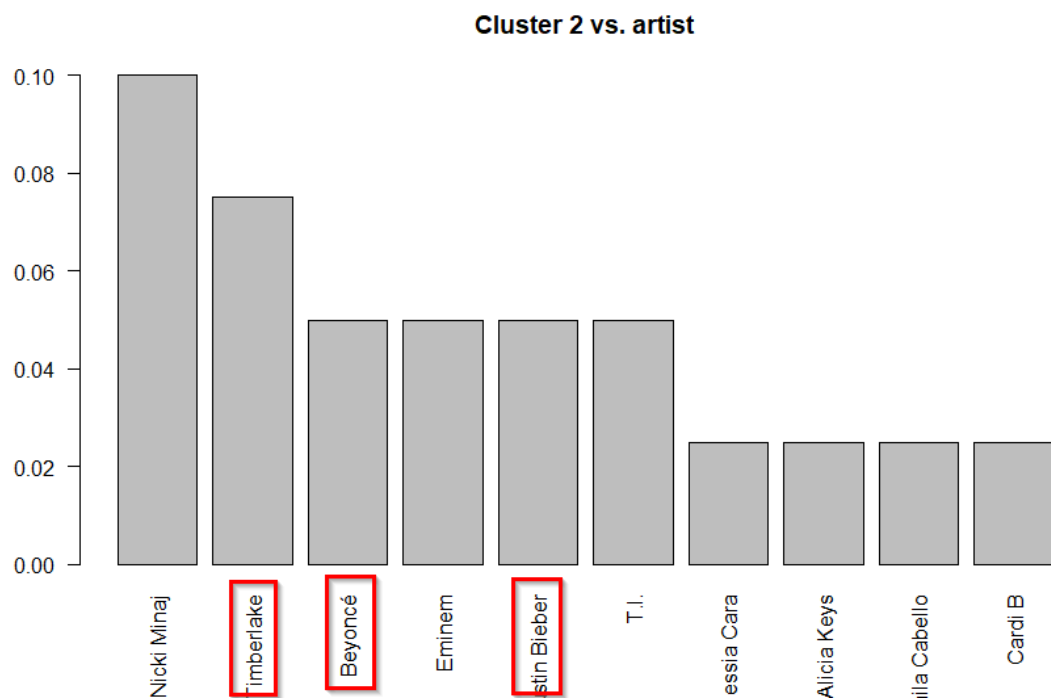
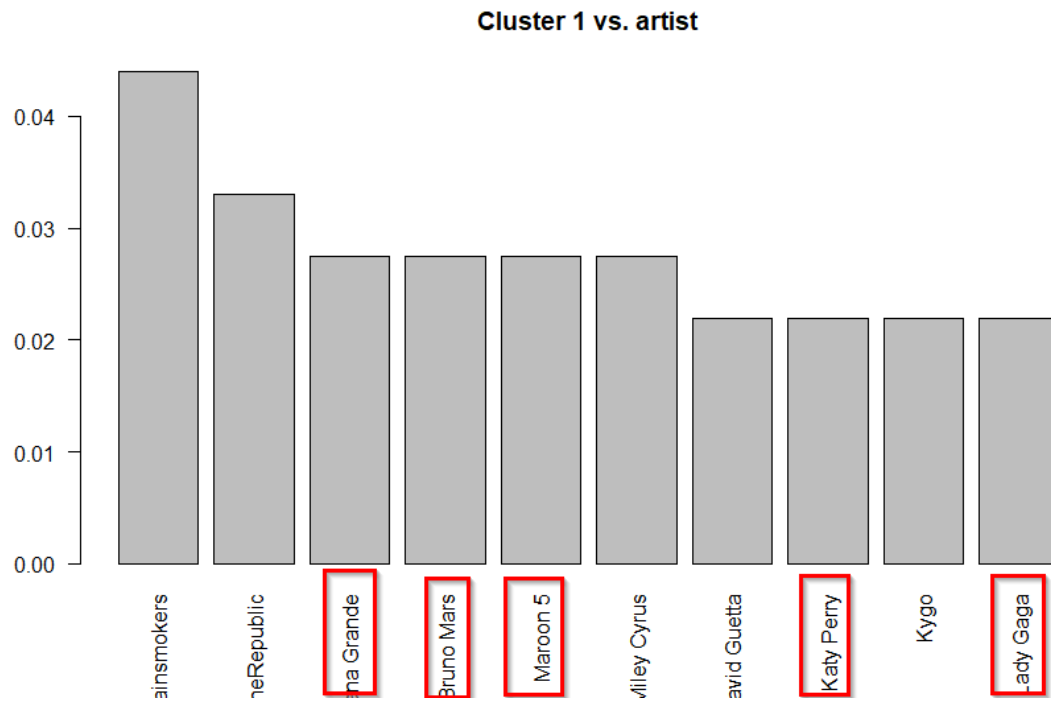
Cluster 5 vs. top.genre



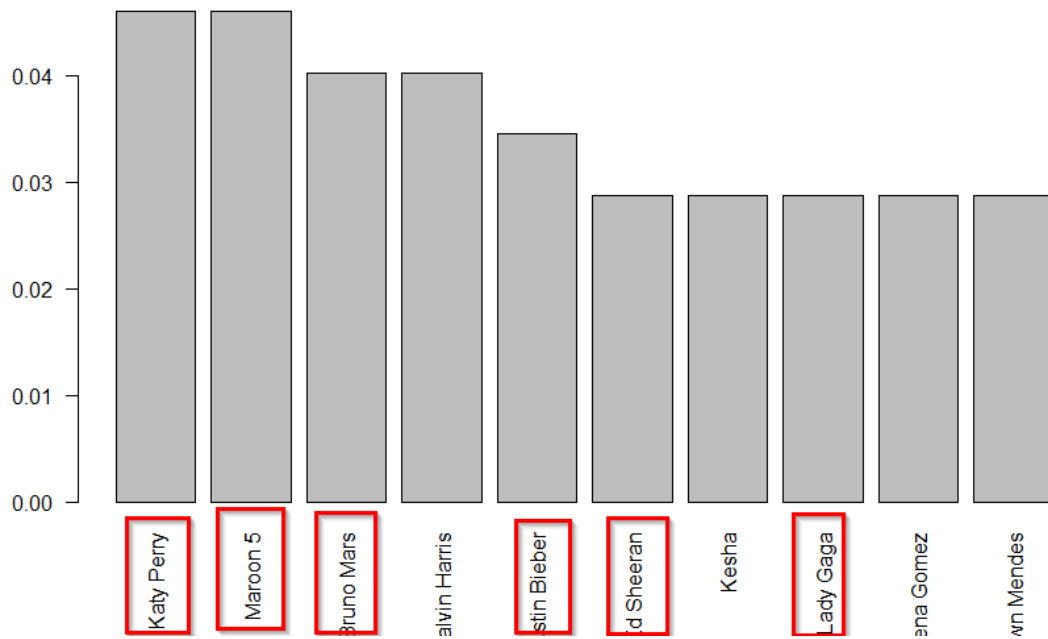
Cluster 6 vs. top.genre



## Anexo VI – Cluster GMM vs Artistas



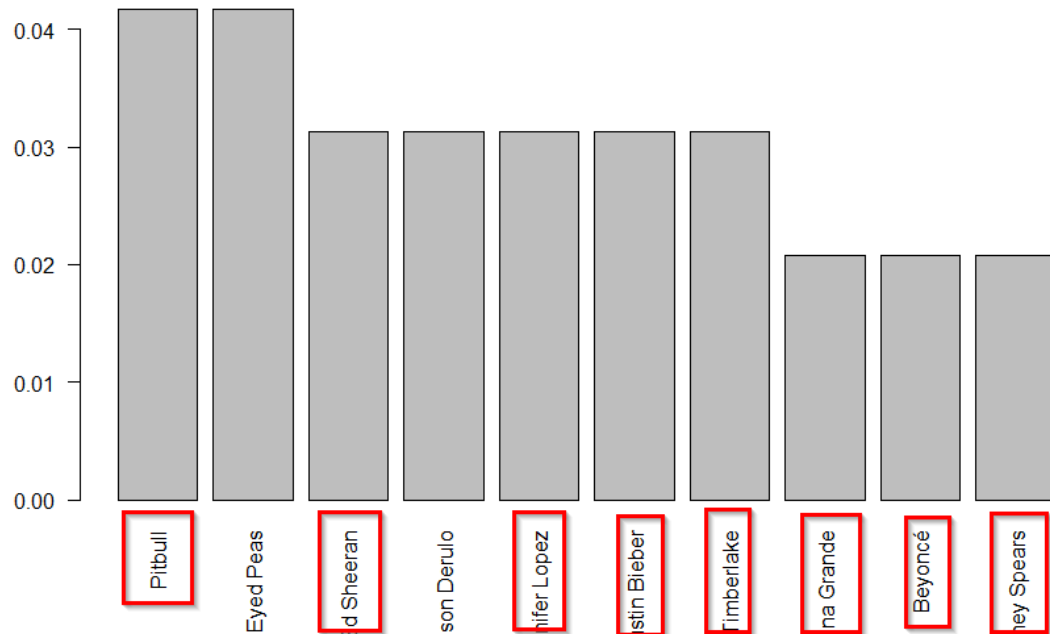
Cluster 3 vs. artist



Cluster 4 vs. artist



Cluster 5 vs. artist



Cluster 6 vs. artist

