**TOP100BILLBOARD – TAREA BASE DE DATOS NOSQL MONGODB**

**Introducción**

El siguiente trabajo tiene por objetivo hacer un análisis de sentimiento del dataset que recoge las canciones de “Billboard Hot 100” es la lista de récords estándar de la industria musical en Estados Unidos para canciones, publicada semanalmente por la revista Billboard. Las clasificaciones de las listas se basan en las escuchas de canciones vía streaming online en la plataforma Spotify en Estados Unidos.

Este dataset proviene de [https://data.world/kcmillersean/billboard-hot-100-1958-2017#](https://data.world/kcmillersean/billboard-hot-100-1958-2017)[[1]](#footnote-1)

creado por [Sean Miller](https://data.world/kcmillersean), [Billboard.com](http://billboard.com/" \t "_blank) and Spotify.

Mediante las consultas de queries, se pretende hacer un análisis de sentimiento musical, respecto a la valencia musical y semántica, lenguaje explícito, modo armónico mayor o menor, tempo, bailabilidad, energía, valencia, entre otras características que poseen las canciones aquí consultadas y que se buscará determinar un “índice de emotividad” de una canción las cuales responden a las características antes mencionadas.   
En función de lo anterior mencionado, podremos valernos de los datos que nos aporta este dataset y a partir de los valores en lo que haré las búsquedas podré determinar que canciones pueden ser consideradas como “tristes” y cuales “alegres”.   
Si bien la percepción musical es subjetiva, y no necesariamente estos filtros y valores respondas a las características de las canciones, si los campos de estos documentos pueden aportarnos un acercamiento al análisis.

**Los Campos que componen el documento de cada Canción o pista y sus respectivas características de audio son: [[2]](#footnote-2)**

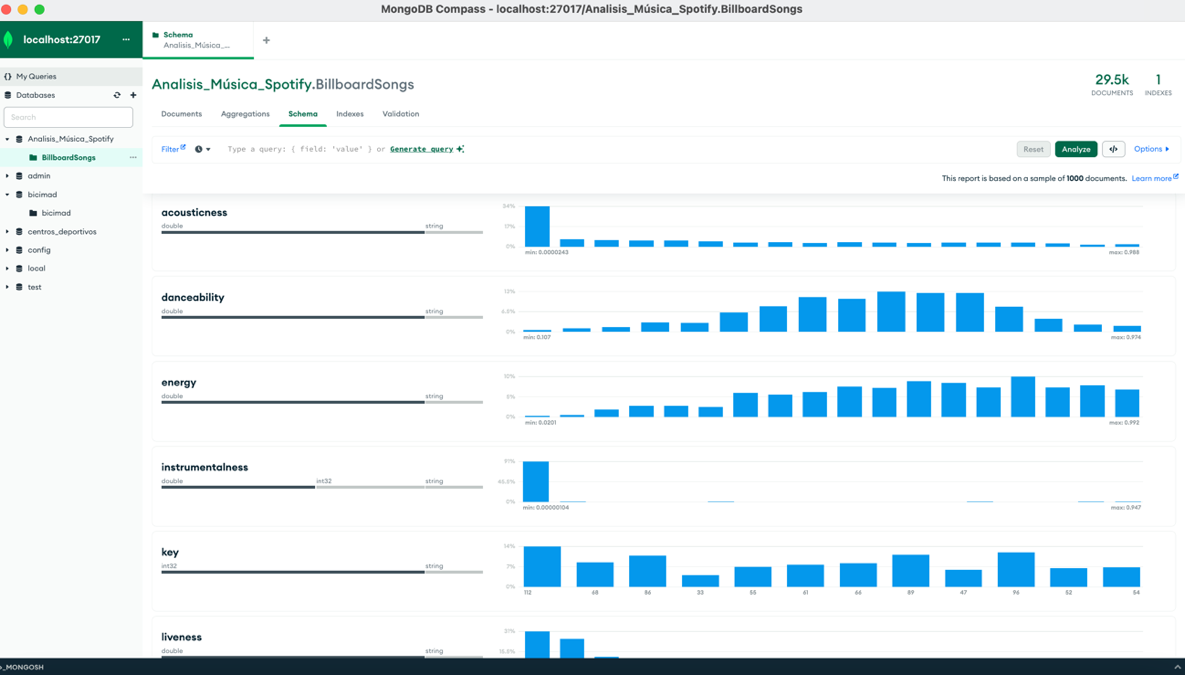
* "\_id" : ObjectId
* "song\_id" : Id canción
* "performer" : Artista
* "song" : Título canción
* "spotify\_genre" : Género musical
* "spotify\_track\_id" : Track Id
* "spotify\_track\_preview\_url" : Spotify URL
* "spotify\_track\_duration\_ms" : Duración de la cancion
* "spotify\_track\_explicit" : si contiene lenguaje explícito
* "spotify\_track\_album" : Nombre del álbum
* "danceability" : la bailabilidad, combinación de elementos musicales como el tempo, la estabilidad del ritmo, la fuerza del compás y la regularidad general. Un valor de 0,0 es el menos bailable y 1,0 el más bailable.
* "energy" : La energía es una medida de 0,0 a 1,0 y representa una medida perceptiva de intensidad y actividad. Las características perceptivas que contribuyen a este atributo incluyen el rango dinámico, el volumen percibido, el timbre, la velocidad de inicio y la entropía general.
* "key" : Tonalidad musical de la canción. Se utiliza el cifrado americano para representar los tonos. (0 = C, 1 = C#, 2 = D, 3 = D#, 4 = E, 5 = F, 6 = F#, 7 = G, 8 = G#, 9 = A, 10 = A#, 11 = B). C = do, D = re, E = mi, F = fa, G = sol, A = la, B = si.  
  Si no se identifica el tono de la canción, el valor será -1.
* "loudness" : La sonoridad global de una pista en decibelios (dB). Los valores de sonoridad se promedian en toda la pista y son útiles para comparar la sonoridad relativa de las pistas. La sonoridad es la cualidad de un sonido que es el principal correlato psicológico de la fuerza física (amplitud). Los valores suelen oscilar entre -60 y 0 db.
* "mode" : El modo indica la modalidad (mayor o menor) de una pista, el tipo de escala del que se deriva su contenido melódico. Mayor se representa con 1 y menor con 0.
* "speechiness" : La locuacidad detecta la presencia de palabras habladas en una pista. Cuanto más exclusivamente hablada sea la grabación (por ejemplo, programa de entrevistas, audiolibro, poesía), más se acercará a 1,0 el valor del atributo. Los valores superiores a 0,66 describen pistas que probablemente estén compuestas en su totalidad por palabras habladas. Los valores entre 0,33 y 0,66 describen pistas que pueden contener tanto música como voz, ya sea en secciones o en capas, incluyendo casos como la música rap. Los valores inferiores a 0,33 representan probablemente música y otras pistas no habladas.
* "acousticness" : Una medida de confianza de 0,0 a 1,0 sobre si la pista es acústica. 1,0 representa una confianza alta en que la pista es acústica.
* "instrumentalness" : Predice si una pista no contiene voces. Los sonidos "ooh" y "aah" se consideran instrumentales en este contexto. Las pistas de rap o spoken word son claramente "vocales". Cuanto más se acerque el valor de instrumental a 1,0, mayor será la probabilidad de que la pista no contenga voces. Los valores superiores a 0,5 representan pistas instrumentales, pero la confianza es mayor a medida que el valor se acerca a 1,0.
* "liveness" : Detecta la presencia de público en la grabación. Los valores de liveness más altos representan una mayor probabilidad de que la pista se haya interpretado en directo. Un valor superior a 0,8 proporciona una gran probabilidad de que la pista sea en directo.
* "valence" : Medida de 0,0 a 1,0 que describe la positividad musical que transmite una pista. Las pistas con valencia alta suenan más positivas (por ejemplo, felices, alegres, eufóricas), mientras que las pistas con valencia baja suenan más negativas (por ejemplo, tristes, deprimidas, enfadadas).
* "tempo" : El tempo global estimado de una pista en pulsaciones por minuto (BPM). En terminología musical, el tempo es la velocidad o el ritmo de una pieza determinada y se deriva directamente de la duración media de los tiempos.
* "time\_signature" : El compás general estimado de una pista. El compás es una convención de notación que especifica cuántos tiempos hay en cada compás. Ej: 4/4, 3/4.
* "spotify\_track\_popularity" : La popularidad de la pista. El valor estará entre 0 y 100, siendo 100 la más popular. La popularidad se calcula mediante un algoritmo y se basa, en su mayor parte, en el número total de reproducciones que ha tenido la canción y lo recientes que son esas reproducciones.

**1 – Carga del fichero CSV**

(base) emvlp@MacBook-Pro-de-Emiliano bin % ./mongod --dbpath /usr/local/mongodb/data/db > /usr/local/mongodb/logs/mongod.log 2>&1 &

[1] 13010

mongoimport --db Billboard\_Songs -- Spotify\_songs --type csv --headerline --ignoreBlanks -- file /Users/emvlp/Downloads/Top100Billboard/audio\_features.csv

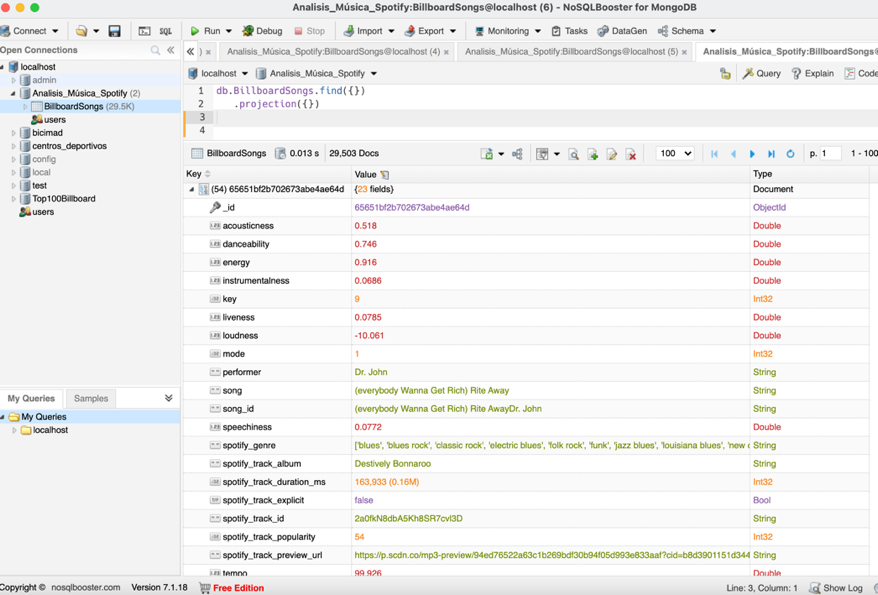


MONGODB COMPASS SCHEMA

Con una consulta para ver todos los documentos, su clave – valor y tipo.

db.BillboardSongs.find({})

.projection({})



Podemos observar que los documentos tienen 23 campos.

**2 – Queries**

Una vez creado el dataset en MongoDB Compass e insertada la colección de documentos, analizamos la información que contienen estos conjuntos de documentos con la herramienta Schema, que nos ayuda a obtener toda la información de los documentos y sus respectivos campos.

Los criterios y valores de búsqueda insertados con **$gte** (mayor que) y **$lt** (menor que) son operadores de comparación para filtrar los documentos basándome en un rango, para pistas tristes he utilizado valores de la media hacia abajo y para pistas alegres valores de la media hacia arriba.

* **Con el siguiente Query, podemos contar cuántos documentos conforman el dataset, un total de 29503.**

db.BillboardSongs.count()

* **Para analizar las canciones con mayor y menor “acousticness”, es decir, aquellas pistas en que predomine o no el formato acústico:**

db.BillboardSongs.find({})

.projection({})

var documentosMenorAcousticness = db.BillboardSongs.aggregate([

{

$match: {

"acousticness": { $gte: 0.05, $lt: 0.3 }

}

},

{

$sort: {

"acousticness": 1

}

},

{

$limit: 100

}

]);

// Imprime el resultado

print("Documentos con menor acousticness en el rango [0.05, 0.3):", documentosMenorAcousticness.toArray());

db.BillboardSongs.find({})

.projection({})

var documentosMayorAcousticness = db.BillboardSongs.aggregate([

{

$match: {

"acousticness": { $gte: 0.5, $lt: 1.0 }

}

},

{

$sort: {

"acousticness": -1

}

},

{

$limit: 100

}

]);

// Imprime el resultado

print("Documentos con mayor acousticness en el rango [0.5, 1.0):", documentosMayorAcousticness.toArray());

* **Analizamos las 100 pistas con menor y mayor “danceability”: Describe lo adecuada que es una pista para bailar basándose en una combinación de elementos musicales como el tempo, la estabilidad del ritmo, la fuerza del compás y la regularidad general.**

db.BillboardSongs.find({})

.projection({})

var pistasMenorDanceability = db.BillboardSongs.aggregate([

{

$match: {

"danceability": { $gte: 0.01, $lt: 0.3 }

}

},

{

$sort: {

"danceability": 1

}

},

{

$limit: 100

}

]);

print("Las 100 pistas con el menor danceability:", pistasMenorDanceability.toArray());

db.BillboardSongs.find({})

.projection({})

var pistasMayorDanceability = db.BillboardSongs.aggregate([

{

$match: {

"danceability": { $gte: 0.7, $lt: 1.0 }

}

},

{

$sort: {

"danceability": -1

}

},

{

$limit: 100

}

]);

print("Las 100 pistas con el mayor danceability:", pistasMayorDanceability.toArray());

* **Pistas con menor y mayor “energy”. Representa una medida perceptiva de intensidad y actividad.**

db.BillboardSongs.find({})

.projection({})

var pistasMenorEnergy = db.BillboardSongs.aggregate([

{

$match: {

"energy": { $gte: 0.1, $lt: 0.3 }

}

},

{

$sort: {

"energy": 1

}

},

{

$limit: 100

}

]);

// Imprime el resultado

print("Las 100 pistas con la menor energy:", pistasMenorEnergy.toArray());

db.BillboardSongs.find({})

.projection({})

var pistasMayorEnergy = db.BillboardSongs.aggregate([

{

$match: {

"energy": { $gte: 0.6, $lt: 1.0 }

}

},

{

$sort: {

"energy": -1

}

},

{

$limit: 100

}

]);

print("Las 100 pistas con la mayor energy:", pistasMayorEnergy.toArray());

* **Consultado el campo “mode”, el cual representa la modalidad mayor = 1 y menor = 0, de una pista, el tipo de escala musical del que deriva su contenido melódico. Con esto veremos cuáles pistas se encuentran en una tonalidad mayor (más “alegres”) y cuáles en una tonalidad menor (más “tristes”).**

db.BillboardSongs.find({})

.projection({})

var pistasTristes = db.BillboardSongs.aggregate([

{

$match: {

"mode": 0

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

mode: 1

}

}

]);

print("Las 100 pistas con modo triste:", pistasTristes.toArray());

var pistasAlegres = db.BillboardSongs.aggregate([

{

$match: {

"mode": 1

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

mode: 1

}

}

]);

print("Las 100 pistas con modo alegre:", pistasAlegres.toArray());

* **La consulta en función de la “valence” (valencia). Describe la positividad musical que transmite una pista. Las pistas con valencia alta suenan más positivas (por ejemplo, felices, alegres, eufóricas), mientras que las pistas con valencia baja suenan más negativas (por ejemplo, tristes, deprimidas, enfadadas).**

db.BillboardSongs.find({})

.projection({})

var pistasmenorValence = db.BillboardSongs.aggregate([

{

$match: {

"valence": { $gte: 0.01 $lt: 0.5 }

}

},

{

$sort: {

"valence": 1

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

valence: 1

}

}

]);

print("Las 100 pistas con menor valence:", pistasmenorValence.toArray());

var pistasmayorValence = db.BillboardSongs.aggregate([

{

$match: {

"valence": { $gte: 0.5, $lt: 1 }

}

},

{

$sort: {

"valence": -1

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

valence: 1

}

}

]);

print("Las 100 pistas con mayor valence:", pistasmayorValence.toArray());

* **La consulta relacionada con el “tempo” se refiere a los beats por minuto (BPM), considerando un ritmo lento entre 60-80 BPM y un ritmo rápido entre 120-160 BPM**.

db.BillboardSongs.find({})

.projection({})

var pistasTempolento = db.BillboardSongs.aggregate([

{

$match: {

"tempo": { $gte: 60, $lt: 80 }

}

},

{

$sort: {

"tempo": 1

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

tempo: 1

}

}

]);

print("Las 100 pistas con tempo lento:", pistasTempolento.toArray());

Esta consulta utiliza la etapa **$project** para proyectar solo los campos "song" y "tempo" en la salida, y luego imprime los resultados.

var pistasTemporapido = db.BillboardSongs.aggregate([

{

$match: {

"tempo": { $gte: 120, $lt: 160 }

}

},

{

$sort: {

"tempo": -1

}

},

{

$limit: 100

},

{

$project: {

\_id: 0,

performer: 1,

song: 1,

tempo: 1

}

}

]);

print("Las 100 pistas con tempo rapido:", pistasTemporapido.toArray());

**En función de estos campos vamos a determinar cuáles pistas son más “tristes” y cuáles son más “alegres”. Para esto haré uso de la sintaxis $match para filtrar solo aquellos documentos que cumplen con las condiciones especificadas.**

**Query de las pistas más tristes en función de la consulta de los documentos con sus respectivos criterios y valores especificados.**

db.BillboardSongs.find({})

.projection({})

var pistasAgrupadasTristes = db.BillboardSongs.aggregate([

{

$match: { // Filtro los documentos

"acousticness": { $gte: 0.01, $lt: 0.5 },

"danceability": { $gte: 0.01, $lt: 0.5 },

"energy": { $gte: 0.01, $lt: 0.5 },

"mode": 0,

"tempo": { $gte: 60, $lt: 80 },

"valence": { $gte: 0.01, $lt: 0.5 },

//"loudness": { $gte: -60, $lt: 0 },

//"spotify\_track\_explicit": true,

//"time\_signature": 3

}

},

{

$group: { // Agrupo los documentos por los valores especificados

\_id: {

acousticness: "$acousticness",

danceability: "$danceability",

energy: "$energy",

mode: "$mode",

tempo: "$tempo",

valence: "$valence",

},

pistas: { $push: "$song" }, // Hago un array para las canciones y artistas

artistas: { $push: "$performer" }

}

},

{

$project: { // Se proyectan los documentos creando un nuevo campo llamado pistas\_y\_artistas, un nuevo array concatenando estos campos anteriores.

\_id: 0,

pistas\_y\_artistas: {

$map: {

input: { $range: [0, { $size: "$pistas" }] },

as: "index",

in: {

$concat: [

{ $arrayElemAt: ["$pistas", "$$index"] },

" - ",

{ $arrayElemAt: ["$artistas", "$$index"] }

]

}

}

},

acousticness: "$\_id.acousticness",

danceability: "$\_id.danceability",

energy: "$\_id.energy",

mode: "$\_id.mode",

tempo: "$\_id.tempo",

valence: "$\_id.valence",

}

},

{

$sort: { // ordena en orden ascendente, primero valores más bajos

"acousticness": 1,

"danceability": 1,

"energy": 1,

"mode": 1,

"tempo": 1,

"valence": 1,

"loudness": 1,

"spotify\_track\_explicit": 1,

"time\_signature": 1

}

}

]);

print("Pistas agrupadas:", pistasAgrupadasTristes.toArray());

La siguiente consulta me devuelve 13 documentos, los cuales refieren a las canciones agrupadas con ciertos criterios que se consideran “tristes” en función de los campos seleccionados y ajustando los valores de la media hacia abajo.

Los criterios de selección para filtrar las pistas tristes han sido la siguientes características, Acousticness, Danceability, Energy, Mode, Tempo y Valence , los cuales han sido descriptos en la primera parte del trabajo.   
Bajo estos criterios, se podría considerar que los artistas y sus pistas responden a que dichas canciones se caracterizan por ser más melancólicas o tranquilas, con bajo tempo, no tan bailables, un modo armónico menor y una valencia con tendencia menos positiva.

**Query de las pistas más alegres considerando la consulta de dichos campos y sus valores. Ordenado de mayor a menor.**

db.BillboardSongs.find({})

.projection({})

var pistasAgrupadasAlegres = db.BillboardSongs.aggregate([

{

$match: {

"acousticness": { $gte: 0.5, $lt: 1 },

"danceability": { $gte: 0.7, $lt: 1 },

"energy": { $gte: 0.6, $lt: 1 },

"mode": 1,

"tempo": { $gte: 120, $lt: 160 },

"valence": { $gte: 0.5, $lt: 1 },

"loudness": { $gte: -20, $lt: 0 },

"spotify\_track\_explicit": false,

"time\_signature": 4

}

},

{

$group: {

\_id: {

acousticness: "$acousticness",

danceability: "$danceability",

energy: "$energy",

mode: "$mode",

tempo: "$tempo",

valence: "$valence",

loudness: "$loudness",

spotify\_track\_explicit: "$spotify\_track\_explicit",

time\_signature: "$time\_signature"

},

pistas: { $push: "$song" },

artistas: { $push: "$performer" }

}

},

{

$project: {

\_id: 0,

pistas\_y\_artistas: {

$map: {

input: { $range: [0, { $size: "$pistas" }] },

as: "index",

in: {

$concat: [

{ $arrayElemAt: ["$pistas", "$$index"] },

" - ",

{ $arrayElemAt: ["$artistas", "$$index"] }

]

}

}

},

acousticness: "$\_id.acousticness",

danceability: "$\_id.danceability",

energy: "$\_id.energy",

mode: "$\_id.mode",

tempo: "$\_id.tempo",

valence: "$\_id.valence",

loudness: "$\_id.loudness",

spotify\_track\_explicit: "$\_id.spotify\_track\_explicit",

time\_signature: "$\_id.time\_signature"

}

},

{

$sort: { // ordena en orden descendente, primero valores más altos

"acousticness": -1,

"danceability": -1,

"energy": -1,

"mode": -1,

"tempo": -1,

"valence": -1,

"loudness": -1,

"spotify\_track\_explicit": -1,

"time\_signature": -1

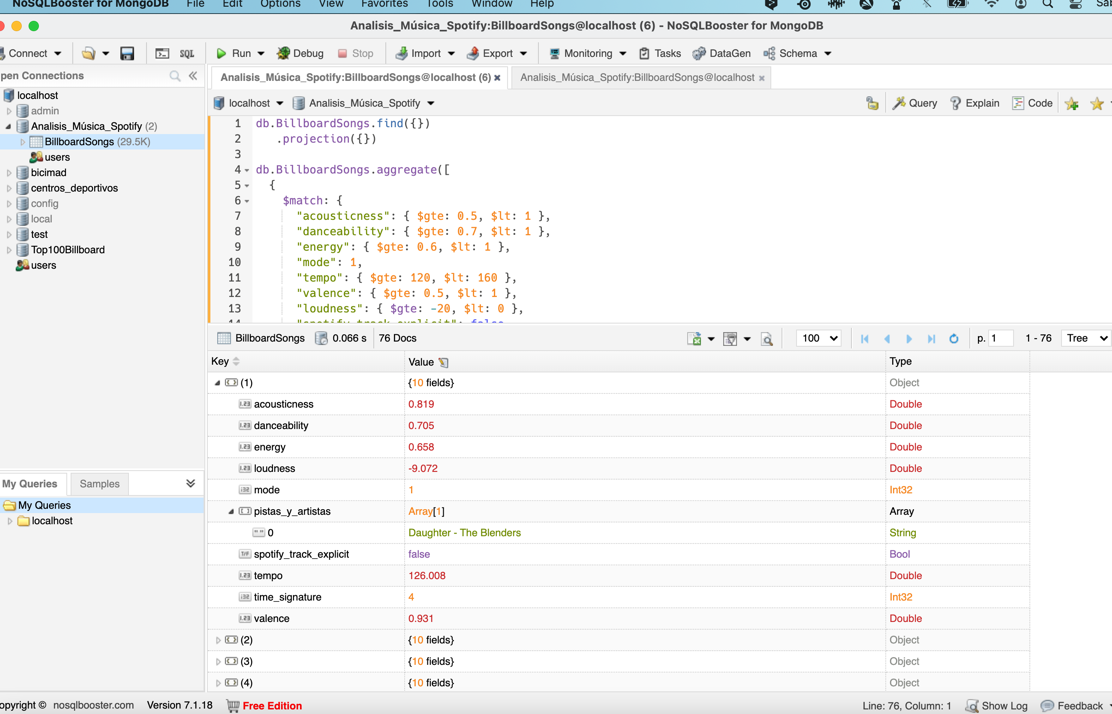
}

}

]);

print("Pistas Alegres agrupadas:", pistasAgrupadasAlegres.toArray());

**Para ver el formato de árbol (tree) directamente en la consola sin utilizar la variable (var) y el print:**

****

**Conteo total de pistas Alegres:**

db.BillboardSongs.find({})

.projection({})

var conteoTotal = db.BillboardSongs.aggregate([

{

$match: {

"acousticness": { $gte: 0.5, $lt: 1 },

"danceability": { $gte: 0.7, $lt: 1 },

"energy": { $gte: 0.6, $lt: 1 },

"mode": 1,

"tempo": { $gte: 120, $lt: 160 },

"valence": { $gte: 0.5, $lt: 1 },

"loudness": { $gte: -20, $lt: 0 },

"spotify\_track\_explicit": false,

"time\_signature": 4

}

},

{

$group: {

\_id: null,

count: { $sum: 1 }

}

}

]);

print("Conteo total de pistas:", conteoTotal.toArray()[0].count);

Conteo Total de pistas Alegres = 76 pistas.

**Imprime la lista de Pista y Artista ordenada de mayor a menor**.

var pistasAlegres = db.BillboardSongs.aggregate([

{

$match: {

"acousticness": { $gte: 0.5, $lt: 1 },

"danceability": { $gte: 0.7, $lt: 1 },

"energy": { $gte: 0.6, $lt: 1 },

"mode": 1,

"tempo": { $gte: 120, $lt: 160 },

"valence": { $gte: 0.5, $lt: 1 },

"loudness": { $gte: -20, $lt: 0 },

"spotify\_track\_explicit": false,

"time\_signature": 4

}

},

{

$project: {

\_id: 0,

pista: "$song",

artista: "$performer",

acousticness: "$acousticness",

danceability: "$danceability",

energy: "$energy",

mode: "$mode",

tempo: "$tempo",

valence: "$valence",

loudness: "$loudness",

spotify\_track\_explicit: "$spotify\_track\_explicit",

time\_signature: "$time\_signature"

}

},

{

$sort: {

"acousticness": -1,

"danceability": -1,

"energy": -1,

"mode": -1,

"tempo": -1,

"valence": -1,

"loudness": -1,

"spotify\_track\_explicit": -1,

"time\_signature": -1

}

}

]);

pistasAlegres.toArray().forEach(function(doc) {

print("Pista:", doc.pista, "Artista:", doc.artista);

});

Esta consulta me devuelve y filtra 76 canciones que pueden ser consideradas como Alegres.

Los criterios de selección para filtrar las pistas Alegres han sido la siguientes características: Acousticness, Danceability, Energy, Mode, Tempo, Valence , y en esta consulta le he sumado Loudness, lo que sugiere que las pistas no son demasiado sileniciosas como tampoco demasiado fuertes, Spotify Track Explicit, indica que no hay lenguaje explícito que pueda incitar a la ira, bronca, insultos, y además el Time Signature, que al ser en 4/4 (cuatro tiempos) es común para pistas comerciales y generalmente asociadas con alegría, a diferencia de tiempo ternario 3/4 que tienen los valses o canciones folclóricas, por ejemplo.

Estos campos agrupados y valorando una media hacia arriba, según los criterios seleccionados, se considerarían alegres y se caracterizan por tener alta energía, ser adecuadas para el baile, ser más acústicas, tener un tempo más rápido y ser positivas en términos de valencia. Se categorizarían como pistas alegres y optimistas.

**En Conclusión**:

Dado la cantidad de documentos de este dataset he decidido filtrar los campos por los valores asignados desde una media hacia abajo para las pistas tristes y una media hacia arriba para las pistas alegres, a modo subjetivo considerando valores que podrían ajustarse a esas categorías de las canciones en función de los campos detallados de cada documento-canción.

Los campos consultados, es decir, los “audio features” son propiedades que Spotify le da a las canciones, en este trabajo es tomado solo las siguiente con el fin de filtrar según mis intenciones y lo que considero que estas características puedan devolvernos las pistas más tristes o más alegres, "acousticness”, "danceability", "energy", "mode", "tempo", "valence", "loudness", "spotify\_track\_explicit", "time\_signature", representan las caracteristicas e información de un audio o pista identificada por un ID único que asigna Spotify.

Este análisis asume que las características musicales seleccionadas se correlacionan directamente con el sentimiento de tristeza o alegría, simplificando a grandes rasgos. Si bien estas características musicales y sus correspondientes emociones dan cuenta de un análisis general, de alguna manera no responde de manera subjetiva y personal. La percepción de tristeza o alegría puede variar según el oyente. También las emociones y sentimientos en la música involucran otras características que no han sido abordadas.

Este tipo de análisis podría ser útil para crear listas de reproducción automáticas basadas en el estado de ánimo del usuario o para análisis más profundos de la relación entre las características musicales y las emociones percibidas.

1. https://data.world/kcmillersean/billboard-hot-100-1958-2017 [↑](#footnote-ref-1)
2. <https://developer.spotify.com/documentation/web-api/reference/get-audio-features> [↑](#footnote-ref-2)