UNIVERSIDAD
COMPLUTENSE
MADRID

Facultad de Informática

Departamento de Ingeniería del Software e Inteligencia Artificial

Trabajo de fin de grado del Grado en Ingeniería Informática

Clasificador de subgéneros de música electrónica

Autores:
Caparrini López Antonio
Pérez Molina Laura

Directores:
Dr. Arroyo Gallardo Javier
Dr. Sánchez Hernández Jaime

16 de junio de 2017

Agradecimientos

En primer lugar, gracias a mi familia, que se interesan y valoran lo que hago mostrando un orgullo que me llena de felicidad, especialmente a mis padres, que siempre me han apoyado en las decisiones que he tomado.

Gracias a mis amigos y los momentos que me han ayudado a continuar. A los cafés con mi amiga Cris, que se han convertido en tradición. A mi amiga Blanca, que siempre escucha con empatía y devuelve las más sinceras palabras de apoyo. A mi amigo David, por esos "triples" que compartimos desde primero de carrera animando el día a día.

Gracias en especial a mi amigo Álvaro, una persona ejemplar que me aporta y critica, en la que me veo muchas veces reflejado y que siempre está presente para apoyar y contribuir. La vida sería más triste sin nuestros debates cuestionándolo todo.

Finalmente, gracias a mi compañera de trabajo, y más aún, gran amiga. Desde que nos encontramos hemos compartido todas las alegrías y desgracias, y en gran parte, nos conocimos en un momento extraño de mi vida, desde el que, todo ha ido siempre a mejor. Gracias por su paciencia, la fuerza que me ha transmitido y los momentos que hemos compartido juntos.

"There's no gene for fate."
— Vincent, *Gattaca (1997)*

Antonio Caparrini López.

Quisiera dar las gracias y dedicar el trabajo a toda mi familia, en especial a mis padres y a mi hermana. A mis padres gracias por todo el esfuerzo y el apoyo que me han dado siempre, por darme esa seguridad que a veces creía que me faltaba y con paciencia me recordaban una y otra vez que sí la tenía. Es una gran suerte tenerles y lo mínimo que quiero es que se sientan orgullosos. A Nerea que iba a estar en mis agradecimientos antes de que me lo dijera, por ser alguien fundamental en mi vida, por dejarme ser su pesada hermana mayor y por soportarme tanto en mis alegrías como en las que no lo son tanto y a mi *yaya* que ha sido mi principal motivación para aprender y un motor para conseguir aquello que me propusiera. A David, que a pesar de ser el que más "cera" me ha dado a lo largo de mi vida, siempre ha sacado un ratito para preguntarme sobre mis exámenes y a ofrecerme su ayuda en cualquier situación que la necesitara y a Nuria por ser modelo sin pretenderlo provocando que estudiase esta carrera. Gracias a mis amigos de siempre, aquellos que construyen su camino y ayudan a construir el mío. A Patri que ha estado

antes de empezar la carrera, ha sido un gran apoyo que ha sabido soportarme en ocasiones que ni yo lo hacía e incluso en los momentos más difíciles, tenía una sonrisa que sacarme. A Alberto y a Rocío por entenderme y darme esas palabras de apoyo justo cuando las necesitaba y a Inma por estar ahí a pesar de los kilómetros de distancia. Y no puedo dejar de agradecer a los compañeros de la universidad por soportar y ayudarme en los momentos más agobiantes que hemos podido vivir a lo largo de la carrera. He aprendido mucho y me llevo experiencias y momentos inolvidables con ellos. A Rafa por su inmensa paciencia y buenos consejos y a Carolina, que quizás fuese por la cantidad de horas en los laboratorios, por compartir el agobio y el estrés de los exámenes, por los momentos, las terapias, las obsesiones o las horas de biblioteca las que han provocado que incumpliese su promesa de no tener amistades en la universidad, gracias por todo amiga. Gracias a mi compañero y aquí sí que que le voy a llamar Capa, porque ha sido con quien he compartido todos los agobios, las alegrías, las decepciones y muchas horas durante este último año con este trabajo. Ha sido una suerte que al final nos hayamos conocido y eso que creía que no me soportaba.

"Desordenando la felicidad me encontré con la vida."

— AJO, *Micropoemas*

Laura Pérez Molina.

Queremos agradecer a nuestros directores, Javier y Jaime, por ayudarnos a poder plantear este proyecto. Gracias por todos los ánimos, los consejos y vuestra labor docente en general. Gracias sobretodo por vuestro tiempo en esas reuniones en las que se concretaba hora de inicio pero no la final. También queremos agradecer a todas las personas que nos han hecho el gran favor de realizar nuestro experimento de clasificación y dedicarnos dos horas sin esperar nada a cambio. Para terminar, gracias a nuestros compañeros de batalla Iván, David, Carolina, Adolfo y Ángela, con los que hemos compartido largas horas de biblioteca y grandes momentos.

"If you try and lose then it isn't your fault. But if you don't try and we lose, then it's all your fault."

— Valentine, *Ender's Game (1985)*

Antonio y Laura.

Resumen

¿Qué hace que nosotros, los humanos, seamos capaces de diferenciar canciones de distintos géneros? Quizás el lector se habrá encontrado alguna vez en la difícil situación de explicar a alguien "cómo suena" el estilo de música que le gusta. Entonces, ¿podría existir una clasificación de géneros automática?

El incremento del contenido digital disponible en diversas áreas nos obliga a buscar formas más rápidas y eficientes de almacenar y ordenar la información. Nunca había sido tan fácil hacer música y miles de canciones nuevas se publican cada día. En sitios web como *Beatport*, cada semana se publican 25.000 canciones nuevas de música electrónica. Probablemente sería de gran ayuda clasificar automáticamente todo este contenido.

En la actualidad, con los algoritmos de aprendizaje automático se buscan patrones comunes para clasificar y facilitar el acceso de datos digitalizados. Durante los últimos 20 años, se ha estudiado el reconocimiento de géneros musicales para predecir automáticamente el género de una canción. En los estudios pasados se han clasificado géneros y subgéneros en distintos estilos musicales, pero hasta donde alcanza nuestro conocimiento, nunca se ha abordado la clasificación de subgéneros de música electrónica. No obstante, existe una amplia variedad de música electrónica y a pesar de que para un oyente ocasional todo podría sonar tremendamente parecido, los fans distinguen entre subgéneros. Por lo tanto tienen que tener características que los definan.

La clasificación en géneros es subjetiva, pero partiendo de un conjunto debidamente clasificado podemos pensar que los diferentes géneros tienen algunas cualidades intrínsecas objetivables que los caracterizan. En este proyecto, nuestro objetivo es clasificar automáticamente subgéneros de música electrónica atendiendo exclusivamente a datos de audio.

Palabras clave: Clasificación de géneros musicales, aprendizaje automático, árbol de decisión, bosque aleatorio, características de audio, MIR[1], música electrónica.

[1] *Music Information Retrieval (Recuperación de la información musical)*

Abstract

What makes us, humans, able to tell apart two songs of different genres? Maybe you have ever been in the difficult situation to explain "how it sounds" the music style that you like to someone. Then, could an automatic genre classification be possible?

The increase in digital content available in several fields forces us to look for faster and more efficient ways to store and sort information. It has never been so easy to make music and thousands of new songs are released every day. There are websites, such as *Beatport*, where every week 25,000 new electronic songs are released. It may be beneficial to classify automatically these songs into genres.

Nowadays, machine learning algorithms are used for searching repeated patterns to classify and make the access to digitalized data easier. Over the last 20 years, musical genre recognition has been studied to automatically predict the genre of a song. In past studies, genres and sub-genres have been classified but as far as we are aware, this has not been done for sub-genres of electronic music. There is a wide variety of electronic music and despite sounding extremely similar for an occasional listener, fans discern between sub-genres. Thus, they probably have features that define them.

Classification into genres is subjective, but based on a properly classified set we can infer that they have some intrinsic aspects that make a difference. In this project, we aim to automatically classify sub-genres of electronic music depending on audio data.

Keywords : Music genre classification, machine learning, decision tree, random forest, audio features, MIR, electronic music.

Índice general

Índice de figuras

Listado de acrónimos

ML	*Machine Learning*
BPM	*Beats per minute*
DFT	*Discrete Fourier Transform (Transformada discreta de Fourier)*
FT	*Fourier Transform (Transformada de Fourier)*
GTZAN	*Conjunto de datos musical creado por George Tzanetakis*
MFCC	*Mel Frequency Cepstral Coefficients*
MIDI	*Musical Instrument Digital Interface*
MIR	*Music Information Retrieval (Recuperación de la información musical)*
ZCR	*Zero Crossing Rate (Tasa de cruce por cero)*

Capítulo 1

Introducción

En los últimos años, el auge de datos digitalizados pertenecientes a distintos campos, ha propiciado la creación de herramientas de análisis automático de dichos contenidos con el fin de facilitar la búsqueda y el acceso a los mismos. En este contexto, la música y las canciones no son ajenas a este fenómeno. La necesidad de organizar la música, incluso antes de la era digital, condujo a la creación de géneros musicales. Los géneros sirven para facilitar la experiencia del usuario ya que pretenden agrupar temas similares, facilitando que encuentre música de su agrado.

La clasificación en géneros se considera un proceso subjetivo que se ve afectado por diversos factores tales como la cultura, la localización geográfica, el espacio temporal o la influencia de mercado (Scaringella et al., 2006). Por esto, uno de los inconvenientes encontrados, es que no existe una forma inequívoca y única de clasificación de géneros. En el presente documento se propone el diseño de un software de clasificación de subgéneros de música electrónica.

1.1. Clasificación musical en géneros

Los géneros musicales son categorías que han surgido ante la necesidad de organizar colecciones de música y caracterizar las similitudes entre músicos y composiciones. Pese a ello, los límites entre géneros siguen siendo difusos así como su definición, haciendo que el problema de la clasificación no sea trivial y siendo aún más difícil la separación de subgéneros dentro de un mismo género.

Actualmente existe una amplia gama de clasificaciones de géneros en la música. Estas diferentes formas de clasificar se pueden observar en tiendas y sitios web relacionados con la música, por ejemplo, en sitios web mundialmente conocidos como *YouTube* (*Latina*, *Reggae*, *Electrónica...*), *last.fm* (*Electronic*, *Indie*, *Folk...*) o *Spotify* (*Latina*, *Pop*, *Trending...*).

1.1.1. Antecedentes de la clasificación musical en géneros

Según Pachet and Cazaly (2000), la industria musical siempre ha creado sus taxonomías de géneros para satisfacer sus propias necesidades. Afirman que no ha habido ningún esfuer-

zo para unificar estas taxonomías, no obstante, esta idea les resulta bastante interesante, ya que mostraría de forma clara las diferencias entre estas taxonomías. Además, presentan una división de la música de acuerdo con los minoristas, es decir, tiendas de música como *Fnac* (empresa francesa especializada en la venta de artículos electrónicos, ordenadores, artículos fotográficos, libros, música y vídeo) que plantean una división musical orientada directamente a los consumidores: un primer nivel con categorías musicales (*música clásica, jazz, rock, etc.*), un segundo nivel con *subcategorías* más específicas (*Hard Rock* dentro de *rock*), un tercer nivel mediante una ordenación alfabética por artistas y un cuarto nivel clasificado por álbumes. Los autores también destacan una división por comercialización (promociones/ventas) o temas ("rock", "mejor colección de canciones de amor"...).

Afirman que en Internet también se encuentran taxonomías destinadas a guiar al usuario a través de catálogos de música de forma similar a las tiendas de discos, pero con más nivel de detalle. Analizan varios sitios web relevantes, entre ellos, *Amazon*[1]. El análisis muestra claramente que no hay mucho consenso en estas clasificaciones. Por ello, partiendo de la base de que existen muchos datos relacionados con la música que pueden explotarse a nivel de software, afirman que debe haber algún tipo de coherencia en la clasificación automática de géneros y que ésta debe lograrse.

1.2. Reconocimiento automático del género musical

La hipótesis principal de la que se parte a la hora del reconocimiento automático de un género es la existencia de características compartidas entre las obras musicales pertenecientes al mismo. A través de estas características, que pueden extraerse automáticamente a partir del audio, se pretende generar un modelo predictivo para reconocer el género. La disciplina que trata este campo se denomina MIR por las siglas en inglés, *Music Information Retrieval* (Recuperación de información musical). La definición de MIR[2] es:

> *La recuperación de información musical (MIR) es la ciencia interdisciplinar encargada de recuperar información de la música. MIR es un pequeño pero creciente campo de investigación con muchas aplicaciones en el mundo real. Aquellos implicados en MIR pueden ser especialistas en musicología, psicología, estudio de música académica, procesamiento de señal, aprendizaje de máquina o alguna combinación de estos.*

Uno de los enfoques principales de la recuperación de información musical es a través de las características del audio (Tzanetakis and Cook, 2000a). La recuperación se realiza

[1] https://www.amazon.es/Musica-Digital/b?ie=UTF8&node=1748200031

[2] Music information retrieval. (s.f.). En Wikipedia. Recuperado el 19 de mayo de 2017 de https://en.wikipedia.org/wiki/Music_information_retrieval

mediante el procesamiento de la señal del mismo, extrayendo características relacionadas con el timbre, el contenido rítmico y el tono (Tzanetakis and Cook, 2002). A continuación, detallaremos las características musicales.

1.2.1. Características musicales

A continuación, mostramos una breve descripción de los tipos de características más relevantes.

Características rítmicas

La mayoría de los autores se refieren al ritmo como una medición de la regularidad temporal, es decir, una forma de sucederse y alternar una serie de sonidos que se repiten periódicamente en un determinado intervalo de tiempo.

Características del tono: Melodía y armonía

El tono está directamente relacionado con la frecuencia del sonido, siendo grave para frecuencias bajas y agudo para altas. Este concepto está ligado a los conceptos de melodía y armonía. La melodía puede definirse como una sucesión de sonidos que es percibida como una sola entidad y la armonía como el estudio del uso de la simultaneidad del tono y de los acordes que viene implícito o no en la música[3]. En consecuencia, el análisis armónico y melódico ha sido utilizado para estudiar las estructuras musicales.

Características del timbre

El timbre es la característica de la música que hace que dos sonidos con el mismo tono y volumen suenen diferente. Las funciones que caracterizan al timbre son globales, es decir, integran la información de todos los instrumentos al mismo tiempo junto con la voz. De una forma más intuitiva, el timbre sería el equivalente en el audio a la textura en el color.

Estas características son cualidades de la música que las personas percibimos directamente y con las que los expertos en música describen las canciones. Los ordenadores no trabajan con ellas del mismo modo, sino que utilizan propiedades que se extraen de la señal digital de ficheros de audio como por ejemplo ".wav". Las propiedades, aunque están relacionadas con las características musicales, no tienen una correspondencia perfecta con ellas. Además aunque las extraigamos, si lo que pretendemos es realizar una clasificación

[3]Melody. (s.f.). En Wikipedia. Recuperado el 20 de abril de 2017 de https://en.wikipedia.org/wiki/Melody Harmony. (s.f.). En Wikipedia. Recuperado el 20 de abril de 2017 de https://en.wikipedia.org/wiki/Harmony

automática, necesitamos que un software o algoritmo realice el proceso hecho por las personas. En este punto, introducimos el aprendizaje automático que cumplirá la función de buscar patrones y similitudes entre estas características para realizar la clasificación de una forma automatizada.

1.2.2. Aprendizaje automático

El aprendizaje automático es una rama de la inteligencia artificial que tiene como objetivo crear algoritmos que permitan a las máquinas aprender. Expresado de forma más concreta, estos algoritmos son capaces de generalizar comportamientos y reconocer patrones a partir de una información suministrada inicialmente en forma de ejemplo. Esto es por tanto un proceso de inducción, donde a partir de los casos particulares aportados, se obtiene una generalización.

Muchos autores han utilizado estas técnicas para procesos de clasificación, inducción o aprendizaje. Según Nilsson (1996), existen varias razones para utilizar el aprendizaje automático. Entre ellas, se encuentra que existen bases de datos muy grandes que pueden ocultar algún tipo de relación entre sus variables y que sería interesante conocer.

Los algoritmos de aprendizaje automático supervisados parten de conjuntos debidamente clasificados y buscan en ellos características afines o patrones de similitud, es decir, las clases están asignadas en el conjunto de datos. Este es el tipo de clasificación que nos interesa, ya que buscamos reproducir la clasificación musical hecha por personas.

La utilización de técnicas de aprendizaje automático aplicadas a la música (MIR) constituyen un área de interés creciente en los últimos años y existen comunidades científicas dedicadas a ello.

Comunidad MIR (Music Information Retrieval)

Entre las aplicaciones del MIR se encuentra la categorización de música según su género. En este contexto, cabe mencionar MIREX (Music Information Retrieval Evaluation eXchange). MIREX es una organización que está intentando unificar esfuerzos y publica cada año varios conjuntos de datos musicales para su clasificación musical. De esta forma, se expone una comparación de algoritmos cada año donde se exploran las diversas técnicas de aprendizaje automático presentadas por los miembros pertenecientes a la misma.

En la misma línea se encuentra ISMIR (International Society for Music Information Retrieval). ISMIR es una organización sin ánimo de lucro que organiza la Conferencia ISMIR. Se lleva a cabo anualmente y es uno de los foros de investigación más importantes del mundo en procesamiento, búsqueda y acceso a los datos relacionados con la música[4].

[4]La conferencia ISMIR 2016 se celebró el 11 de agosto del año pasado en Nueva York contando con la

1.3. Motivación de nuestro trabajo

La clasificación de géneros musicales es subjetiva, ya que como ha quedado expuesto, depende de aspectos culturales, temporales y personales. Sin embargo, si se determina un conjunto que ha sido clasificado previamente por usuarios que son auténticos aficionados a ese tipo de música, ¿sería descabellado pensar que existe algo realmente intrínseco a esas canciones que las caracteriza?

La respuesta en un primer momento puede parecer más que evidente, incluso caer en el pensamiento: "*sin tener ningún tipo de conocimiento musical, hay algo que distingue la música clásica de la música rock. Parece evidente*". Hay estudios actuales que han querido sacar conclusiones al respecto (véase sección 2) y que resaltan que es muy importante partir de un buen conjunto de datos debidamente etiquetado por profesionales. A través del procesamiento de música se pretende automatizar la clasificación de géneros y así precisar más las fronteras entre ellos. Siendo esta división entre géneros, en muchas ocasiones, algo polémico o complejo, ¿qué sucede con los subgéneros?

En nuestro trabajo abordaremos precisamente la clasificación entre subgéneros dentro de un género en constante evolución en los últimos años gracias a la digitalización de la música. Presentamos la distinción entre subgéneros de la música electrónica mediante la implementación de un clasificador automático.

Nuestro principal referente será ***Beatport***[5], una tienda *online* de música electrónica que tiene todas sus canciones clasificadas en subgéneros de manera que cada canción, únicamente está etiquetada en un sólo género.

En la siguiente entrevista a representantes de *Beatport* en agosto del 2016[6] se habla de su forma de clasificar géneros. Reconocen que los géneros son subjetivos y su interés no es dar una clasificación impuesta por ellos, sino una que facilite a sus clientes el acceso a la música. En ella, hablan de dos nuevos géneros que añadieron por petición popular. Además, argumentan que si el público llama de determinada forma a un conjunto de canciones es más fácil para ellos encontrar música nueva en la tienda si está clasificada de esa manera.

> *DJTT: ¿Qué está ocurriendo con el nuevo enfoque de géneros musicales en Beatport?*
>
> *Beatport: Esto es algo que el público ha estado pidiendo y nosotros simplemente les hemos escuchado. Pero lo que hace falta decir es que no hay una definición correcta de género. Solo queremos que nuestros clientes encuentren buena música con facilidad.*

organización de la universidad de Nueva York junto con la universidad de Columbia (universidad privada estadounidense ubicada en Alto Manhattan, Nueva York). MIREX 2016 formaba parte de esta conferencia.

[5] https://beatport.com

[6] http://djtechtools.com/2016/08/12/beatports-re-approach-to-genre-tagging/

> *DJTT: ¿Llegará un día en que los géneros resulten innecesarios, como en el caso del Glitch-hop?*
>
> *Beatport: "Creo que los DJs siempre necesitarán géneros como forma de encontrar música nueva. Tenemos 25.000 nuevas publicaciones la mayoría de las semanas y los géneros ayudan a acotar la búsqueda de los temas. Nosotros tenemos que permanecer relevantes en términos de lo que la gente está haciendo sonar. También tendremos que quitar géneros y crear otros. Lo principal es seguir a la cabeza de las tendencias, y trabajar con los artistas y los sellos discográficos en esa dirección."*

La evolución de la tecnología ha facilitado la creación de música electrónica y como consecuencia se produce más cantidad y más variada. Por este motivo, una clasificación automática de la música nueva que aparece cada semana sería una propuesta interesante y una forma de ahorrar esfuerzos.

1.4. Objetivos del trabajo

En el presente trabajo se pretende clasificar subgéneros de música electrónica expuestos en la página de *Beatport*. Además, se pretende comprobar si realmente hay cualidades intrínsecas en las canciones que las hacen pertenecer a dichos subgéneros. *Beatport* clasifica con arreglo a las preferencias de sus usuarios, las cuales intenta objetivar. El clasificador trabajará con propiedades de la señal de audio que están relacionadas con las propiedades musicales pero que no son las mismas, por lo que es interesante comprobar hasta qué punto la clasificación automática y humana alcanzan resultados comparables. Para ello, los objetivos concretos que planteamos serán:

- Implementar un clasificador de subgéneros de música electrónica.
- Comprobar la validez del clasificador.
- Estudiar las posibles confusiones entre subgéneros. Analizar si son debidas a la aproximación seguida (no inclusión de variables más descriptivas, sesgo del algoritmo) o si, por el contrario, se deben a que son géneros difícilmente "objetivables".
- Comparar la clasificación automática con la realizada por personas para comprobar si la clasificación humana y automática resultan similares o no.

1.5. Estructura de la memoria

A continuación se pasa a detallar la disposición de los contenidos de este trabajo:

1. **Introducción**

   Se contextualiza el contenido del trabajo en general y la motivación para el desarrollo de un clasificador automático de subgéneros de la música electrónica. Además se ofrece un apartado de los objetivos que se desean cumplir dentro del trabajo.

2. **Estado del arte**

   En este capítulo se muestra el estado actual de la clasificación automática de géneros, desde el punto de vista de la extracción de características del audio, el aprendizaje automático, los conjuntos de datos y el estado actual de este sector.

3. **Fundamentos conceptuales del trabajo**

   Se ofrece una visión más descriptiva de los conceptos aplicados en el proyecto. En este apartado se detalla el conjunto de datos, las características utilizadas (definición y proceso de extracción), algoritmos de aprendizaje automático o el software utilizado.

4. **Desarrollo**

   En este apartado se exponen los aspectos más técnicos del trabajo. Se presentan detalles como los géneros escogidos, la duración de cada pista de audio, el tamaño de la ventana elegida para extraer las muestras de audio, la composición del vector de características, el criterio de validación de los algoritmos escogidos entre otros.

5. **Resultados**

   Se muestran los resultados generados de las pruebas realizadas más significativas, mostrando un análisis de los mismos.

6. **Conclusiones y trabajo futuro**

   Se exponen las líneas abiertas y propuestas del trabajo por continuar y un resumen de las conclusiones finales de este trabajo de fin de grado.

7. **Apéndices**

   En esta sección se presentan: **Contribuciones al proyecto**, según la normativa de TFG de curso 2016-2017 indicamos en este apartado de la memoria la contribución de cada uno al proyecto. **Introduction** y **Conclusions and future work** e **Información adicional del trabajo** con detalles adicionales del trabajo como la prueba del clasificador con el conjunto GTZAN[7] y una descripción de los subgéneros de nuestro trabajo.

[7] *Conjunto de datos musical creado por George Tzanetakis*

Capítulo 2

Estado del Arte

"There is nothing new under the sun. It has all been done before."
— Sherlock Holmes, *A Study in Scarlet (1887)*

Según Guaus (2009), uno de los primeros enfoques en la clasificación automática de audio fue propuesto por Wold et al. (1996). El autor propone la clasificación de distintas familias de sonidos tales como animales, instrumentos de música, voz y máquinas. En este estudio, se extraen características del audio y se calculan medidas estadísticas como la mediana o la varianza entre otras, utilizando un clasificador *gaussiano* (también denominado clasificador de Bayes[1]). Un estudio relevante para la clasificación automática de géneros, que ha servido de punto de partida de muchos otros desde el punto de vista de los parámetros de audio extraídos para analizar, fue el de George et al. (2001) dejando paso a un estudio más sistemático un año después (Tzanetakis and Cook, 2002). En este último, calculan tipos de características como el timbre, el ritmo y el tono. En este trabajo utilizan clasificadores *gaussianos* con los siguientes géneros: *clásica, country, disco, hip-hop, jazz, rock, blues, pop y metal.*

2.1. Reconocimiento del género musical

Si nos centramos en las técnicas necesarias para el reconocimiento del género musical, hay dos enfoques principales: la recuperación musical a partir del audio y la recuperación de la información musical a partir de representaciones simbólicas (Ponce de León Amador, 2011).

- En las representaciones simbólicas se utilizan conceptos explícitos, es decir, se usa información musical que se encuentra directamente escrita, como las notas con altura, duración, intensidad... Los formatos digitales que contienen este tipo de información

[1] Un clasificador de Bayes es un clasificador probabilístico fundamentado en el teorema de Bayes donde se asume que la presencia o ausencia de una característica particular no está relacionada con la presencia o ausencia de cualquier otra. Naive Bayes classifier. (s.f.). En Wikipedia. Recuperado el 19 de mayo de 2017 de https://en.wikipedia.org/wiki/Naive_Bayes_classifier

pueden ser archivos de texto, como *MusicXML* que es un formato que facilita el uso en Internet para representar partituras y notación musical (Good, 2001), o archivos binarios como los ficheros MIDI[2] (Selfridge-Field, 1997) entre otros.

- En la recuperación de la información musical (*audio information retrieval*) utilizado por muchos autores (Tzanetakis and Cook, 2000a), tratan la señal digital de audio, extrayendo información a partir de ella, sin que ésta se encuentre representada explícitamente. Este enfoque, utiliza la señal de audio directamente y extrae características musicales propias del audio (ritmo, tono...).

En este trabajo tratamos este segundo enfoque cuyo proceso queda descrito en la sección 3.2. Para abordar el problema del reconocimiento del género musical hay dos partes, la extracción de la información musical y la descripción estadística global (Ponce de León Amador, 2011). Para la extracción de la información musical se usa el concepto de ventana. Esta ventana, se mueve a lo largo del audio, tomando muestras de una determinada longitud y a intervalos concretos (más información sobre este proceso en las secciones 3.2.1 y 4.2). A partir de este contenido, se calcula un conjunto de características estadísticas que ofrecen una descripción global de la pieza. Las características extraídas en este estudio son las planteadas por Giannakopoulos (2015) y Bogdanov et al. (2013) que se basan en estudios anteriores como el de Tzanetakis and Cook (2002) entre otros. A partir de este contexto, esta información debe ser tratada para la clasificación.

2.2. Clasificación de géneros musicales mediante aprendizaje automático

La hipótesis principal de este trabajo es que la música de un mismo género debe compartir rasgos comunes que una vez sean identificados, permitan clasificar automáticamente nuevas canciones según su género. Este problema ha sido abordado a través del aprendizaje automático, principalmente mediante la aplicación de técnicas de aprendizaje supervisado (más detalles en la sección 3.3). En los últimos años, la clasificación de géneros musicales desde la perspectiva del aprendizaje automático, ha suscitado un gran interés. Sin entrar en detalles específicos de los algoritmos, muchos autores proponen mezclas gaussianas (Aucouturier and Pachet, 2002; E. Pampalk and Widmer, 2005), redes neuronales (Sigtia and Dixon, 2014), máquinas de soporte vectorial (Scaringella and Zoia, 2005) o árboles decisión y bosques aleatorios (Ponce de León Amador, 2011; Creme et al., 2016).

En este trabajo se implementan árboles de decisión y bosques aleatorios (Breiman et al., 1984; Breiman, 2001) que quedarán detallados en las secciones 3.3.1 y 3.3.2.

[2] *Musical Instrument Digital Interface*

2.3. Conjuntos de datos

En general, existe una variedad amplia en conjuntos de datos para su clasificación en géneros. Como se argumenta en la introducción, todos los años se presentan conjuntos de datos distintos[3], por ejemplo, el año pasado presentaron un conjunto de música con 1894 canciones etiquetadas en 7 géneros: *Ballad, Dance, Folk, Hip-hop, R&B, Rock* y *Trot*. El audio estaba en formato *.wav* mono 22,05 *kHz*, con muestras de 30 segundos de duración. El procedimiento a seguir era la extracción de características del audio y el uso de un algoritmo de aprendizaje automático. Para la evaluación de los algoritmos de aprendizaje presentados, se requería una validación cruzada y una matriz de confusión final entre otras[4].

Por otra parte, Tzanetakis and Cook han creado el conjunto de datos *GTZAN*[5] que hasta la fecha se considera un estándar para la clasificación de géneros. Las características innovadoras, los nuevos algoritmos de clasificación o las diferentes estrategias publicadas suelen probarse en él por ser una referencia en la clasificación automática de géneros. En este trabajo, utilizamos un conjunto de datos que no ha sido analizado hasta el momento según nuestro conocimiento. Por esto, para probar la metodología y verificar que el clasificador calcula buenos resultados, lo hemos utilizado con este conjunto de datos siguiendo todo el proceso descrito en la sección 3.

Aunque el *GTZAN* es ampliamente utilizado, su validez como conjunto de prueba ha sido puesta en duda en varios trabajos (Sturm, 2012, 2013). El *GTZAN* data del 2002, por lo que hay que tener en cuenta que en ese momento la música no estaba tan digitalizada. Además, dentro del conjunto, encontramos diferentes tipos de grabaciones que provienen de distintos formatos y en distintas calidades (grabaciones de CD, grabaciones de la radio, *radio-cassetes*, ...) añadiendo un nivel de complejidad mayor a la clasificación. *GTZAN* consta de 1000 pistas de audio de 30 segundos de duración cada una. Las pistas son archivos de audio de 16 bits de 22050 Hz mono en formato *.wav*. Contiene 10 géneros, cada uno representado por 100 pistas. No obstante como muestra la tabla 2.1, se utiliza hoy día y una vez desarrollada nuestra aproximación para realizar la clasificación automática, decidimos probarla con este conjunto de datos para estimar la fiabilidad del método. Más información sobre esta prueba en el apéndice D.

[3] Los conjuntos presentados cada año, se pueden encontrar en http://www.music-ir.org/mirex/wiki/MIREX_HOME.

[4] Más información sobre este conjunto de datos en http://www.music-ir.org/mirex/wiki/2016:Audio_K-POP_Genre_Classification.

[5] *GTZAN* se encuentra disponible en http://marsyasweb.appspot.com/download/data_sets/

2.4. Herramientas software

Hemos utilizado *Python* como lenguaje anfitrión para nuestra aplicación. Este lenguaje es muy fácil de utilizar y tiene librerías de código libre que permiten un "prototipado" rápido. Además es un lenguaje muy utilizado y hay multitud de herramientas, tanto de análisis de audio como de aprendizaje automático implementados con este lenguaje de programación. Por otro lado, hay una comunidad muy amplia de gente utilizándolo, de manera que ante cualquier error en el desarrollo, es fácil encontrar una buena solución. Las librerías que hemos utilizado en este trabajo han sido:

- **Extracción de características**
  - **PyAudioAnalysis**

    PyAudioAnalysis es una biblioteca de código abierto para *Python*. Proporciona muchas funcionalidades como la extracción de características, clasificación, segmentación y visualización del audio.
  - **Essentia**

    Essentia es una biblioteca de C++ de código abierto desarrollada para el análisis y la recuperación de la información musical a partir del audio. También está disponible para *Python*, incluyendo un número predefinido de extractores del audio que facilitan el prototipado rápido.
  - **Librosa**

    Es un paquete de *Python* de código abierto destinado al análisis de música y audio. Proporciona los componentes básicos necesarios para crear sistemas de recuperación de información musical.
- **Aprendizaje automático**
  - **Pandas**

    Pandas es una librería de código abierto, que proporciona estructuras de datos y herramientas de análisis de éstos para *Python*.
  - **Scikit-learn**

    Scikit-learn de código abierto, ofrece herramientas para la minería y el análisis de datos. Contiene implementaciones de algoritmos de aprendizaje automático y herramientas para facilitar su uso y entrenamiento.
  - **DEAP**

    Facilita el "prototipado" rápido de optimizaciones con algoritmos genéticos. Es libre, bien documentado y está implementado en Python.
- **Representación de resultados**

- **Matplotlib**
  Matplotlib es una biblioteca de *Python* que genera gráficos en 2D. Usado para la representación gráfica de resultados.

Para representar los resultados a través de los grafos de confusión, hemos utilizado **Gephi** que es un software libre para la exploración y la manipulación de redes. En este proyecto se utiliza para la creación de grafos de decisión.

El entorno de programación que hemos utilizado ha sido **PyCharm**, un IDE o entorno de desarrollo integrado multiplataforma utilizado para desarrollar en *Python*.

2.5. Estado actual sobre la clasificación musical en géneros

Hasta donde nuestro conocimiento alcanza, la mayoría de la clasificación automática de la música ha sido mediante géneros. Por ejemplo, en el *GTZAN* o en los conjuntos propuestos por la comunidad MIR, siendo menos habitual la clasificación dentro de un mismo género. Como se puede observar en la figura 2.1, mostramos varios ejemplos donde el conjunto *GTZAN* ha sido usado. Existen varios estudios sobre el estudio de los subgéneros:

Referencia	**Tasa de aciertos (*accuracy*)**
Lim et al. (2012)	87.90 %
Baniya and Lee (2016)	87.40 %
de Sousa et al. (2016)	79.70 %
Li et al. (2003)	78.50 %
Panagakis et al. (2008)	78.20 %
Lidy et al. (2007)	76.80 %
Benetos and Kotropoulos (2008)	75.00 %
Holzapfel and Stylianou (2008)	74.00 %
Tzanetakis and Cook (2002)	61.00 %

Cuadro 2.1: Tabla comparativa de tasas de aciertos con GTZAN.

una clasificación de la música *folk*, que sobre estas líneas se podrían considerar una división entre subgéneros de este tipo de música ya que distingue según el país (Alemania, Austria e Irlanda) al que pertenece (Chai, 2001), una clasificación de subgéneros de música latina por Lopes (2010) y una clasificación en subgéneros de *Heavy Metal* por Tsatsishvili (2011), pero ninguno que trate la división de los que se encuentran dentro de la música electrónica (hasta donde nosotros sabemos). Por ello, esta clasificación en subgéneros de la música electrónica es nuestra contribución en este trabajo.

Capítulo 3

Fundamentos conceptuales del trabajo

En este capítulo, se muestra el material utilizado así como una explicación descriptiva de los conceptos clave que se han tenido cuenta en el trabajo.

Para la consecución de los objetivos mencionados en la introducción, es necesario el planteamiento de un modelo predictivo para la clasificación automática de géneros de música electrónica. El trabajo debe cumplir las siguientes fases:

- Elección del conjunto de datos.
- Elección de las características del audio.
- Elección de los algoritmos de aprendizaje automático.
- Implementación de la extracción de características y del clasificador.
- Validación de los resultados.

3.1. Conjunto de datos

Nuestra referencia es la clasificación que ofrece ***Beatport***. Cada canción es etiquetada en un único género, lo cual nos da un conjunto debidamente clasificado. Además, este conjunto está etiquetado por profesionales del mundo de la música, artistas, sellos discográficos e incluso, el público atraído por este tipo de música. Esto se debe a que *Beatport* está muy interesado en que la música esté debidamente etiquetada para facilitar el acceso y la búsqueda de canciones a sus usuarios, por lo que este sitio es una buena referencia.

Cada semana, la colección de música de *Beatport* se actualiza con más de 25000 temas exclusivos de los mejores artistas de música electrónica del mundo, considerando las ventas, la popularidad, las novedades o el número de escuchas de las canciones. Por ello, vamos a considerar que las canciones pertenecientes al *top100* son una buena muestra para cada género. Esta lista de las 100 mejores cambia con el paso del tiempo debido a las nuevas canciones que aparecen o fluctuaciones de popularidad. Por este motivo, cabe destacar que el conjunto de datos de este trabajo sigue la clasificación propuesta en *Beatport* el día **23 de noviembre de 2016**.

- El conjunto de datos consta de fragmentos de 120 segundos de las 100 mejores canciones de los 23 géneros de *Beatport* publicados el 23 de noviembre de 2016 (2300 canciones en total). Más información descriptiva sobre estos 23 géneros en el apéndice E y a través de la figura 4.1 que ofrece enlaces a las listas de *Beatport* de cada género.

- Estos 120 segundos son cogidos de forma aleatoria de la canción original, puede coincidir con el principio, el final o partes intermedias de la canción.

- El hecho de que sólo cojamos 120 segundos de la canción no es por ninguna razón técnica, simplemente decidimos coger la duración total de la muestra que ofrece *Beatport*.

- Las muestras son archivos *.wav*, con frecuencia de muestreo 22050 y mono, ya que es el estándar utilizado a la hora de realizar extracción de características.

3.2. Características del audio

Una vez que disponemos de muestras de audio para analizar, es necesario extraer sus características. Estas características se extraen a partir de la señal digital del audio y no tienen una correlación directas con las cualidades que perciben los seres humanos, por lo que su explicación en ocasiones, es difícil de argumentar incluso para los expertos en la materia.

3.2.1. Proceso de extracción de características

Para la clasificación de una señal de audio, es necesaria la extracción de sus características. Una forma de conseguirlo (y la elegida por nosotros), es a través de muestras pequeñas que se consiguen por la división de la señal en ventanas. El tamaño de estas ventanas suele ser diferente en cada estudio. Algunos eligen ventanas de 10 a 40 *milisegundos* (Subramanian et al., 2004), otros entre 10 y 60 *milisegundos* (Peeters, 2004), en función de las que ofrecen mejores resultados. En nuestro caso, utilizamos ventanas de 50 *milisegundos* ya que nos han dado mejores resultados.

Una vez que la señal ha sido dividida, se calcula el valor de cada característica en cada ventana. Para agregar esta información se calcula la media y la desviación típica estándar de las características calculadas para las ventanas (más adelante se cuenta este proceso en detalle en la sección 4.2). En la siguiente sección, se muestra una lista detallada de las características que se calculan.

3.2.2. Definición de las características del audio

Una pieza musical se caracteriza principalmente por el timbre, el tono y el ritmo. Para calcular estas propiedades, buscamos distintas métricas que caractericen distintos aspectos de la señal de audio. En la siguiente sección hablaremos del significado de estas características.

Esto ha sido planteado por varios autores entre ellos, Tzanetakis and Cook (2002) definen:

- **ZCR - Zero Crossing Rate (Tasa de cruce por cero)**

  Característica definida por Kedem (1986) y Saunders (1996) entre otros, el ZCR mide la velocidad con la que la señal cambia de positivo a negativo. Es decir, cuenta las veces en la que la señal pasa por amplitud igual a cero como se muestra en la figura 3.1.

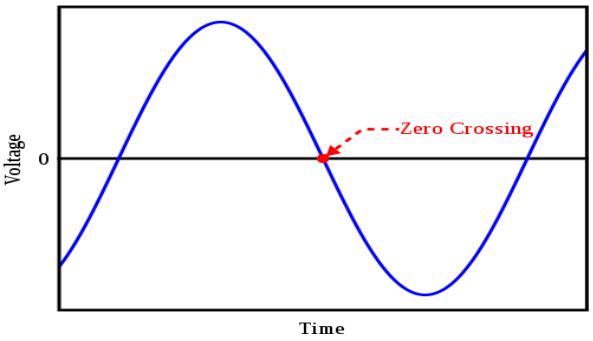


Figura 3.1: Zero Crossing Rate

En general, los valores mayores de ZCR indican sonidos más agudos, y más bajos, indican frecuencias más graves. Uno de sus usos es diferenciar sonidos de percusión. Entiéndase como sonidos de percusión a los sonidos producidos por el bombo (graves), la caja (medios) y platillos (altos).

Este parámetro se define como:

$$ZCR = 1/2 \cdot \sum_{n=1}^{N} |sign(x[n]) - sign(x[n-1])|$$

donde la función *sign* es 1 para los argumentos positivos y 0 para los argumentos negativos, $x[n]$ es el dominio de tiempo de entrada de datos y N es la longitud de $x[n]$ (en ventanas).

- **Spectral Centroid (Centroide espectral)**

  Representa el centro de gravedad del espectrograma, concretamente, el punto de equilibrio de la distribución espectral (Scheirer, 1998). Es el valor de la frecuencia a partir de la cual, la media de las energías de las frecuencias mayores y menores son iguales.

  Desde un punto de vista matemático, el *Spectral Centroid* se calcula como:

  $$C_t = \frac{\sum_{n=1}^{N} M_t[n] \cdot n}{\sum_{n=1}^{N} M_t[n]}$$

  donde $M_t[n]$ es la magnitud de la transformada de Fourier en el fragmento t y frecuencia n.

- **Spectral Rolloff (Atenuación espectral)**

  Es la frecuencia R_t por debajo de la cual el 85-90 % del espectro está concentrado. Es un término ampliamente utilizado por diversos autores que fijan el umbral en diferentes porcentajes, pero en este trabajo se tiene en cuenta el propuesto por Tzanetakis and Cook que lo fijan en el 85 %. Al igual que el ZCR, es una medida de la forma del espectro. Se utiliza para determinar diferencias en el timbre.

  $$\sum_{n=1}^{R_t} M_t[n] = 0{,}85 \cdot \sum_{n=1}^{N} M_t[n]$$

- **Spectral Flux (Flujo espectral)**

  El *Spectral Flux* es el cuadrado de la diferencia entre los valores normalizados de la magnitud del espectro de dos ventanas sucesivas. Esta característica mide la rapidez con la que cambia la potencia del espectro (cuánta energía está concentrada alrededor de una ventana de tiempo) de una señal. Entre sus aplicaciones, se utiliza para determinar el timbre de la señal de audio. El *Spectral Flux* se define como:

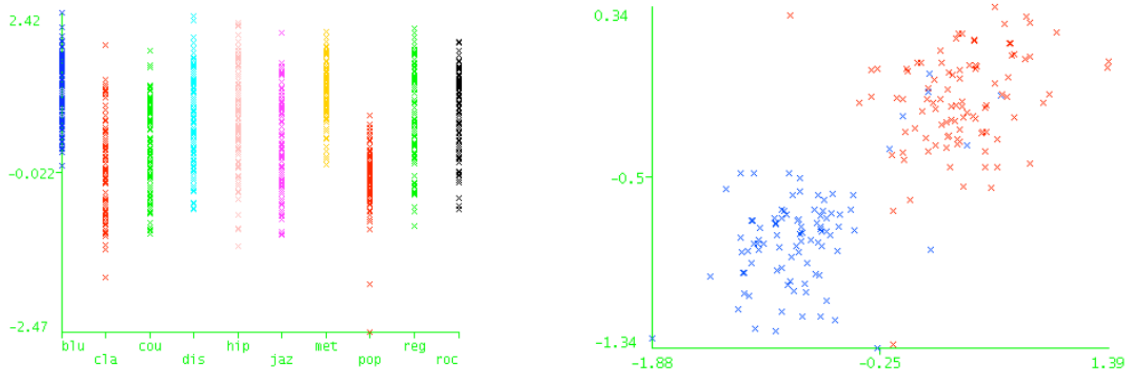
  $$F_t = \sum_{n=1}^{N} (N_t[n] - N_{t-1}[n])^2$$

  Considerando $N_t[n]$ y $N_{t-1}[n]$ como las transformadas de Fourier normalizadas en el fragmento t y $t-1$ respectivamente. Es una medida de la cantidad de cambio local espectral.

- **MFCC**[1] **(Coeficientes cepstrales de Mel)**

[1] *Mel Frequency Cepstral Coefficients*

Estos coeficientes se utilizan para concentrar los datos del espectro y reducirlos a características más cercanas a lo que el oído humano es capaz de percibir. Particularmente, en el análisis de género de una canción se utiliza para la identificación de contenido relevante y es una forma de aportar a la máquina las frecuencias más importantes para un ser humano. El primer coeficiente, que no se almacena es proporcional a la energía, los siguientes 12 coeficientes se almacenan por cada ventana analizada. En la figura 3.2 (Guaus, 2009) se muestran dos de los coeficientes aquí mencionados. En la parte izquierda, se muestra el comportamiento del MFCC5 para los géneros musicales de GTZAN (*blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock*). A la derecha, se muestra en el eje X el coeficiente MFCC6 y en el eje Y el coeficiente MFCC10 para valores de *Pop* en color rojo y *Metal*, en color azul.



(a) Coeficiente MFCC5 para diferentes géneros musicales.

(b) Coeficiente MFCC6 en el eje X y coeficiente MCC10 en el eje Y para valores dados por *Metal* (azul) y *Pop* (rojo).

Figura 3.2: Valores MFCC.

El proceso documentado por Aizawa et al. (2004) que se sigue para extraer estos parámetros es:

1. Separar la señal ventanas.
2. A cada tramo aplicarle la Transformada de Fourier discreta y obtener la potencia espectral de la señal.
3. Aplicar el banco de filtros correspondientes a la Escala Mel al espectro obtenido en el paso anterior y sumar las energías en cada uno de ellos.
4. Tomar el logaritmo de todas las energías de cada frecuencia Mel.
5. Aplicarle la transformada de coseno discreta a estos logaritmos.

- **Energy (Energía)**

  La energía se define como la suma de los cuadrados de los valores de la señal normalizados al tamaño de la ventana que estamos analizando. Este valor identifica secciones dentro de la señal de audio, que tienen mayor o menor amplitud y viene definido por:

$$E = \sum_{n=0}^{N} x[n]^2$$

  donde x[n] es el dominio de tiempo de entrada de datos y N es la longitud de x[n] (en muestras).

Otras características del audio dadas por *PyAudioAnalysis* (Giannakopoulos, 2015) son:

- **Entropy of Energy (Entropía de la energía)**

  Es la entropía de las energías de las ventanas que analizamos. Entendemos entropía como una medida de cambios abruptos. Esta medida es un concepto derivado de la física, al igual que *Spectral Entropy* (que veremos más adelante) donde se utiliza como información complementaria a la energía y al espectro generado respectivamente como medida de desorden.

- **Spectral Spread (Propagación espectral)**

  Es una medida del ancho de banda del espectro de la señal. Indica cómo está dispersa la señal en la frecuencia y complementa a la medida anterior.

- **Spectral Entropy (Entropía del espectro)**

  Entropía del espectro de energías normalizadas para un conjunto de ventanas. Nos sirve para detectar cuándo se produce un cambio en el espectro de energía de la señal. Se utiliza generalmente, en el reconocimiento de voz, en nuestro caso, lo tratamos como información complementaria a *Spectral Spread*.

- **Chroma Vector (Vector de croma)**

  Las características del croma consisten en la representación de un vector de 12 elementos de la energía del espectro donde cada elemento representa la intensidad asociada a cada uno de los 12 tonos de la escala cromática (en música occidental), es decir, asociada a un espacio de un semitono, independientemente de la octava.

  En el estudio de Ellis (2007) queda reflejado que las características obtenidas del croma, son menos informativas para discriminar clases por sí solas, pero que en combinación con las características espectrales se puede encontrar datos relevantes para la clasificación musical.

- **Chroma Deviation (Desviación del croma)**

  Es la desviación estándar de los 12 coeficientes del croma mencionados en el apartado anterior.

- **Beats per minute (Pulsaciones por minuto)**

  Es una unidad de medida del tiempo utilizada en música que indica los pulsos que hay en un minuto. Música más rápida tendrá valores mayores, mientras que música más lenta los tendrá menores. Es la única característica de largo término (entiéndase como característica de largo término que se tiene en cuenta toda la muestra). Se calcula utilizando las ventanas pero de una forma diferente, buscando golpes o "ticks". Para las descripciones de los algoritmos en concreto más en información en Giannakopoulos (2015) y en Bogdanov et al. (2013).

3.2.3. Significado de las características extraídas

Las características de audio extraídas (sin contar las de largo término como el BPM[2]) no tienen una correlación directa con las capacidades cognitivas de los humanos al clasificar. En multitud de ocasiones durante la realización del trabajo nos han preguntado sobre la definición exacta de alguna característica en concreto y qué aspecto de la música refleja. Sin embargo, una respuesta clara y concisa a esta pregunta es difícil ya que por ejemplo, en el caso del *Spectral Spread*, la definición determina que es una visión del espectro de audio que ayuda a diferenciar timbres. No obstante, esta definición puede parecer una respuesta vaga.

Para aclarar este aspecto, podemos considerar las palabras de Tzanetakis and Cook en una clase *online*[3] donde presenta la mayoría de las características que mencionamos anteriormente en la sección 3.2.2. Cabe destacar una frase cerca del minuto 16:00 del vídeo en el que el autor cita textualmente:

> *"Unlike measured pitch, audio features do not necessarily have a direct perceptual correlated. In other words, we would be computing these numbers over audio, but if you asked me, what exactly does this number mean? I can give you a handwaving explanation but I cannot define it precisely, like pitch."*
>
> "A diferencia del tono, las características de audio no tienen necesariamente un concepto directo asociado. En otras palabras, estaríamos estos números sobre el audio, pero si me preguntaras, ¿qué significa exactamente este número? Puedo darte una explicación medianamente razonable pero no puedo definirlo con precisión, como con el tono."

[2] *Beats per minute*

[3] El vídeo se encuentra disponible en: https://www.youtube.com/watch?v=cd2Jfi0PE2Y.

Otro punto de referencia, lo encontramos en un trabajo de Aucouturier and Bigand (2013), donde exponen los siete motivos por los que la clasificación automática de sonidos no es atractiva para la neurociencia. Es decir, que los resultados obtenidos con las características extraídas no son aceptados por estudios de psicología que tratan de entender cómo los humanos clasifican la música. Algunos de estos motivos son:

- **Las características de audio extraídas no tienen una función asociable cognitiva.** Podría parecer que las características extraídas son sustraídas a partir del sistema auditivo humano. Por ejemplo, los MFCCs intentan reproducir la escala no lineal de la cóclea pero esta afirmación sólo es parcialmente correcta, ya que partes de este algoritmo se añadieron después para mejorar los valores destinados al aprendizaje automático y en ningún momento para mejorar la relevancia cognitiva.

- **Falta de validación psicológica de bajo nivel.** Suponiendo que añadir una característica mejora la clasificación dentro del algoritmo (comparado con no añadirla), terminamos añadiéndola. Sin embargo, esta metodología es diferente a la práctica que se haría en neurociencia, donde no se añaden variables por su poder explicativo *a posteriori*, sino por corresponderse con una teoría o hipótesis probada anteriormente *a priori*. Es decir, para que una característica pueda ser añadida en neurociencia, ésta tiene que tener algún tipo de base teórica.

Por estas razones, consideramos que si alguna característica no está del todo clara en este documento es debido a que no existe una correspondencia clara de qué parte del problema de la clasificación resuelve cada característica.

3.3. Aprendizaje automático

El aprendizaje automático es un área de la inteligencia artificial que proporciona a las máquinas la capacidad de aprender. Se centra en el desarrollo de programas que permitan reconocer patrones, identificar clases o predecir resultados.

Hay dos tipos de aprendizaje que se diferencian según el tipo de datos de entrada:

1. **Aprendizaje supervisado**

   Estos algoritmos producen una correspondencia entre la entrada y la salida deseada. En clasificación, el sistema trata de etiquetar (clasificar) nuevos datos seleccionando una entre varias categorías. En nuestro caso, el sistema clasifica una canción según un vector de características eligiendo entre varios géneros. Estos algoritmos reciben un conjunto de datos debidamente clasificados y busca características o patrones comunes para generalizar las clases y poder clasificar nuevos datos.

2. **Aprendizaje no supervisado**

   El aprendizaje se lleva a cabo sobre un conjunto de datos sin etiquetar, es decir, sin tener información sobre las categorías a las que pertenecen esos ejemplos. Estos algoritmos lo que hacen es buscar una estructura en los ejemplos presentados y agrupan los datos que se parezcan más entre sí atendiendo a las variables que los describen.

Los algoritmos de aprendizaje que se plantearon en este trabajo pertenecen al grupo de aprendizaje supervisado.

3.3.1. Árboles de decisión

Utilizamos árboles de decisión[4] ya que permiten una visualización gráfica de los resultados y además han sido utilizados en otros trabajos de clasificación automática de géneros (Ponce de León Amador, 2011). Este tipo de algoritmo genera un gráfico en forma de árbol donde se puede ver las características que han sido más relevantes a la hora de la clasificación.

Un árbol de decisión proporciona un método de clasificación que construye un modelo a partir de un conjunto de datos debidamente etiquetado con la clase a la que corresponden para posteriormente, clasificar nuevos datos desconocidos (Breiman et al., 1984).

El funcionamiento del árbol es el siguiente: en primer lugar, partiendo del conjunto de datos inicial, se busca la característica que pueda generar una regla para partir dicho conjunto en dos subconjuntos siendo éstos lo más "puros" posibles. Entiéndase "puro" como el agrupamiento donde la mayor proporción de elementos del conjunto pertenecen a una misma clase. Y estos dos subconjuntos resultantes, vuelven a dividirse a su vez, buscando la nueva característica y la regla que los separará nuevamente. Un ejemplo de regla puede ser $x > 7$ donde x es una característica.

Existen varias formas para determinar la pureza (función de impureza), en nuestro caso es Gini (Breiman et al., 1984). Gini es una medida de desigualdad que se calcula sumando la probabilidad f_i de los valores con etiqueta i por la probabilidad $1 - f_i$ del error en etiquetarlo. Alcanza su mínimo (cero) cuando todos los casos del conjunto pertenecen a una sola categoría.

Una vez que el árbol está formado, podemos hacer predicciones con él. Dado un nuevo ejemplo para el cual queremos saber a qué clase pertenece, lo colocamos en la parte superior del árbol y analizamos la regla que divide dicho nodo para ver en qué nodo hijo cae el ejemplo y repetimos hasta llegar un nodo hoja que defina la clase en la que se encuentra. Por ejemplo, en la figura 3.3 vemos un pequeño árbol de decisión donde en cada nodo se comprueba una característica y se avanza (baja) por la estructura del árbol. De manera

[4]Referencia: http://scikit-learn.org/stable/modules/tree.html

que un caso que tenga como características $sex = male$ y $age = 9{,}6$ sería pronosticado como un no superviviente[5] .

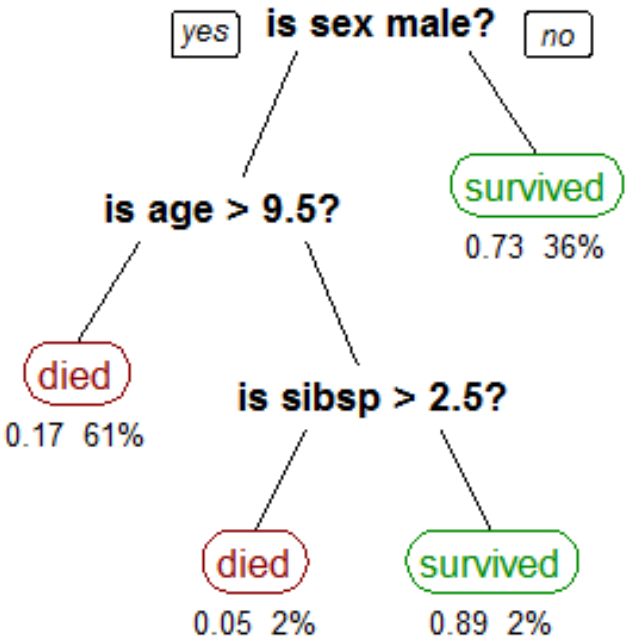


Figura 3.3: Ejemplo de árbol de decisión

Los árboles de decisión tienen como ventajas:

1. Son simples de entender e interpretar y además dan una descripción gráfica de los resultados.

2. No hay que seleccionar las variables de entrada más relevantes porque de eso se encarga el algoritmo de generación del árbol.

3. Son rápidos, tanto a la hora de construirse como de hacer predicciones.

4. Son capaces de manejar datos continuos o discretos.

5. Pueden ser usados para hacer no solamente clasificación binaria, sino en múltiples clases.

[5]Este ejemplo está extraído del tutorial en la página de *kaggle* sobre predicción de supervivientes en el Titanic https://www.kaggle.com/c/titanic

6. Usan un modelo de caja blanca. La estructura del árbol explica de forma comprensible cómo se realiza la clasificación.

Sin embargo, también nos encontramos con algunas desventajas:

1. Los árboles de decisión pueden "sobreaprender" y no generalizar bien. De manera que puede que sea necesario limitar el crecimiento del árbol, por ejemplo, usando estrategias de validación cruzada.

2. Pueden ser inestables debido a que pequeñas variaciones en los datos pueden dar como resultado árboles completamente distintos.

3. Hay conceptos que son difíciles de aprender para el algoritmo ya que no los expresa con facilidad como por ejemplo la lógica *XOR*.

3.3.2. Bosques aleatorios

Para complementar los resultados y lograr una precisión mayor de la clasificación utilizamos otro algoritmo en este trabajo: el bosque aleatorio[6].

El bosque aleatorio[7] es una agregación de árboles de decisión. Las agregaciones de clasificadores son utilizadas ya que normalmente funcionan bien debido a que cada clasificador aprende en detalle diferentes aspectos del conjunto de datos y luego ponen en común las predicciones teniéndose en cuenta la mejor o más repetida. En algoritmos de agregación tenemos métodos de "*bagging*" que se basan en entrenar diferentes instancias de algoritmos (como en este caso árboles) con partes aleatorias del conjunto de datos para entrenar. La forma de elegir estas partes aleatorias depende de los distintos agregados, en el caso del bosque aleatorio utilizamos la **selección con reemplazamiento**, que siendo n el conjunto de datos para entrenar coge n muestras y en cada uno tiene una probabilidad de $(1 - \frac{1}{n})^n$ de ser omitido (ya que es elegido varias veces).

El objetivo de agregar distintos árboles entonces es construir un modelo que mejora la generalización y robustez frente a un solo árbol. Para ellos construimos los árboles independientemente y luego promediamos las predicciones para dar la predicción final del bosque.

Cada árbol se crea de la siguiente manera:

- Si el número de casos para entrenar es n, se cogen n casos al azar con reemplazamiento.

[6]Referencia: http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[7]Página de Breiman donde explica los bosques aleatorios: https://www.stat.berkeley.edu/ breiman/RandomForests/cc_home.htm

- Si hay M variables de entrada (características), un número $m < M$ se especifica de manera que en cada nodo se seleccionan m características al azar de las M totales. Este valor de m es constante en la creación del bosque.
- Cada árbol crece lo máximo posible, sin poda.

La predicción del bosque sobre una muestra es el promedio de las predicciones de los árboles independientes. Cuanto mayor sea el número de árboles, mayor será la tasa de aciertos del bosque, pero tanto el entrenamiento como la predicción serán más lentos en tiempo de ejecución. Con un mayor número de características es necesario un mayor número de árboles y la forma de encontrar el valor perfecto depende del problema que se quiere resolver y las condiciones especiales (mayor importancia en tiempo o aciertos).

Características destacables de este modelo:

1. Tiene buena precisión comparado con otros algoritmos de aprendizaje automático.
2. Es eficiente con conjuntos de datos amplios.
3. Puede soportar miles de variables de entrada sin que sea necesario que el usuario seleccione las mejores.
4. Nos da información de qué variables son importantes.
5. Al aumentar el número de árboles no se "sobreentrena", pero el clasificador en conjunto sí que se puede "sobreentrenar".
6. A diferencia de los árboles de decisión, la clasificación generada es difícil de interpretar visualmente.

Importancia de las variables

En principio, las variables no utilizadas en un árbol de decisión no son importantes, aunque puede no ser cierto en algunos casos, como en el caso en el que sea una variable correlacionada con otra o redundante. La importancia de las variables utilizadas en un árbol se pueden medir de forma individual utilizando la impureza Gini. Cada vez que partimos un nodo con una variable resultan dos nodos hijos con menos impureza. La importancia de la variable la conseguimos agregando el decrecimiento que se produce en la impureza al partir para esa variable a lo largo del árbol cuando haya sido utilizada.

En los bosques aleatorios, que están formado por un conjunto de árboles de decisión, se puede medir la importancia de las variables de cada árbol, de manera que una característica es importante a partir de cómo decrece la impureza del árbol. Este decrecimiento de la impureza se promedia para todos los árboles del bosque y se consigue la importancia en

el clasificador global. Este método se denomina como la media del decrecimiento de la impureza (*mean decrease impurity*)[8].

3.4. Validación de la clasificación

A la hora de ajustar un algoritmo de aprendizaje automático se busca que generalice bien el conjunto de datos que se ha usado para entrenarlo, es decir, que aprenda únicamente aquellas características que son extrapolables a otros datos que no hayan sido usados para entrenar el algoritmo. Hay que evitar que aprenda características irrelevantes del conjunto de datos de entrenamiento. Para ajustar un algoritmo de forma correcta se deben usar estrategias de validación cruzada como la que explicamos a continuación.

3.4.1. Validación cruzada de K iteraciones

Si tenemos un conjunto inicial de 100 elementos y queremos generar un modelo predictivo, puede parecer razonable partir el conjunto en uno de 80 y otro de 20. Podemos utilizar entonces el conjunto de 80 para entrenar, y el de 20, para comprobar la precisión del algoritmo (conjunto de prueba). Lo normal es encontrar un error menor en el conjunto de entrenamiento que en el de prueba, pero el de prueba nos aporta una orientación más realista sobre cómo se comportará el modelo ante datos nuevos.

Sin embargo, puede darse el caso de que esta partición nos dé una precisión más alta de la realmente esperable, ya que puede haberse dado que las muestras más "fáciles" de clasificar han caído en el conjunto de prueba, o puede haber sucedido lo contrario. En definitiva, ante particiones diferentes es esperable que encontremos valores distintos de entrenamiento y prueba.

Para obtener una estimación más fiable del rendimiento de nuestro modelo en el conjunto de validación usaremos la técnica conocida como validación cruzada de K iteraciones (*K-Fold cross validation*) que mitiga el problema de condicionar los resultados a la partición de entrenamiento y prueba. Esta técnica permite evaluar y garantizar que la precisión obtenida es independiente de los subconjuntos seleccionados como entrenamiento y prueba. En definitiva, se estima cómo de exacto es el modelo generado. El procedimiento (como se indica en la figura 3.4, donde cada color representa una clase) es el siguiente:

1. El conjunto del que se parte se divide en K subconjuntos donde uno se asigna como el conjunto de prueba y la unión del resto (*K-1*) se asignan como entrenamiento.

[8]Más información y formulación matemática en el capítulo 5.3.4 del libro escrito por Breiman et al. (1984) que es la fuente que inspira la implementación en *scikit learn*

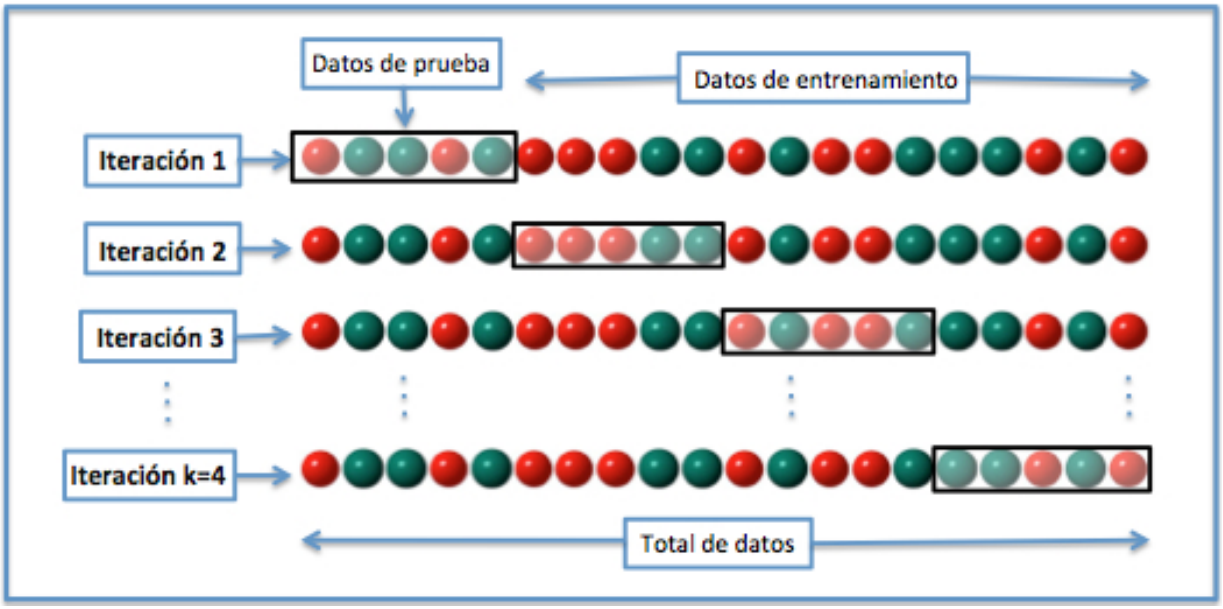


Figura 3.4: Validación cruzada de K iteraciones

2. El proceso de validación se repite K veces, de manera que todos los K subconjuntos hayan pasado por ser el conjunto de prueba.

3. Una vez terminado el proceso, se calcula la media aritmética de las tasas de aciertos (*accuracy*) de cada iteración con su desviación típica estándar. Esto nos da una tasa de aciertos media con una desviación que podemos tomar como la exactitud aproximada que tiene el algoritmo con el conjunto de datos utilizado.

La validación cruzada se puede hacer **estratificada**. Esto quiere decir que las particiones, a pesar de ser aleatorias, forzosamente tienen la misma proporción de elementos de cada clase que la proporción de clases en el conjunto original.

3.4.2. Matrices de confusión y terminología

A lo largo de este proceso de clasificación se calculan matrices de confusión. Estas tienen mucha importancia y son muy utilizadas en el capítulo 5. Por lo tanto, vamos a explicar en qué consisten y la terminología y métricas derivadas a partir de ellas.

La matriz de confusión es una forma de visualización de los resultados de un algoritmo de clasificación. El algoritmo recibe clases con una etiqueta real y devuelve una etiqueta pronosticada. Siendo n el número de clases utilizadas, la matriz de confusión tiene dimensiones $n \times n$. Para visualizar cómo de buena ha sido la clasificación, formamos esta matriz donde las filas se corresponden con las etiquetas reales de los datos y las columnas con las

etiquetas predichas por el algoritmo. De esta forma, se puede observar cómo los números que aparecen en la diagonal de la matriz se corresponden con la cantidad de muestras acertadas por el algoritmo, es decir, el número de muestras cuya etiqueta real es igual a la pronosticada. Por el contrario, los números que aparecen fuera de esta diagonal se corresponden con las muestras que no han sido bien clasificadas por el algoritmo y apuntan confusiones.

Las matrices de confusión aportan información relevante como se muestra en la figura 3.5, además generalizan para más clases de forma trivial. En esta figura se muestran un total de 165 muestras etiquetadas en dos clases (YES, NO), indicando cuatro métricas (TN,FN,FP,TP). A continuación, presentamos estas cuatro métricas que se calculan por cada clase, de forma que serán indicativas del rendimiento del clasificador para esa clase:

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Figura 3.5: Matriz de confusión de dos clases

- **TP - Verdaderos positivos**

  Los verdaderos positivos son los aciertos, de manera que la clase pronosticada y la real coinciden. Estos valores se corresponden con la diagonal de la matriz de confusión. Siendo n el número de clases, cm una matriz de dimensiones $n \times n$ e i la fila de la matriz de la clase que evaluamos. El valor se calcula:

$$\mathrm{TP}_i = cm[i][i]$$

- **FN - Falsos negativos**

  Los falsos negativos se dan cuando el elemento pertenece a la clase que estamos evaluando, pero no lo identificamos como tal. Dentro de la matriz de confusión, el

valor sería la suma de todos los valores de la fila de la clase exceptuando la columna de la clase (que serían aciertos). Siendo n el número de clases, cm una matriz de dimensiones $n \times n$ e i la fila de la matriz donde se da el valor de la clase original que se evalúa. El valor se calcula:

$$\mathrm{FN}_i = \sum_{j=0}^{n-1} cm[i][j] \text{ (Siendo } j \neq i)$$

- **TN - Verdaderos negativos**

  Los verdaderos negativos se dan cuando la clase predicha no es la que estamos evaluando ni originalmente ni al pronosticarla. Son poco descriptivos en un experimento con multitud de clases. Dentro de la matriz de confusión, es la suma de todos los valores salvo los pertenecientes a la fila y la columna de la clase evaluada. Siendo n el número de clases, cm una matriz de dimensiones $n \times n$ e i_c y j_c la fila y la columna respectivamente de la clase evaluada. El valor se calcula:

$$\mathrm{TN}_{i_c} = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} cm[i][j] \text{ (Siendo } i \neq i_c \wedge j \neq j_c)$$

- **FP - Falsos positivos**

  Los falsos positivos, son la cantidad de muestras que han sido detectadas como la clase que se está evaluando, pero cuya clase original era otra distinta. Dentro de la matriz de confusión este valor es la suma de los valores de la columna etiquetada con la clase exceptuando la fila de esa misma clase. Siendo n el número de clases, cm una matriz de dimensiones $n \times n$ y j la columna de la matriz donde se predice la clase que se evalúa. El valor se calcula:

$$\mathrm{FP}_j = \sum_{i=0}^{n-1} cm[i][j] \text{ (Siendo } i \neq j)$$

A partir de estas métricas más sencillas calculamos las siguientes que son más explicativas:

- **Precisión (*Precision*)**

  Es una métrica que simboliza cuando el clasificador predice una clase, como es de probable que haya acertado y que la clase original fuera esa. Se calcula como los verdaderos positivos entre todos los valores predichos como la clase que se evalúa:

$$\text{Precisión} = \frac{TP}{(TP + FP)}$$

- **Sensibilidad (*Recall*)**

  La sensibilidad mide la capacidad del clasificador de detectar una clase. Nos indica cuando una muestra de una clase es introducida en el clasificador qué probabilidad hay de que sea clasificada correctamente. Se calcula como los verdaderos positivos entre la suma entre los verdaderos positivos y los falsos negativos:

$$\text{Sensibilidad} = \frac{TP}{(TP + FN)}$$

- **F1 score (*F1 score*)**

  Esta métrica es la media armónica de la precisión y la sensibilidad. Nos da información de cómo de buena es el desempeño del algoritmo sobre una clase en concreto. Se calcula:

$$\text{F1score} = \frac{2 \cdot TP}{(2 \cdot TP + FP + FN)}$$

- **Tasa de aciertos (*Accuracy*)**

  La tasa de aciertos la definimos para la matriz de confusión completa y es el porcentaje de predicciones correctas sobre el total de muestras predichas. Para calcularlo sumamos la diagonal principal (todos los TP) y dividimos entre el número de muestras totales (que es la suma de todos los TP y FN). Como fórmula:

$$\text{Tasa de aciertos} = \frac{\sum TP}{\sum TP + \sum FN}$$

3.4.3. Grafos de confusión

Las matrices de confusión son un estándar a la hora de representar resultados dados por un modelo predictivo. Sin embargo, a pesar de ser un tipo de representación muy útil, en situaciones donde el número de clases es muy grande, una matriz de confusión no parece ser una forma muy eficiente para visualizar los resultados. Esto nos lleva a proponer otra forma de representación basada en la matriz de confusión: los grafos de confusión.

En un grafo de confusión, los nodos se corresponden con las clases utilizadas en la clasificación y las aristas se crean a partir de la matriz de confusión. Cuando se da una confusión de un género a otro en n muestras, la arista unirá el género original de las muestras al género pronosticado incorrectamente con un peso de arista de n. La dirección de la arista es importante, ya que une un nodo a otro con el que se produce confusión, por ello, el grafo es dirigido, es decir, una arista no implica relación bidireccional.

En nuestros grafos, el tamaño de los nodos está relacionado con la tase de aciertos, donde los de mayor tamaño serán los que se predicen con mayor acierto. La distribución espacial se forma a partir de *Force Atlas* 2 y *Frutchman Reingold*, dos algoritmos que sólo

es necesario entender que aproximan o alejan nodos considerando las uniones mediante aristas como si fueran fuerzas, siendo éstas mayores con aristas de mayor peso (Bastian et al., 2009).

Para realizar está representación gráfica se ha utilizado *Gephi* (Bastian et al., 2009). *Gephi* es un software para la exploración, manipulación y visualización de grafos (o redes). En la figura 3.6 se muestra un ejemplo de grafo sencillo, siguiendo las especificaciones descritas. En él vemos como la confusión de *ElectroHouse* en *BigRoom* es mayor que la de *BigRoom* a *ElectroHouse* y además vemos que la mayor tasa de aciertos la tiene el *BigRoom*.

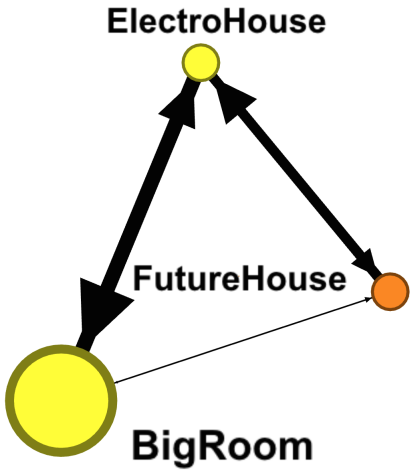


Figura 3.6: Ejemplo de grafo

3.4.4. Comparación humano-máquina

En el estudio de Seyerlehner et al. (2010), se comprueba la capacidad del ser humano para clasificar muestras de distintos géneros musicales frente a la clasificación automática. En este trabajo, se utiliza un conjunto formado por 19 géneros con 10 canciones por cada uno, es decir, un total de 190 muestras. Afirman que el reducido número de canciones del estudio es debido a la complejidad de escuchar y clasificar cada canción. Concluyeron que la precisión que se conseguía con las personas era mayor que la realizada por la máquina en este estudio.

Por este motivo, consideramos que un estudio como éste puede ser una buena forma de validación de la clasificación obtenida por aprendizaje automático. El experimento explicado en detalle se encuentra en la sección 4.4.

Capítulo 4

Desarrollo

En este capítulo detallamos el proceso llevado a cabo para la realización del trabajo. A modo de resumen, en la figura 4.1 se muestra las fases que sigue este trabajo. El código de la implementación se encuentra disponible en el repositorio de *Github*, *PyGenreClf* (https://github.com/Caparrini/pyGenreClf) con una documentación detallada del mismo. A continuación detallamos las fases del proyecto así como los archivos *.py* que hemos generado para cumplirlas:

1. **Generación de conjuntos de datos**

   En primer lugar, se elige un conjunto de datos. La referencia que se tiene en cuenta para la extracción del conjunto viene detallada en la sección 3.1.

2. **Extracción de características**

   Extraemos las características del conjunto de datos almacenándolas en un fichero. Todo lo referido a las características del audio viene definido en la sección 3.2.2. El proceso que se utiliza en este trabajo para la extracción de las mismas queda reflejado en las secciones 4.2 y siguientes.

   En nuestro proyecto el archivo ***featuresExtraction.py*** lo utilizamos para extraer las características de los archivos de audio. Utiliza las librerías *PyAudioAnalysis* y *Essentia* para extraer las características y *Pandas* para guardar los resultados en formato *.csv*. Más información sobre estas librerías en la sección 2.4.

3. **Generación de modelos**

   Para cada conjunto, utilizamos árboles de decisión y bosques aleatorios para generar el mejor modelo predictivo posible. En la sección 3.3 se proporcionan más detalles sobre estos algoritmos.

   El archivo ***classifier.py*** de nuestro proyecto contiene funciones para generar y probar el clasificador. En este archivo se encuentra las implementaciones para los árboles de decisión y los bosques aleatorios con los parámetros finales.

   En el archivo ***optimize.py*** se encuentra la implementación de un optimizador de árboles de decisión y bosques aleatorios usando la librería *DEAP*. Más información sobre esta librería en la sección 2.4.

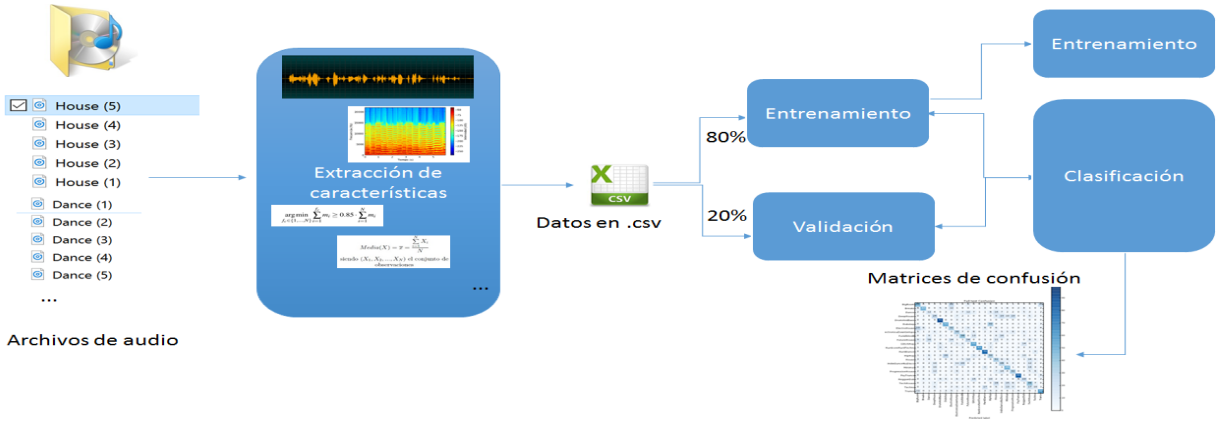


Figura 4.1: Esquema procesamiento del trabajo

4. **Validación de modelo**

   A través de los métodos de validación de los modelos los descartamos o aceptamos hasta llegar finalmente al mejor. Para conocer más en detalle la forma de validación, el proceso se encuentra explicado en la sección 3.4.

   En ***tools.py*** se encuentran varias funcionalidades como el cálculo de métricas y grafos de confusión a partir de una matriz de confusión.

   Para llevar a cabo una validación con usuarios, hemos implementado ***humanExperiment.py*** para la generación y evaluación de una clasificación por usuarios similar al de la máquina.

4.1. Conjunto de datos del trabajo

El conjunto de datos está formado por 23 géneros con 100 canciones de 2 minutos (120 segundos) de duración.

La mayor parte de estudios sobre clasificación automática de géneros musicales utiliza entre 2 y 10 clases. Por ello, llevamos a cabo diferentes experimentos, empezando por uno con 7 géneros de los 23 totales que tiene *Beatport*.

4.1.1. Conjunto de datos de 7 géneros

Este conjunto se utilizó como una primera aproximación. Decidimos que sería conveniente empezar con un conjunto más pequeño para poder detectar más fácilmente rasgos comunes entre géneros así como las características más importantes. Cuánto más pequeño es el conjunto, más sencillos e interpretables son los gráficos que mostramos.

Los géneros son *BigRoom, Dance, DrumAndBass, Dubstep, ElectroHouse, FutureHouse* y *HipHop*, con 100 canciones de cada género. Dentro de los 23 géneros, estos 7 se podrían considerar más populares y podrían ser reconocidos por oyentes poco o nada habituales de música electrónica.

Conjunto de datos de 7 géneros del experimento humano-máquina

Es un subconjunto del conjunto de datos de 7 géneros que se presentó anteriormente. Lo hemos utilizado para el experimento de clasificación con personas (más información en la sección 3.4.4). Está formado por 140 canciones pertenecientes a los 7 géneros (20 canciones por género) del primer conjunto (*BigRoom, Dance, DrumAndBass, Dubstep, ElectroHouse, FutureHouse* y *HipHop*).

4.1.2. Conjunto de datos de 23 géneros

En este conjunto de datos se analizan 23 géneros: *BigRoom, Breaks, Dance, DeepHouse, DrumAndBass, Dubstep, ElectroHouse, ElectronicaDowntempo, FunkRAndB, FutureHouse, GlitchHop, HardcoreHardTechno, HardDance, HipHop, House, IndieDanceNuDisco, Minimal, ProgressiveHouse, PsyTrance, ReggaeDub, TechHouse, Techno, Trance.*

Es el conjunto total de datos y está compuesto por 2300 canciones (100 canciones por cada género).

A continuación a modo de resumen, se muestra en la tabla 4.1, cada subgénero junto el BPM y un enlace a la lista de cada subgénero. Para facilitar la comprensión del lector, puede pulsar sobre el nombre de la lista y escuchar canciones etiquetadas en ese subgénero desde la página oficial de *Beatport*. Cabe destacar que los datos referidos al BPM son meras aproximaciones ya que se puede considerar que un estilo en concreto oscila en un determinado rango. Además pueden existir canciones etiquetadas en ese subgénero que tengan un BPM posicionado fuera de él. El BPM es una característica muy importante para DJs ya que lo utilizan para mezclar temas, es decir, para sincronizar canciones una a continuación de otra.

4.1.3. Conjunto de datos de validación de 23 géneros

Es un conjunto destinado a probar el clasificador final entrenado a partir del conjunto de datos inicial de 23 géneros. Este es un conjunto que sólo será expuesto al modelo predictivo cuando esté finalmente entrenado y analizado para probar la tasa de aciertos con un conjunto completamente nuevo de datos.

Este conjunto de datos está formado por 60 canciones de cada género, es decir, 1380 en

Subgénero	BPM	Lista
BigRoom	126-132	https://www.beatport.com/genre/big-room/79
Breaks	110-150	https://www.beatport.com/genre/breaks/9
Dance	100-150	https://www.beatport.com/genre/dance/39
DeepHouse	120-125	https://www.beatport.com/genre/deep-house/12
DrumAndBass	160-192 (172)	https://www.beatport.com/genre/drum-and-bass/1
Dubstep	140-150 (160-175)	https://www.beatport.com/genre/dubstep/18
ElectroHouse	125-135	https://www.beatport.com/genre/electro-house/17
Electronica / Downtempo	<120	https://www.beatport.com/genre/electronica-downtempo/3
Funk	100-125	https://www.beatport.com/genre/funk-soul-disco/40
FutureHouse	120-130	https://www.beatport.com/genre/future-house/65
GlitchHop	110-160	https://www.beatport.com/genre/glitch-hop/49
HardcoreHardTempo	150-250	https://www.beatport.com/genre/hardcore-hard-techno/2
HardDance	150-230	https://www.beatport.com/genre/hard-dance/8
HipHop	90-160	https://www.beatport.com/genre/hip-hop-r-and-b/38
House	120-135	https://www.beatport.com/genre/house/5
IndieDanceNuDisco	110-126	https://www.beatport.com/genre/indie-dance-nu-disco/37
Minimal	118-124 o 128	https://www.beatport.com/genre/minimal-deep-tech/14
Progressive House	120-130	https://www.beatport.com/genre/progressive-house/15
Psy-Trance	140-150	https://www.beatport.com/genre/psy-trance/13
Reggae/Dub	90-125	https://www.beatport.com/genre/reggae-dancehall-dub/41
Tech House	120-128	https://www.beatport.com/genre/tech-house/11
Techno	120-125	https://www.beatport.com/genre/techno/6
Trance	125-160	https://www.beatport.com/genre/trance/7

Cuadro 4.1: Tabla resumen subgéneros

total (60 canciones×23 géneros). Estas canciones pertenecen a las 100 mejores canciones de *Beatport* publicadas en abril de 2017. Como es natural, las canciones que se han mantenido entre las 100 mejores canciones entre octubre de 2016 (la fecha en la que se extrajo el conjunto de 23 géneros) y abril de 2017 no han sido incluidas en este conjunto de validación, ya que se usaron para ajustar el clasificador.

4.2. Extracción de características

La clave para una buena clasificación es conseguir una representación exacta del audio. Necesitamos que el algoritmo pueda "escuchar" la canción. Para ello, es necesario extraer las características del audio (más información sobre las características en la sección 3.2.2) y almacenarlas en un vector.

El **vector de características** que usaremos para caracterizar cada señal **tendrá 71 elementos**. Primero, es necesario desarrollar el proceso de extracción de características con las librerías utilizadas, *PyAudioAnalysis* y *Essentia*. Más información sobre estas librerías en la sección 2.4.

4.2.1. PyAudioAnalysis

Esta librería ha sido la seleccionada para calcular las características de audio casi en su totalidad (70 de 71). A continuación explicaremos el proceso de extracción con esta librería, sin embargo, para más información se encuentra disponible en el trabajo de Giannakopoulos (2015).

Este proceso está aplicado a cada canción, la cual tienen una duración de dos minutos, consiguiendo finalmente un vector con 70 características. Para empezar diferenciamos tres fases del proceso:

1. Extracción de características en ventanas cortas.
2. Promedio de las características de corto plazo en una ventana de textura.
3. Métricas estadísticas de todas las ventanas de textura

Extracción de características en ventanas cortas

El audio es una señal de dominio en el tiempo. En primer lugar, la señal es dividida en pequeños fragmentos temporales (ventanas) de 50 *milisegundos*, donde el mismo fragmento de audio no puede estar en dos ventanas a la vez, por lo que no hay solapamiento. A partir

Número	Característica
1	Zero Crossing Rate
2	Energy
3	Entropy of Energy
4	Spectral Centroid
5	Spectral Spread
6	Spectral Entropy
7	Spectral Flux
8	Spectral Rollof
9-21	MFCCs
22-33	Chroma Vector
34	Chroma Deviation

Cuadro 4.2: Tabla vector de características.

de cada una de ellas se calculan todas las características descritas en la figura 4.2. Más detalle sobre ellas en la sección 3.2.

Teniendo en cuenta que tenemos 120 segundos de muestra y que la ventana es de 50 *milisegundos*, observamos que el número de ventanas cortas es de 2400:

$$\text{Número de ventanas cortas} = \frac{120s}{0{,}05s} = 2400$$

Además, como se calculan 34 características para cada una, en este punto del proceso tenemos un vector que nada se parece al vector final. Tenemos un vector de 2400 elementos, que a su vez, son vectores de 34 características. Por esta razón, tenemos una matriz de dimensiones 2400×34.

Medias por ventana de textura

La ventana de textura es otro fragmento temporal (ventana) de mayor duración que incluye dentro de sí ventanas cortas. En nuestro caso, la ventana de textura es de 1 segundo de duración, sin solapamiento. En la figura 4.2 se presenta de forma esquemática la información sobre la ventana de textura.

Para cada ventana de textura, las características son la media de los valores de las ventanas cortas incluidas en ella. Es decir, cada ventana de textura tiene también 34 valores, pero éstos se calculan a través de las medias de las características de las ventanas cortas que están contenidas en ella.

Dado que en 1 segundo tenemos 20 ventanas de 50 *milisegundos*, tras realizar este

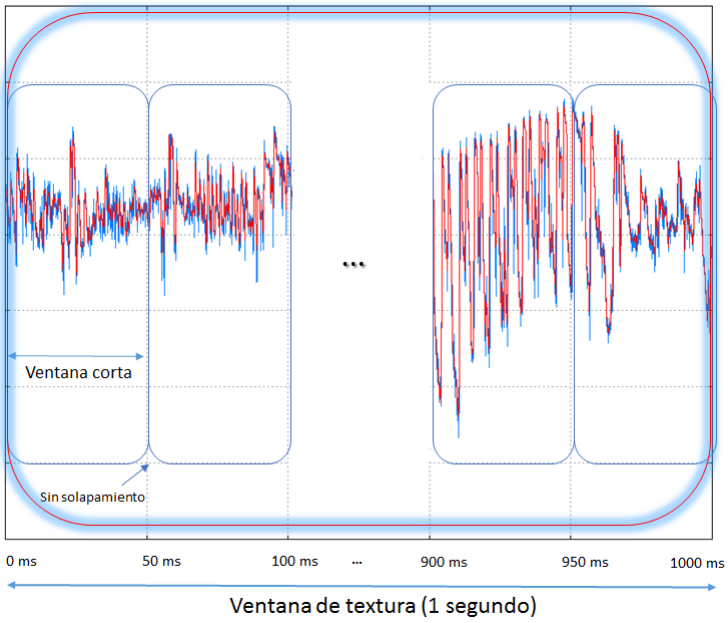


Figura 4.2: Detalle de la ventana de textura

proceso, obtenemos un total de 120 ventanas de textura:

$$\text{Número de ventanas de textura} = \frac{2400}{20} = 120$$

Por lo tanto, en este punto tenemos un vector de 120 elementos que a su vez, tienen 34 características.

Métricas estadísticas finales

Finalmente, no estamos interesados en tener un valor por cada ventana de textura de las muestras que queremos clasificar sino que necesitamos un único valor que explique cada característica. Para ello, las métricas elegidas concretamente son la media aritmética y la desviación típica estándar. Estás métricas se aplican a todas las ventanas de textura de manera que finalmente tenemos 34 medias de las características y 34 desviaciones estándar por cada una de las canciones. Estos 68 elementos ya sí forman parte del vector final de características.

A estos valores sumamos el BPM y su valor de confianza (grado de seguridad de que el valor calculado es correcto[1]). Por lo tanto el vector de características final se compone de

[1]El método implementado para calcular el BPM y su valor de confianza, utiliza un procedimiento de detección de máximos, aplicado en las características de ventanas cortas. Para más información sobre el algoritmo utilizado y la extracción de características en https://github.com/tyiannak/pyAudioAnalysis/wiki/3.-Feature-Extraction (Giannakopoulos, 2015)

71 elementos tal y como se indica en la figura 4.3. En cada fila se detalla la posición del vector junto con el nombre de cada característica (más detalle de las características en la sección 3.2.2), la métrica utilizada, el valor almacenado en esa posición es calculado como una media (media) o como una desviación típica (std), y el rango de valores en el que se mueve cada valor.

Número	Característica	Métrica	Rango
1	Zero Crossing Rate	Media	0.017-0.247
2	Energy	Media	0.005-0.282
3	Entropy of Energy	Media	2.746-3.254
4	Spectral Centroid	Media	0.083-0.374
5	Spectral Spread	Media	0.152-0.301
6	Spectral Entropy	Media	0.34-1.959
7	Spectral Flux	Media	0.003-0.0533
8	Spectral Rollof	Media	0.012-0.484
9-21	MFCCs	Media	—
22-33	Chroma Vector	Media	—
34	Chroma Deviation	Media	—
35	Zero Crossing Rate	Std	0.004-0.155
36	Energy	Std	0.03-0.158
37	Entropy of Energy	Std	0.039-3.23
38	Spectral Centroid	Std	0.017-0.129
39	Spectral Spread	Std	0.07-0.062
40	Spectral Entropy	Std	0.026-0.955
41	Spectral Flux	Std	0.001-0.07
42	Spectral Rollof	Std	0.006-0.299
42-55	MFCCs	Std	—
56-67	Chroma Vector	Std	—
68	Chroma Deviation	Std	—
69	BPM *PyAudioAnalysis*	Ritmo	63-600
70	Confianza del BPM	—	0.073-0.423
71	BPM *Essentia*	Ritmo	61-188

Cuadro 4.3: Tabla vector de características.

4.2.2. Essentia - BPM

A través de *Essentia* (Bogdanov et al., 2013) se calcula el BPM, para complementar el vector de características extraídas con *PyAudioAnalysis*. El BPM calculado por *PyAudioAnalysis* producía resultados atípicos. Estos valores no se aproximaban al BPM "real" impuesto a las personas sino que eran múltiplos de él que en ocasiones oscilaban entre 200 y 500. Por ello, decidimos añadir el calculado por *Essentia* para completar la información del vector de características. En un principio el objetivo era sustituir el valor dado por *PyAudioAnalysis* por el de *Essentia*, pero al utilizar los dos conjuntamente, vimos que producía mejores resultados de clasificación, por lo que decidimos dejar ambos. En la sección de resultados veremos esta relación, pero básicamente se han mantenido las dos medidas porque los valores que aportan no son erróneos y ofrecen más información rítmica de la muestra.

4.3. Entrenamiento y optimización de los algoritmos de aprendizaje automático

Llegados a este punto, con las características extraídas de cada conjunto, es necesario la creación del modelo predictivo. El proceso de generación de cada modelo y de validación están relacionados y se dan simultáneamente. Por ello, para conseguir la tasa de aciertos y las matrices de confusión siempre hemos utilizado una **validación cruzada de 5 iteraciones estratificada** (este procedimiento se presenta en detalle en la sección 3.4.1).

4.3.1. Árboles de decisión

La implementación utilizada del **árbol de decisión** es la presente en la librería *Scikit-learn* (más información en la sección 2.4).

En dicha implementación se pueden pasar dos parámetros que permiten controlar el aprendizaje del árbol:

- *Min_ samples_ split (muestras mínimas a partir)*: es el número mínimo de muestras que tiene que haber en un nodo para que el algoritmo lo pueda partir.
- *Min_ samples_ leaf (muestras mínimas por hoja)*: es el valor mínimo requerido de muestras que tiene que tener un nodo para formar una hoja final.

Para encontrar valores óptimos hemos usado un algoritmo genético (véase sección 4.3.3) y la mejor combinación de valores se muestra en la figura 4.4:

Árbol de decisión	*min_ samples_ split*	*min_ samples_ leaf*
7 géneros	40	9
23 géneros	61	15

Cuadro 4.4: Valores óptimos para los árboles de decisión.

4.3.2. Bosques aleatorios

La implementación utilizada del **bosque aleatorio** es la presente en la librería *Scikit-learn* (más información en la sección 2.4).

Al igual que en el apartado anterior, en los bosques se pueden pasar cuatro parámetros que permiten controlar el aprendizaje del bosque:

- *Min_ samples_ split (Muestras mínimas a partir)*: este valor es el número mínimo de muestras que tiene que haber en un nodo para que el algoritmo lo llegue a partir (aplicado a todos los árboles internos).
- *Min_ samples_ leaf (muestras mínimas por hoja)*: es el valor mínimo requerido de muestras que tiene que tener un nodo para formar una hoja final (aplicado a todos los árboles internos).
- *max_ features (características máximas)*: es el número de características que se consideran a la hora de partir un nodo. En caso de usar un valor entre 0 y 1, se aplica un porcentaje sobre el número de características.
- *n_ estimators (número de estimadores)*: es el número de árboles en el bosque.

En este caso, igual que en los árboles para encontrar valores óptimos hemos usado un algoritmo genético (véase sección 4.3.3). La mejor combinación de valores se muestra en la figura 4.5:

Bosque aleatorio	*min_ samples_ split*	*min_ samples_ leaf*	*max_ features*	*n_ estimators*
7 géneros	15	2	0.59	84
23 géneros	2	2	0.498	150

Cuadro 4.5: Valores óptimos para los bosques aleatorios.

4.3.3. Optimización con algoritmo genético

Con el objetivo de buscar el mejor modelo posible, utilizamos la librería **DEAP** (más información en la sección 2.4) para *Python* que permite la rápida implementación de algoritmos genéticos para optimizar un resultado. Un algoritmo genético es un algoritmo de

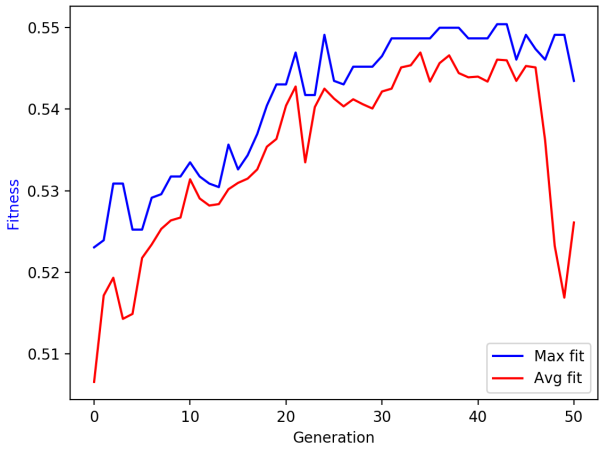


Figura 4.3: Optimización de bosque aleatorio.

búsqueda del mejor resultado basado en las ideas de la selección natural y la genética. De esta forma, diferentes soluciones al problema, son generadas de forma aleatoria al principio. Esta sería la primera generación, y las mejores soluciones, comparten parámetros entre ellas formando nuevas generaciones. Además en las diferentes generaciones se introducen mutaciones para generar aleatoriedad. De esta forma, vamos consiguiendo cada vez mejores resultados, y aunque no garantiza el mejor resultado, proporciona un resultado bueno ahorrando computación y recursos[2].

En nuestro proyecto, hemos implementado este algoritmo para afinar el aprendizaje de los árboles de decisión y los bosques aleatorios. El motivo por el cual se ha utilizado, es para evitar el "sobreajuste" (*overfitting*) de los algoritmos. Esto se produce cuando un sistema se entrena demasiado (se "sobreentrena") con el conjunto de entrenamiento, de manera que el modelo predice con gran exactitud muestras con las que haya sido entrenado, pero se comporta peor de lo esperado con muestras nuevas. En definitiva, el "sobreentrenamiento" causa que el algoritmo no generalice bien.

En la figura 4.3 podemos ver un gráfico de ejemplo con información sobre la optimización del bosque aleatorio en el conjunto de 23 géneros. En ella hemos usado una generación inicial de 30 bosques que evoluciona a lo largo de 50 generaciones. La línea azul indica la máxima tasa de aciertos de cada generación, mientras que la línea roja es la media de las tasas de aciertos de todos los bosques de la generación. Observamos cómo a partir de la generación 40 dejan de mejorar tanto las tasas de acierto máxima como la media de la generación.

[2]Más información en https://www.doc.ic.ac.uk/ nd/surprise_96/journal/vol1/hmw/article1.html

4.4. Diseño del experimento humano-máquina

Como se explica en la sección 3.4.4, probar la clasificación automática con seres humanos es una buena forma de validación. En este trabajo, de igual manera, elegimos una fracción de los géneros, un conjunto de 7 géneros ya que la clasificación de los 23 géneros totales para una persona puede ser demasiado tediosa. Sin embargo, a diferencia del otro estudio mencionado, facilitamos un conjunto de "entrenamiento" de la misma forma que le damos al algoritmo pero en menor cantidad. Es decir, el usuario cuenta con 7 carpetas pertenecientes a distintos géneros (cuyos nombres son anónimos), con 10 canciones en cada una. El requisito fundamental es que los usuarios deben escuchar absolutamente todas las canciones, no hacía falta que fuese entera, al menos un fragmento significativo de cada una de ellas.

En resumen, la estructura del experimento es:

- El conjunto de datos está formado por 7 géneros. Más información sobre el conjunto en la sección 4.1.1).
- Por cada género, se ofrece una carpeta con 10 muestras que pueden escuchar. Estas carpetas tendrán un nombre genérico (A, B, C,...) para así evitar sesgos en la clasificación debido al conocimiento anterior de las etiquetas por parte de los oyentes.
- Un conjunto de 70 canciones sin clasificar, 10 canciones por género, para que sean clasificadas dentro de los diferentes géneros. Cabe destacar que el número de canciones por género no fue concretado a ningún usuario.

El proceso del experimento sería por tanto:

- El sujeto tiene acceso a una carpeta de entrenamiento con una carpeta por cada género que contiene 10 canciones debidamente etiquetadas. Debe escucharlas antes del experimento para poder clasificar posteriormente. Además cuenta con papel para apuntar etiquetas que le ayuden a identificar los distintos géneros. Durante el resto del experimento siempre podrá volver a esta carpeta para escuchar estas canciones de referencia.
- En otra carpeta de experimento, se encuentra una carpeta vacía por cada género, con la etiqueta anónima idéntica a la carpeta de entrenamiento, y 10 muestras por género, anónimas y diferentes de las de entrenamiento, que deben mover a cada carpeta. Estas muestras las pueden escuchar libremente y sin límite de tiempo.

Capítulo 5

Resultados

"Data! Data! Data" he cried impatiently. "I can't make bricks without clay."
— Sherlock Holmes, *The Adventure of the Copper Beeches (1892)*

En este capítulo presentamos los resultados que hemos obtenido a partir de los conjuntos de datos presentados en la sección 4.1, utilizando los procesos de aprendizaje automático explicados en detalle en la sección 4.3. La forma de mostrar los resultados para cada algoritmo será ligeramente diferente:

- Árboles de decisión

  En los árboles presentaremos la matriz de confusión final a partir de la validación cruzada de 5 iteraciones (explicado en la sección 3.4.1). Además presentamos un gráfico del árbol para analizar la toma de decisiones.

- Bosques aleatorios

  En los bosques, ya que los resultados son similares pero mejores, aportaremos además de la matriz de confusión, la tabla de métricas y los grafos de confusión. La parte equivalente a visualizar la toma de decisiones, será explicada mediante un gráfico con la importancia de las características.

5.1. Conjunto de datos de 7 géneros

En primer lugar, para realizar una primera aproximación al problema, utilizamos un conjunto de 100 canciones de 7 géneros, pertenecientes a los 23 totales (más información en la sección 4.1.1). Los géneros contenidos en esta prueba son *BigRoom, Dance, DrumAndBass, Dubstep, ElectroHouse, FutureHouse y HipHop*. En la tabla 4.1 hay enlaces para escuchar una lista de cada subgénero y en el apéndice E hay descripciones detalladas de cada género.

5.1.1. Árbol de decisión

La principal ventaja de los árboles de decisión es que su interpretación resulta muy intuitiva. Por esto, aunque existen algoritmos de clasificación más sofisticados y precisos, el árbol permite visualizar las características que han sido relevantes en la clasificación así como la toma de decisiones.

Matriz de confusión

Después de realizar la validación cruzada de 5 iteraciones con el árbol usando los parámetros descritos en la sección 4.3.1, el promedio de las tasas de aciertos de cada conjunto de prueba nos da una media y una desviación típica estándar:

Tasa de aciertos media: 0.60 $\pm$ 0.07

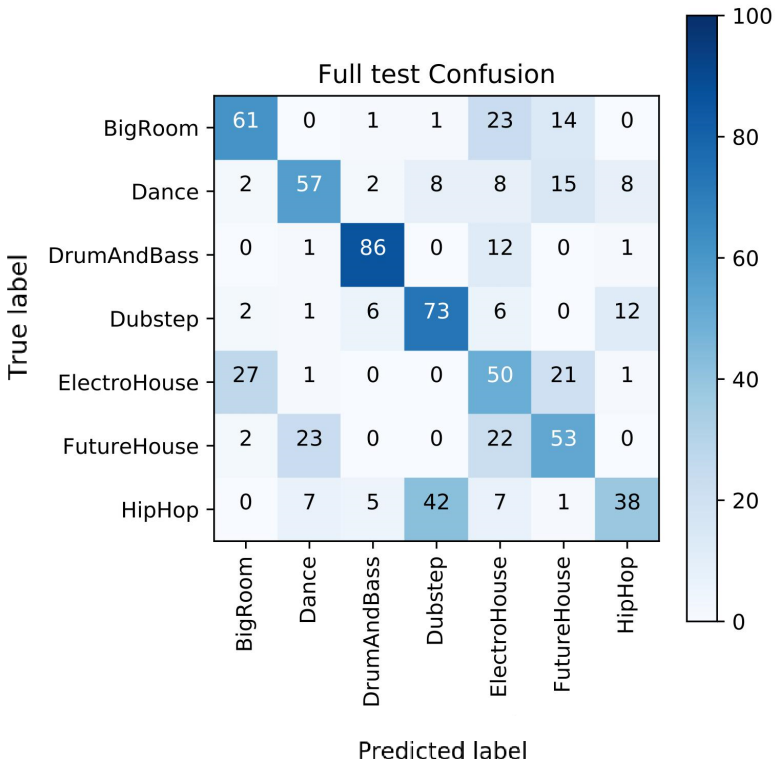


Figura 5.1: Matriz Confusión - Árbol (7 géneros)

La matriz de confusión de la figura 5.1 tiene las 700 canciones del conjunto cuando eran parte del conjunto de prueba en la validación cruzada de 5 iteraciones descrita en la sección 3.4.1.

La diagonal principal de la matriz refleja los aciertos, mientras que el resto de celdas

reflejan las confusiones entre géneros (más información sobre matrices de confusión en la sección 3.4.2).

Concretamente, en la matriz de la figura 5.1 podemos ver que:

- El *DrumAndBass* (86/100) se clasifica muy bien y las confusiones se dan con el *ElectroHouse* (12/100) pero no se dan a la inversa. El hecho de que *DrumAndBass* se clasifique con mayor tasa de aciertos es razonable, ya que es el estilo que menos se parece a los demás si hablamos en términos del ritmo.
- El *Dubstep* (73/100) aunque no se clasifica tan bien tiene buen ratio de acierto, teniendo la mayor parte de confusiones con *HipHop* (12/100).
- El *HipHop* por su parte, se detecta peor que ninguno (38/100), pero muchas de las muestras pronosticadas erróneamente se clasifican como *Dubstep* (42/100).
- El *Dubstep* y el *HipHop* parecen tener una clara relación en estos resultados. Esta relación entre ambos es bastante razonable a ojos de un aficionado, ya que el ritmo y los tipos de sonidos de ambas son muy similares.

Respecto a los cuatro géneros restantes, se deduce que:

- *BigRoom* y *ElectroHouse* se parecen. Las confusiones se producen entre ellos en cantidades similares, tanto en un sentido como en el otro.
- *BigRoom* y *FutureHouse* también se relacionan, pero sólo hay confusión en sentido *BigRoom* hacia *FutureHouse*. Teniendo en cuenta que el *BigRoom* es un género más enérgico, esto parece indicar que las canciones más tranquilas del *BigRoom* pueden confundirse con *FutureHouse*, pero este último no llega a tener canciones tan potentes que puedan generar confusión con *BigRoom*.
- Por otro lado, vemos como el *Dance* tiene una confusión muy fuerte con *FutureHouse*, tanto en un sentido como en otro. Es verdad que en *Dance* encontramos canciones que pueden parecer bastante diferentes y de hecho se ve cómo es el que tiene más confusiones con todos los géneros, pero también destaca por tener canciones que son muy similares al *FutureHouse*.
- Además tanto *FutureHouse* como *BigRoom* son géneros cuya creación es muy reciente, concretamente en 2016 clasificaron alrededor de 500000 canciones[1]. Esto quiere decir que canciones que antes pertenecían a *Dance* o *ElectroHouse* ahora están en estos otros grupos por "características" que los harían claramente diferentes. Sin embargo,

[1]Noticia en *Beatport*: https://www.label-worx.com/news/beatport-adds-big-room-future-house-genre-classifications/7573/

debido a este origen a partir de otros géneros es normal que puedan tener aspectos en común.

Con esto, podemos decir que el *BigRoom*, el *ElectroHouse* y el *FutureHouse* son similares y el *Dance* se parece mucho a *FutureHouse*. Todos juntos parecen crear una jerarquía de manera que el *BigRoom* y el *Dance* están separados entre ellos, ya que el *BigRoom* sería el género más potente y el Dance el que menos, pero aún así, quedan conectados a través del *ElectroHouse* y el *FutureHouse*.

Árbol de decisión

Tras realizar las 5 iteraciones para la validación cruzada tenemos en realidad 5 árboles distintos de entrenamiento. Para obtener un árbol final de clasificación, lo que haremos será entrenar un árbol con el conjunto de canciones completo y este árbol será el que analizaremos.

En la figura 5.2 descubrimos que la característica más ampliamente usada es el BPM, tanto el calculado con *PyAudioAnalysis* como el conseguido con *Essentia* (más información sobre las librerías en la sección 2.4), por lo que vemos que la información rítmica es muy importante en la clasificación de música electrónica.

Hay una decisión que podemos considerar relevante, la partición por *49-MFCCs7std* entre *BigRoom* y *ElectroHouse*. Las demás particiones donde aparecen otras características, las consideramos redundantes ya que parten por canciones del mismo género (por ejemplo, *27-ChromaVector6m* parte en dos grupos de *FutureHouse*). Por el contrario, en algunos nodos finales vemos que hay numerosos elementos de varios géneros y esto da la sensación de que el árbol podría seguir entrenándose. Sin embargo, al utilizar la optimización con algoritmo genético explicada en la sección 4.3.3, en este árbol sabemos que es el que mejor generaliza. Es decir, otra configuración del árbol podría haber permitido que creciera aprendiendo más por esos nodos, pero lo que ganaría en entrenamiento, lo perdería en validación.

5.1.2. Bosque aleatorio

Tras el uso del árbol de decisión, para mejorar los resultados recurrimos al bosque aleatorio (más información en la sección 3.3.2). Éste combina varios árboles para aprender mejor el conjunto de datos y mejorar la tasa de aciertos.

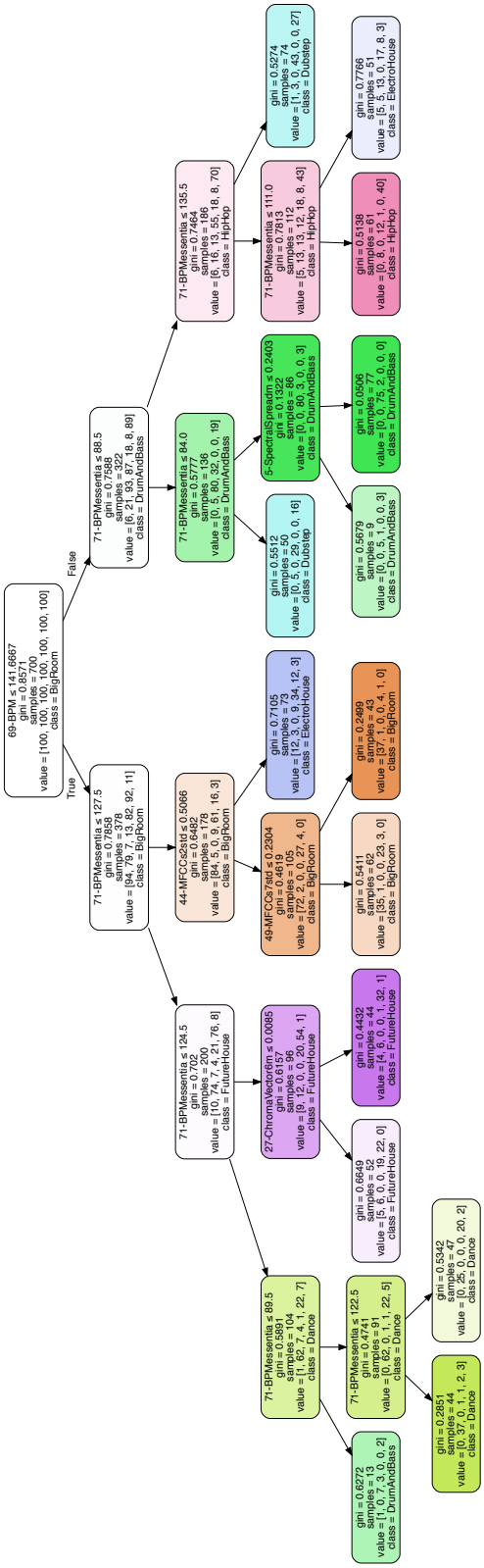


Figura 5.2: Árbol de decisión (7 géneros)

Matriz de confusión

Después de realizar la validación cruzada de 5 iteraciones (descrita en la sección 3.4.1) con el bosque (descrito en la sección 4.3.2) el promedio de las tasas de aciertos de cada conjunto de prueba nos da una media y desviación típica estándar:

Tasa de aciertos media: 0.64 $\pm$ 0.05

En la matriz 5.3 vemos que los resultados han mejorado, ya que todos los valores de la diagonal son más altos. La tasa de aciertos ha aumentado un 4 % (0.60 a 0.64) y se mantienen algunas confusiones aunque han reducido en intensidad respecto a la matriz de confusión del árbol de decisión (como se ve en la figura 5.1).

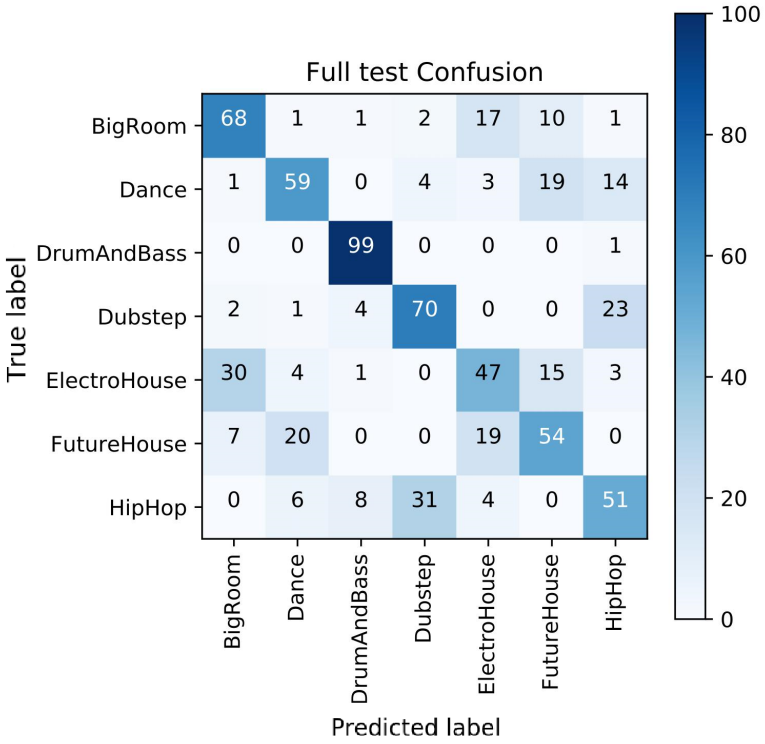


Figura 5.3: Matriz Confusión - Bosque (7 géneros)

- *DrumAndBass* sigue siendo el que mejor se clasifica (99/100). Nos hemos deshecho de la confusión que había con *ElectroHouse*, que no era esperable desde el punto de vista de la música, puesto que no tienen el mismo BPM.
- *Dubstep* y *HipHop* siguen teniendo una relación de similitud, siendo el *HipHop* el que más se confunde con *Dubstep* (31 canciones de *HipHop* clasificadas como *Dubstep*).
- Al igual que antes, *BigRoom*, *ElectroHouse* y *FutureHouse* forman un grupo con confusiones entre ellos.

- *Dance* y *FutureHouse* tienen aspectos en común. Además, el *Dance* se clasifica erróneamente como *HipHop* con una cantidad que consideramos no despreciable (14/100).

Las métricas de esta matriz de confusión se encuentran en la tabla de métricas en la siguiente sección. La explicación del significado de cada métrica la encontramos en la sección 3.4.2.

La tasa de aciertos con este algoritmo ha sido de un 64 %. En la figura 5.1.2 encontramos las mejores mediciones con el *DrumAndBass* que tiene una precisión del 88 % y un valor de F1 de 93 %. Por otra parte, el *ElectroHouse* tiene una precisión del 52 % y un valor de F1 del 49 %.

Tabla de métricas

Clase	TP	FN	TN	FP	Precisión	Sensibilidad	Valor F1
BigRoom	68	32	560	40	0.63	0.68	0.65
Dance:	59	41	568	32	0.65	0.59	0.62
DrumAndBass	99	1	586	14	0.88	0.99	0.93
Dubstep	70	30	563	37	0.65	0.7	0.68
ElectroHouse	47	53	557	43	0.52	0.47	0.49
FutureHouse	54	46	556	44	0.55	0.54	0.55
HipHop	51	49	558	42	0.55	0.51	0.53

La precisión mide la probabilidad de si el clasificador dice que algo pertenece a un género es ese género (probabilidad de que si digo *Dance* es *Dance*), mientras que la sensibilidad indica la probabilidad que se identifique correctamente las canciones de un género (probabilidad de que identifique todas las canciones *Dance* de una muestra). El valor de F1 es una media de los dos anteriores (precisión y sensibilidad) dando idea de lo bien que clasifica para cada género en concreto. Lo más destacable de las medidas de la tabla en la figura 5.1.2 es que el mejor clasificado es el *DrumAndBass*. Aunque la sensibilidad es muy alta (99 %) la precisión es peor (88 %), esto quiere decir que el *DrumAndBass* es detectado casi siempre y cuando introduzcamos una canción que originalmente es *DrumAndBass* en el clasificador, casi siempre pronosticará correctamente. Sin embargo, que una canción sea pronosticada como *DrumAndBass* no nos garantiza que sea realmente su género, aunque la precisión es alta y es bastante probable (0.88) que lo sea. Podemos ver otro dato curioso como el *Dance* que tiene más precisión que sensibilidad, implica que la probabilidad de detectar *Dance* es menor que la de asegurar que algo pronosticado como *Dance* sea *Dance*.

Grafo de confusiones

A la vista de la matriz de confusión, hemos mencionado que existen errores de clasificación que parecen coherentes entre diferentes géneros. No obstante, en la representación

dada por la matriz no se percibe tan claramente, por lo que buscamos una representación gráfica que lo exprese con mayor facilidad.

Hemos creado un grafo de confusión donde los nodos son los 7 géneros que manejamos en el experimento y las aristas que unen los géneros, indican si ha habido error de clasificación entre ellos. Las aristas tienen pesos que vienen dados por el número de canciones incorrectamente clasificadas. Este grafo es dirigido, es decir, las conexiones entre nodos no tienen porqué ser recíprocas.

En la matriz de la figura 5.3, hay 23 canciones del género *Dubstep* que se clasifican erróneamente como *HipHop*, por esto, en el grafo nos da una arista de *Dubstep* a *HipHop* con un peso de 23 (número de canciones clasificadas erróneamente). En el otro sentido, hay 31 canciones de *HipHop* que son clasificadas como *Dubstep*, formando otra arista de *HipHop* a *Dubstep* de peso 31.

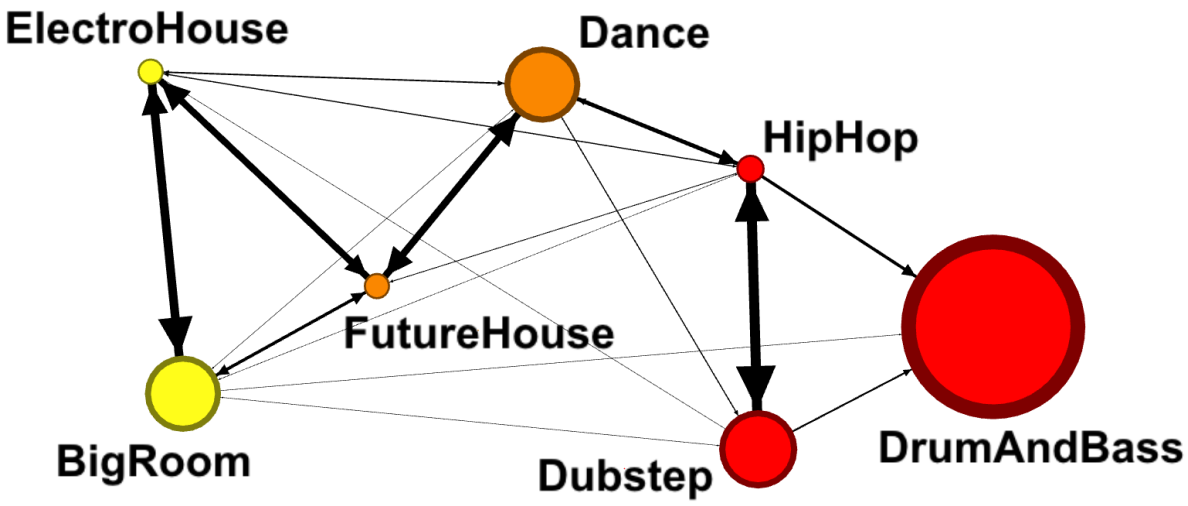


Figura 5.4: Grafo de confusiones (7 géneros)

En el grafo de la figura 5.4 los nodos tienen un tamaño inversamente proporcional a su grado con pesos. Dado que el grado viene dado por las aristas y los pesos asociados a éstas, los nodos más grandes son los géneros clasificados con mayor tasa de aciertos, mientras que los más pequeños tienen mayor confusión. Además el grosor de las aristas está en proporción del peso, siendo las flechas más gruesas las que muestran una confusión mayor hacia otro género. La posición de los nodos viene determinada por las aristas que tienen entre ellos actuando como si fueran "fuerzas". Los colores vienen determinados por su *clase de modularidad*, que agrupa elementos que están más relacionados entre ellos.

Podemos apreciar en la figura 5.4 que:

- El *DrumAndBass* es el género clasificado con mayor tasa de aciertos y aunque no

se confunde prácticamente con otros géneros, sí hay otros como el *HipHop, Dubstep*, que se confunden ligeramente con él.

- El *Dubstep, Dance y BigRoom* se clasifican bien pero tienen confusiones muy marcadas con otros géneros.
- El *Dubstep* y *HipHop* tienen una gran confusión que se percibe en las flechas, siendo éstas muy marcadas, que los unen.
- Entre los géneros *ElectroHouse, BigRoom, Dance* y *FutureHouse* hay relaciones que es necesario resaltar:
  - *BigRoom* se confunde con *ElectroHouse* y a su vez, el *ElectroHouse* y *FutureHouse* tienen una relación parecida. Pero entre *BigRoom* y *FutureHouse* no hay una relación tan marcada.
  - Por otro lado, el *Dance* se confunde con *FutureHouse* y otra vez nos encontramos que el *Dance* y el *ElectroHouse* no se confunden tanto.

Estos datos reflejan algo que intuitivamente ya pronosticábamos: las relaciones de confusión parecen coherentes, ya que dentro de un subgénero hay canciones más parecidas a otros. Esto ocurre porque los orígenes de ciertos subgéneros se han visto influenciados por otros.

En el grafo de la figura 5.4 además hay 3 grupos bastante bien diferenciados. Por un lado tenemos *DrumAndBass, Dubstep* y *HipHop*, en otro lado, *BigRoom* y *ElectroHouse* y en el medio, nos encontramos con *Dance* y *FutureHouse*. Observamos cómo los estilos tienen una relación de confusión con los próximos y se van uniendo mediante otros que tienen influencias de varios estilos. De esta manera, vemos que los que están muy alejados son bien distinguibles, mientras que los que tenemos en el centro, tienen más conexión con géneros vecinos y son más propensos a ser confundidos.

Dando una visión musical, el *DrumAndBass* es el que más se diferencia de los demás, ya que tiene un BPM muy marcado y unos sonidos muy metálicos. Aunque está bien clasificado, hay algunas pequeñas confusiones con el *Dubstep* y el *HipHop*. Esto lo consideramos natural, ya que tanto en *Dubstep* como en *HipHop* hay algunas canciones que tienen el BPM del *DrumAndBass* y además comparten el mismo tipo de sonidos. El *BigRoom* por su parte, nada tiene que ver con el *DrumAndBass*, y se parece poco al *Dubstep* y al *HipHop*, tanto por los sonidos, como por el ritmo o las voces que hay en la canción. No obstante, sí que tiene aspectos muy parecidos al *ElectroHouse* ya que es uno de los géneros a partir de los que se originó en 2016 (en este caso, nos referimos a la creación de la etiqueta en *Beatport*). De nuevo, refiriéndonos a la etiqueta, el *FutureHouse* por su parte, también se originó en 2016 y surgió como una mezcla de *Dance* y *ElectroHouse* que es justo lo que nos encontramos en el grafo. Y para acabar, en el caso del *Dance*, la mayor parte de las

canciones pueden ser escuchadas en distintos establecimientos por lo que en este género, podemos encontrar canciones muy dispares. Estas últimas están relacionadas con los demás géneros, ya que lo que es el *Dance* cambia con tendencias y adopta influencias de otros estilos. Por esta razón, cabe esperar que el *Dance* será uno de los géneros peor clasificados.

Importancia de las características

A pesar de no tener un árbol donde visualizar fácilmente cómo es la toma de decisiones en función de las características, este algoritmo permite extraer la importancia de éstas gracias a la librería utilizada a través del proceso que detallamos en la sección 3.3.2.

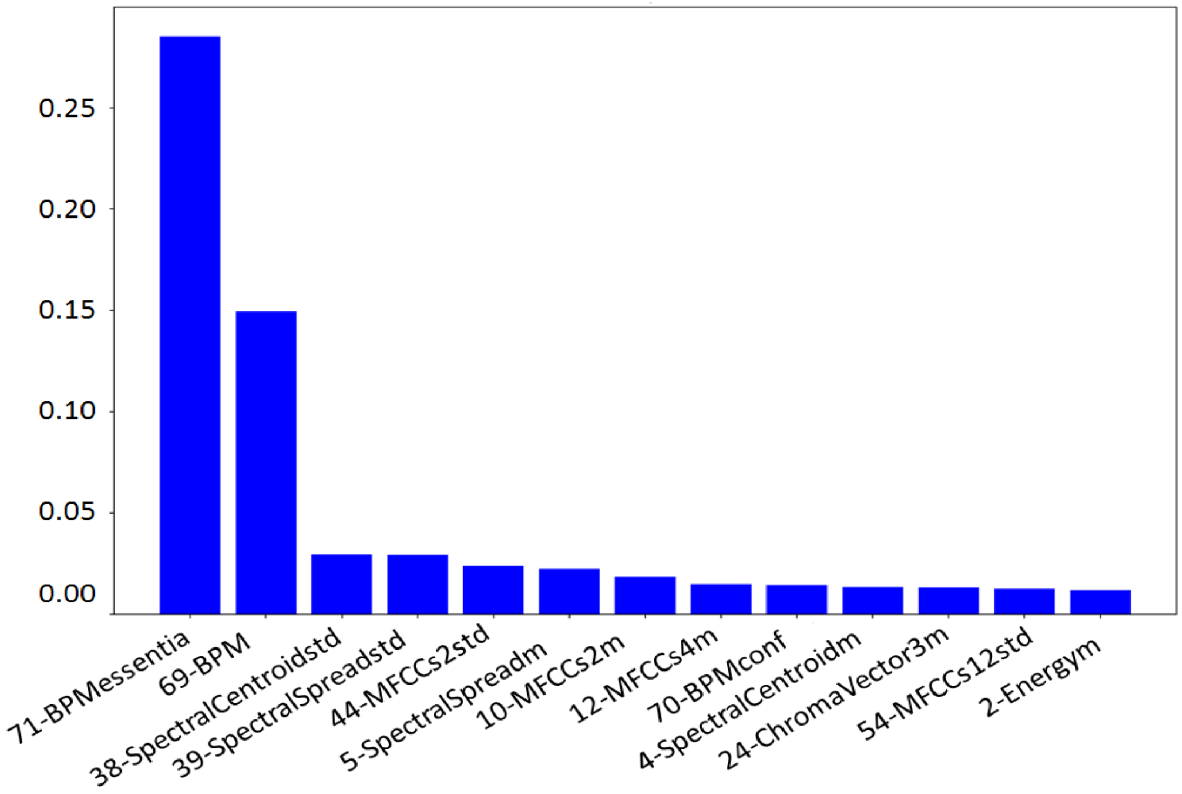


Figura 5.5: Características importantes - Bosque (7 géneros)

En la figura 5.5 se aprecia que la característica más relevante para la clasificación ha sido *71-BPMEssentia* seguida del *69-BPM*. Este resultado muestra lo mismo que en el árbol de decisión donde la característica más importante era el BPM.

Entre las siete características más importantes nos encontramos:

Las características 71 y 69 son las dos más importantes y miden el BPM. Esto deja claro que el ritmo de la canción es muy importante a la hora de clasificar y diferenciar la

música electrónica. La característica 70 aparece y puede parecer extraño, pero el valor de confianza del BPM es importante porque teniendo dos canciones del mismo BPM, si una tiene el ritmo muchísimo más marcado que otra es una diferencia muy a tener en cuenta. Si esto ocurre en conjuntos grandes a la hora de partirlos es una característica relevante (más información sobre el BPM en la sección 3.2.2).

Por un lado tenemos la 38 que es la desviación estándar del centroide espectral (*Spectral Centroid*). Esta medida mide cuánto varía el centro del espectro a lo largo de la canción, es decir, es una característica importante para ver cómo cambia el equilibrio de las frecuencias a lo largo de la canción. La presencia de voces o la aparición de sonidos graves o agudos repentinamente, produce que el *Spectral Centroid* cambie, por lo que esta característica está mostrando cambios claves que se dan en géneros en concreto (más información en la sección 3.2.2). La dispersión del espectro, *Spectral Spread* (más información de esta característica en la sección 3.2.2), es utilizada tanto en las características 39 como la 5. Esto quiere decir que diferencian las canciones dependiendo de dónde está la fuerza en el espectro de audio.

Los MFCC aproximan cómo los humanos escuchamos, por lo que el hecho de se encuentre entre las características más relevantes hace que pensemos que estos valores, que han sido ampliamente usados en la música, descartan muy bien entre los subgéneros. Más información sobre los MFCC en la sección 3.2.2.

5.2. Conjunto de datos 23 géneros

Una vez realizadas las pruebas con el conjunto de 7 géneros, utilizamos el conjunto completo. En este conjunto de datos, se analizan 23 géneros con los mismos criterios explicados en la sección anterior.

Los géneros analizados son: *BigRoom, Breaks, Dance, DeepHouse, DrumAndBass, Dubstep, ElectroHouse, ElectronicaDowntempo, FunkRAndB, FutureHouse, GlitchHop, HardcoreHardTechno, HardDance, HipHop, House, IndieDanceNuDisco, Minimal, ProgressiveHouse, PsyTrance, ReggaeDub, TechHouse, Techno* y *Trance* (descripciones de los géneros en el apéndice E y enlaces a las listas de estos géneros en la figura 4.1).

5.2.1. Árbol de decisión

Como mencionamos anteriormente, la principal ventaja de los árboles de decisión es su interpretación intuitiva (más información descriptiva del algoritmo en la sección 3.3.1). Sin embargo, el árbol tiene unas dimensiones considerables para los 23 géneros, por esta razón, hemos analizado las partes que hemos considerado significativas del árbol.

Matriz de confusión

Después de realizar la validación cruzada de 5 iteraciones (descrita en la sección 3.4.1) con el árbol (descrito en la sección 4.3.1), el promedio de las tasas de aciertos de cada conjunto de prueba nos da una media y desviación típica estándar:

Tasa de aciertos media: 0.42 $\pm$ 0.02

La matriz de confusión de 23 géneros se muestra en la figura 5.6. Sin embargo, no explicaremos esta matriz ni sacaremos grafo de confusión de ella ya que lo haremos con la que resulta de usar el bosque aleatorio, que es muy similar pero con mayor tasa de aciertos.

Árbol de decisión

El gráfico del árbol de decisión tiene unas dimensiones considerables por lo que no queda claro en este documento[2]. Pese a ello, hemos decidido aportar información sobre particiones relevantes:

- En la figura 5.7 observamos en detalle la partición del *Spectral Centroid* (más información en la sección 3.2.2).

  Se puede observar cómo esta característica parece un factor determinante para separar el *ReggaeDub* del *Dubstep*. Cuando el *4-SpectralCentroidm*, que es una media, es menor o igual a 0.2763, esa canción se clasifica en *ReggaeDub*. Por el contrario, si este es mayor, clasifica dentro de *Dubstep*. Por lo tanto, el *Dubstep* tiene un espectro de la señal de media más alto, lo que quiere decir que las energías de las frecuencias más graves son más fuertes que las del *ReggaeDub*.

- En la parte derecha de la figura 5.7, observamos cómo el BPM es una característica muy relevante tal y cómo queda reflejado en la sección anterior con el conjunto de 7 géneros. En esta partición, observamos cómo diferencia por el *BPMconf*, el valor de confianza del BPM entre *DrumAndBass* y *HardcoreHardTechno*. Esto puede parecer extraño, pero el tener un valor de confianza del BPM da mucha información sobre el ritmo en una canción, en este caso, el valor de confianza más alto clasifica como *DrumAndBass*, dando a entender que tiene el ritmo más marcado que el *HardcoreHardTechno*.

- En la figura 5.8 observamos varias particiones que resultan interesantes: En primer lugar, se ve que el *SpectralFlux* (más información sobre el *Spectral Flux en la sección 3.2.2*) es una característica importante y definitoria de *PsyTrance*, ya que separa

[2] Disponible en https://github.com/Caparrini/pyGenreClf/blob/master/Examples/FinalTree.pdf

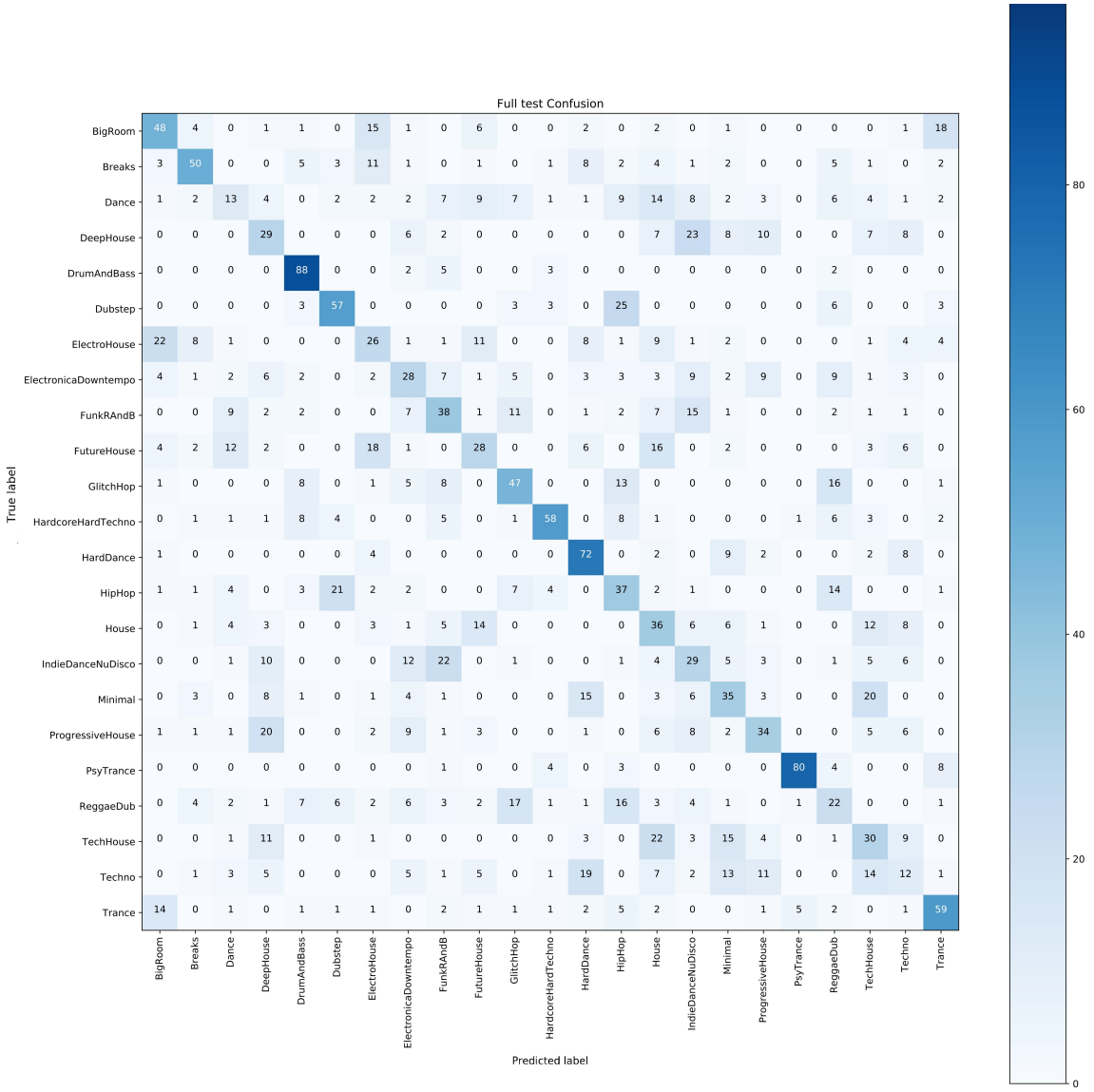
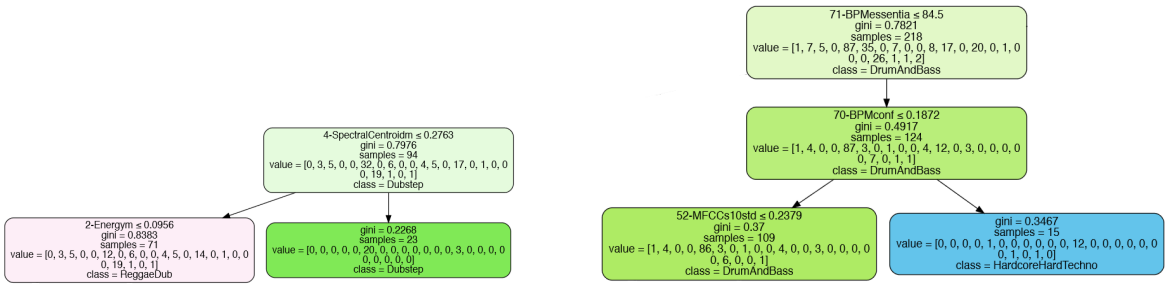


Figura 5.6: Matriz Confusión - Árbol de decisión (23 géneros)

un gran número de canciones de *PsyTrance* (85) de un conjunto variado. Esta característica mide la rapidez con la que cambia la potencia del espectro, dando a entender que este género tiene sonidos muy potentes como se aprecia al escucharlo. Además si seguimos bajando por esa rama, nos encontramos otra característica, *44-MFCCs2std* que distingue entre *HardcoreHardTechno* y *Dubstep*. Encontramos el *4-SpectralCentroidm* (más información en la sección 3.2.2) cuyo valor determina una buena partición entre *HipHop* y *Dubstep*. Anteriormente ya habíamos usado esta misma característica para separar canciones de *Dubstep* y *ReggaeDub*, y ahora vuelve a



(a) Detalle *Spectral Centroid*. (b) Detalle BPM.

Figura 5.7: Detalle árbol de decisión (23 géneros)

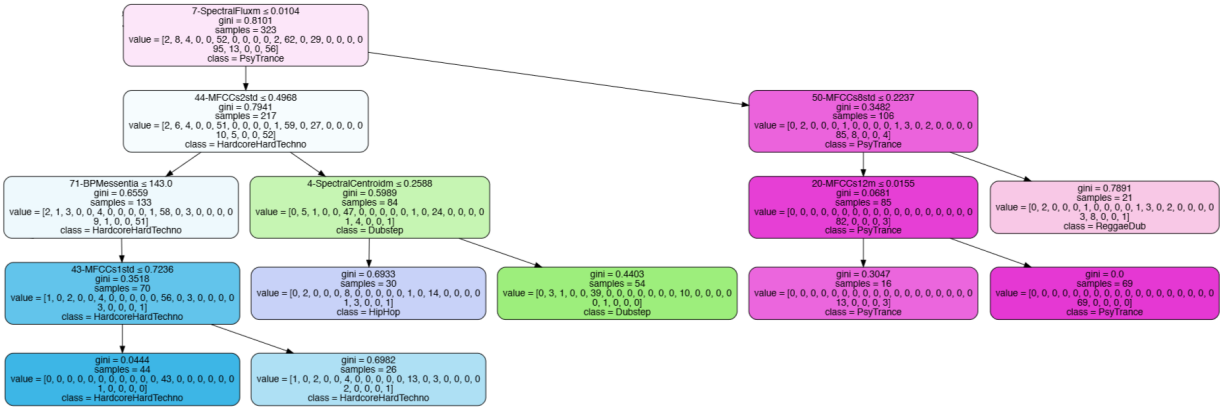


Figura 5.8: Detalle árbol de decisión (*SpectralFlux* y MFCC).

hacerlo pero con *HipHop*. Vemos claro que el *Dubstep* tiene el centroide más alto lo cuál es debido a sus bajos más energéticos.

5.2.2. Bosque aleatorio

Utilizamos el bosque aleatorio para el conjunto de 23 géneros igual que lo hicimos con el conjunto de 7 géneros para generar nuestro clasificador final.

Matriz de confusión

Después de realizar la validación cruzada de 5 iteraciones (descrita en la sección 3.4.1) con el bosque (descrito en la sección 4.3.2), el promedio de las tasas de aciertos de cada conjunto de prueba nos da una media y desviación típica estándar:

Tasa de aciertos media: 0.54 $\pm$ 0.01

Teniendo en cuenta que el número de géneros es 23 (que son bastantes clases) y que los

géneros tienen fronteras difusas entre ellos, parece un buen resultado. Este valor es el que está previsto que consigamos con el conjunto de validación.

En la matriz de confusión de la figura 5.9 se ven géneros que forman agrupaciones y otros que claramente se confunden entre ellos. Lo primero que nos llama la atención de la matriz es que las confusiones que teníamos con tan sólo 7 géneros se ven también tras completar hasta 23 géneros:

- *DrumAndBass* sigue clasificándose con una tasa de aciertos muy alta (96 %).
- Las dos confusiones más marcadas de *BigRoom* son *ElectroHouse* y *FutureHouse*.
- El *ElectroHouse* se confunde con *BigRoom* y *FutureHouse*. En esta ocasión, se añade una confusión nueva con *Breaks*.
- El *Dance* se confunde con *FutureHouse* y además aparece una nueva confusión con *House*.
- El *Dubstep* sigue confundiéndose con *HipHop*.
- El *HipHop* se confunde con *Dubstep*, aunque añadimos nuevas confusiones con *Glitch-Hop* y *ReggaeDub*.

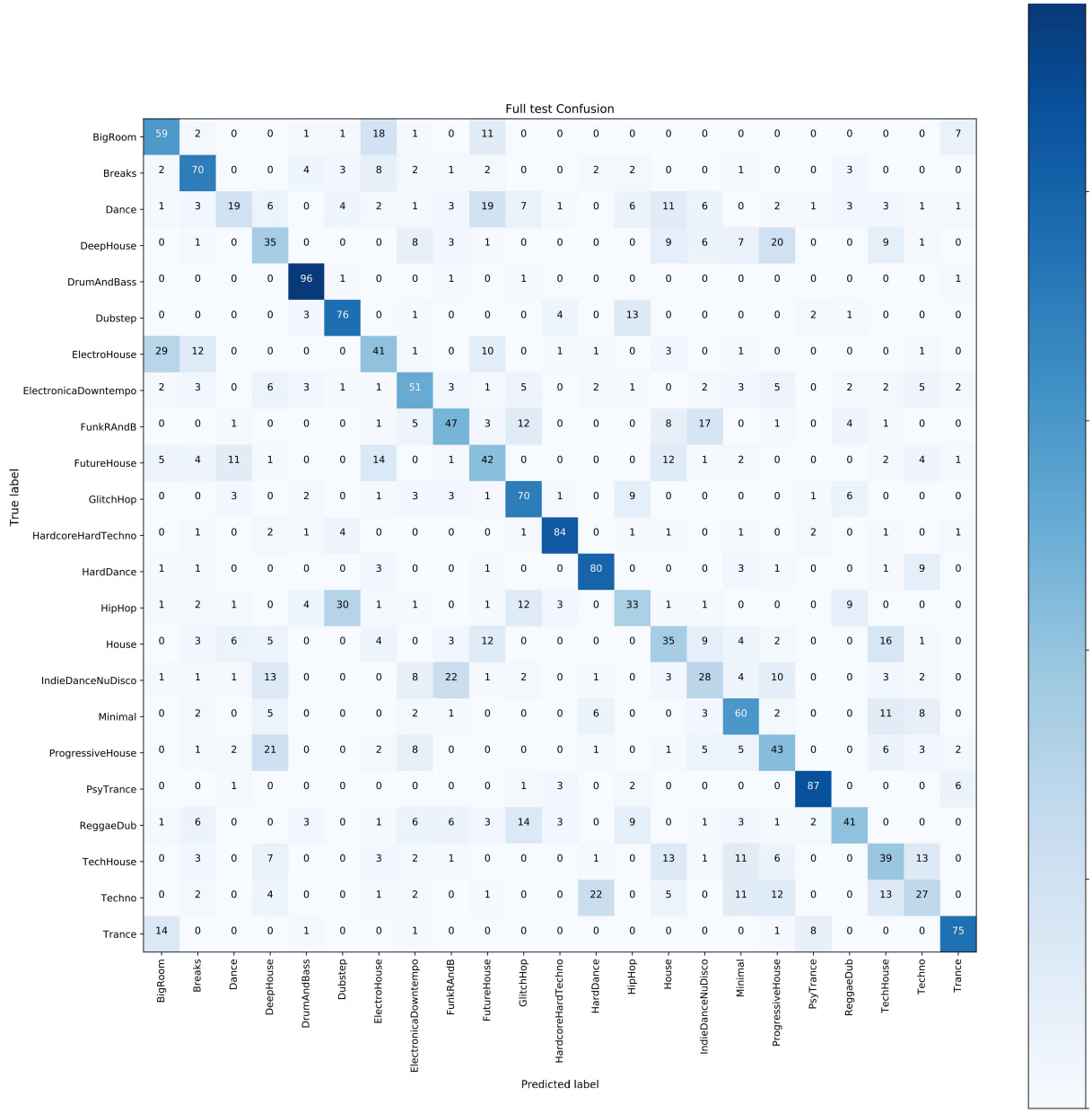


Figura 5.9: Matriz Confusión - Bosque (23 géneros)

Tabla de métricas

Clase	TP	FN	TN	FP	Precisión	Sensibilidad	Valor F1
BigRoom	59	41	2143	57	0.51	0.59	0.55
Breaks	70	30	2153	47	0.6	0.7	0.65
Dance	19	81	2174	26	0.42	0.19	0.26
DeepHouse	35	65	2130	70	0.33	0.35	0.34
DrumAndBass	96	4	2178	22	0.81	0.96	0.88
Dubstep	76	24	2156	44	0.63	0.76	0.69
ElectroHouse	41	59	2140	60	0.41	0.41	0.41
ElectronicaDowntempo	51	49	2148	52	0.5	0.51	0.5
FunkRAndB	47	53	2152	48	0.49	0.47	0.48
FutureHouse	42	58	2133	67	0.39	0.42	0.4
GlitchHop	70	30	2145	55	0.56	0.7	0.62
HardcoreHardTechno	84	16	2184	16	0.84	0.84	0.84
HardDance	80	20	2164	36	0.69	0.8	0.74
HipHop	33	67	2157	43	0.43	0.33	0.38
House	35	65	2133	67	0.34	0.35	0.35
IndieDanceNuDisco	28	72	2148	52	0.35	0.28	0.31
Minimal	60	40	2144	56	0.52	0.6	0.56
ProgressiveHouse	43	57	2137	63	0.41	0.43	0.42
PsyTrance	87	13	2184	16	0.84	0.87	0.86
ReggaeDub	41	59	2172	28	0.59	0.41	0.49
TechHouse	39	61	2132	68	0.36	0.39	0.38
Techno	27	73	2152	48	0.36	0.27	0.31
Trance	75	25	2179	21	0.78	0.75	0.77

En la tabla 5.2.2 anterior, tenemos las métricas de la matriz de confusión. Las medidas y su explicación se encuentran en la sección 3.4.2. Como mencionamos anteriormente, precisiones altas indican que las predicciones son más certeras, es decir, que realmente el género pronosticado es correcto, mientras que sensibilidades más altas indican que el clasificador va a detectar el género.

Al igual que en la tabla de métricas del bosque aleatorio de 7 géneros, el mejor clasificado es el *DrumAndBass*. Aunque la sensibilidad es muy alta (0.96), la precisión es peor (0.81). Al igual que en el conjunto de datos anterior, esto quiere decir que el *DrumAndBass* es detectado casi siempre y cuando introduzcamos una canción que originalmente es *DrumAndBass* en el clasificador, casi siempre la pronosticará correctamente.

El *Dance* tiene una sensibilidad del 19 % mientras que tiene una precisión del 42 %, siendo el valor de la sensibilidad menos de la mitad de la precisión. Esto quiere decir que

al clasificador le cuesta mucho detectar el *Dance* y es posible que una canción *Dance* la clasifique como de otros subgéneros.

El *PsyTrance* y el *HardcoreHardTechno* están equilibrados con valores F1 altos. Tienen unos valores similares de sensibilidad y precisión, por lo que son unos géneros que se detectan y pronostican con exactitud.

Los géneros *IndieDanceNuDisco*, *House*, *HipHop*, *Techno* y *DeepHouse* son los que salen peor parados en la clasificación por lo que consideramos que puede deberse a que son géneros menos característicos y que tienen similitudes muy fuertes con otros géneros.

Grafo de confusión

Aunque la matriz de confusión nos da mucha información, es difícil ver y explicar las relaciones de confusión, por lo que pasamos a realizar un grafo de confusiones análogo al realizado anteriormente con el experimento de 7 géneros. Cabe destacar que en este grafo se han filtrado las aristas de grado 1 y 2 que muestran unas confusiones mínimas del 1 %, para conseguir una representación más clara.

En la figura 5.10 se observan los géneros mejor clasificados en los nodos que aparecen más grandes. Comparando con el grafo que teníamos de 7 géneros de la figura 5.4, vemos cómo los géneros que estaban en el mismo grupo dentro del grafo de 7, están también en el mismo grupo dentro del grafo de 23.

- En color rojo nos encontramos con: *HardcoreHardTechno, Dubstep, HipHop, GlitchHop, ReggaeDub* y *DrumAndBass*.

  En el grafo de 7 ya teníamos 3 de estos géneros, *DrumAndBass*, *Dubstep*, *HipHop*. Nos encontramos que los subgéneros que se encuentran más alejados del centro de este grupo son más rápidos en ritmo, siendo más lento el *ReggaeDub* y el *GlitchHop*. Estos géneros tienen sonidos vibrantes y metálicos muy similares.

- En amarillo: *PsyTrance, Trance, BigRoom, ElectroHouse* y *Breaks*.

  En el conjunto de datos anterior, teníamos *BigRoom* y *ElectroHouse*. Todos estos géneros son característicos de festivales de música electrónica ya que son rápidos, energéticos con partes muy claras y nítidas y voces muy editadas. Como ya habíamos comentado anteriormente, el *BigRoom* es un género que apareció nuevo en 2016 y la estrecha relación con *ElectroHouse* da una clara muestra de dónde ha surgido. Además vemos que *Breaks* es el género que está más próximo al grupo anterior, esto es debido a que los sonidos que tiene y el ritmo son parecidos a los géneros más cercanos a este otro grupo.

- En morado: *Dance, House* y *FutureHouse*.

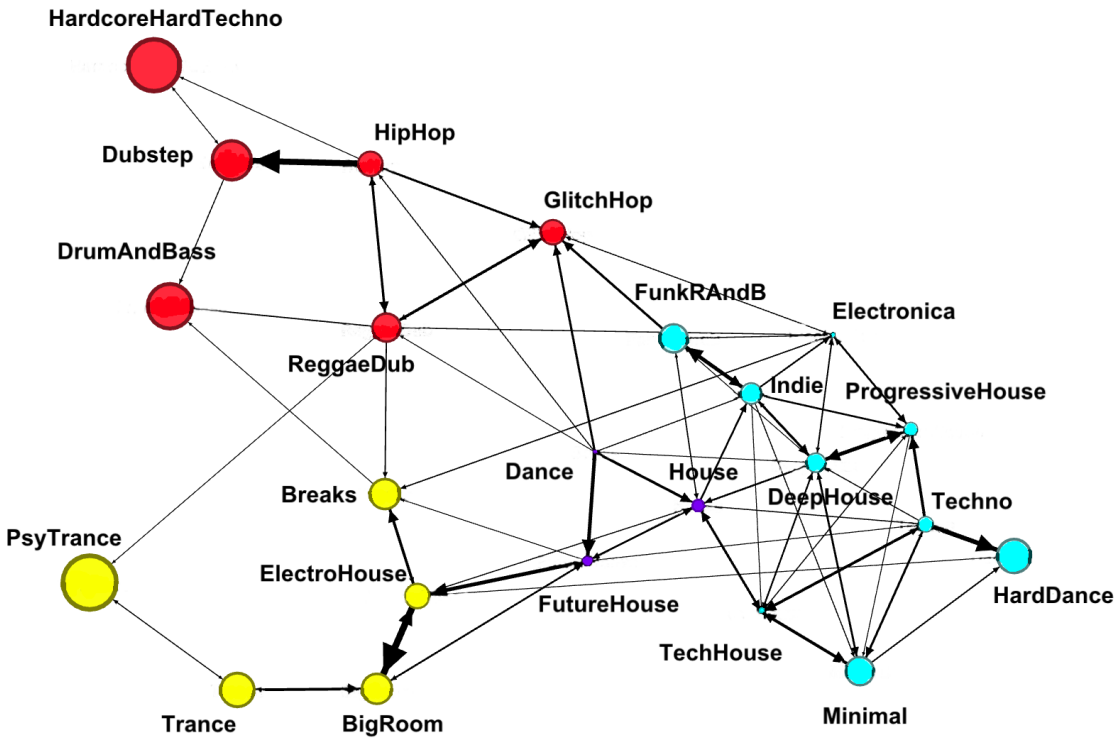


Figura 5.10: Grafo de confusión (K iteraciones)

En el primer conjunto analizado, *Dance* y *FutureHouse* pertenecían a un mismo grupo. Este grupo está muy céntrico dentro del grafo y tiene mucho sentido ya que el *Dance* cambia con el paso del tiempo debido a que es el subgénero de la electrónica que actualmente está de moda, y *FutureHouse* es otro género de nueva creación parecido tanto al *ElectroHouse* como a los géneros que pertenecen al grupo rojo.

- En color azul: *FunkRAndB, IndieDanceNuDisco, DeepHouse, ElectronicaDowntempo, ProgressiveHouse, Techno, HardDance, Minimal* y *TechHouse*.

  Según nuestra apreciación, este grupo podría haberse dividido en dos más, tiene tanto los géneros más puros del *Techno* y del *Minimal*, muy relacionados entre ellos porque son muy parecidos, como algunos que son más melódicos como el *ProgressiveHouse* y *DeepHouse*.

En una vista general del grafo, el grupo rojo formado por géneros con sonidos muy metálicos, el grupo amarillo podríamos considerarlo como música de "festival", el grupo

morado es la música de "moda" y el grupo azul claro la música *House* y *Techno* más difícil de escuchar junto con otra más melódica como *ProgressiveHouse* y *DeepHouse*.

Se puede observar que los mejores clasificados son el *HardcoreHardTechno*, el *DrumAndBass* y el *PsyTrance*. Estos géneros son bastante particulares y muy descriptivos, al escuchar las listas de canciones se nota que los temas son más parecidos entre ellos y hay menos variedad.

Entre los géneros que crean más confusión, nos encontramos *ElectronicaDowntempo, Dance, FutureHouse, TechHouse, Techno* o *ProgressiveHouse*.

Además cabe resaltar que:

- *HipHop* tiene una gran confusión con *Dubstep* y se confunde ligeramente con el *Glitch-Hop*. Estos tres géneros tienen temas muy parecidos y comparten ritmos y sonidos.
- *ElectroHouse* se confunde mucho con *BigRoom* y *Trance*, siendo este último con el que produce una confusión más débil. Son las típicas canciones que se pueden escuchar en festivales y tienen muchos aspectos en común.
- *FunkRAndB* parece tener algún tipo de relación común con el *IndieDanceNuDisco*. Son géneros que que parecen *canciones de discoteca de los 80* pero con sonidos y tendencias actuales.
- *Techno* se equivoca mucho con el *HardDance* y tiene relación con el *TechHouse*.
- *TechHouse* tiene muchas confusiones con otros géneros como el *Techno* o el *House* entre otros.
- *ProgressiveHouse* tiene una relación bastante fuerte con *DeepHouse*.

Importancia de las características

Una vez más encontramos que los valores del BPM han sido las características más relevantes, seguida de otras como el *SpectralFluxm*, el *MFCCs2std* o *SpectralSpreadstd*.

Entre las siete características más importantes nos encontramos:

Las características 71, 69 y 70 son las más importantes y miden el BPM y su confianza. Dejando claro que el ritmo de la canción es muy importante a la hora de clasificar y diferenciar música electrónica como ya llevamos viendo en secciones anteriores.

La característica 7, el *Spectral Flux*, es la media de la rapidez con la que cambia la potencia del espectro. Vimos en el árbol que esta característica por ejemplo nos diferenciaba el género *PsyTrance*, por lo que consideramos que esta característica al aparecer tantas veces, es una característica muy relevante para la clasificación de subgéneros musicales. La

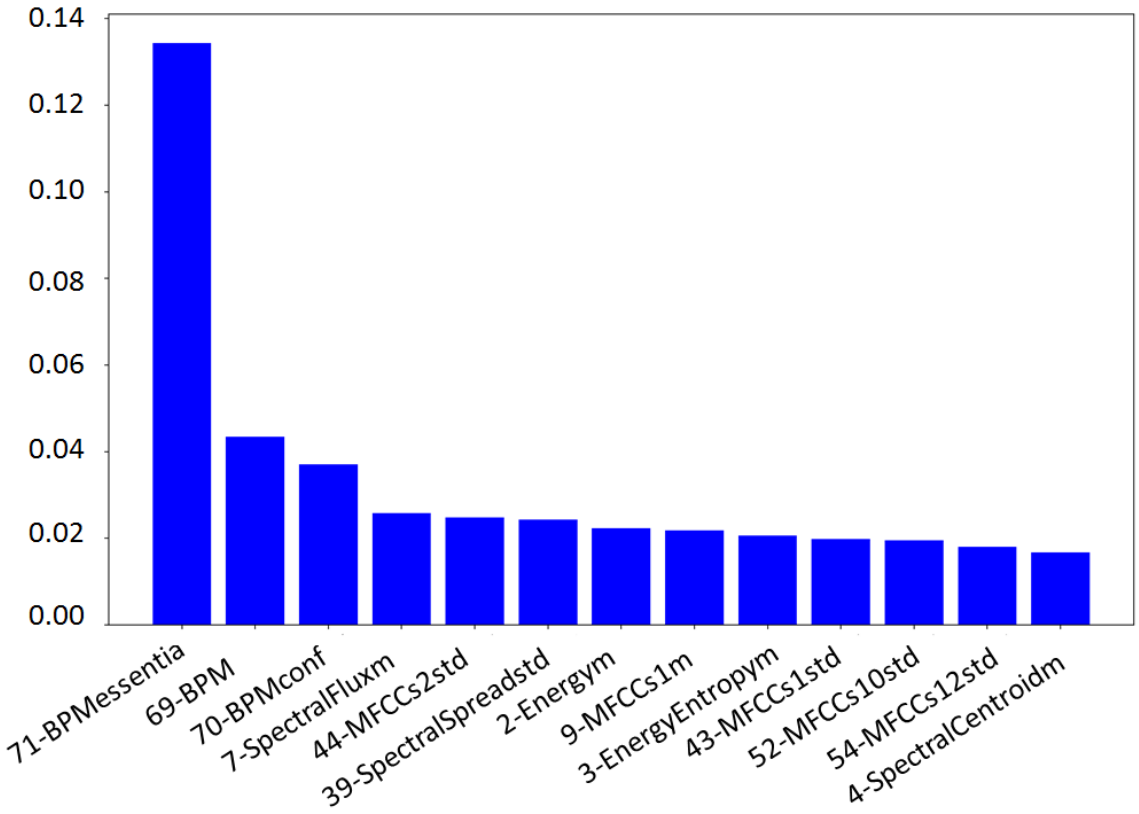


Figura 5.11: Características importantes - Bosque (23 géneros)

característica 39 que es la desviación estándar del *Spectral Spread* que mide la dispersión del espectro. Como dijimos anteriormente, esto quiere decir que diferencia las canciones dependiendo de dónde esta la fuerza en el espectro de audio. Más información sobre estas características en las sección 3.2.2.

Igual que ha ocurrido con el bosque aleatorio en el otro conjunto, aquí también aparece los valores de MFCC. Más información sobre estas medidas en la sección 3.2.2.

5.3. Conjunto de datos 23 géneros - Validación final

Una vez que el clasificador de 23 géneros ha sido entrenado, decidimos probarlo con un conjunto de datos distinto al que entrenamos en un principio (más detalles sobre este conjunto en la sección 4.1.3). Éste tiene 60 canciones de cada uno de los 23 géneros (1380 canciones).

Tasa de aciertos media (validación): 0.49

El valor de la tasa de aciertos media es menor que el esperado que era de 0.54 ± 0.01. Nos parece un valor bastante aceptable, ya que las confusiones siguen manteniéndose (como veremos más adelante de esta sección). En la matriz de confusión de la figura 5.12 hemos normalizado los resultados para que sean porcentajes, de manera que el valor de cada casilla es el porcentaje de las 60 canciones de cada género. Los géneros que clasificaba bien en un principio se siguen manteniendo con este conjunto: *DrumAndBass* (68 %), *HardDance* (87 %), *HardcoreHardTechno* (78 %), *PsyTrance* (87 %) y *Trance* (78 %).

- Aunque se visualice en la matriz que el *DrumAndBass* tiene en su diagonal un número de aciertos menor respecto al experimento anterior, es una buena clasificación porque la mayoría de las clasificadas erróneamente, sólo las introduce en *Dubstep* (62 %). Este cambio puede deberse a que de las canciones con las que se entrenó el conjunto seis meses atrás respecto a las de validación, el *DrumAndBass* haya evolucionado de forma que ahora tiene un parecido mayor con *Dubstep*.

- Al igual que ha pasado anteriormente, *Dance* es un género que no se reconoce especialmente con un 12 %, pero era algo que ya podíamos prever, al igual que con el *IndieDanceNuDisco*, *HipHop* y *House*.

- El *Techno* se clasifica muy mal y aparece con una confusión muy grande con *ProgressiveHouse* que antes no existía. Nuestra intuición es que en los últimos meses ha cambiado el *Techno* y está más influenciado por el *ProgressiveHouse*.

- Otro género que cambia mucho es el *ElectroHouse*. De un 41 % de tasa de acierto que era esperable tenemos sólo un 18 %.

Visualizando el grafo correspondiente en la figura 5.13, en la matriz de confusión encontrábamos los mismos grupos que vimos anteriormente pero hemos "hilado más fino" para ver agrupaciones más pequeñas dentro de esos grupos.

Estos grupos, salvando algunas relaciones, tienen una distribución muy similar, por lo que pensamos que los cambios en este grafo indican los movimientos en los últimos seis meses debidos a nuevas tendencias. Un ejemplo de esto, es la confusión entre el *DrumAndBass* y *Dubstep*.

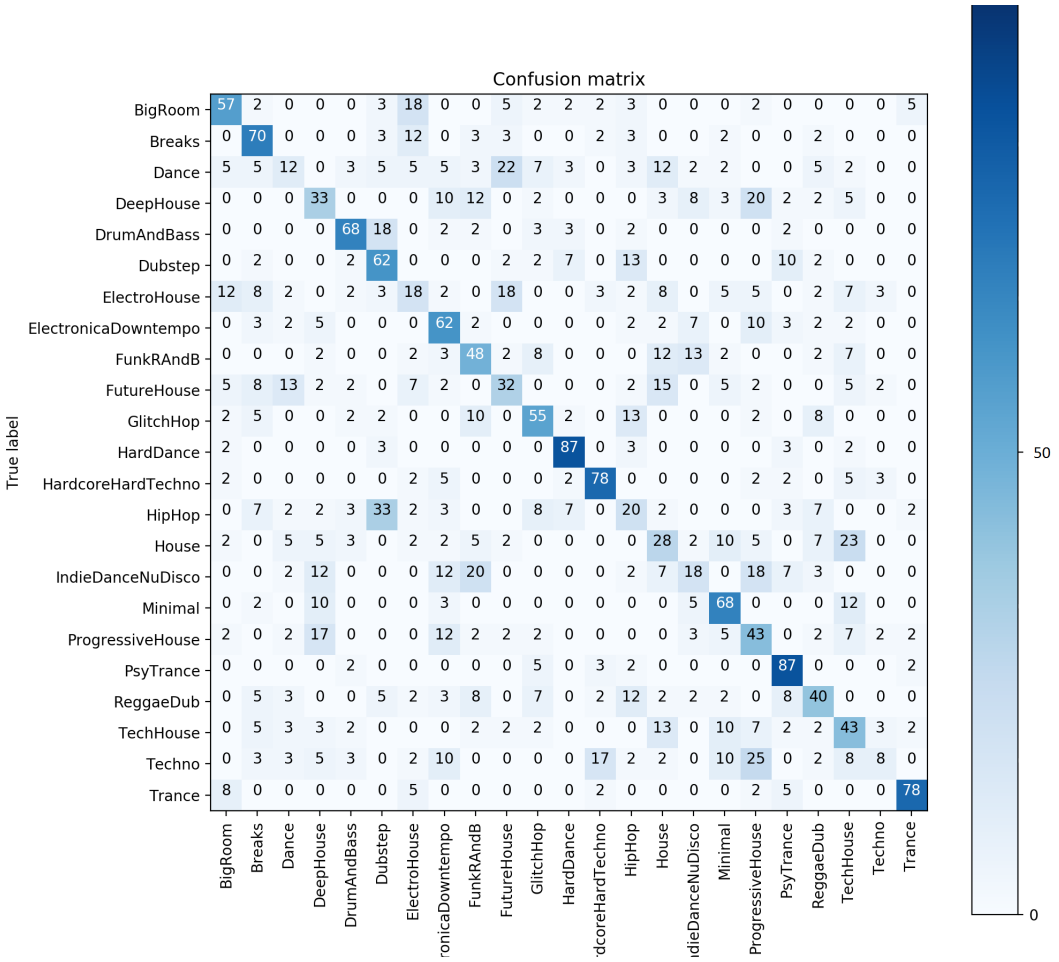


Figura 5.12: Matriz Confusión - Validación

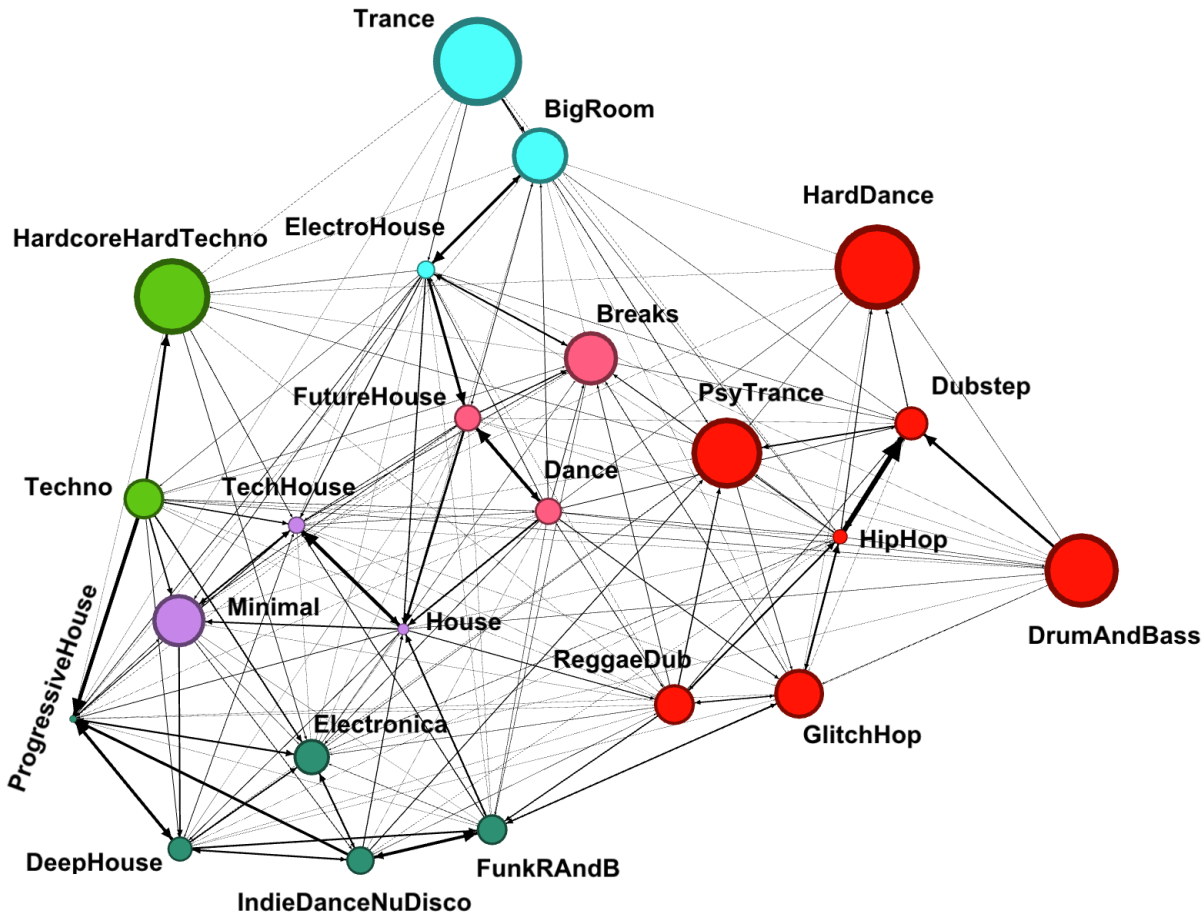


Figura 5.13: Grafo de confusión (validación)

5.4. Experimento humano-máquina

Una vez analizados los resultados viendo las grandes confusiones entre géneros, nos planteamos si esta clasificación automática tenía sentido o no. Decidimos probar el conjunto de datos de los 7 géneros con personas (más información sobre este conjunto en la sección 4.1.1). Nuestro principal objetivo era ver si tanto las buenas predicciones como los géneros que aparecían con más confusiones se mantenían entre las clasificaciones de las personas. El diseño del experimento se encuentra en la sección 4.4 aunque hacemos un pequeño resumen a continuación.

Los usuarios tenían que clasificar 70 canciones en 7 géneros distintos, sin especificar el número de canciones por género. Al igual que la máquina, se les ofrecía un conjunto de entrenamiento. Este conjunto estaba formado por otras canciones distintas a las que había que clasificar, que pertenecían a cada género, un total de 10 canciones por género, para que pudieran detectar similitudes entre canciones de un mismo conjunto.

Se realizó el experimento a 16 usuarios, de edades comprendidas entre 16 y 26 años. No todos tenían conocimientos musicales aunque todos eran oyentes de música (no necesariamente electrónica). El tiempo que les llevó completar el experimento ha oscilado entre una y dos horas. En la figura 5.14 podemos observar un gráfico de barras donde las primeras 16 columnas (de la 0 a la 15) son las tasas de acierto de los 16 participantes ordenadas de menor a mayor. En la columna con índice 16 tenemos la media de todos los experimentos junto con una línea verde vertical que muestra la desviación típica estándar de esta media, y en la columna 17 la tasa de aciertos considerando a todos los participantes como una "comunidad aleatoria", esto último lo explicaremos más adelante. La tasa media de aciertos de todos los participantes fue $\mathbf{0.48} \pm 0.1$.

El resultado obtenido es bajo comparado con el del bosque aleatorio sobre estos mismo géneros de la sección 5.1.2 donde se consigue una tasa de aciertos de $\mathbf{0.64} \pm 0.05$. Sin embargo, antes de hacer una comparación directa entre la clasificación automática y las personas, es necesario tener en cuenta más información:

En la matriz 5.15 se muestran los resultados agregados de las 16 personas que realizaron el experimento, pero en porcentaje para facilitar su comprensión. Cada persona clasificó 70 canciones y todos ellos escucharon y clasificaron las mismas. Por tanto:

16 Personas $\times$ 70 Canciones = 1120 Clasificaciones.

En el grafo de confusiones 5.15 vemos que hay bastantes confusiones, pero las más marcadas son las que ya hemos visto en el experimento de 7 géneros con el bosque aleatorio:

- *Dubstep* y *HipHop*.
- *BigRoom* y *ElectroHouse*.
- *Dance* y *FutureHouse*.

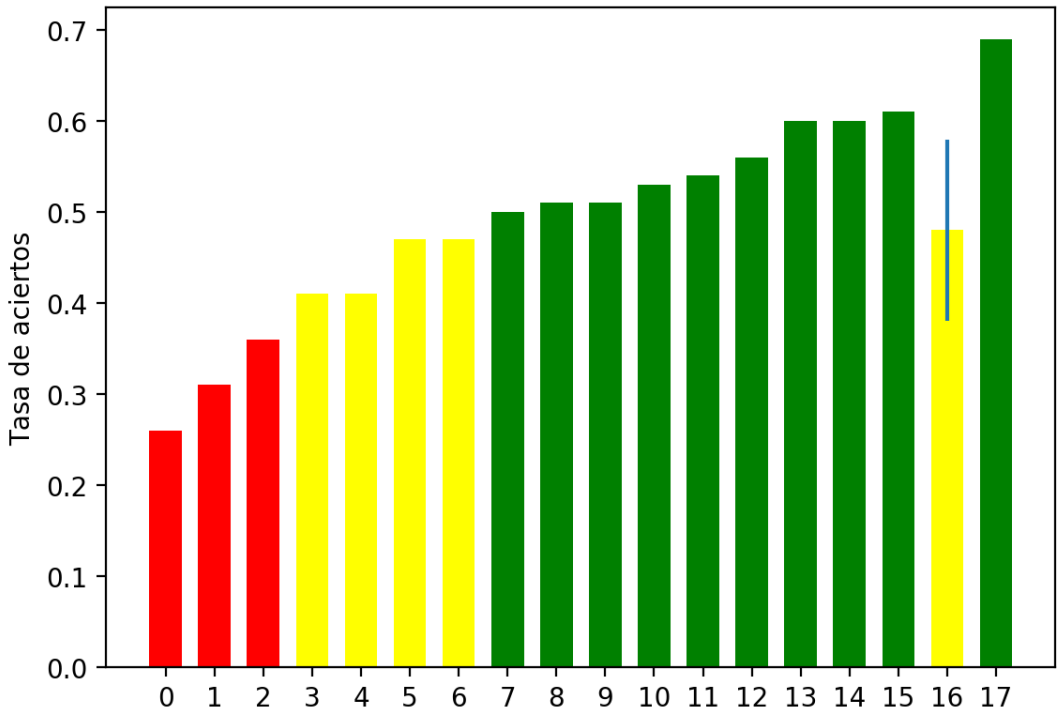
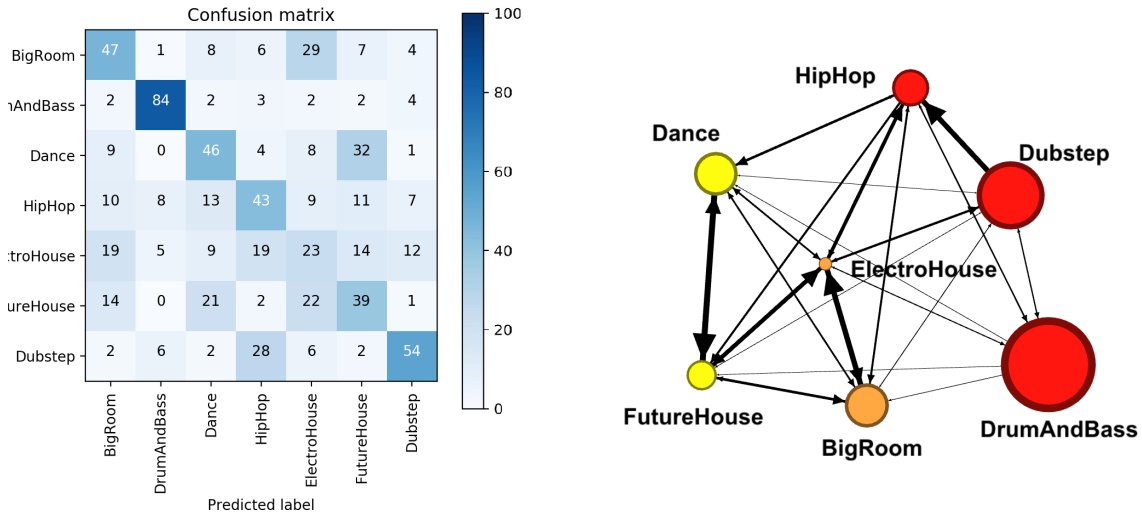


Figura 5.14: Gráfico de barras - Resultados clasificación con personas



(a) Matriz de confusión

(b) Grafo de confusión

Figura 5.15: Experimento humano-máquina

- *FutureHouse* y *ElectroHouse*.

En este caso, además de los anteriores contamos una nueva que no habíamos tenido antes, una confusión de *ElectroHouse* a *HipHop*.

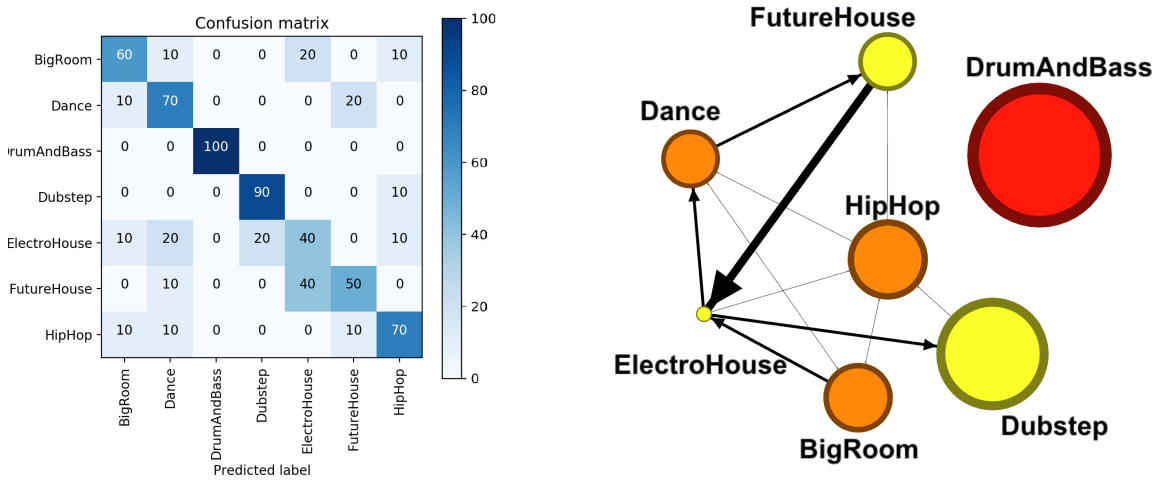
En definitiva, el grafo esta lleno de confusiones entre todos los géneros, excepto en el caso del *DrumAndBass* que se clasifica bastante bien y es realmente razonable porque es el que menos se parece a los demás, tanto rítmicamente como en los sonidos que utiliza. Por este motivo para exponer de forma clara los resultados, pasamos a considerar cada una de las 16 personas como un árbol de decisión y crear una "comunidad aleatoria" parecida al funcionamiento de un bosque aleatorio. Vamos a explicarlo:

- Cada usuario ha clasificado según su criterio, es decir, cada uno ha tenido en cuenta unas "características" que le han hecho tomar la decisión. Sin embargo, dado que los 16 participantes han hecho el experimento individualmente, este criterio es completamente personal y probablemente distinto para cada uno, aunque no descartamos que en algún caso, hayan podido tener en cuenta aspectos comunes.
- Tomando como referencia la agregación que hace el bosque aleatorio de los árboles de decisión que contiene, aplicamos esta agregación análoga a las personas. De esta manera tenemos una "comunidad aleatoria" de 16 personas en la que cada uno ha pronosticado la clase de cada una de las 70 canciones. Consideramos las predicciones de todos ellos sobre las mismas 70 canciones como votaciones y para cada una de ellas, asignamos como predicción de la "comunidad" el género más votado.

En la matriz de confusión 5.16 observamos cómo el resultado no sólo se ve más claro, sino que mejora considerablemente. En esta matriz los valores están presentados en tantos por ciento para facilitar las comparaciones con el resto de resultados. La tasa de aciertos de nuestra "comunidad aleatoria" es de **0.69**. Este resultado ya sí que es comparable al resultado del bosque aleatorio: $\mathbf{0.64 \pm 0.05}$.

En el grafo de confusiones hecho sobre la matriz 5.16, las confusiones son bastante pequeñas salvo una muy destacable entre *FutureHouse* y *ElectroHouse*. Estas confusiones son similares a las realizadas por el bosque aleatorio:

- *BigRoom* con *ElectroHouse*
- *Dance* con *FutureHouse*
- *ElectroHouse* con *Dance*
- *FutureHouse* con *ElectroHouse* (más notable)
- *ElectroHouse* con *Dubstep*. Aunque no se ve tan clara con 7 géneros, al hacer el experimento de 23 se observa claramente esta confusión en la matriz de confusión.



(a) Matriz de confusión

(b) Grafo de confusión

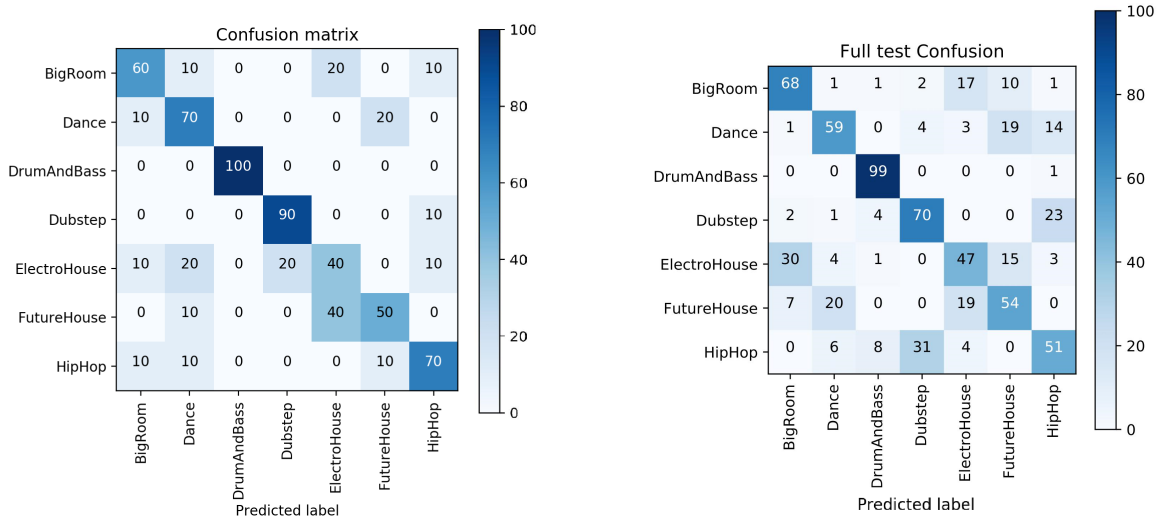
Figura 5.16: Experimento humano-máquina (bosque aleatorio)

A lo largo del experimento se le ofrecía a cada persona un folio donde podía apuntar etiquetas de cada género que les sirviera como referencia a la hora de clasificar. A modo de curiosidad, presentamos algunas etiquetas relevantes para los usuarios (muchas de ellas, han sido comunes en varios usuarios):

- **Género A (*BigRoom*)** - *rápido y lento, piano, technoHouse, house duro, electrónica, animado, discoteca.*
- **Género B (*ElectroHouse*)** - *más duro que el anterior, pum pum, techno, continuación de la otra?, remix, voces retocadas.*
- **Género C (*DrumAndBass*)** - *drumAndBass, dubstep, psicodélico*
- **Género D (*Dubstep*)** - *dubstep, música videojuego, GTA.*
- **Género E (*HipHop*)** - *drumAndBass, HipHop, dubstep, trap, dubstep más lento y repetitivo, bullicioso.*
- **Género F (*Dance*)** - *sin distorsión de la voz, música tienda, más comercial, tropical.*
- **Género G (*FutureHouse*)** - *deep, house, house repetitivo, música gym, movida, notas muy altas.*

En la figura 5.17 podemos ver a la izquierda la matriz de confusión de las personas consideradas como "comunidad aleatoria" y a la derecha la matriz de confusión del bosque aleatorio.

Vemos como hay confusiones que son comunes para personas y el bosque aleatorio:



(a) Matriz personas (b) Matriz bosque aleatorio

Figura 5.17: Experimento humano-máquina (bosque aleatorio)

- *BigRoom* - *ElectroHouse*
- *Dance* - *FutureHouse*
- *Dubstep* - *HipHop*
- *FutureHouse* - *ElectroHouse*

Las tasas de aciertos (0.64, 0.69) y confusiones son razonablemente similares en cantidad, de manera que la capacidad de abstracción del algoritmo parece razonablemente buena. Sin embargo, el experimento con personas y el experimento con el bosque aleatorio han tenido cantidades distintas de datos, tanto para entrenar como para probar, lo cuál hace injusta realmente la comparación. Cabe destacar que para que las personas hubieran podido realizar el experimento en las mismas condiciones que la máquina, sólo para escuchar las canciones habrían tenido que emplear:

$$\frac{2 \text{ minutos } \times 7 \text{ géneros } \times 100 \text{ canciones}}{60 \text{ minutos}} = 23{,}33 \text{ horas}$$

Por lo tanto, consideramos que una clasificación automática sería muy factible ya que emplearía mucho menos tiempo tanto en entrenarse como en clasificar una gran cantidad de canciones de distintos géneros consiguiendo resultados similares.

Capítulo 6

Conclusiones y trabajo futuro

La clasificación de música en géneros es un proceso subjetivo que se ve afectado por diversos factores. Aunque existen trabajos relacionados con la clasificación automática de géneros hasta donde nosotros sabemos, no se ha tratado la clasificación automática de los subgéneros propuestos de música electrónica. La música electrónica en particular, aparentemente es un género que se presta mucho a este tipo de clasificación automática debido a su auge en las últimas décadas.

Nuestro trabajo se ha centrado en la extracción de características y el aprendizaje automático. A lo largo de la carrera, no hemos tenido ninguna asignatura que trabajase en profundidad temas relacionados con la música, por lo que esto nos ha llevado a investigar en este terreno.

A través de las pruebas realizadas a lo largo de este proyecto, hemos comprobado que sin duda, el aprendizaje automático es una gran herramienta que ha resultado muy útil para nuestro propósito. Queremos resaltar que particularmente en la clasificación automática en música, creemos que no debería primar el conseguir una tasa de aciertos muy alta, sino que el resultado sea coherente. Con esto queremos decir que si el fallo en la confusión conduce a otro género que realmente es parecido, el clasificador puede seguir siendo muy útil. En nuestro caso, una clasificación con una tasa de aciertos en torno al 50 % cuando contamos con 23 géneros posibles no es una mala clasificación. De hecho algunas confusiones tienen sentido, porque los subgéneros, y más tal y como los usa *Beatport*, son cambiantes y en algunos casos muy subjetivos. Además, hay subgéneros concretos que se relacionan muy estrechamente con otros, lo cual hace que no sea exactamente un error de clasificación o uno excesivamente relevante. Llegados a este punto, queda la duda de si al partir de un conjunto de datos subjetivo, los errores de clasificación se deben al software o a la propia clasificación propuesta por *Beatport*.

A la hora del estudio y la presentación de los resultados, consideramos que los grafos de confusión, que no han sido muy utilizados en trabajos de este tipo, son muy útiles e informativos. Gracias a ellos, podemos visualizar de forma más clara las relaciones entre distintos géneros, cuya información estaba presente de manera menos evidente en las matrices de confusión.

Algo que vale la pena mencionar es el resultado bastante satisfactorio que hemos obtenido al probar la clasificación con usuarios. Aunque la máquina y el usuario utilicen formas

de clasificar completamente diferentes, los resultados son comparables. Por ejemplo, *DrumAndBass* es un género fácilmente detectable por las dos clasificaciones mientras que el *ElectroHouse*, en la mayoría de los casos, parece confuso de clasificar. Nuestro experimento ha sido una pequeña prueba y sólo podemos hacer conjeturas, pero da la impresión de que lo que es confuso para el algoritmo lo es también para las personas. Esto refrenda la utilidad de la clasificación automática, la cual es instantánea y ofrece una muy buena primera aproximación. También quiere decir que las propiedades extraídas aunque no tienen una explicación clara a partir de conceptos musicales, son perfectamente útiles para clasificar subgéneros de música electrónica. Este método de clasificación automática podría usarse por productores musicales para ver el estilo de sus canciones, por las tiendas y sellos musicales para crear un primer filtro y facilitar el trabajo de clasificación o para recomendaciones musicales tanto para consumidores habituales de música como para profesionales DJs.

A continuación pasamos a detallar las posibles líneas a seguir que han quedado abiertas en este trabajo:

- Hacer un seguimiento de las 100 mejores canciones de cada subgénero que vayan apareciendo en un futuro próximo y repetir las técnicas empleadas para descubrir si las relaciones y confusiones entre subgéneros se mantienen en el tiempo o evolucionan según las tendencias.
- Probar distintas técnicas de aprendizaje automático que han utilizado otros autores más actuales, filtrando por las características que en este estudio han sido más importantes, y comprobar cuál es el algoritmo que da más tasa de aciertos.
- Usar redes de *Kohonen*, que sirven para hacer *clustering* y que ofrecen una representación en 2D de las canciones, cuya taxonomía se puede comparar con la inferida por la clasificación.
- Utilizar una representación simbólica del audio como MIDI como alternativa a la extracción de características de la señal de audio y comprobar si los resultados son similares.
- Investigar sobre el tiempo mínimo necesario de una muestra para su clasificación y comprobar la viabilidad de un reconocedor de género en tiempo real.
- Diseñar una aplicación web o móvil que informe sobre el género de una canción o los distintos géneros que detecta a lo largo de la canción usando 2 minutos.
- Investigar más a fondo la comparación entre la clasificación automática y la realizada por personas.

Apéndice A

Introduction

Over the last few years, the rise of digitized data belonging to different fields has led to the creation of automatic analysis tools for this content in order to facilitate search and access to them. In this context, music and tracks are affected by this phenomenon. Music needs to be classified, even before the digital era, and this led to the creation of musical genres. Genres are used to facilitate the user's experience, as they aim to group similar tracks, which makes easier finding music liked which makes easier for the user to find music of their taste.

Genre classification is considered a subjective process that is affected by different factors such as culture, geographical location, time period or market trends (Scaringella et al., 2006). For this reason, we have found the drawback that there is not one and only classification of genres. The present work proposes the design of a software electronic music sub-genre classification.

Music genre classification

Musical genres are categories that have emerged due to the need of classifying music collections and establishing similarities between musicians and compositions. Despite this, boundaries between genres are still fuzzy as well as their definition, making the problem of classification not at all and sub-genres even more difficult to tell apart.

Nowadays there is a wide range of genre taxonomies in music. These different ways to classify can be seen online and virtual shops related to music, for instance, on web sites well-known websites as *YouTube* (*America*, *Reggae*, *Electronic*...), *last.fm* (*Electronic*, *Indie*, *Folk*...) or *Spotify* (*America*, *Pop*, *Trending*...).

Precedents of musical genre classification

According to Pachet and Cazaly (2000), the music industry has always created their taxonomies of genres to meet their own needs. They claim that there has been no effort has been made to unify. There is the idea that this unification would be quite interesting, as it would clearly show the differences between these taxonomies. In addition, music up to now

has been classified according to retailers' needs, that is to say, music shops as *Fnac* (French company specializing in the sale of electronics, computers, photography equipment, books, music, and video) that pose a music classification aimed directly to consumers: a first level, with musical categories (*classical music, jazz, rock,* etc.), a second level with more specific *subcategories*("*Hard Rock*" within "*rock*"), a third level using an alphabetical sort by artists and a fourth level sorted by albums. The authors also highlight a classification by marketing (promotional/best hits) or by theme ("Rock", "best collection of love songs"...).

They claim that multiple taxonomies have appeared on the Internet, designed to help users navigate through catalogs of music in a similar way to record stores but with more detail. After analyzing several relevant websites such as ***Amazon***, the results clearly show that there is not much consensus in these classifications. Therefore, on the basis that there are many music-related data that can be exploited at a software level, they claim that there must be some kind of coherence must be in automatic musical genre classification and that this should be achieved.

Automatic recognition of musical genre

The initial hypothesis when it comes to automatic recognition of a genre, is the existence of shared features between musical works belonging to the same. Through these features, which can be extracted automatically from audio, we aim to generate a predictive model to recognize their genre The discipline which covers this field is called MIR for its acronym in English, *Music Information Retrieval Information Retrieval (musical)*. The definition of MIR[1] is:

> *recovery of musical information (MIR) is the interdisciplinary science which was responsible for retrieving information from music. MIR is a small but growing field of research with many applications in real world. Those involved in MIR may be specialists in musicology, psychology, academic study of music, signal processing, machine learning, or some combination of these.*

One of the main approaches to musical information retrieval consists in analyzing audio features: retrieve of musical information from audio (*Audio Information Retrieval* mentioned in one paper by Tzanetakis and Cook (2000a)). This retrieve of musical information is obtained from audio, through processing of the signal of the same. Retrieval takes place through processing of the signal, thus, features related to timbre, rhythm and pitch can be extracted (Tzanetakis and Cook, 2002).

[1]Music information retrieval. (s.f.). About Wikipedia. Recuperado el 19 de mayo de 2017 de https://en.wikipedia.org/wiki/Music_information_retrieval

Musical features

Below, we show a brief description of the most relevant musical features.

Pitch features: melody and harmony

Pitch is directly related to sound frequency, with low frequencies originating low sounds and high ones originating high sounds. This term is linked to the concepts of melody and harmony; melody can be defined as a succession of sounds perceived as an only entity, while harmony is the study of pitch simultaneity and chords, which can be implicit in the piece of music or not. Consequently, harmonic and melodic analysis have been used to study musical structures.

Rhythm features

The majority of the authors refer to the rhythm as a measurement of the regular time, that is to say, it is a form of succession and toggle a series of sounds that are repeated periodically in a certain time interval.

Timbral features

Timbral features makes two sounds with the same tone and volume sound different. The features that characterize the bell are global, that is to say, integrate the information of all the instruments at the same time along with the voice. In a more intuitive, the ringer would be the equivalent of the audio to the texture in the color.

These characteristics are qualities of the music that we perceive directly the people and with the music experts describe the songs. The computers do not work directly with them, but they use properties that are extracted from the digital signal from audio files such as ".wav". The properties, although they are related to the musical characteristics, do not have a perfect correspondence with them. In addition, although the extract, if what we want is to perform an automatic classification, we need a software or algorithm perform the process made by the people. At this point, we introduce the automatic programming that will fulfill this function to search for patterns and similarities between these features to perform the classification in an automated way.

A.0.1. Machine learning

Machine learning is a branch of artificial intelligence that aims to create algorithms that allow the machines to learn. Expressed in a more practical way, these algorithms are

able to generalize behaviors and recognize patterns from a information provided initially in the form of example. This is therefore a process of induction, where, on the basis of individual cases provided, you get a generalization.

Many authors have used these techniques to processes of classification, induction or learning. According to Nilsson (1996), there are several reasons to use automatic learning. Among them, there is that there are very large databases that can hide some kind of relationship between its variables and that it would be interesting to know.

The supervised machine learning algorithms are based on sets properly classified, and seek in them similar characteristics or patterns of similarity, that is to say, the classes are assigned in the data set. This is the type of sort that interests us, as we seek to reproduce the music classification made by people.

The use of machine learning techniques applied to the music (MIR) are an area of growing interest in the past few years, and there are scientific communities dedicated to it.

Comunity MIR (Music Information Retrieval)

Applications of the MIR is the categorization of music according to their genre. In this context, it is worth mentioning MIREX (Music Information Retrieval Evaluation eXchange). MIREX is an organization that is trying to unify efforts and published every year several musical data sets for classification of music. In this way, a comparison of algorithms each year exploring the various machine learning techniques presented by the members belonging to the same.

In the same line is ismir (International Society for Music Information Retrieval). ISMIR is a non-profit organization that organizes the Conference ISMIR. Takes place annually and is one of the most important research forums of the world in processing, search and access to the data related to the music[2].

A.1. Motivation

The music clasification of genres is subjective, as has been exposed, it depends on cultural aspects, temporary and personal. However, if it is determined a set that has been classified previously by users who are true fans of this kind of music, it would be crazy to think that there is something intrinsic to those songs that characterized?

The answer at first may seem self-evident, even to fall into the thinking: "*without having any sort of musical knowledge, there is something that distinguishes the classical music of*

[2]ISMIR 2016 conference was held on 11 August last year in New York with the organization of the university of New York in conjunction with the University of Columbia (US private university located in Upper Manhattan, New York). MIREX 2016 was part of this conference.

rock music. It seems clear". There are current studies that have wanted to draw conclusions about this issue (see section **??**) and to emphasize that it is very important from a good set of data properly labeled by professionals. Through the processing of music is intended to automate the classification of genera and more precise borders between them. Being this division between genres, on many occasions, something controversial or complex, what happens to the subgenera?

In our work we will discuss precisely the classification between genres within a genre in constant evolution in recent years thanks to the digitalization of music. We present the distinction between genres of electronic music through the implementation of an automatic classifier.

Our main reference will be ***Beatport***[3], a *online* of electronic music that has all songs classified in sub-genres so that each song, only is labeled in a single genre.

In the following interview with representatives of *Beatport* in August 2016[4] speaks of his way of classifying genre equality. Recognize that the genera are subjective and his interest is not to give a classification imposed by them, but a to provide its customers access to the music. In it, they speak of two new genera, which added by popular demand. In addition, they argue that if the public called in a certain way to a set of songs is easier for them to find new music in the shop if it is classified that way.

> *DJTT: What's happening with Beatport's new approach to genres?*
>
> *Beatport: This is something that the public has been asking for, and hands up, we've heard them. The important thing to say is that there is not one correct definition of a genre. We just want to make finding good music easier for our customers.*
>
> *DJTT: Is there a time when you think genres will become redundant, like Glitch-Hop?*
>
> *Beatport: I think DJs will always need genres as a way of finding new music. We have 25,000 new releases most weeks and genres help to narrow down the tracks you'll need to search through. We have to stay relevant in terms of what people are playing. We will also have to retire genres and bring in new ones. The main thing is staying ahead of the trends, and working with the artists and labels in those areas.*

The evolution of technology has facilitated the creation of electronic music and as a result occurs more and more varied. For this reason, an automatic classification of the new music that appears every week, it would be an interesting proposal and a way to save efforts.

[3] https://beatport.com

[4] http://djtechtools.com/2016/08/12/beatports-re-approach-to-genre-tagging/

A.2. Objectives

The present work seeks to classify sub-genres of electronic music shown on *Beatport* website. In addition, the aim is to verify whether there are real intrinsic qualities in the songs that make them belong to these sub-genres. *Beatport* classification is based upon its users preferences, in an attempt to objectify them. The classifier will work with properties from the audio signal that are related to musical properties but are not equivalent, so it is interesting to see to what extent automatic and human classification render comparable results. To this end, we set the following objectives:

- To implement a classifier of sub-genres of electronic music.
- To check the validity of the classifier.
- To study possible confusion between genres. To analyze whether this is due our particular approach (non-inclusion of descriptive variables, bias of the algorithm) or on the other hand, is due to the fact that some genres are hard to objectify.
- To compare automatic classification with the classification made by people to check whether human and automatic classificatiosn are similar or not.

A.3. Structure of the document

Now we will briefly outline the contents of this work:

1. **Introduction**

   It contextualizes the contents of this work in general and justifies our motivation for the development of an automatic classifier of sub-genres of electronic music. It also provides the objectives to accomplish throughout this work.

2. **State of the art**

   This section shows you the current status of automatic genre classification from the point of view of audio feature extraction, machine learning, data sets, and the current state of this field.

3. **Conceptual foundations**

   It offers a more descriptive view of the concepts applied in the project. This section details the data set, the features used (definition and extraction process), machine learning algorithms or the software.

4. **Development**

   This section sets out the most technical aspects of our work. Details are provided such as chosen genres, length of each audio track, the size of the chosen window to extract audio samples, the composition of the feature vector, the validation criteria of the algorithms chosen among others.

5. **Results**

   Most relevant test results, together with their analysis, are shown in this section.

6. **Conclusions and future work**

   Open lines and proposals of the work to continue and a summary with final conclusions from this final degree project.

7. **Appendices**

   Presented in this section: **Introduction, Conclusions and future work** and **Contributions to the project** in accordance with regulations for the final degree project for academic year 2016/2017. **Additional information of the work.** Details of this work such as test of the classifier with GTZAN and descriptions of chosen sub-genres.

Apéndice B

Conclusions and future work

Genre classification is a subjective process that is affected by different factors. Although there are works related to automatic classification of genres, as far as we know, there has not been an automatic classification of the electronic music sub-genres included in our proposal. Electronic music in particular appears to be a genre that lends itself to this type automatic classification due to its boom the last few decades.

Our work has focused on feature extraction and machine learning. Throughout our degree, we have not had the chance to further study issues related to music, so this has led us to investigate in this field.

Through tests performed over the course of this project, we have found that without a doubt, machine learning is a great tool that has successfully served our purpose. We want to stress that particularly in automatic classification of music, we believe that the hit rate does not necessarily have to be very high as long as the results are consistent. By this, we imply that even if an error leads to another genre which is quite similar, the classifier can remain very useful. In our case, a classification with a hit rate of around 50 % when we count among 23 possible genres is not a bad classification, in addition to some confusion makes sense, because sub-genres, especially the way *Beatport* manages them. In addition, there are specific sub-genres that relate very closely with others, which makes it not exactly a classification error or one excessively a very relevant one. At this point, and starting from a set of subjective data, doubt remains as to whether classification errors are due to software or to *Beatport* classification proposal.

When it comes to study and presentation of results, we believe that confusion graphs, which have not been widely used in work of this kind of work, are very useful and enlightening. Thanks to them, we can see more clearly relationships between different genres, whose information was present in confusion matrices, but in a less intuitive way.

Something that is worth mentioning is the quite satisfactory result that we have obtained when testing classification by users. Even though machine and user use completely different ways to classify, the results are comparable. For example, *DrumAndBass* is easily detectable in both classifications while *ElectroHouse*, in most cases, seems confusing to classify. Our experiment has been a small test and we can only guess, , but there is a hint that if something is confusing for the algorithm, it is for humans as well. This confirms the usefulness of automatic classification, which is instantaneous and offers a very good first

approximation. It also means that properties extracted may well not have an explanation based on musical concepts, but they are perfectly valid to classify genres of electronic music. This method of automatic classification could be used by music producers to to know the style of their songs, by shops and record labels to create a first filter therefore facilitate work of classification or for music recommendations for both regular consumers of music and for professional DJs.

Some of the potential lines of research that have remained open:

- To make a follow-up of the 100 best songs from each genre the near future and to replicate the techniques used to discover whether relations and confusion between genres are kept remain the same or rather they evolve.
- To try different machine learning techniques that have been used by current authors, filtering the most important features in this study have been more important, andto check which algorithm provides the highest hit rate.
- To use *Kohonen* networks for *clustering* and thus having a 2D representation of songs, whose taxonomy can be compared with the mininum to the inferred by classification.
- To use a symbolic representation of audio as MIDI as an alternative to extraction of features from audio signal and to check whether the results are similar.
- To investigate on minimum time necessary of a sample for classification and to test feasibility of a genre recognizer in real time.
- To design a web or mobile application which provides the genre or genres of a song using a 2-minute sample.
- To further investigate the comparison between automatic classification and and the one carried out by people.

Apéndice C

Contribuciones al proyecto

C.1. Antonio Caparrini López

El trabajo realizado y las aportaciones, tanto mías como de mi compañera Laura, han sido compartidas en todas las partes relativas al proyecto desde que lo empezamos. A continuación paso a comentar mis aportaciones más destacables. Originalmente buscamos realizar algún tipo de proyecto relacionado con el reconocimiento de piezas musicales, aunque al no haber estudiado nada relacionado con audio digital en la carrera no partíamos de una base firme.

Participé junto con mi compañera en la investigación sobre los trabajos de reconocimiento musical. Empecé probando librerías que simulaban el comportamiento de *Shazam*. Pero al final llegamos al trabajo de George Tzanetakis sobre reconocimiento automático de géneros musicales que fue lo el que nos impulsó en este campo. Estudié la documentación de *Marsyas* y probé su uso para realizar el reconocimiento automático de géneros, y aunque fue el inició que nos impulsó en este proyecto, la herramienta la acabamos desechando.

Como músico y aficionado a la música electrónica planteé el uso de los subgéneros de ésta y el conjunto de datos en concreto, las 100 mejores canciones de cada género en *Beatport*. A la hora de conseguir el conjunto de datos colaboré con la creación del programa que recuperaba el audio (2 minutos) de las 2,300 canciones de *Beatport*. Lo usamos para conseguir tanto el conjunto de canciones inicial como el de validación final.

A pesar de no haber tratado de análisis de audio digital ni aprendizaje automático en la carrera, dediqué esfuerzo en investigar y aprender sobre el tema. Para el estudio de los fundamentos básicos de aprendizaje automático seguí un curso *online* en *Udacity* sobre *Introduction to machine learning* que fue muy útil y fácil de seguir. El curso trataba todos los ejemplos con código en *Python* y la librería *scikit learn* que tiene implementaciones y herramientas para el uso de aprendizaje automático, y al comprobar la facilidad y lo bien documentado que estaba en Internet decidimos utilizarlo.

Necesitábamos extraer las características que habíamos encontrado en los trabajos anteriores sobre reconocimiento de género, por ello, participé buscando librerías de audio para *Python*. Encontramos gran variedad y *pyAudioAnalysis* que fue la que fundamentalmente utilizamos ya que es libre, actual y muy completa. Dediqué tiempo a la comprensión de la herramienta y lo que internamente realizaba, para aportar los fundamentos teóricos sobre

los que se apoya este proyecto.

Participé del proceso cíclico de extraer características, entrenar los clasificadores y estudiar los resultados. Para ello colaboré en la codificación de los módulos de nuestro software donde extraemos las características y creamos y entrenamos los clasificadores. Investigué y estudié el proceso de validación cruzada para el uso en la generación de nuestros modelos. También el uso de matrices de confusión y el significado de las métricas, así como implementé la utilidad para calcularlas a partir de una matriz de confusión. Realicé y documenté con gráficos e imágenes los experimentos de clasificación sobre árboles y bosques aleatorios y participé de su análisis.

He buscado formas de mantener en ficheros las características extraídas junto con Laura, hasta que llegamos a la librería pandas (escrita en *Python*) que nos permitía fácilmente exportar e importar el conjunto de características.

Llegado el momento de realizar la optimización de los algoritmos de aprendizaje automático, árboles de decisión y bosques aleatorios, investigué el campo de los algoritmos genéticos. Estudié la documentación de la librería *DEAP*, que es una librería de *Python* con implementaciones de estos algoritmos para ser adaptadas a otros proyectos, y aporté en la implementación de su uso en nuestro software.

Cuando surgió la idea del experimento de clasificación supervisada con personas participé del diseño de los bocetos iniciales de experimento. Colaboramos conjuntamente con nuestros directores hasta llegar a uno que consideramos factible. Con el experimento ya diseñado contribuí al desarrollo de un programa que genera y evalúa automáticamente los resultados. Me encargué, junto con mi compañera Laura, de buscar y realizar el experimento en las personas dispuestas, supervisando la realización y analizando los resultados.

Gracias a mi experiencia musical y que soy un aficionado a la música electrónica como mencioné anteriormente, contribuí activamente tratando de dar una visión intuitiva de las confusiones de los resultados. A pesar de no dejar de ser especulaciones, espero que faciliten al lector una comprensión de los resultados sin necesidad de escuchar los 2,300 temas en *Beatport*, lo cuál llevaría más de setenta horas.

Finalmente, contribuí a todo el proceso de documentación. He contribuido a la presente memoria, para lo que fue necesario el aprendizaje de *LaTeX* y la síntesis de todo el trabajo realizado. Ayudé con el *README* del repositorio y la creación de los tutoriales básicos de instalación y uso. Para ello probé la instalación en diferentes sistemas operativos para comprobar la funcionalidad. Y para terminar, aporté la documentación de los módulos, para lo que fue necesario el aprendizaje y uso de la librería *Sphinx* que facilita la generación de ésta.

C.2. Laura Pérez Molina

En esta sección paso a detallar cuáles han sido mis aportaciones a lo largo del desarrollo de este trabajo de fin de grado. Es necesario resaltar que este trabajo ha sido planteado desde cero, es decir, desde un primer momento no había materiales definidos para su realización por lo que ha sido un constante aprendizaje por parte de los dos. Por este motivo desde el principio del proyecto, de forma conjunta con Antonio estuvimos investigando sobre las líneas existentes del terreno musical, su aplicación y la realización del proyecto en general. En partes del trabajo donde uno de nosotros ha puesto más iniciativa y empeño en su realización, ha sido el otro el que ha suplido esto dedicándose en mayor medida a otros puntos del mismo (cabe destacar que esto ha sido posible debido a la gran extensión de nuestro TFG).

Desde un primer momento, supimos que sería algo relacionado con la música por lo que buscamos aplicaciones existentes relacionadas con ella. La aplicación por excelencia actualmente es *Shazam*, por lo que nuestra primera idea fue una aplicación como ésta que aportase más información de la que tiene actualmente, de hecho, replicamos un reconocedor de "juguete" pero que no satisfacía nuestros propósitos para desarrollar en el TFG. Durante el proceso de búsqueda del estado actual de este tipo de aplicaciones dimos con la posibilidad de dedicar este trabajo a un proceso más dedicado a la investigación que a la implementación de una interfaz. Con ayuda de nuestros profesores, empezamos un proceso de investigación equitativa hasta que dimos con un documento que nos inspiró y nos dio la idea de lo que es hoy nuestro proyecto, el documento de George Tzanetakis y Perry Cook (este documento ha sido ampliamente nombrado en el proyecto).

Descubrimos que el género musical era algo abstracto y que no existía consenso de clasificación, por lo que nos planteamos una clasificación automática musical. En mi caso particular, mis conocimientos a alto nivel de la música eran más limitados que los de mi compañero, por lo que en este aspecto, él ha sido el que más luz ha aportado en todo lo relacionado con el tema musical. De ahí surgió la idea de la música electrónica y su clasificación.

El documento de George Tzanetakis y Perry Cook lamentablemente, no da suficientes detalles técnicos por lo que en ese momento empezó una búsqueda sobre el software existente. Encontramos *Marsyas* que fue un "dolor de cabeza" entender su funcionamiento y ponerlo en marcha. Esto nos llevó, de forma paralela, a la búsqueda de otras opciones más actuales que nos ofrecieran más detalles sobre la extracción de características del audio. Todas las ilusiones que teníamos con empezar a investigar sobre las características del audio se desvanecieron al comprobar que este software, a nuestro parecer, era de todo menos intuitivo. Además nos encontramos un obstáculo más, el documento había sido citado por muchísimos autores y de hecho, pero nos faltaba conocimientos sobre los puntos que trataba. Entendíamos qué hacían y el porqué pero no llegábamos a ver cómo lo hacían. Por esta

razón, replicar el trabajo de estos dos autores a través de un software nuevo desde cero, iba a resultar algo difícil, para nada exacto y se desviaba de nuestro objetivo final sobre el reconocimiento de género.

Nuestros profesores nos aconsejaron varias librerías relacionadas con el audio, pero que al final también descartamos por ser herramientas que fueron actualizadas por última vez, hace bastantes años. No cesamos en nuestro empeño por lo que decidimos buscar librerías más actuales pero que tuvieran sentido utilizar y que se apoyaran en este documento que parece un "pilar" importante para la clasificación automática de géneros. Por suerte y tras varias horas de trabajo, dimos con una librería de *Python*. La verdad, es que a pesar de no haber utilizado este lenguaje en ninguna asignatura cursada en la carrera, observamos que la curva de aprendizaje no era muy pronunciada por lo que nos aventuramos a utilizarlo. Ahí empezó otro aprendizaje paralelo con este lenguaje a través de información disponible en Internet y de bibliografía de la biblioteca de la facultad. La librería que encontramos fue *PyAudioAnalysis* y sin duda, ha sido un gran acierto para este trabajo, por tener muchas de las funcionalidades que necesitábamos. Dimos con otras, pero que descartamos ya sea por su funcionalidad o por su falta de documentación. *PyAudioAnalisys* ha sido nuestro principal aliado ya que además su fecha de creación es relativamente reciente, es de código libre y existe un documento detallado de dicha librería. Además las características que extraía del audio, eran las propuestas por Tzanetakis y Cook junto a otras más que planteaba. Estuvimos investigando acerca de esta librería, aprendimos a utilizarla y a entender cómo funcionaba y así adquirir más conocimientos sobre cómo funcionaría el reconocimiento. Empezamos a preguntarnos sobre el tamaño de la ventana y otros factores. Buscamos documentación sobre cómo lo habían hecho en otros estudios y la verdad, es que fue complicado de encontrar ya que toda la información que mostramos no estaba a un sólo golpe de ratón, si no que había que adentrarse e ir más allá para encontrar de verdad información útil para ampliar nuestro conocimiento.

Una vez localizada la librería, empezamos una nueva línea de aprendizaje a través del aprendizaje automático. Gracias a nuestros profesores que nos ofrecieron material y a cursos que realizamos *online*, empezamos a aprender sobre ello para poder realizar una clasificación. En las siguientes reuniones, fuimos llegando a un acuerdo sobre los algoritmos que utilizaríamos por ser más didácticos y sencillos de explicar pero eso no zanjó nuestra curiosidad y probamos distintas técnicas, buscando más información.

Un momento crítico fue cuando nos preguntaban por las características que utilizamos y qué median exactamente ya que a pesar de todo el conocimiento que adquirimos, ninguno de los dos tenía una respuesta rotunda. Tanto mi compañero como yo iniciamos un proceso de búsqueda exhaustivo que no hacía más que generarnos más dudas sobre ellas, hasta que encontramos que los propios autores, cuyo criterio nos fiamos completamente por ser expertos en la materia, tampoco sabían explicar qué característica medían la propiedad musical.

Junto con mi compañero participamos conjuntamente en todo el proceso relacionado con el experimento de clasificación con personas. Una vez concretados los puntos a tratar de la memoria, empezamos con una redacción más detallada conducida por todo el proceso de investigación anterior, de igual manera con el reparto de las tareas de forma equitativa.

Apéndice D

Resultados con GTZAN

D.1. Conjunto de datos GTZAN

Este conjunto de datos creado por Tzanetakis and Cook (2000b) se utilizó para el estudio del que se parte principalmente en este trabajo, ampliamente utilizado en la clasificación automática de género musical (Tzanetakis and Cook, 2002).

D.1.1. Bosque aleatorio

Usando un bosque aleatorio análogo al utilizado en la obtención de lo resultados 5 obtenemos la siguiente información.

Matriz de confusión

Tasa de aciertos media: 0.70 +- 0.02

Clase	TP	FN	TN	FP	Precisión	Sensibilidad	ValorF1
blues	73	27	869	31	0.7	0.73	0.72
classical	90	10	889	11	0.89	0.9	0.9
country	70	30	863	37	0.65	0.7	0.68
disco	64	36	858	42	0.6	0.64	0.62
hiphop	71	29	868	32	0.69	0.71	0.7
jazz	70	30	882	18	0.8	0.7	0.74
metal	86	14	874	26	0.77	0.86	0.81
pop	72	28	878	22	0.77	0.72	0.74
reggae	61	39	867	33	0.65	0.61	0.63
rock	47	53	856	44	0.52	0.47	0.49

Conclusión

En la matriz de confusión D.1 vemos en la diagonal todos los aciertos. Hay géneros que destacan por estar mejor clasificados que otros (como *classical* y *metal*) y algunas

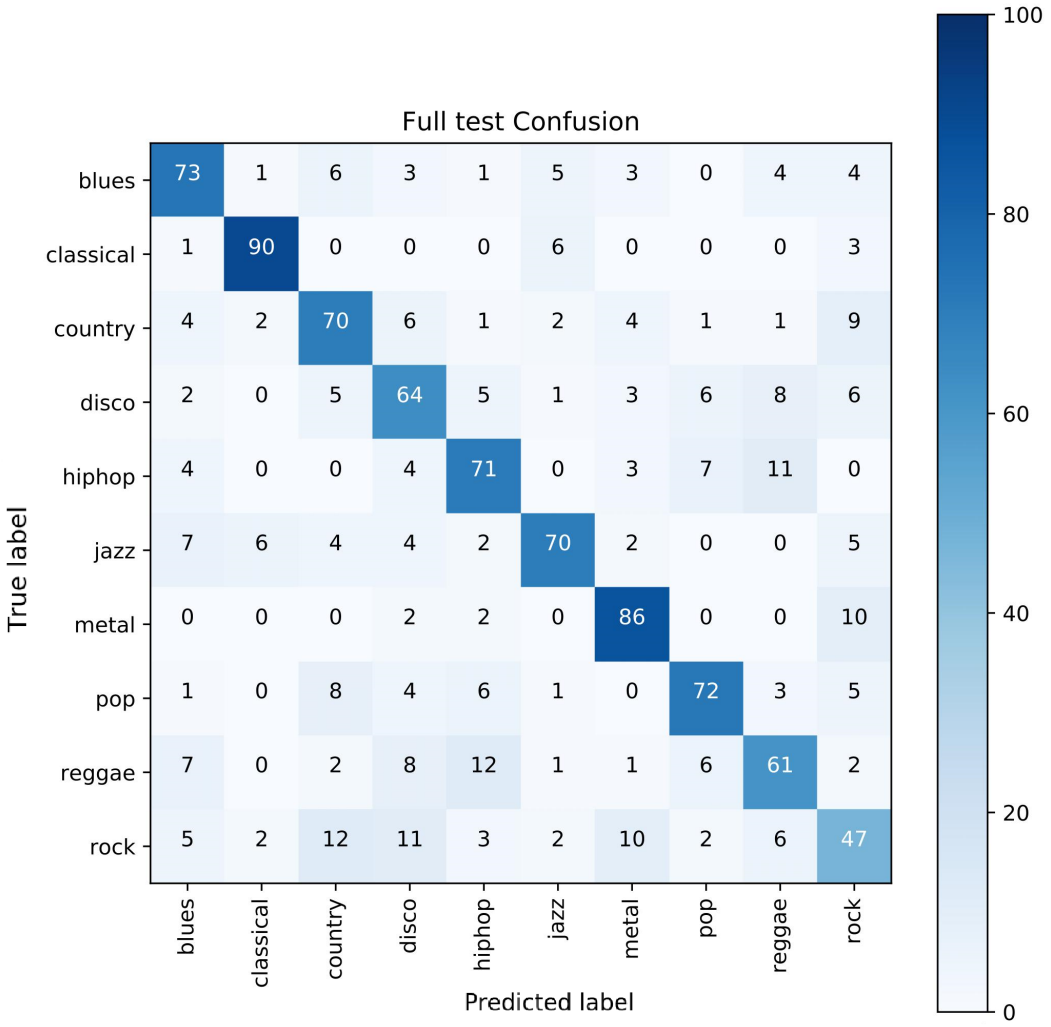


Figura D.1: Matriz de confusión del bosque aleatorio (GTZAN)

confusiones que se podrían indagar más a fondo (*hiphop* y *reggae*, *rock* y *country*).

A través del algoritmo propuesto, se ha conseguido un nivel de precisión del 70 %. Es un resultado que nos parece razonable ya que nuestro objetivo no era presentar un método nuevo o innovador de clasificación, ni tampoco conseguir superar la mayor precisión conseguida hasta entonces con este conjunto sino probar nuestra aproximación para clasificar en subgéneros de música electrónica contra este conjunto de referencia.

Apéndice E

Descripción de los 23 géneros

A continuación, se procede a describir los subgéneros que se utilizan en este estudio que son aceptados en la actualidad a través de *Beatport*[1]. Con esto, se pretende aproximar o etiquetar el tipo de música perteneciente a cada grupo y dar una aproximación al lector de cómo sería la escucha sin entrar en detalles más técnicos. Cabe destacar que los datos referidos a la velocidad del tempo o BPM son meras aproximaciones ya que se puede considerar que un estilo en concreto oscila en un determinado rango, pero eso no excluye que existan canciones etiquetadas en ese subgénero que tengan un BPM posicionado fuera de él. Esto se debe a que el rango es una forma de generalizar pero como el etiquetado es algo subjetivo, pueden existir temas que se posicionen fuera de dicho intervalo:

E.0.1. Big Room

Etiquetas descriptivas de Beatport: Electro, Melbourne Bounce, Moombahton, Progressive.

También denominado como *Big Room House* que a menudo se caracteriza por Big Room house (o simplemente abreviado como *Big Room*) es un subgénero del *Electrohouse* con influencias del *Progressive House* y el *Trance* y cuya etiqueta apareció por petición de los usuarios de *Beatport*[2].

A menudo se caracteriza por la introducción de subidas y bajadas (con piano, cuerdas, vocales...), seguido de un estado altamente energético y una caída de bajo y bombo generalmente simplistas. Normalmente, cada pista está compuesta a una velocidad de 126-132 BPM.

E.0.2. Breaks

Etiquetas descriptivas de Beatport: Bass.

[1]About Beatport Genres and Sub-genres(s.f). En beatport. Recuperado el 17 de abril de 2017 de https://beatportops.zendesk.com/hc/en-us/articles/223805367-Beatport-Genres-and-Sub-genres

[2]Beatport agregó dos etiquetas nuevas a petición de los usuarios como se puede encontrar en http://djtechtools.com/2016/08/12/beatports-re-approach-to-genre-tagging/

Breaks (también denominado *Breakbeat*) suele estar caracterizado por patrones rítmicos constantes en contraposición con el *House*. Su BPM suele oscilar entre 110 y 150.

E.0.3. Dance

Etiquetas descriptivas de Beatport: Pop, Tropical House, Future Bass.

Esta etiqueta ha surgido por la combinación de varios estilos, por lo que al no encontrar una definición exacta en *Beatport*, no se precisa los géneros relacionados ni características musicales ya que cada fuente aporta una visión diferente de ella. Tomando como referencia a *Beatport* se encuentra que las canciones etiquetadas en este género tienen un rango entre 100 y 150 de BPM.

E.0.4. Deep House

Este tipo de música tuvo su momento de máximo esplendor a últimos de los años ochenta y los primeros noventa. Está caracterizado por tener un sonido suave, cálido envolvente y bailable a la vez. Está estrechamente relacionado con el *House* donde las canciones duran entre 6 y 10 minutos moviéndose en un rango de 120 a 125 de BPM. Este género ha ganado popularidad a lo largo de los años[3].

E.0.5. DrumAndBass

Etiquetas descriptivas de Beatport: Liquid, Jump Up, Jungle, Deep, Neurofunk, Bass.

Tiene tantas facetas rítmicas que se puede bailar de formas muy diversas. Unos bailan al ritmo de la línea de bajos, otros al son de la percusión y otros al tempo (BPM entre 160 y 190, siendo normalmente 172).

E.0.6. Dubstep

Etiquetas descriptivas de Beatport: Deep, Bass.

Es un estilo relacionado con el *DrumAndBass* o el *Reggae* entre otros. Su BPM suele mantenerse en 140 o 150 y en casos más particulares, oscila de 160 a 175. Una de sus técnicas sonoras es la presencia de voz modificada de forma digital en muchos casos, al principio de la canción.

[3] http://www.digitalmusicnews.com/2014/08/22/beatports-top-selling-genres-year-year/

E.0.7. Electro House

Etiquetas descriptivas de Beatport: Complextro.

Se considera un grupo dentro del *House* que creció hasta convertirse en un género muy relevante en la actualidad. Combina ritmos comúnmente encontrados en música *House* con bases de sintetizadores con guitarras fuertemente distorsionadas y ocasionalmente el sonido de piano entre otros. El tempo tiene un rango de 125 a 135 pulsaciones por minuto.

E.0.8. Electronica / Downtempo

Etiquetas descriptivas de Beatport: Ambient, Downtempo, Bass.

Electronica o *Downtempo* caracteriza por su sonido relajado con un BPM que no supera los 120. El término música *chill out* se ha empleado en ocasiones para referirse a esta música, pero es una palabra que se aplica también a otros géneros.

E.0.9. Funk R&B / Soul / Disco

El *Funk* reduce el protagonismo de la melodía y de la armonía y dota, a cambio, de mayor peso a la percusión y a la línea de bajo eléctrico. El BPM oscila entre 100 y 130 más o menos.

E.0.10. Future House

Etiquetas descriptivas de Beatport: G-House, Bass House.

Está relacionado con el *DeepHouse* y otros como el *BigRoom* o *ElectroHouse*. Inicialmente se decía que era simplemente *DeepHouse* pero la notable diferencia es que el *FutureHouse* posee un arreglo musical más electrónico y más melódico asemejándose a sonidos más computarizados y metálicos. El BPM se encuentra entre 120 y 130 aproximadamente.

E.0.11. Glitch Hop

El *Glitch hop* o *Glitchstep* es un género con influencias con géneros como el *HipHop* y el *Dubstep* aunque también ha ido adquiriendo elementos de estilos como el *Breaks* o el *DrumAndBass*.

Normalmente tiene un BPM entre 110 y 160 BPM. Aunque no necesariamente use vocales o rap, generalmente fusiona con sonidos de *Dubstep* distorsionados y enfatiza sonidos

como el sonido de "videojuego.º *chiptune*[4].

E.0.12. Hardcore / Hard Techno

Se caracteriza por un tiempo rápido o la distorsión de ritmos y muestras grabadas. Lo más destacado del subgénero es la presencia de un potente y distorsionado bombo. Su BPM puede variar de 150 a 250.

E.0.13. Hard Dance

Etiquetas descriptivas de Beatport: Hardstyle,Hard House,Hard Trance.

Las canciones de este estilo suele oscilar normalmente entre los 150 y los 230 BPM y suele confundirse con *PsyTrance*.

E.0.14. Hip-Hop / R&B

Etiquetas descriptivas de Beatport: Trap, Grime, Bass, Twerk, Future Bass.

En *Hip-Hop* o *R&B (Rhythm and blues)* se encuentran sonidos sintetizados y cajas de ritmos. Su BPM puede comprenderse entre los 90 y 160.

E.0.15. House

Etiquetas descriptivas de Beatport: Tribal, Acid, Soulful.

El *House* es un estilo de música electrónica de baile que se considera uno de los precursores de la música electrónica en general (es un estilo de ésta pero también es uno de sus primeros géneros y precursores). Suele imitar la percusión del disco con un especial uso de un prominente golpe de bombo. También puede incluir líneas de bajo sintetizadas o baterías electrónicas entre otros. Actualmente se suele considerar como música rápida de baile, oscilando en la mayoría de los casos, entre los 120 y los 135 BPM.

E.0.16. Indie Dance / Nu Disco

Etiquetas descriptivas de Beatport: Indie Dance, Nu Disco.

[4]Chiptunes o chip musics (a veces llamado 8-bit music) es música escrita en formatos de sonido donde todos los sonidos son sintetizados en tiempo real por el chip de sonido de una videoconsola.

El *Dance alternativo* o *Indie dance* es un subgénero musical que mezcla y fusiona varios subgéneros del rock con la música electrónica de baile. El BPM, generalmente, se encuentra entre los 110 y 126 aproximadamente.

E.0.17. Minimal / Deep Tech

Caracteriza a un tipo de música que utiliza muy pocos sonidos y ritmos repetitivos y que, sin embargo, suele arrojar resultados creativos asombrosos. Tiene influencias del *Trance, House* o *Techno*. Además de su simplicidad, otra característica de este estilo es que se reduce los BPMs a velocidades muy bajas o se utiliza bombos con muy poca presión. Por este motivo, el BPM está comprendido entre unas velocidades que van desde los 118 a 124, pero se puede encontrar sonidos con una presencia abundante de bajos y percusiones muy agudas con un BPM en torno a los 128.

E.0.18. Progressive House

Estilo de música electrónica proveniente de la música *House*, diferenciándose de éste por una estructura más compleja que progresa a lo largo de la canción. Se caracteriza también por la presencia de bajos y elementos de muchos otros géneros, como el *Trance*, mezclados en su estructura dándole un sonido más electrónico. Desde 2005, la popularidad de este género ha bajado cediéndole terreno a estilos como el *Electro House* y el *Minimal* provocando cambios que han hecho aún más difusa la frontera entre estos subgéneros con un BPM similar entre 120 y 130.

E.0.19. Psy-Trance

Etiquetas descriptivas de Beatport: Full-On, Progressive.

El *Trance psicodélico*, comúnmente llamado *Psychedelic Trance* o *Psytrance* es un género de música electrónica caracterizado por arreglos de ritmo hipnóticos y melodías de sintetizador complejas con un BPM entre 140 y 150 BPM. Algunas canciones que se engloban en este estilo incluyen toques de *Minimal*.

E.0.20. Reggae / Dancehall / Dub

Etiquetas descriptivas de Beatport: Dancehall, Dub, Reggae.

Por extensión, el *Dub* es un género que utiliza las bases del *Reggae* para crear efectos electrónicos, ecos y reverberaciones. En las canciones que pertenecen a este estilo, se da

especial protagonismo al bajo y a la batería y se aparta más la parte vocal. Muchas veces también se incluyen otros efectos sonoros como tiros, sonidos de animales, sirenas de policía,cantar de pájaros, rayos y relámpagos, caer de agua etc. Se suele utilizar para la creación del *remix* de una canción. Su BPM, normalmente se mueve en un rango entre 90 y 125.

E.0.21. Tech House

Se trata de la fusión entre *Techno* y *House*. Comparte la estructura básica del *House* introduciendo elementos característicos del *Techno* como la presencia de bombos más profundos, a menudo con distorsión. El rango de BPM oscila entre 120 y 128.

E.0.22. Techno

Etiquetas descriptivas de Beatport: Melodic, Detroit, Dub, Electro, Industrial.

Se trata de un estilo principalmente musical con escasa presencia de vocales.Se suele escuchar en una sesión continua de DJ dando especial énfasis al ritmo moviéndose en un rango de velocidad de 120 a 125 de BPM.

E.0.23. Trance

Etiquetas descriptivas de Beatport: Progressive, Tech, Uplifting, Vocal.

El *Trance* está caracterizado por un tempo entre 125 y 160 BPM con una forma musical que sube y baja durante cada tema. Los ritmos que se emplean son patrones de la música *Techno*.

Es el resultado de la combinación de diferentes estilos musicales, como el *House, Techno*entre otros por lo que a veces puede ser muy ambiguo debido a esta amplia variedad.

Bibliografía

Aizawa, K., Nakamura, Y., and Satoh, S. (2004). *Advances in Multimedia Information Processing - PCM 2004: 5th Pacific Rim Conference on Multimedia, Tokyo, Japan, November 30 - December 3, 2004, Proceedings*. Number parte 3 in Lecture Notes in Computer Science. Springer Berlin Heidelberg.

Aucouturier, J. and Pachet, F. (2002). Music similarity measures: What's the use? In *Proc. 3rd Int. Symp. Music Information Retrieval*.

Aucouturier, J.-J. and Bigand, E. (2013). Seven problems that keep mir from attracting the interest of cognition and neuroscience. *Journal of Intelligent Information Systems*, 41(3):483–497.

Baniya, B. K. and Lee, J. (2016). Importance of audio feature reduction in automatic music genre classification. *Multimedia Tools and Applications*, 75(6):3013–3026.

Bastian, M., Heymann, S., Jacomy, M., et al. (2009). Gephi: an open source software for exploring and manipulating networks. *ICWSM*, 8:361–362.

Benetos, E. and Kotropoulos, C. (2008). A tensor-based approach for automatic music genre classification. In *Signal Processing Conference, 2008 16th European*, pages 1–4. IEEE.

Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J. R., and Serra, X. (2013). Essentia: An audio analysis library for music information retrieval. In *ISMIR*, pages 493–498. Citeseer.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Chai, W., . V. B. (2001). Folk music classification using hidden markov models. In *International Conference on Artificial Intelligence (Vol. 6, No. 6.4). sn.*

Creme, M., Burlin, C., and Lenain, R. (2016). Music genre classification.

de Sousa, J. M., Pereira, E. T., and Veloso, L. R. (2016). A robust music genre classification approach for global and regional music datasets evaluation.

E. Pampalk, A. F. and Widmer, G. (2005). Improvements of audio based music similarity and genre classification? In *6th Int. Symp. Music Information Retrieval, London, UK*.

Ellis, D. P. (2007). Classifying music audio with timbral and chroma features. In *ISMIR*, volume 7, pages 339–340.

George, T., Georg, E., and Perry, C. (2001). Automatic musical genre classification of audio signals. In *Proceedings of the 2nd international symposium on music information retrieval, Indiana*.

Giannakopoulos, T. (2015). pyaudioanalysis: An open-source python library for audio signal analysis. *PloS one*, 10(12).

Good, M. (2001). Musicxml for notation and analysis. *The virtual score: representation, retrieval, restoration*, 12:113–124.

Guaus, E. (2009). Audio content processing for automatic music genre classification: descriptors, databases, and classifiers.

Holzapfel, A. and Stylianou, Y. (2008). Musical genre classification using nonnegative matrix factorization-based features. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):424–434.

Kedem, B. (1986). Spectral analysis and discrimination by zero-crossings. *Proceedings of the IEEE*, 74(11):1477–1493.

Li, T., Ogihara, M., and Li, Q. (2003). A comparative study on content-based music genre classification. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 282–289. ACM.

Lidy, T., Rauber, A., Pertusa, A., and Inesta, J. M. (2007). Mirex 2007 combining audio and symbolic descriptors for music classification from audio. *MIREX 2007—Music Information Retrieval Evaluation eXchange*.

Lim, S.-C., Lee, J.-S., Jang, S.-J., Lee, S.-P., and Kim, M. Y. (2012). Music-genre classification system based on spectro-temporal features and feature selection. *IEEE Transactions on Consumer Electronics*, 58(4).

Lopes, M., G. F. K. A. L. . O. (2010). Selection of training instances for music genre classification. In *In Pattern Recognition (ICPR), 2010 20th International Conference on (pp. 4569-4572). IEEE.*

Nilsson, N. J. (1996). Introduction to machine learning. an early draft of a proposed textbook.

Pachet, F. and Cazaly, D. (2000). A taxonomy of musical genres. In *Content-Based Multimedia Information Access-Volume 2*, pages 1238–1245. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

Panagakis, I., Benetos, E., and Kotropoulos, C. (2008). Music genre classification: A multilinear approach. In *ISMIR*, pages 583–588.

Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the cuidado project.

Ponce de León Amador, P. J. (2011). *A statistical pattern recognition approach to symbolic music classification*. Universidad de Alicante.

Saunders, J. (1996). Real-time discrimination of broadcast speech/music. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 2, pages 993–996. IEEE.

Scaringella, N. and Zoia, G. (2005). On the modeling of time information for automatic genre recognition systems in audio signals. In *6th Int. Symp. Music Information Retrieval, London, UK*.

Scaringella, N., Zoia, G., and Mlynek, D. (2006). Automatic genre classification of music content: a survey. *IEEE Signal Processing Magazine*, 23(2):133–141.

Scheirer, E. D. (1998). Tempo and beat analysis of acoustic musical signals. *The Journal of the Acoustical Society of America*, 103(1):588–601.

Selfridge-Field, E. (1997). *Beyond MIDI: the handbook of musical codes*. MIT press.

Seyerlehner, K., Widmer, G., and Knees, P. (2010). A comparison of human, automatic and collaborative music genre classification and user centric evaluation of genre classification systems. In *International Workshop on Adaptive Multimedia Retrieval*, pages 118–131. Springer.

Sigtia, S. and Dixon, S. (2014). Improved music feature learning with deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6959–6963. IEEE.

Sturm, B. L. (2012). An analysis of the gtzan music genre dataset. In *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 7–12. ACM.

Sturm, B. L. (2013). The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*.

Subramanian, H., Rao, P., and Roy, S. (2004). Audio signal classification. *EE Dept, IIT Bombay*, pages 1–5.

Tsatsishvili, V. (2011). Automatic subgenre classification of heavy metal music.

Tzanetakis, G. and Cook, P. (2000a). Audio information retrieval (air) tools. In *Proc. International Symposium on Music Information Retrieval*.

Tzanetakis, G. and Cook, P. (2000b). Marsyas: A framework for audio analysis. *Organised sound*, 4(3):169–175.

Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5):293–302.

Wold, E., Blum, T., Keislar, D., and Wheaten, J. (1996). Content-based classification, search, and retrieval of audio. *IEEE multimedia*, 3(3):27–36.