

Music Information Retrieval: Similitud de géneros musicales

Clustering

Daniel Caicedo, Ignacio Chiapella, Miguel Guerrero, Juan Knebel

Daniel Caicedo

UAH

djcc710@gmail.com

Ignacio Chiapella

FCEyN

ignacio.chiapella@gmail.com

Miguel Guerrero

CUFM

miguelgh72@gmail.com

Juan Knebel

FCEyN

juanknebel@gmail.com

Abstract

Escribir resumen del artículo

Keywords: Cluster, DataMining, Música, KMeans, KMedoids, DBSCAN, Hierarchical, Silhouette, PCA, Distancias, MIR

Introducción

La recuperación de información musical, Music Information Retrieval en inglés o simplemente **MIR** de ahora en adelante, es la ciencia interdisciplinaria encargada de recuperar información de la música. Decimos que es interdisciplinaria ya que principalmente los especiales en musicología, en procesamiento de señales o en aprendizaje automático son los que más implicados en el tema están.

Actualmente **MIR** es un área de estudio nuevo pero en crecimiento y muchos de sus estudios y resultados están siendo utilizados en muchas aplicaciones comerciales desde sistemas de recomendación, búsqueda de contenido, interfaces de usuarios para navegar por grandes colecciones de música, detección de instrumentos, categorización automática y hasta generación de música.

Si bien el estudio de la música y la importancia que tuvo y sigue teniendo a lo largo de la historia no tiene discusión, en los últimos años el estudio de la técnica cobró mayor notoriedad. Las causas de lo anteriormente mencionado son, (i) el avance tecnológico que posibilitó a todos tener acceso a prácticamente cualquier pista de audio gracias a aplicaciones como *Napster*, *Grooveshark* y recientemente *Spotify*, (ii) el incremento del poder de computo para aplicar las técnicas de estudio de la música, solo por nombrar dos de ellas.

En este trabajo se realizará un estudio sobre los atributos de más de 2000 canciones, los cuales fueron obtenidas de la interfaz que ofrece Spotify. En la primer sección se comentarán los atributos.

El trabajo consistirá en aplicar diferentes técnicas de **Clustering** sobre los conjuntos de datos previamente mencionados. El objetivo será agrupar diferentes canciones según su similitud en términos del género musical al que pertenecen.

Las técnicas que se utilizarán serán el clásico método *KMeans*, seguido por el también clásico pero más robusto al ruido **PAM** o **KMedoids**, también utilizaremos una implementación jerárquica o **Hierarchical**. Por último mencionaremos una experiencia fallida con el método **DBSCAN**, anticipando que en principio y para este conjunto de datos las pistas de audio no tienen un agrupamiento por densidad.

Trabajo previo

Antes de comenzar con el trabajo actual, es necesario describir el conjunto de datos con el cual se realizaron las experimentaciones. El primer archivo llamado "audio_features" contiene los atributos llamados de alto nivel de los cuales solo tuvimos en cuenta los siguientes: acousticness, danceability, energy, instrumentalness, liveness, loudness, speechiness, tempo y valence. El resto fueron dejados lado ya que no aportaban ningún valor para el agrupamiento de las canciones. Por ejemplo: analysis_url, track_href, type y uri fueron inmediatamente descartadas ya que es información vinculada a la extracción de datos. El resto: duration_ms, key, mode y time_signature no ayudan en nada a discriminar grupos de canciones, como puede verse en el siguiente gráfico.

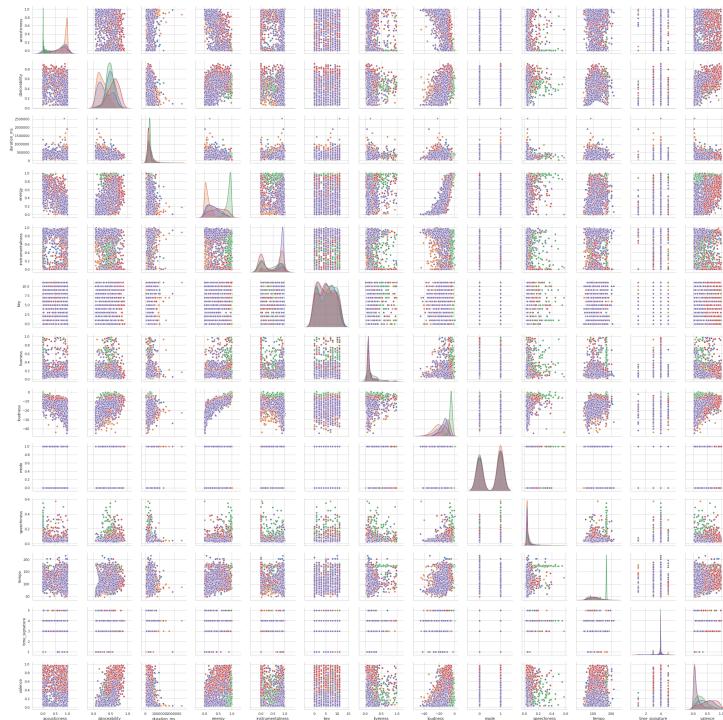


Figure 1. Scatter plot de los atributos de alto nivel.

El segundo conjunto de archivos llamados "audio_analysis" contienen 12 atributos de bajo nivel, uno para cada una de las notas musicales. Se cuenta con un archivo por pista tanto para el timbre como para el pitch separados en ventanas de tiempo. Para no trabajar con series de tiempo y porque las pistas tiene duraciones distintas, se resumieron estos atributos en 2 medidas distintas tanto para los pitches como para los timbres. Las medidas elegidas fueron: para cada atributo de cada canción se calculo la media total y el desvío estándar. Entonces por cada uno de los temas musicales obtuvimos 2 valores resúmenes para cada uno de los 12 atributos. Para el trabajo posterior no se descartó ningún atributo ya que todos son importantes para el agrupamiento.

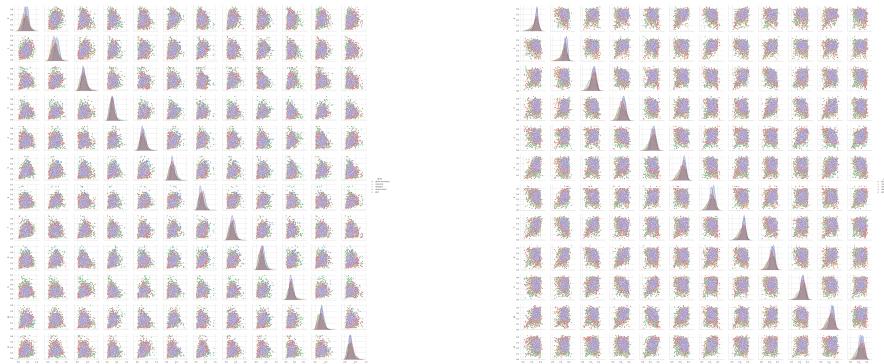


Figure 2. Scatter plot de los atributos sobre el pitch.

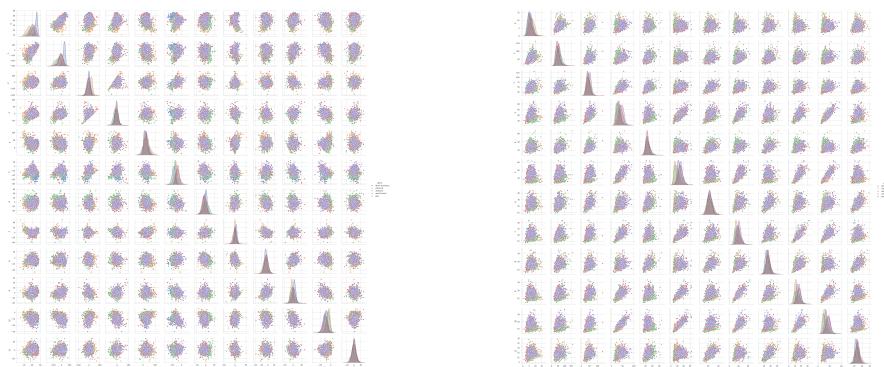


Figure 3. Scatter plot de los atributos sobre el timbre.

Todas las canciones de este conjunto de datos pertenecen a solo 5 géneros musicales: ambient, classical, drum and bass, jazz y world music.

Experiencia KMeans

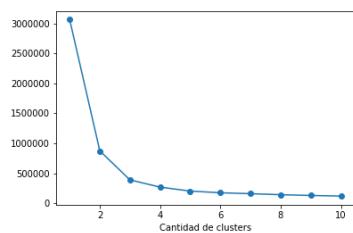
Se utilizó el algoritmo **KMeans** de la librería **Scikit Learn (scikit-learn)** con la utilización de sus parámetros por defecto (**sklearn\api**). Al tratarse de un método sin supervisión necesitamos indicarle la cantidad de clusters en las cuales se quiere que estén agrupadas las observaciones.

Para medir que tan "bien" resulta la agrupación vamos a utilizar diferentes medidas. La primera para identificar cual será la cantidad de clusters será utilizar la suma de los errores cuadrados medios, o **SSE** por sus siglas en inglés. También nos apoyaremos en la medida de **silhouette** para observar que cantidad de clusters obtuvo el mejor valor y también para validar la consistencia de los clusters obtenidos. Como métrica adicional para validar la consistencia de los grupos generados **Rand index**.

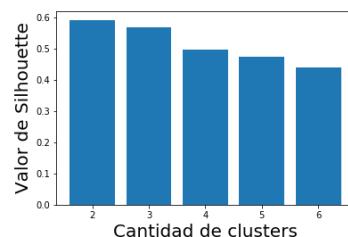
Comenzaremos con generar entre 2 a 6 clusters para los atributos de alto nivel, luego elegida la cantidad de clusters vamos a proceder a agrupar las mismas canciones pero utilizando los datos de bajo nivel, recordar que usaremos las 4 medidas que resumen la información y obtuvimos en el trabajo previo. Con estos nuevos resultados se elegirá cual fue la medida que obtuvo mejores métricas y procederemos a concatenar los valores de alto nivel con esta última. Con este nuevo conjunto de generado vamos a recrear el experimento de generar de 2 a 6 clusters y observar si los resultados se repiten o no.

Resultados

Vamos a comenzar analizando cual sería la mejor cantidad de clusters para los atributos de alto nivel.



(a) Método elbow



(b) Silhouette score

Figure 4. Cantidad de clusters óptima

Si solo tuviésemos en cuenta el método elbow podríamos elegir 4 clusters como el óptimo para realizar el agrupamiento pero, al observar los valores que se obtienen con el método del silhouette, se decidió que el número de clusters sea 3.

Cantidad de clusters	Indice de Rand
2	0.098
3	0.122
4	0.141
5	0.146
6	0.152

Table 1. Rand index

Para reforzar la elección de 3 como cantidad de clusters se puede apreciar que quitando un agrupamiento de 2 clusters, el resto de valores fue cambiando pero no tan abruptamente.

A continuación se muestra la tabla que se obtiene de evaluar los el método con 3 clusters en que grupo clasificó a cada pista musical.

Género	Clusters		
	0	1	2
ambient	224	58	178
classical	253	32	120
drum-and-bass	39	359	53
jazz	159	56	212
world-music	182	62	219

Table 2. Géneros musicales y grupos**Discusión**

Experiencia Jerarquico

Introduccion

Se utilizó el algoritmo **AgglomerativeClustering** de la librería sklearn extra.cluster y con la métrica de distancia euclidean.

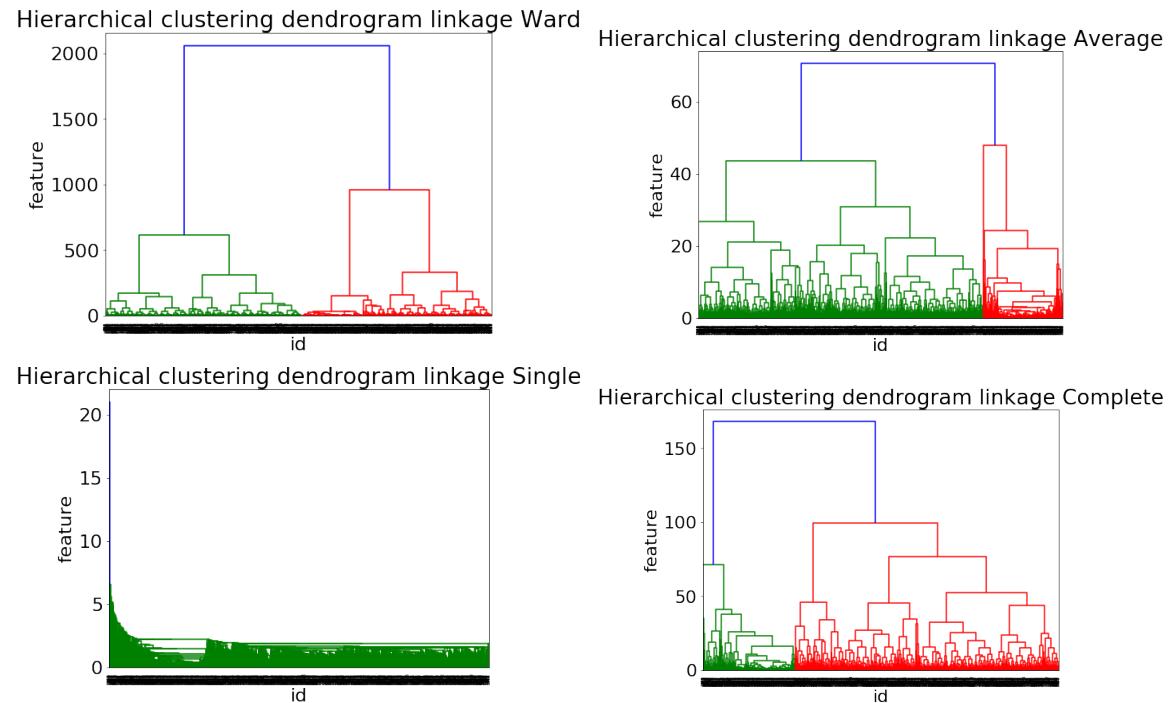
Audio Features

Dendogramas

En un primer análisis, tuvimos que decidir qué tipo de distancia utilizaremos para formar los clusters, la función AgglomerativeClustering nos daba cuatro opciones de linkage criterion:

- **ward** Minimiza la varianza de los clusters que son mergeados.
- **average** Computa la distancia media de cada observación de dos conjuntos.
- **single** Usa la máxima distancia entre todas las observaciones de dos conjuntos.
- **complete** Usa la máxima distancia entre todas las observaciones de dos conjuntos.

Se generan los 4 dendogramas para estudiar qué criterio agrupa mejor los datos.



Al revisar la salida de los distintos linkages, lo que podemos observar es que el método **ward** es quien está manteniendo un mayor balance entre los distintos clusters. Revisando en detalle dicho dendograma, podemos concluir en principio que una separación natural podría ser en 2, 3 o 4 clusters ya que determinamos que con esta cantidad se podría generar un balanceo entre el tamaño de los mismos.

Hierarchical clustering dendrogram linkage Ward

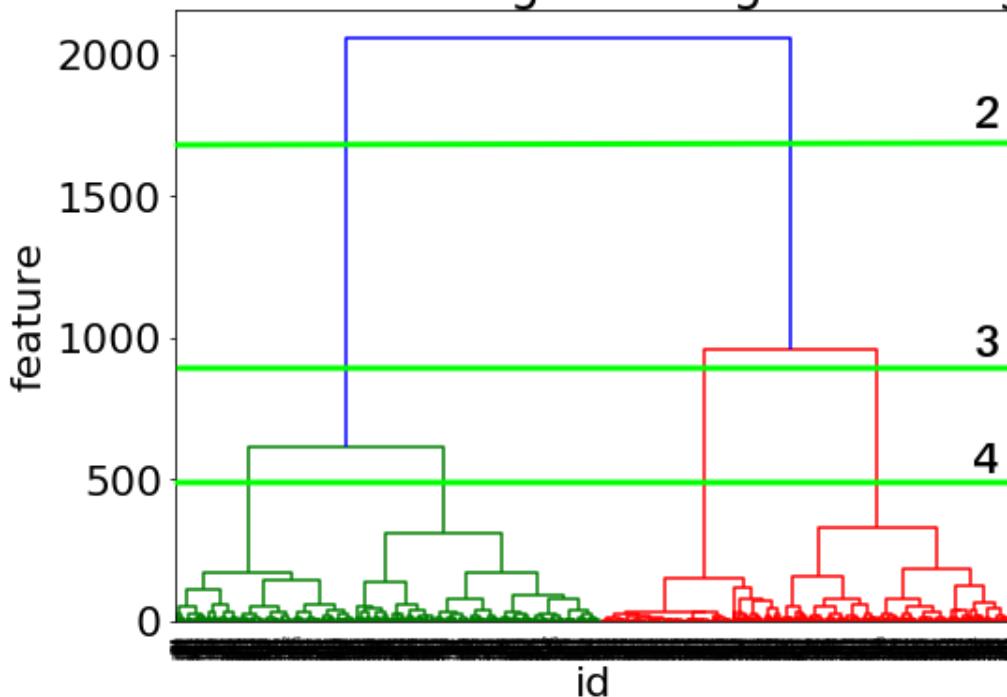
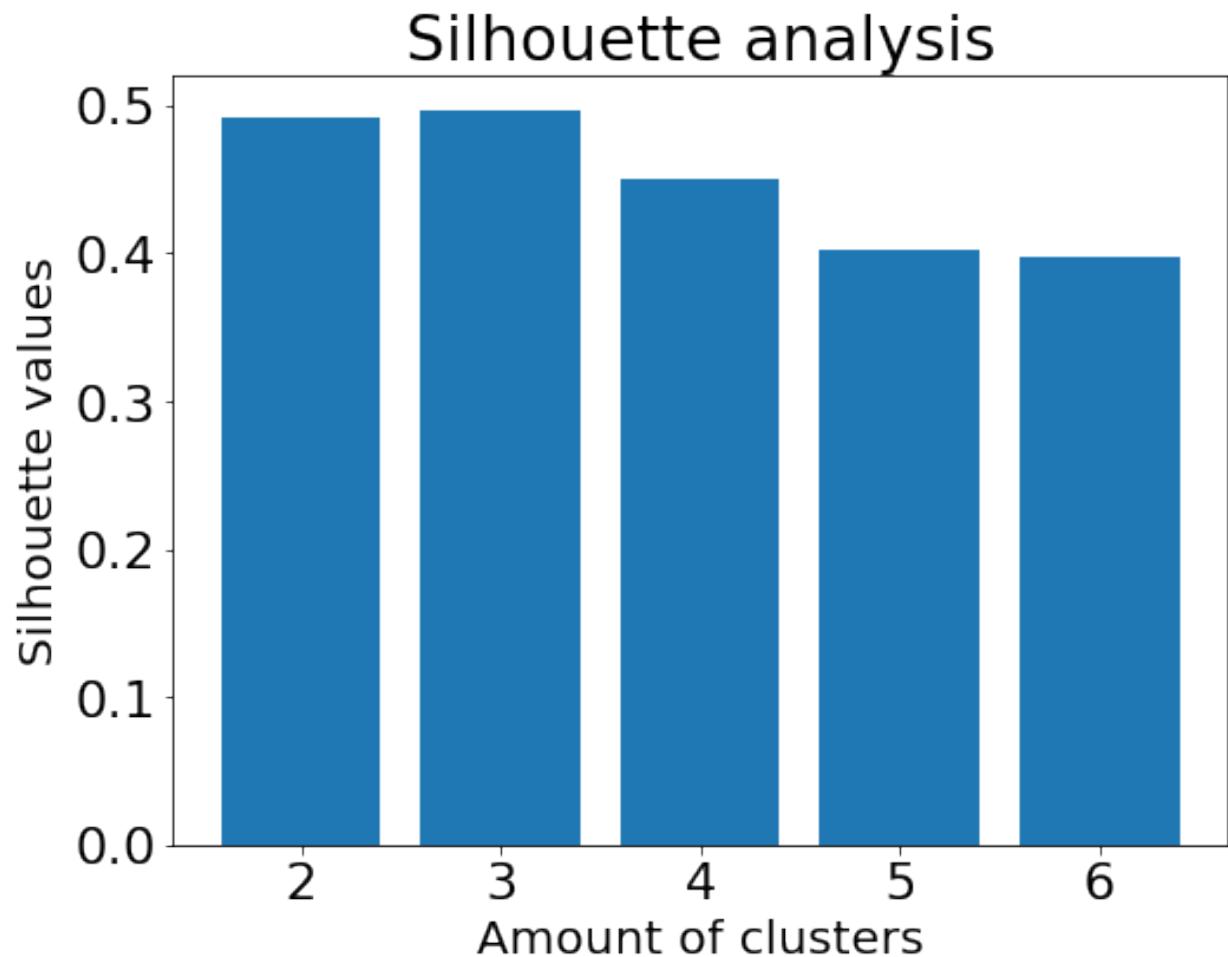


Figure 5. Posibles separaciones en Clusters

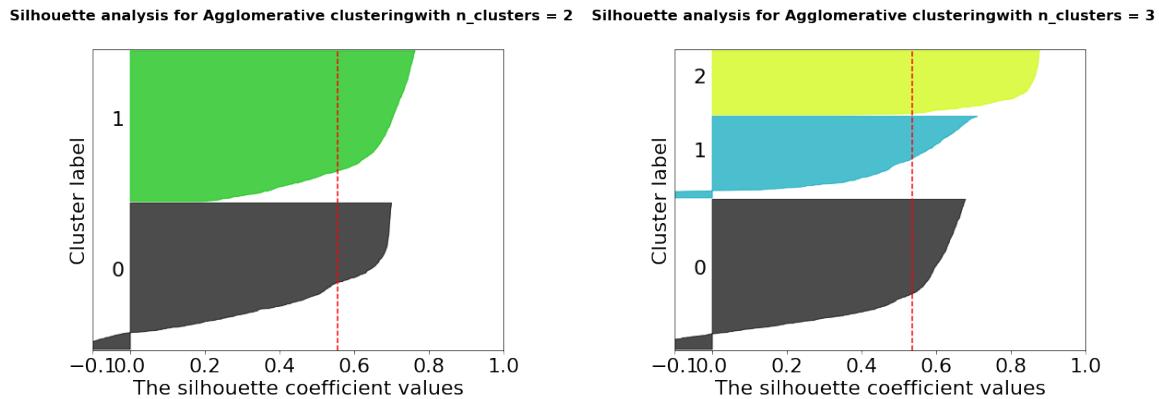
Si bien dicha apreciación podría ser verdadera, nos disponemos a estudiar distintos coeficientes y técnicas que nos permitirán entender cuál es el mejor **K** para este método.

Coeficiente de Silhouette

Para determinar en qué cantidad de clusters conviene más realizar la separación, nos basamos en el coeficiente de Silhouette, el cual nos da una medida para saber cuál es el **K** óptimo para construir la separación. Se realizó el cálculo de dicho coeficiente para 2, 3, 4, 5 y 6 divisiones como para tener una imagen más completa sobre cómo se están comportando los datos.



Como podemos ver en el resultado, claramente con un **K=2** o **K=3** se obtiene el mejor coeficiente, nos disponemos ahora a realizar un análisis un poco más profundo sobre estos valores para tomar una decisión, graficaremos los coeficientes comparando ahora para 2, 3 divisiones.



Cantidad de clusters	Silhouette Score
2	0.5566
3	0.5362

Table 3. Silhouette Score

Si bien se obtiene un mejor coeficiente para un **K=2**, como la diferencia entre este y el **K=3** es muy pequeña seleccionaremos al **K=3** ya que nos permite contar con más agrupaciones y poder lograr una separación en clusters más interesante.

Cross table

Para entender más a detalle que tan bien se realizó la clasificación se genera la siguiente tabla donde se aprecia el agrupamiento de géneros en los distintos clusters.

Género	Clusters		
	0	1	2
ambient	274	154	32
classical	308	72	25
drum-and-bass	42	55	354
jazz	240	153	34
world-music	255	175	33

Table 4. Géneros musicales y grupos

Podemos observar que el mejor agrupamiento lo tenemos en el cluster 2, con el género **drum-and-bass**. Al igual que con K-means y con PAM dicho género es quien concentró en un cluster la mayor cantidad de observaciones. Se dispone a analizarlos índices de RAND y VanDongen para evaluar si los agrupamientos son similares para los distintos conjuntos de datos.

Cantidad de clusters	Indice RAND	Indice VanDongen
2	0.1918	0.7805
3	0.3085	0.7575

Si bien tomamos la decisión de elegir un **K=3**, algunos análisis los haremos con **K=2** también para comparar en todo momento que nuestra elección haya sido adecuada.

Con el índice de VanDongen se observa un alto grado de pureza con los clusters analizados, más de un 75%. Para el índice RAND podemos comentar que el porcentaje de decisiones correctas del cluster es de 30% sin embargo es de esperable tener un porcentaje bajo ya que se están realizando agrupaciones de 3 clusters a diferencia de las 5 etiquetas de género en los datos.

Componentes principales

A continuación se realizó análisis de componentes principales con la finalidad de reducir la dimensionalidad del data set estudiado y así poder lograr una representación gráfica de la división en 3 clusters.

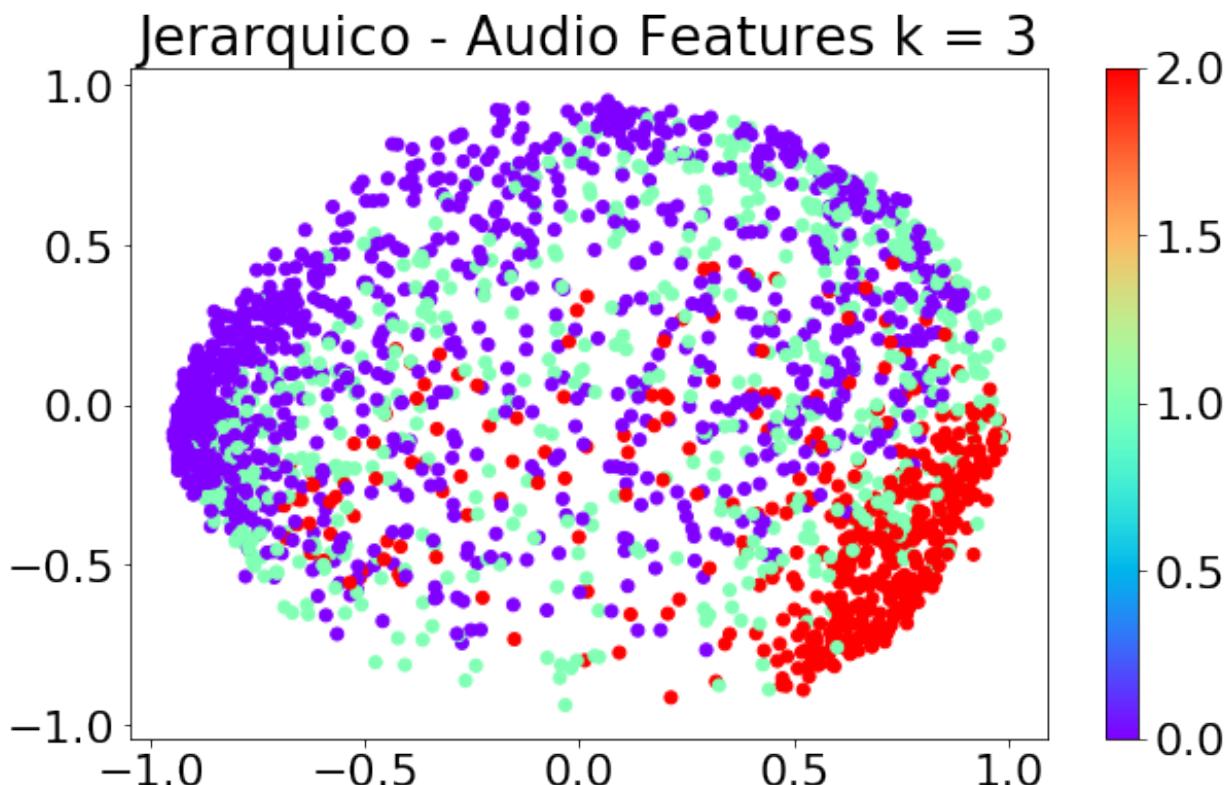
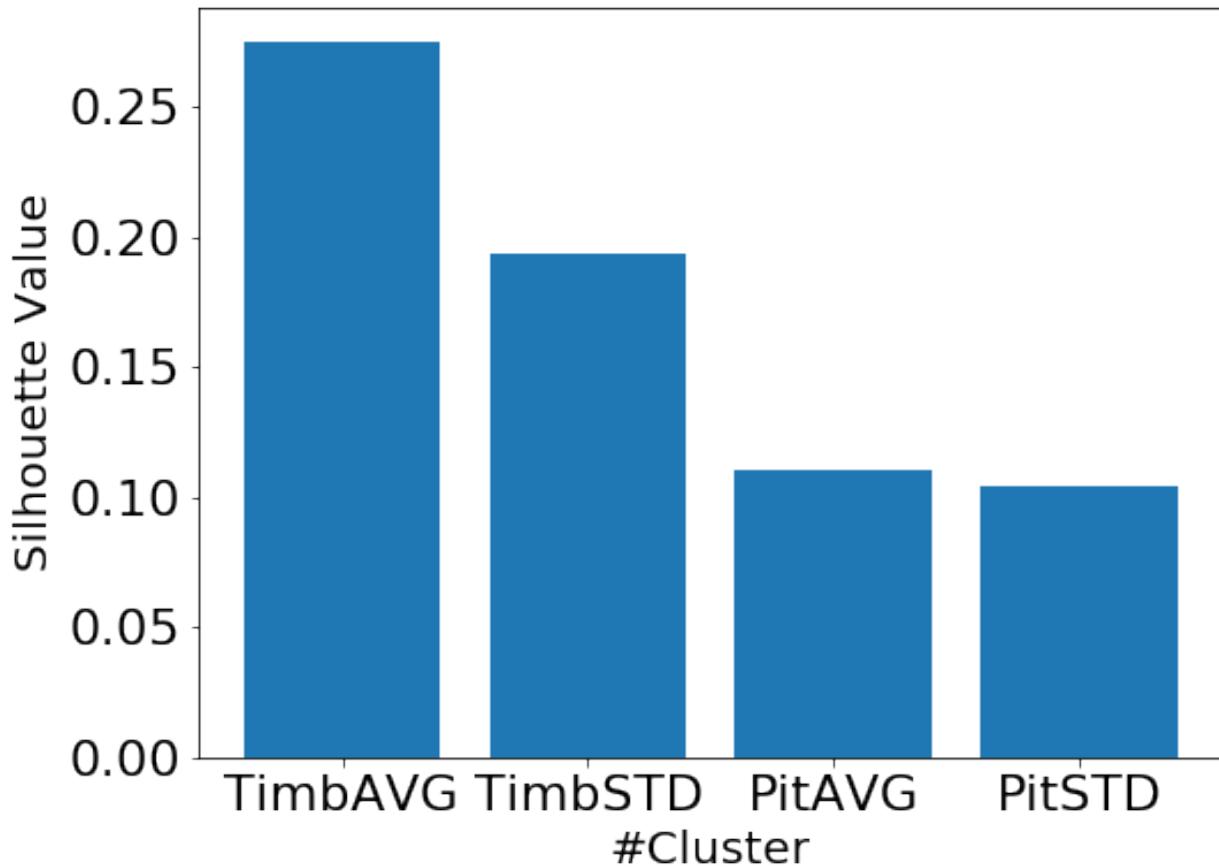


Figure 6. PCA

Dadas las componentes 1 y 2 se observa un buen conglomerado diferenciado en especial para el cluster 2, drum-and-bass, el cual presenta su mayor densidad en la parte positiva de la primera componente y negativa de la segunda. Lo sigue en densidad el cluster 0, y por último el cluster 1 el cual presenta valores más distribuidos y casi sin compactarse. Cabe destacar que en el centro del gráfico, próximo al valor (0,0) se pueden encontrar elementos de los 3 clusters.

Audio Analysis

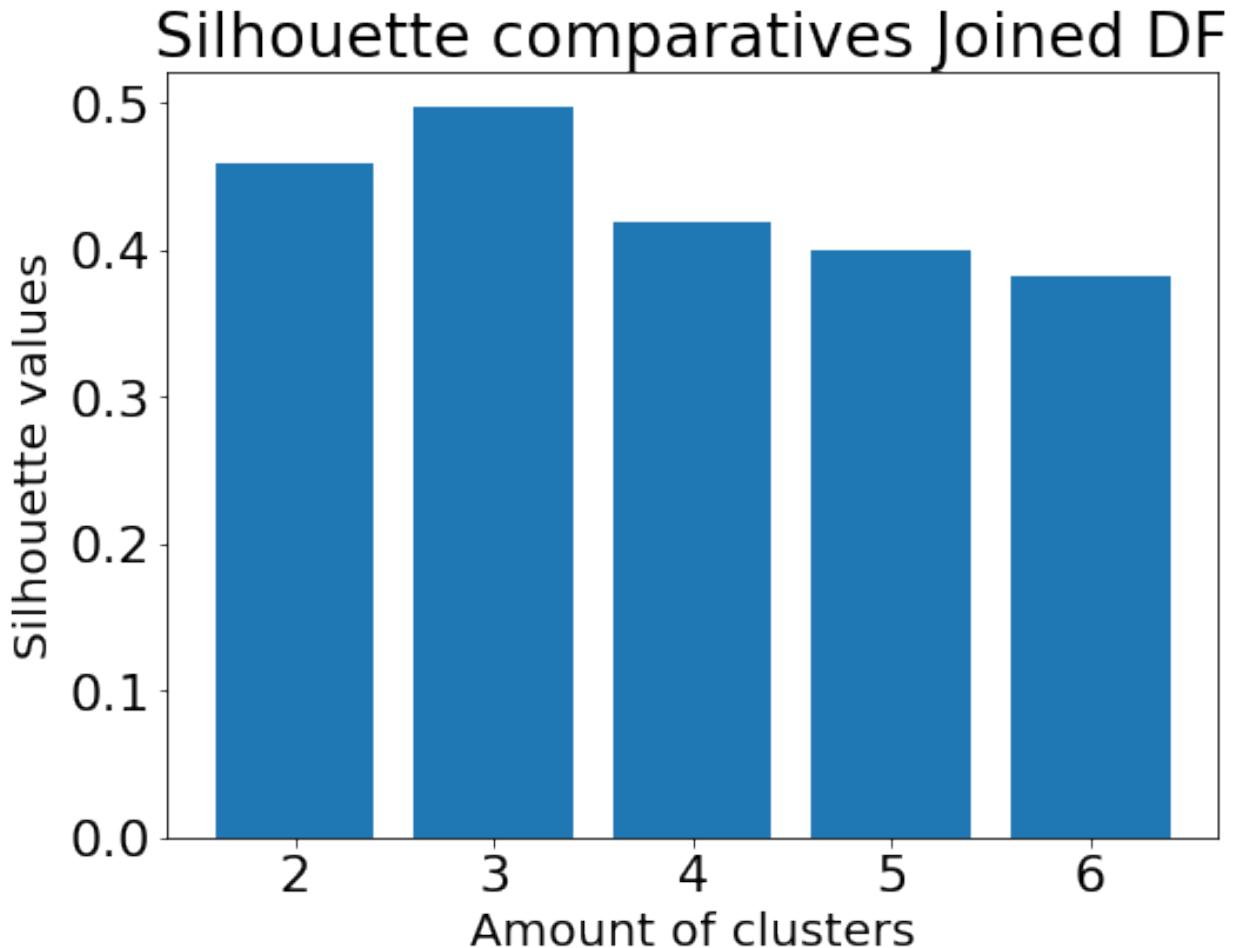
Comenzamos realizando una análisis de Silhouette sobre los distintos tipos de data set *timbres_avg*, *timbres_std_dev*, *pitches_avg* y *pitches_std_dev*. Para hacer esto, nos basamos en la experiencia obtenida con Audio Features, y utilizamos la misma medida Ward y la misma cantidad de clusters óptimos encontrada hasta el momento de 3.



Comparativa general

Data Set	Cantidad de clusters	Indice RAND
timbres avg	3	0.1690
timbres std	3	0.0666
pitches avg	3	0.0754
pirches std	3	0.0231

Observando los valores de Silhouette, la mejor manera de clasificar a las canciones para este dataset es a traves de la media de **timbres average**. Como proximo paso, se procede a realizar una junta entre el dataset de *audio_features* con *audio_analysis* que ya contenia el promedio de los timbres. Para realizar un analisis mas profundo sobre este nuevo data set.

**Figure 7. Silhouette Joined DF**

Cantidad de clusters	Silhouette Score
2	0.4582
3	0.4968
4	0.4192
5	0.3997
6	0.3822

Table 5. Sil full data set

Nuevamente hacemos un balance entre el valor que nos arroja el Silhouette y la mejor configuracion de cantidad de K. Al igual que antes, tanto para **K=2**, **K=3**, **K=4** los valores son muy similares decidimos ir por **K=3** ya que tiene el valor mayor y ademas poder tener cierto grado de comparacion con el trabajo realizado antes. Nos proponemos ahora a calcular los indices RAND y VanDongen

Cantidad de clusters	Indice RAND	Indice VanDongen
2	0.1855	0.6502
3	0.3160	0.5194

El índice de VanDongen da un grado de pureza medio, aproximadamente un 51% para el **K=3**, para el indice

RAND podemos afirmar que el porcentaje de decisiones correctas del cluster es de un 31%, muy similar al anterior análisis y esperable ya que estamos teniendo 3 clusters vs las 5 etiquetas que hay.

Cross table

Para entender a más detalle que tan bien se realizó la clasificación se genera la siguiente tabla donde se aprecia el agrupamiento de géneros en los distintos clusters.

Género	Clusters		
	0	1	2
ambient	334	24	102
classical	386	17	2
drum-and-bass	0	22	429
jazz	158	232	36
world-music	93	334	36

Table 6. Géneros musicales y grupos full df

Al revisar la conformación de los clusters, nuevamente podemos apreciar que el género **drum-and-bass** es quien se concentra en su mayoría en el cluster 3.

Componentes principales

Nuevamente realizamos un análisis de componentes principales con la finalidad de reducir la dimensionalidad del data set estudiado y así poder lograr una representación gráfica de la división en 3 clusters.

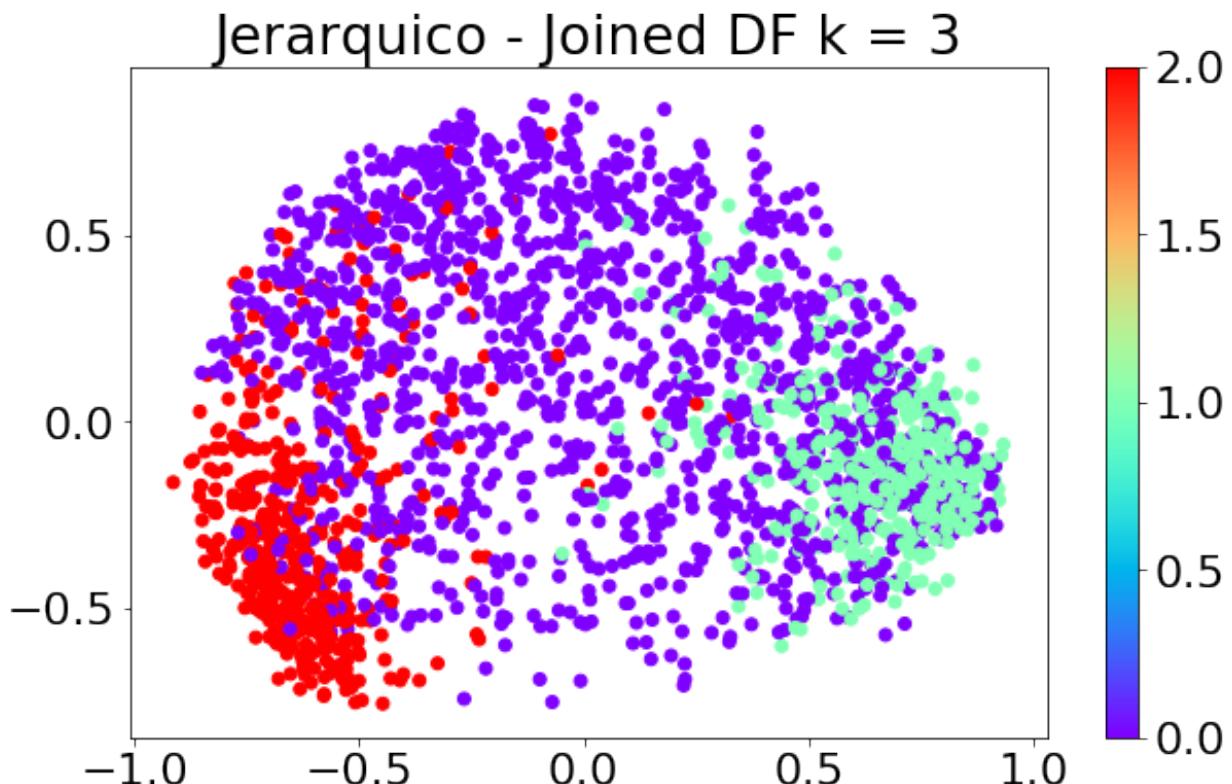


Figure 8. PCA Joined DF

Dadas las componentes 1 y 2 se observa un buen conglomerado diferenciado en especial para el cluster 2, drum-and-bass, el cual tiene su mayor cantidad de valores en la zona de proyección negativa de la primer y segunda componente. Un caso algo diferente es el del cluster 1 que si bien logra algo de aglomeración, se proyecta casi en su totalidad en la parte positiva de la primer componente e intercalando entre positivo y negativo para la segunda componente. Por ultimo esta el cluster 0, el cual muestra mayor esparcimiento y ocupando valores positivos y negativos de ambas componentes. Se aprecia por ultimo, una separación más marcada entre el cluster 1 y 2, mientras que el 0 esta más desparramado.

Experiencia algoritmo PAM

Se utilizó el algoritmo **KMedoids** de la librería sklearn extra.cluster y con la métrica de distancia euclidean.

Métodos

Con el dataset de audio features ejecutamos el algoritmo para distintos k midiendo en cada uno de ellos el promedio del coeficiente de Silhouette para así determinar el k óptimo a utilizar en el análisis.

Los k utilizados fueron [2, 3, 4, 5, 6, 8, 10] y resultados obtenidos los siguientes:

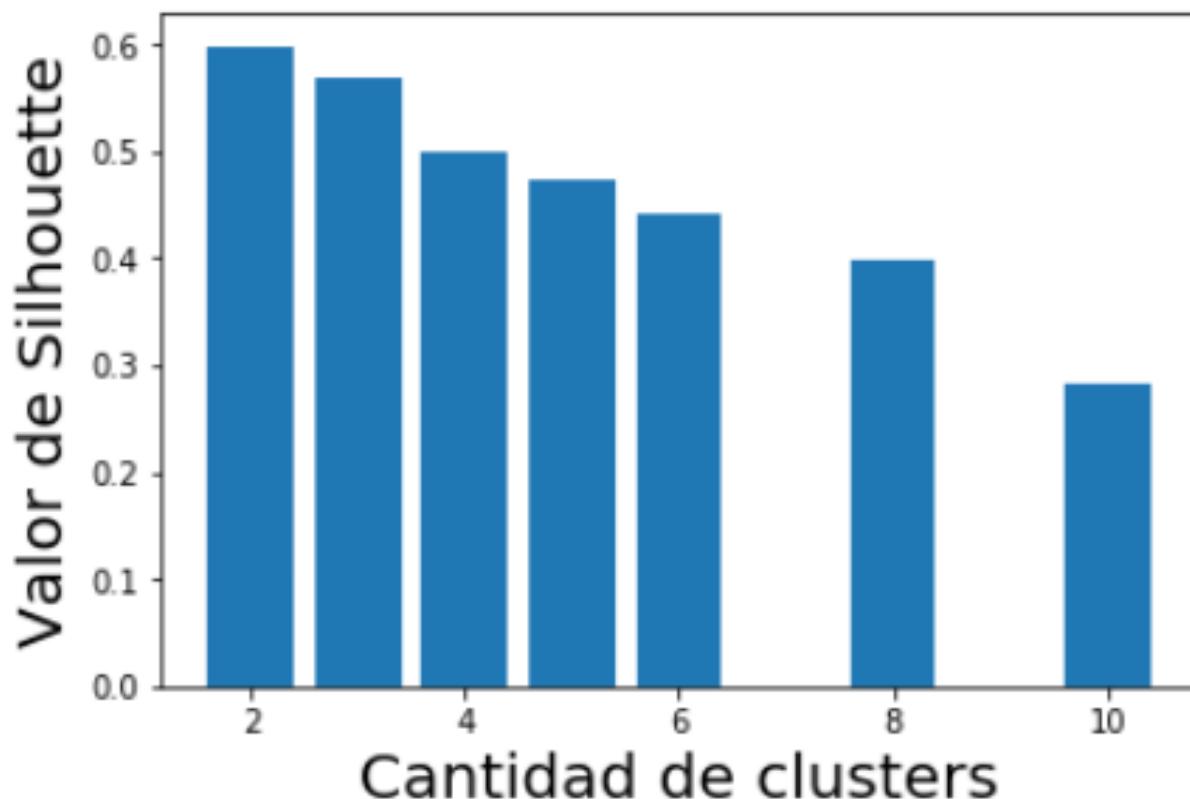


Gráfico valores de silhouette

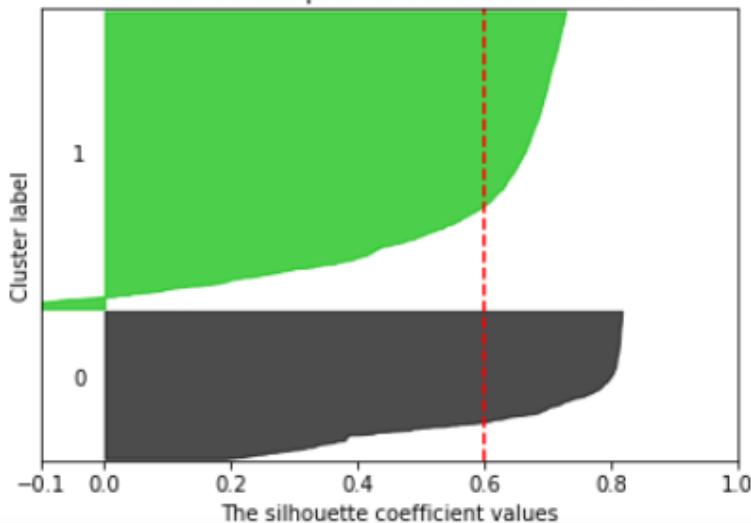
```
For n_clusters = 2 The average silhouette_score is : 0.5985589893752628
For n_clusters = 3 The average silhouette_score is : 0.5689552534692909
For n_clusters = 4 The average silhouette_score is : 0.49849838557654014
For n_clusters = 5 The average silhouette_score is : 0.47295738434050993
For n_clusters = 6 The average silhouette_score is : 0.4413225766604276
For n_clusters = 8 The average silhouette_score is : 0.39911869398929767
For n_clusters = 10 The average silhouette_score is : 0.2824609721868065
```

Coeficiente de Silhouette para varias ejecuciones.

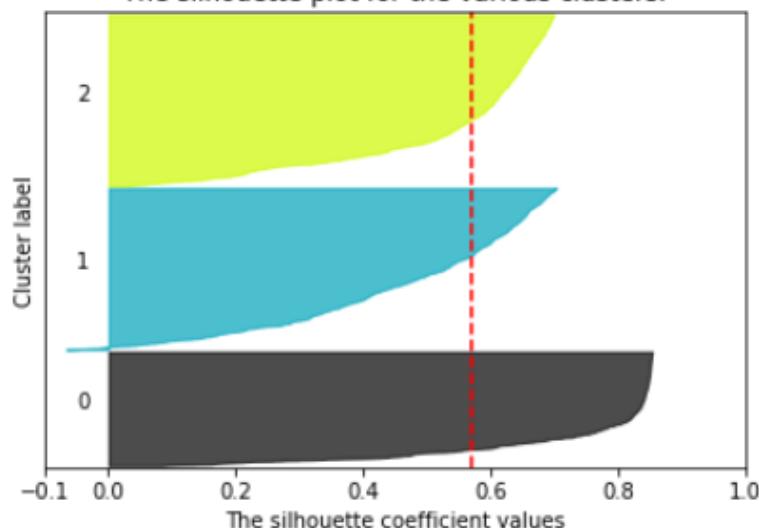
Con el coeficiente de Silhouette decidimos continuar con el k=3 debido a que presentaba un coeficiente cercano al n=2 y así podríamos observar más cantidad de agrupaciones, de igual manera se realizó gráfico para k=2 y k=3 para dejar evidencia de los parecido de los algoritmos con ambos k.

Silhouette analysis for PAM clustering on sample data with n_clusters = 2

The silhouette plot for the various clusters.

**Silhouette analysis for PAM clustering on sample data with n_clusters = 3**

The silhouette plot for the various clusters.



Importante acotar que para las ejecuciones se utilizaron las variables con sus escalas originales, no se aplica normalización ni estandarización.

Con k=3 logramos un coeficiente de 0.5689 lo cual es un número bastante aceptable a nivel general. Para conocer el nivel de concentración de cada cluster obtenemos el cálculo agrupado.

silho_v

Clusters

0	0.745832
1	0.472737
2	0.542215

Silhouette por cluster.

Se observa que el cluster "0" posee un coeficiente más alto que el resto con lo cual la observaciones pertenecientes a se encuentran mejor agrupadas. Para ver que tan bien clasificó nuestro clustering con respecto a los géneros de música que se tenían en el dataset realizamos una crosstable.

Clusters	0	1	2
Genero			
ambient	55	184	221
classical	32	120	253
drum-and-bass	359	53	39
jazz	56	213	158
world-music	60	228	175

Se valida que el género mejor agrupado es **drum-and-bass** el cual presenta gran concentración en el cluster 0. Esta buena agrupación para este género coincide con los resultados observados en el análisis realizado con el método de clustering K-Means.

Utilizamos los índices de RAND y VanDongen para evaluar si los agrupamientos son similares para los distintos conjuntos de datos.

Con el algoritmo de KMedoids y la configuración antes indicada se obtuvieron los siguientes valores.

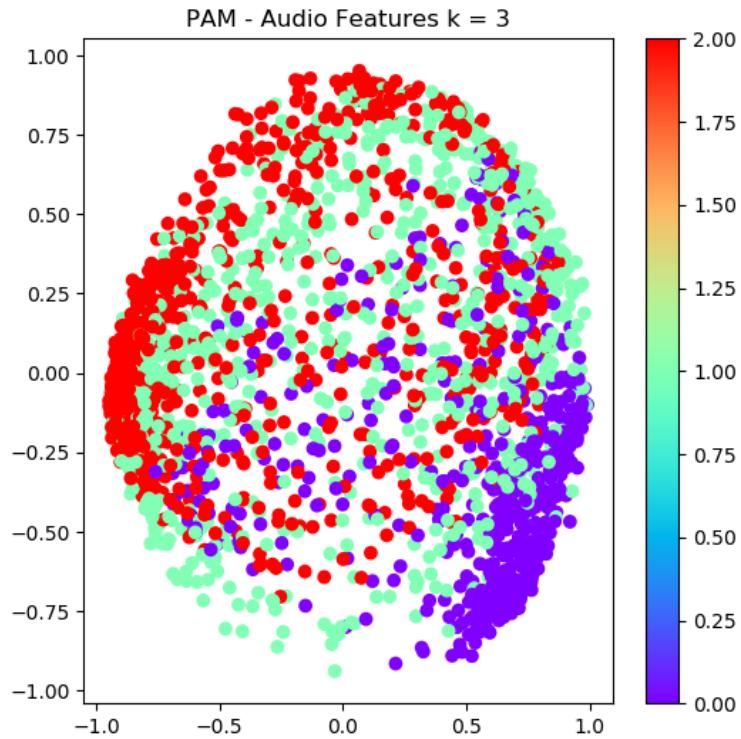
Índice Rand para Algoritmo PAM con 3 clusters: 0.1240

Índice VanDongen para Algoritmo PAM con 3 clusters: 0.7405

Con el índice de VanDongen se observa un alto grado de pureza con los clusters analizados.

Para el índice RAND podemos comentar que el porcentaje de decisiones correctas del cluster es de 12% sin embargo es de esperable tener un porcentaje bajo ya que se están realizando agrupaciones de 3 clusters a diferencia de las 5 etiquetas de género en los datos.

A continuación se realizó Análisis de Componentes con la finalidad de reducir la dimensionalidad de nuestro dataset para poder realizar una representación gráfica de nuestros clusters.



Con las componentes 1 y 2 se observa un buen conglomerado diferenciado en los cluster 2 y 0, sin embargo el cluster 1 se encuentra más distribuido a lo largo y ancho del gráfico. Podemos decir que el cluster "0" drum-and-bass se caracteriza en su mayoría por tener valores positivos en la primera componente (eje x) y negativos en la segunda componente (eje y), mientras que el cluster 2 presenta en su mayoría negativos en x y positivos cercanos a cero en la segunda componente.

También podemos ver una separación un poco más clara entre el cluster 0 y 1 que el cluster 2 con respecto al resto.

Resultados

Discusión

Experiencia Algoritmo DBSCAN

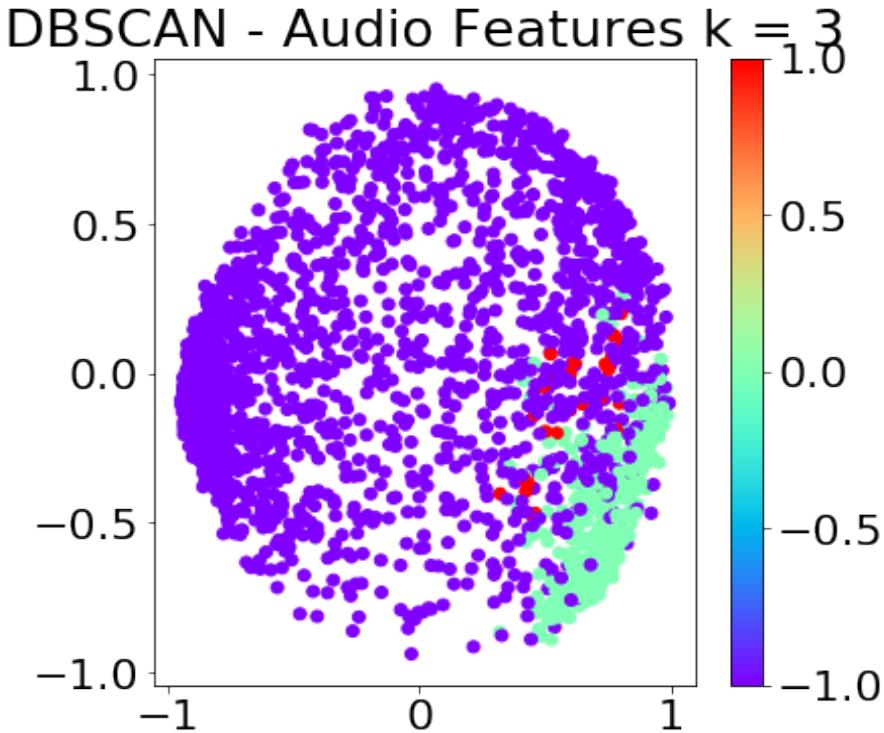
Como acercamiento y validación de otros métodos de clustering decidimos probar con el algoritmo DBSCAN el cual al estar basado en densidad permite agrupar de mejor manera datos que no posean una forma particular.

Con una configuración de $\text{eps} = 2$ un $\text{min sample} = 25$ y métrica euclidean se obtuvieron 3 clusters los cuales presentaban una métrica de silhouette de 0.1564 y la siguiente crossTable:

Clusters	-1	0	1
Genero			
ambient	165	279	16
classical	46	359	0
drum-and-bass	90	6	355
jazz	156	269	2
world-music	158	293	12

Vemos una muy buena clasificación para el género drum-and-bass en el cluster 1, sin embargo el resto de los géneros se encuentran muy distribuidos por el resto de los cluster lo cual puede ser el motivo de obtener un coeficiente de silhouette tan bajo.

En el siguiente gráfico podemos observar las clasificación del algoritmo en baja dimensionalidad (Utilizamos PCA para reducir las dimensiones).



Tal como se comentó anteriormente la clasificación del género drum-and-bass se encuentra concen-

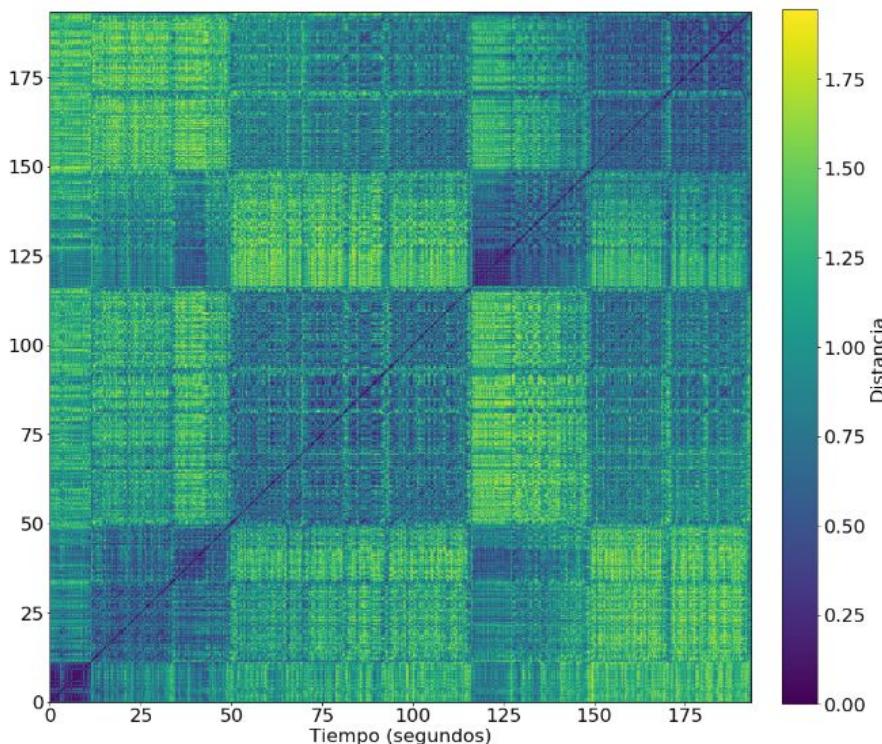
trada en el cluster 1 y lo podemos observar en el gráfico de las componentes principales.

Podemos comentar que las causas de la baja performance de este algoritmo se debe a que precisamente se trata de un método basado en densidad y según lo observado durante la elaboración del pre TP1 en la sección de scatter plot's muchos de los features del dataset presentaban altas concentraciones y se mezclaban los distintos géneros musicales.

Clustering de secciones dentro de una pista

Para este análisis se seleccionó una pista de Timbres de los datos de Audio Analysis, específicamente la pista cuyo id es “ooAt7PWydsdg7g5xgaYang”, es una canción de género Electro/Pop: All I Know - Matrix & Futurebound ft. Luke Bingham.

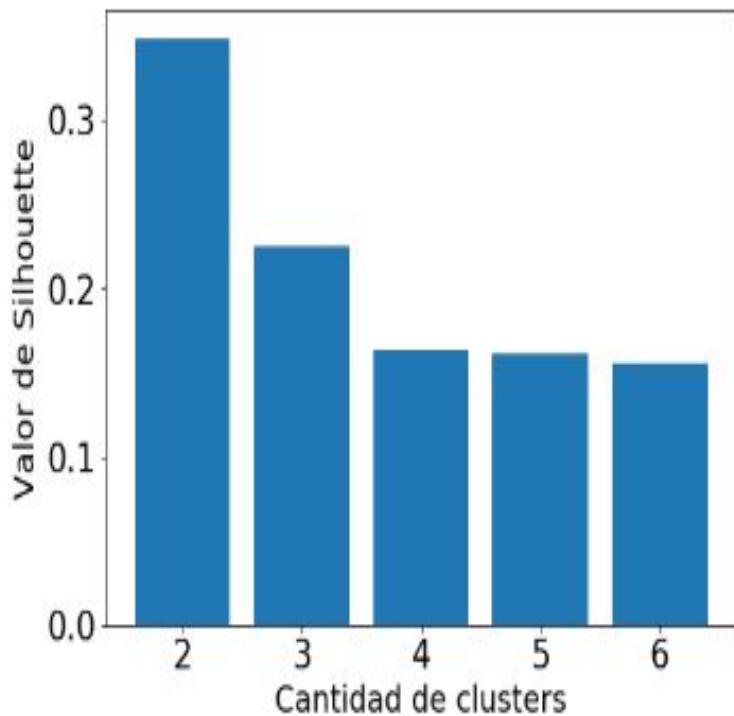
Se realizó una matriz de recurrencia con los datos normalizados y la serie temporal de la pista interpolada obteniendo los siguientes resultados:



Lo primero que se puede destacar es en los primeros segundos de la canción, se ve claramente una intro que es bastante distinta al resto, seguida de un par de secciones parecidas entre sí (estribillos y coros). Se puede observar también en el segundo 125 un segmento de la canción con características similares al intro.

Para validar los resultados del análisis se escuchó la pista validando que la intro se caracteriza por un sonido más instrumental/electrónico, sin vocalización, similar al del segundo 125. El resto de la canción tiene vocalizaciones con tonos muy uniformes.

Así mismo, para corroborar este análisis se utilizó el algoritmo de clustering de KMeans para identificar distintos grupos dentro de la pista, obteniéndose los siguientes resultados medidos con la métrica de Silhouette:



Los resultados obtenidos indican una clara identificación de dos clusters, que según lo escuchado en la pista y lo visto en la matriz de recurrencia serían las secciones instrumentales/electrónicas y las secciones donde hay vocalización. Si bien la mejor distinción es la de $k=2$, con $k=3$ también se observa una leve mejora con respecto a los $k > 3$, esto puede ser a la diferenciación de la vocalización escuchada en los estribillos y el coro.

Conclusiones

- El análisis de cluster permite...
- Es fundamental contar con técnicas no solo numéricas sino gráficas
- El análisis por PCA
- Distintas medias de distancia entre los clusters
- Entendimiento del dominio