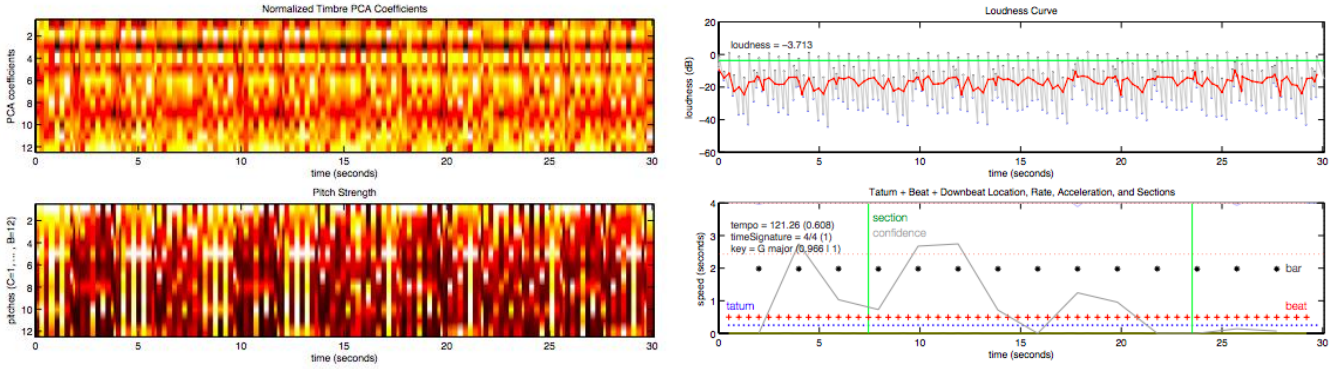
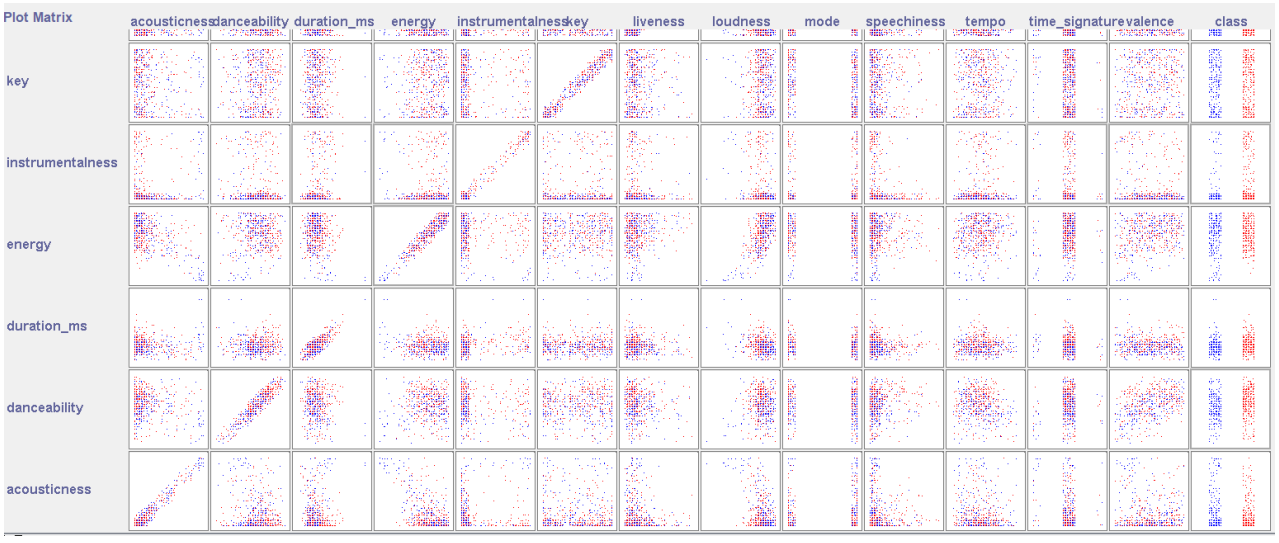
Estudio comparativo de precisión y explicabilidad en algoritmos de cajas blancas, negras y grises sobre modelos de recomendación musical





Autora: Judit González Prol

Coordinadores: Jose Maria Alonso Moral, Alberto Jose Bugarín Diz, Alejandro Catala Bolos

Nome do centro de investigación: Centro Singular de Investigación en Tecnologías (CITIUS)

Titor: Jorge Gómez (profesor do IES Rosalía de Castro)

Centro: IES Plurilingüe Rosalía de Castro

Curso escolar: 2021-2022 (2º BACH)

Resumen

Hoy en día, una de las problemáticas más recurrentes en relación con la inteligencia artificial (IA) y machine learning, es la capacidad de poder dar una explicación entendible en lenguaje natural a los resultados dados por los diferentes algoritmos.

Esta investigación pretende desarrollar una comparativa entre una serie de algoritmos frecuentemente utilizados de cajas negras, blancas y grises. Determinaremos para un problema de recomendación musical a través de variables de audio en bruto, que algoritmos nos ofrecen una mayor precisión y cuales nos ofrecen una mayor explicabilidad en lenguaje natural. Para ello utilizaremos un conjunto de canciones con sus correspondientes métricas de audio en bruto, obtenidas de Spotify, las cuales están clasificadas en canciones que “suelen gustar”, y otras que “no suelen gustar”.

Los algoritmos de cajas negras, como son las redes neuronales, ofrecen una gran precisión en los porcentajes de acierto, y pueden resolver generalmente problemas de gran complejidad en poco tiempo, pero no ofrecen una explicación al resultado obtenido. En cambio, los algoritmos de cajas blancas, son algoritmos de clasificación que ofrecen una menor precisión (dependiendo del problema a tratar), pero sus resoluciones son entendibles y pueden ser explicadas. El hecho de no poder ofrecer una explicación a ciertos problemas, hace que en ciertos casos , la inteligencia artificial puede recaer en resultados racistas o que no se corresponden con las normas sociales. Por lo tanto, es necesario cuestionar a estos algoritmos y no confiar ciegamente en ellos.

Durante el proceso utilizaremos una herramienta de uso profesional de la cual obtendremos las diferentes métricas que nos indican la precisión del algoritmo, así como su desarrollo interno. Y a continuación, a través del programa Expliclas, obtendremos una representación gráfica de las clasificaciones además de una explicación del resultado de cada algoritmo en lenguaje natural entendible.

Abstracts

Nowadays, one of the most recurrent problems in relation to artificial intelligence and machine learning is the ability to give an understandable explanation in natural language to the results given by the different algorithms.

This research aims to develop a comparison between a series of frequently used black-box and white-box algorithms. We will determine, for a music recommendation problem using raw audio variables, which algorithms offer greater precision and which offer greater explainability in natural language. To do so, we will use a set of songs with their corresponding raw audio metrics, obtained from Spotify, which are classified into songs that are "usually liked", and others that are "not usually liked".

Black-box algorithms, such as neural networks, offer high accuracy in hit rates, and can generally solve highly complex problems in a short time, but do not offer an explanation for the result obtained. White-box algorithms, on the other hand, are classification algorithms that offer lower precision (depending on the problem to be dealt with), but their resolutions are understandable and can be explained. The fact of not being able to offer an explanation for certain problems means that in certain cases, artificial intelligence can lead to racist results or results that do not correspond to social norms. It is therefore necessary to question these algorithms and not to trust them blindly.

During the process we will use a professional tool from which we will obtain the different metrics that indicate the accuracy of the algorithm, as well as its internal development. And then, through the Expliclas programme, we will obtain a graphical representation of the classifications as well as an explanation of the result of each algorithm in understandable natural language.

Tabla de contenidos

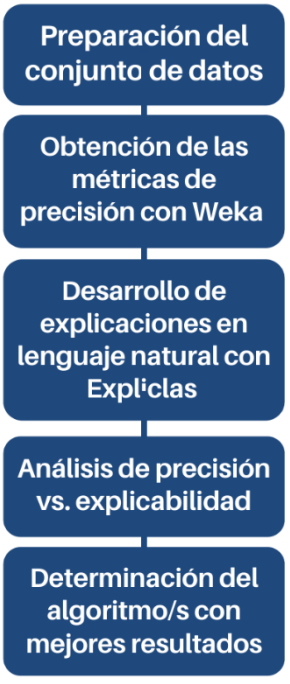
1. Introducción

En los últimos años, el progreso y desarrollo de IA y el machine learning ha crecido exponencialmente, siendo en la actualidad, una parte crucial de nuestra vida diaria así como partidaria de muchas de las decisiones que tomamos. Por ello, debemos comprender su comportamiento, así como permitir a cualquier persona poder comprender las decisiones que estos modelos realizan, algo que con ciertos tipos de modelos algorítmicos dificultan.

Para poder hacer una comprensión completa de estos algoritmos es necesario tener en cuenta otra métrica relativa a la interpretabilidad. En un algoritmo, esta permite conocer el por qué de su toma de decisiones y explicar el funcionamiento del mismo de forma legible para una persona o usuario. De esta forma no nos centramos simplemente en saber el resultado que se predice, si no, la forma en la que esto se predice y su por qué. Este enfoque puede llegar a ser muy resolutivo para ciertos problemas reales y en los cuales entra a tomar en cuenta la visión ética y moral de la que las máquinas y las matemáticas suelen carecer.

Por supuesto, en este proyecto no pretendemos resolver el problema de forma global, pero a través de un ejemplo de modelo de recomendación musical. Esto relacionado con Spotify sería una forma sencilla de explicar su funcionamiento, lo cual nos serviría para poder entender mejor otros problemas más complejos.

Fases del proyecto



1. *Preparación del conjunto de datos*

2. *Obtención de las métricas de precisión con Weka*

   El programa *Weka* nos proporcionará todos los datos relacionados con la precisión de cada uno de los algoritmos a partir de ahí seleccionaremos los 3 algoritmos de los cuales podríamos obtener una información más útil para la comparativa. (Uno por cada categoría)

3. *Desarrollo de explicaciones en lenguaje natural con Expliclas*

   A continuación, elegiremos al azar dos canciones, una que suele gustar y otra que no suele gustar, y obtendremos las explicaciones en lenguaje natural que proporcionan los tres algoritmos elegidos, sobre ellas en Expliclas.

4. *Análisis de precisión vs. explicabilidad*

2. Antecedentes

En este proyecto trabajaremos con uno de los modelos de recomendación utilizados por Spotify. Generalmente, Spotify utiliza sistemas de cajas negras como redes neuronales para determinar cuales serían las canciones que nos gustarían. De esta forma, la red neuronal, no procede a dar explicación de porqué esas canciones que nos ha recomendado nos podrían gustar. La verdadera cuestión yace en el sentido de

esas elecciones. Si utilizásemos arboles de decisión, podríamos saber la razón y de como fueron seleccionadas esas canciones y que variables son las que más se adaptan a nuestros gustos.

A. Algoritmos de recomendación musical de Spotify

Spotify crea sus recomendaciones a través de tres mecanismos y modelos principales:[i]

En este proyecto, trabajaremos con el último método de recomendación, (Audio sin procesar) ya que dados los recursos y medios accesibles y libres, es el único con el que sería posible trabajar. Este modelo obtiene sus métricas y variables del audio en bruto de las canciones. Para ello mide distintas cuestiones como el tiempo estimado de duración, la clave, el modo, el tempo y el volumen. Estas mediciones le permiten a Spotify trazar similitudes entre canciones y así ver para qué usuarios son apropiadas en función de su propio historial de escucha.

Al contrario que Spotify, que simplemente muestra un conjunto de canciones las cuales nos podrían gustar, nosotros pretendemos ofrecer una explicación del porqué de esa recomendación, basándonos en las métricas de las canciones ofrecidas por el propio Spotify y con las que ellos trabajan, a partir del uso de algoritmos accesibles a cualquier usuario común y de libre acceso.

Para ello, hemos obtenido un conjunto de 2.017 canciones con sus respectivas variables musicales designadas por Spotify, además de la variable *target*, que viene definida del conjunto de datos *(1 – Canción que gusta, 2- Canción que no gusta) e* intentaremos a través de la utilización de cajas negras y sistemas de cajas blancas determinar cual sería la mejor opción a la hora de decidir, si las redes neuronales o los árboles de decisión.

La forma en la que podremos decidir que algoritmo es el más adecuado, será basándonos en dos aspectos fundamentales, y que a día de hoy sigue siendo una cuestión que se plantean en una gran cantidad de problemas diferentes de clasificación y de machine learning, pero que también está presente en el proyecto de recomendación musical con el que estamos trabajando a modo de ejemplo.

*Estos dos aspectos fundamentales y principales, son la **precisión** y la **interpretabilidad**.*

3. Hipótesis de trabajo y objetivos de la investigación

El presente trabajo pretende dar respuesta a las siguientes cuestiones:

A. ¿Cuál de los algoritmos de recomendación usados más frecuentemente tiene una mayor precisión?
B. ¿Y cual una mayor explicabilidad?
C. ¿Cuál de estos algoritmos daría el mejor resultado combinando precisión y explicabilidad?
D. ¿En este tipo de problemas y cuestiones que convendría anteponer, una mayor precisión, o una mejor explicabilidad?

Se espera que:

A. Los algoritmos de cajas negras SMO, RandomForest y MultylayerPerception) presenten una mayor precisión y una menor explicación.
B. Los algoritmos de cajas blancas presenten una menor precisión y una explicación más detallada.
C. Probablemente sería una caja blanca porque en este problema tenderemos a darle más importancia a la explicabilidad del gusto musical.
D. Que el factor crucial en este problema sea la explicabilidad, y la diferencia de precisión entre los algoritmos pueda ser casi despreciable.

4. Materiales y métodos

i. Herramientas / programas utilizados

Para la realización de este proyecto se han utilizado dos programas principales:

- Weka,[ii] desde una perspectiva más técnica y de obtención de resultados sobre la precisión. Contiene las herramientas necesarias para realizar transformaciones sobre los datos, tareas de clasificación, regresión, clustering, asociación y visualización. [iii]
- Expliclas[iv], que nos ofrece una representación más legible y gráfica sobre la explicabilidad de cada algoritmo.

i. Algoritmos analizados

Los algoritmos que utilizaremos para realizar esta clasificación para este problema en concreto se clasifican en tres tipos principales según como trabajan y la forma en la que expresan los resultados:[v]

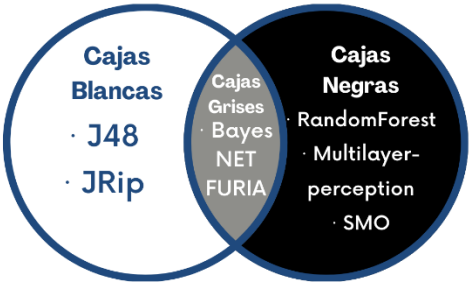


Diagrama de Venn de las categorías de algoritmos: (Material original)

a. Algoritmos de cajas blancas (árboles de decisión)

Los algoritmos caja blanca son aquellos que normalmente trabajan sobre funciones matemáticas o de lógica que el programador conoce muy bien y que va modificando para mejorar el rendimiento del algoritmo.

Estos suelen ser algoritmos de una menor precisión que los algoritmos de cajas negras, pero podríamos preferir los algoritmos de cajas blancas para este problema, ya que nos permiten poder expresar al usuario el porqué de ese resultado positivo (le gusta la canción) o negativo (no le gusta la canción).

b. Algoritmos de cajas negras (redes neuronales)

Por el contrario, un algoritmo caja negra es aquel cuya fórmula para clasificar dichos datos no es conocida. Para este tipo de algoritmos se utilizan, generalmente, redes neuronales con diferentes nodos.

Un nodo puede recibir, por ejemplo, tres datos de entrada que puede ser ponderados con un peso, y sobre los que se aplica una sencilla fórmula matemática para dar una única salida, que podría ser un "si" o un "no". Si a todo ese conjunto de nodos, que llamamos red neuronal, le damos unos datos

de entrada, y nos produce una salida, no sabremos exactamente en que se ha basado el algoritmo para dar dicha salida.

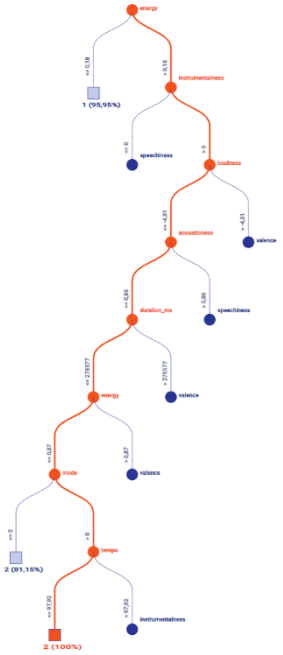
Este modelo se entrena con miles de ejemplos que luego queremos que el algoritmo haga por sí sólo. Además, en un algoritmo de caja negra no sabremos exactamente cuál fue el camino lógico que siguió para dar una salida concreta, pero los ingenieros podrán en todo momento cambiar tanto la estructura de la red, como la fórmula de cada nodo, como los pesos de las variables de entrada de cada nodo para mejorar su rendimiento.

c. *Algoritmos de cajas grises (métodos conjuntos)*

1. *Algoritmos de cajas blancas:*

J48 (C.45)

J48 es la implementación en Weka del algoritmo C4.5[1] introducido por primera vez por Quinlan. Es un clasificador C4.5 podado. Este algoritmo se considera una caja blanca interpretable porque al recorrer el árbol desde la raíz hasta las hojas es posible entender la clasificación de cada instancia de datos. J48 se basa en una estrategia descendente. En primer lugar se selecciona qué atributo se va a dividir en el nodo raíz, y luego se crea una rama para cada posible valor de atributo, y eso divide las instancias en subconjuntos, uno por cada rama que se extiende desde el nodo raíz. Así pues, es uno de los mejores algoritmos de aprendizaje automático para examinar los datos de forma categórica y continua.



Ejemplo del árbol de decisión J48 para una canción elegida: (Material original)

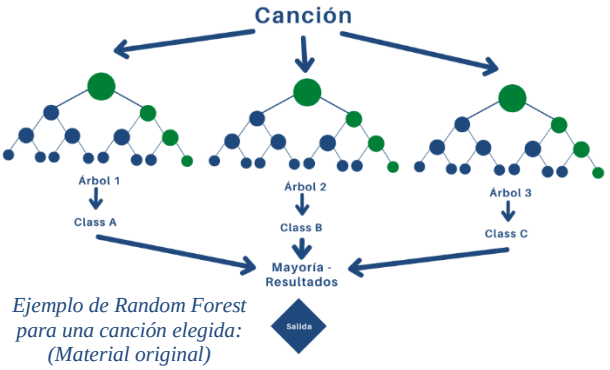
JRip

JRip implementa un aprendiz de reglas proposicionales llamado "Repeated Incremental Pruning to Produce Error Reduction (RIPPER)" y utiliza algoritmos de cobertura secuencial para crear listas de reglas ordenadas. El algoritmo pasa por 4 etapas: Construcción de una regla, poda, optimización y selección
El algoritmo Ripper es un algoritmo de clasificación basado en reglas. Deriva un conjunto de reglas a partir del conjunto de entrenamiento.

[1] C4.5 es una serie de algoritmos utilizados en problemas de clasificación de minería de datos y aprendizaje automático. Su objetivo es supervisar el aprendizaje: dado un conjunto de datos, cada tupla en él se puede describir mediante un conjunto de valores de atributo, y cada tupla pertenece a una determinada categoría en una categoría mutuamente excluyente. El objetivo de C4.5 es encontrar una relación de mapeo de valores de atributo a categorías a través del aprendizaje, y este mapeo se puede usar para clasificar nuevas entidades con categorías desconocidas. *Programador clic. (2022). Retrieved 27 January 2022, from https://programmerclick.com/article/79041108425/*

2. Algoritmos de cajas negras:

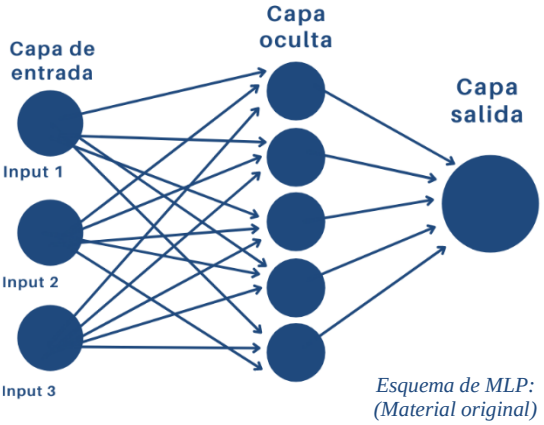
RandomForest



Ejemplo de Random Forest para una canción elegida: (Material original)

Random Forest es un método de aprendizaje de conjunto que crea una combinación de árboles de decisión C4.5. Aunque los clasificadores individuales se consideran cajas blancas interpretables, su combinación aleatoria es difícilmente interpretable y se considera una caja negra. Este es un algoritmo de clasificación formado por muchos árboles de decisión. Utiliza el ensacado[2] y la aleatoriedad de las características al construir cada árbol individual para intentar crear un bosque no correlacionado de árboles cuya predicción por comité sea más precisa que la de cualquier árbol individual.

Multilayerperception (MLP)



Esquema de MLP: (Material original)

Un perceptrón multicapa (MLP) es una clase de reMd neuronal artificial (RNA) de tipo feedforward. Un MLP consta de al menos tres capas de nodos: una capa de entrada, una capa oculta y una capa de salida. Excepto los nodos de entrada, cada nodo es una neurona que utiliza una función de activación no lineal.

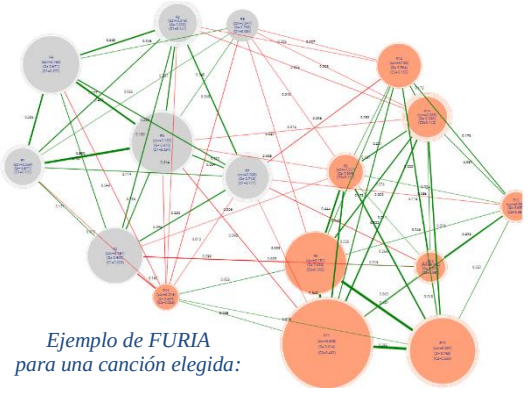
Support Vector Machine (SMO)

Las máquinas de vectores de soporte son un conjunto de algoritmos de aprendizaje supervisado propiamente relacionados con problemas de clasificación y regresión.

Los vectores de apoyo son puntos de datos que están más cerca del hiperplano y que influyen en la posición y la orientación del hiperplano. Utilizando estos vectores de soporte, maximizamos el margen del clasificador.

3. Algoritmos de cajas grises:

FURIA



Ejemplo de FURIA para una canción elegida:

FURIA es el acrónimo de *Fuzzy Unordered Rule Induction Algorithm*. Genera reglas de clasificación difusas IF-THEN con conjuntos difusos de forma trapezoidal en el antecedente de cada regla. Estos conjuntos difusos carecen de interpretabilidad lingüística. Este se considera un clasificador de caja gris porque produce un conjunto de reglas que pueden ser interpretadas.

[2] El ensacado es un método para mejorar la estabilidad y precisión de los algoritmos de aprendizaje automático y puede reducir la varianza del modelo para evitar el sobreajuste. *Programador clic. (2022). Retrieved 28 January 2022, from https://programmerclick.com/article/24171214522/*

Bayes NET

Una red bayesiana (BN) es un modelo gráfico probabilístico para representar el conocimiento sobre un dominio incierto en el que cada nodo corresponde a una variable aleatoria y cada arista representa la probabilidad condicional de las variables aleatorias correspondientes.

5. Resultados

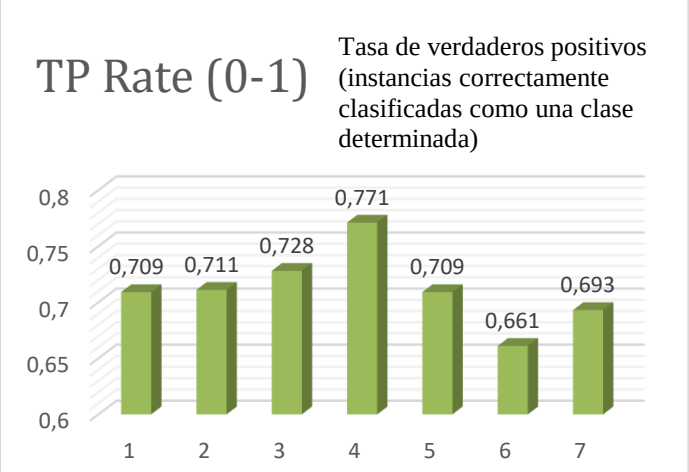
a. Resultados de precisión (Weka)

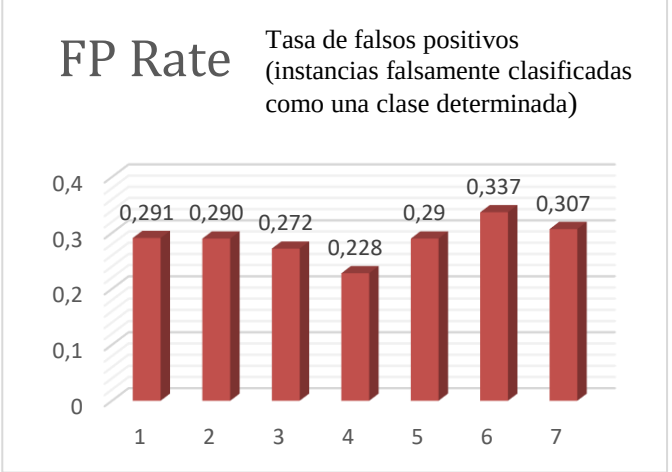
Para obtener estos resultados hemos utilizado métodos de validación cruzada y matrices de confusión en el programa de Weka obteniendo diferentes valores para una serie de métricas sobre las cuales analizamos la precisión. [vi] [vii] [viii]

i. Análisis gráfico de precisión

Tras pasar los datos a la consola de *Weka* y realizar los modelos con cada uno de los algoritmos propuestos obtenemos los siguientes resultados:

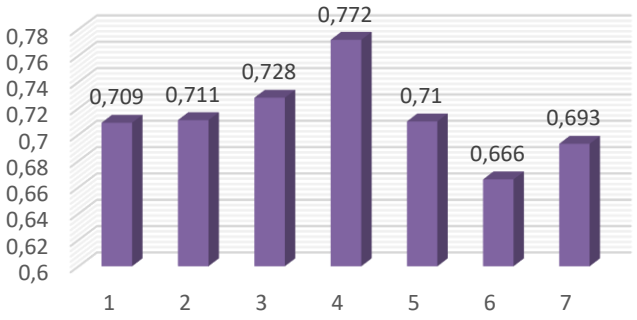
Denominación en las gráficas	1	2	3	4	5	6	7
Nombre	J48	JRip	Furia	RandomForest	MLP	SMO	Bayes NET





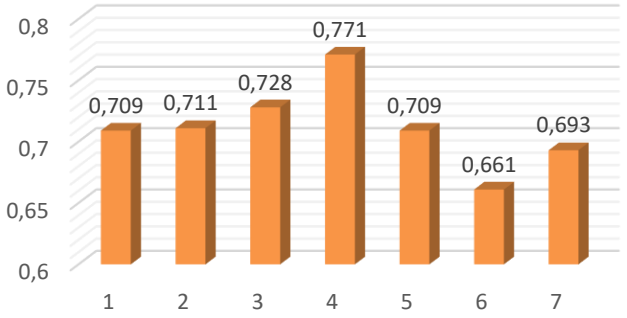
Precision

Proporción de instancias que son realmente de una clase dividida por el total de instancias clasificadas como esa clase.



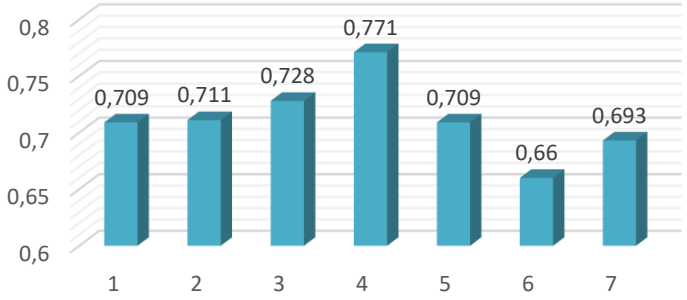
Recall

Proporción de instancias clasificadas como una clase determinada dividida por el total real de esa clase (equivalente a la tasa TP)

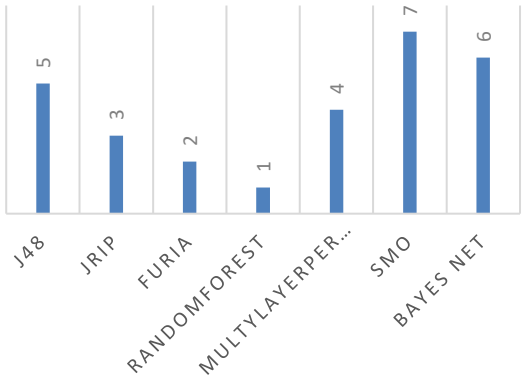


F-Measure

Una medida combinada de precisión y recuperación calculada como 2 * Precision * Recall/ (Precision + Recall)

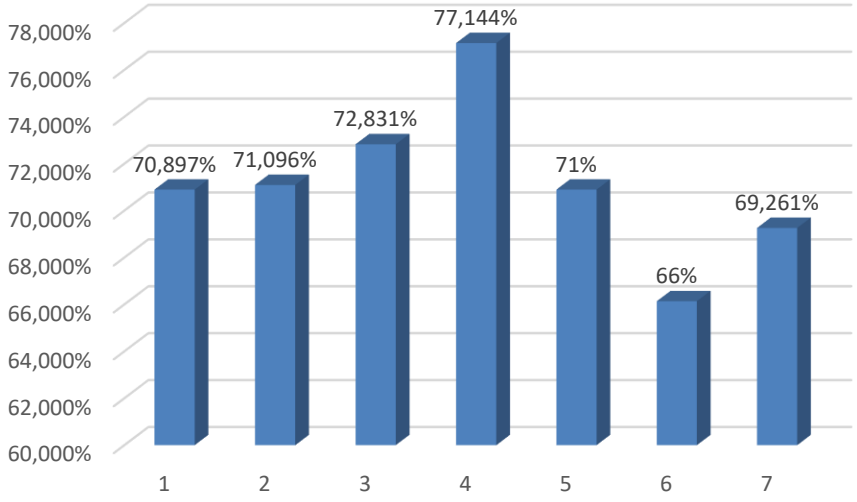


RANKING



Correct Cl. (%)

El porcentaje de instancias clasificadas correctamente suele denominarse precisión o exactitud de la muestra. Destaca RandomForest.



ii. *Resultados de explicabilidad (Expliclas)*

Tras obtener los resultados sobre la precisión de cada algoritmo para el problema, escogeremos tres de categorías diferentes para poder expresar la explicabilidad de cada uno de los algoritmos de forma práctica. Para ello, vamos a utilizar varios ejemplos concretos de canciones, con sus respectivos datos e intentar ver cual de ellos podría expresar al usuario una mejor explicación del por que de si le gusta (2) o no (1) esa canción.[ix] [x]

a. *Unforgettable, French Montana* [3] [xi]

- *J48: – Anexo: Ver árboles de decisión generados para J48* [xii]

El ejemplo es 1 porque la bailabilidad y la valencia son altas, la duración_ms, la instrumentalidad, la locución y el tempo son bajos y la energía es media. Para estos valores específicos es igual de probable que sea 2. Además, es probable que sea 2 porque 1 se confunde con este tipo al menos en un 10%. El 2 es posible debido a la proximidad de la bailabilidad con el valor de división (0,79). Sin embargo, esta clasificación es errónea porque el tipo debería ser 2 en lugar de 1 según la información del conjunto de datos.

- *FURIA:*

El clasificador realiza un estiramiento para determinar que el ejemplo es 1. Sin embargo, esto es incorrecto porque el tipo debería ser 2 en lugar de 1 según la información del conjunto de datos.

- *RandomForest:*

El ejemplo es 2 porque la bailabilidad, el tempo y la valencia son altos, la acústica, la instrumentalidad y la sonoridad son bajos y la duración_ms, la vivacidad y el discurso son medios. Además, es probable que sea 1 porque 2 se confunde con este tipo al menos en un 10%. Pero es poco probable que sea 1. El 1 es posible debido a la proximidad del tempo con el valor de división (111,0).

b. *Get Lucky, Daft Punk* [4] [xiii]

- *J48:*

[3] https://youtu.be/CTFtOOh47oo
[4] https://youtu.be/5NV6Rdv1a3I

El ejemplo es 1 porque acousticness, duration_ms, instrumentalness, loudness y mode son bajos y energy y tempo son medios. Para estos valores específicos es igual de probable que sea 2. Además, es probable que sea 2 porque 1 se confunde con este tipo al menos en un 10%. 2 es posible debido a la proximidad del tempo con el valor de división (97,923).

- *FURIA:*

Tenemos una gran confianza en el resultado de la clasificación. Es muy probable que este ejemplo sea 2 porque la sonoridad y la vivacidad son bajas y la instrumentalidad y la energía son medias. Sin embargo, esto es erróneo porque el tipo debería ser 1 en lugar de 2 según la información del conjunto de datos.

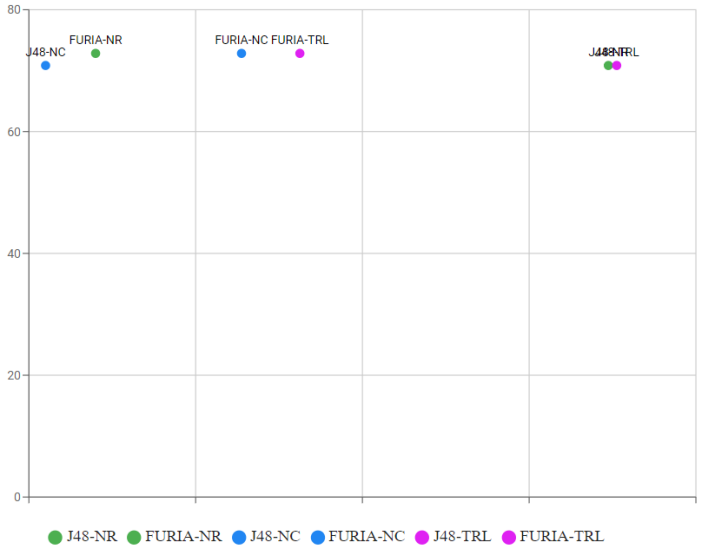
- *Random Forest:*

Existe una confusión relacionada con todos los tipos de ejemplo. El ejemplo es 1 porque acousticness, duration_ms, instrumentalness, loudness y mode son bajos y energy y tempo son medios. Para estos valores específicos es igual de probable que sea 2. Además, es probable que sea 2 porque 1 se confunde con este tipo al menos en un 10%. 2 es posible debido a la proximidad del tempo con el valor de división (97,923)

ii. *Análisis gráfico de precisión vs. Interpretabilidad*

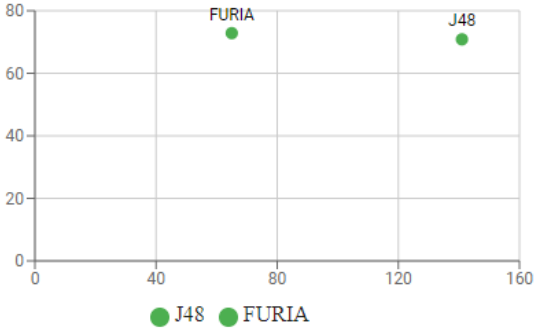
Comparando la información dada por Expliclas con los tres algoritmos que hemos elegido, obtenemos las siguientes gráficas comparativas. *(En esta gráfica solo representa los datos de los algortimos J48 y FURIA, ya que son los únicos por los que puede analizar los siguientes aspectos/variables)*

Precisión vs Interpretabilidad (Generall)

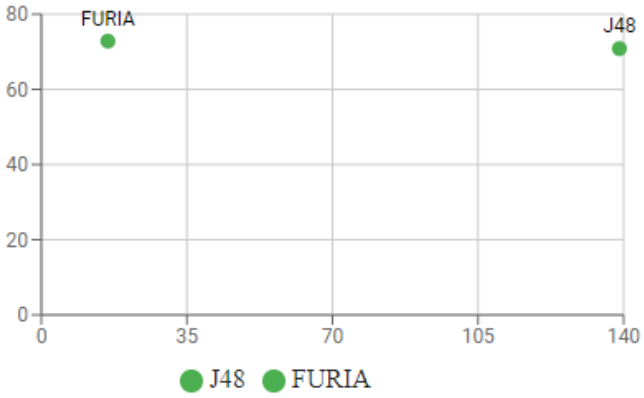


NR: Numero de reglas
NC: Numero de conceptos

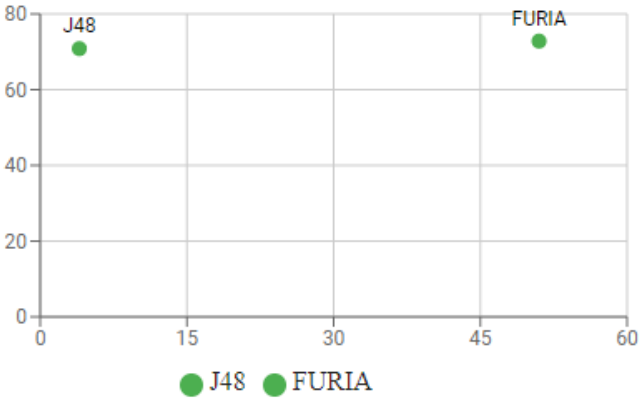
Precision (CCI)[1] vs Interpretabilidad (TRL)



Precision (CCI)[1] vs Interpretabilidad (Número de reglas)



Precision (CCI)[1] vs Interpretabilidad (Número de conceptos)



6. Conclusión

Sobre las gráficas de precisión vs. Interpretabilidad (J48 vs. FURIA):

- El mejor algoritmo entre J48 y FURIA, es **FURIA** porque, con el menor número de reglas tiene el mayor valor de CCI.
- El algoritmo con menor número de conceptos es J48. Sin embargo, el algoritmo FURIA con el menor valor de TRL tiene mayor valor de CCI.

Sobre las métricas de precisión: **RandomForest** presenta un mayor porcentaje de clasificaciones correctas.

El algoritmo que presenta una mejor resolución en las métricas de precisión es RandomForest con un 77,14% de efectividad, pero a pesar de que no presenta una legibilidad tan alta como J48 o FURIA, si ofrece una explicación entendible. Por otra parte, otros algoritmos como J48, tienen una diferencia de precisión no muy notable frente a RandomForest, y desarrolla una mejor explicación de la resolución dada en los ejemplos utilizados.

Siendo un problema real, una tasa de acierto del 70% es alta pero no completamente fiable por lo que algoritmos como RandomForest, en ciertos casos como en los ejemplos mostrados, clasifica incorrectamente, por lo que ver la explicación de esa recomendación sería crucial para obtener un resultado correcto y corregirlo centrándonos en aquellas canciones que fallan.

La solución para este problema sería confiar de la explicación de las cajas blancas J48 y FURIA, solo cuando hay consenso con las cajas negras como RandomForest, o cuando los 3 coincidan.

7. Agradecimientos

Quiero agradecer a Jose Maria Alonso Moral, Alberto Jose Bugarín Diz y Alejandro Catala Bolos Centro Singular de Investigación en Tecnologías (CITIUS), y también a Jorge Gómez Suarez, profesor del IES Rosalía de Castro, por la ayuda ofrecida.

8. Bibliografía

a. Obtención de datos y herramientas

- https://www.kaggle.com/geomack/spotifyclassification
- https://opendatascience.com/a-machine-learning-deep-dive-into-my-spotify-data/
- https://developer.spotify.com/documentation/web-api/reference/#/operations/get-several-audio-features

b. Webgrafía y bibliografía

PromocionMusical.es. Cómo Funcionan los Algoritmos de Recomendación en Spotify. (2021). Retrieved 7 October 2021, from https://promocionmusical.es/como-funcionan-algoritmos-recomendacion-spotify

Black-box vs. white-box models. Most machine learning systems require… | by Lars Hulstaert | Towards Data Science. (2021). Retrieved 8 October 2021, from https://towardsdatascience.com/machine-learning-interpretability-techniques-662c723454f3

Accuracy vs Explainability of Machine Learning Models [NIPS workshop poster review]. (2021). Retrieved 8 October 2021, from https://www.inference.vc/accuracy-vs-explainability-in-machine-learning-models-nips-workshop-poster-review/

By: Sebastian Klovig Skelton. Solving the AI black box problem through transparency. (2021). Retrieved 18 November 2021, from https://searchenterpriseai.techtarget.com/feature/How-to-solve-the-black-box-AI-problem-through-transparency

The Black Box Problem - When AI Makes Decisions That No Human Can Explain. (2021). Retrieved 18 November 2021, from https://www.interceptinghorizons.com/post/the-black-box-problem-when-ai-makes-decisions-that-no-human-can-explain

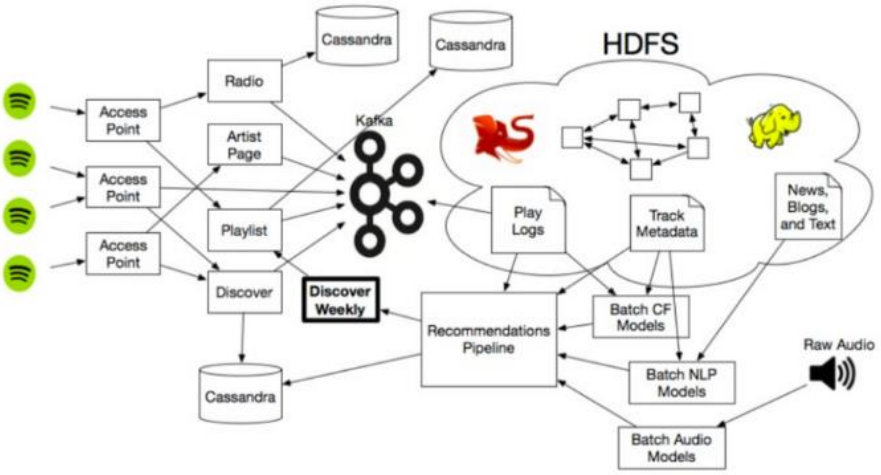
The "black box" problem | ETH Zurich. (2021). Retrieved 18 November 2021, from https://ethz.ch/en/news-and-events/eth-news/news/2020/09/the-black-box-problem.html

Aprendizaje Supervisado y No Supervisado - Fernando Sancho Caparrini. (2021). Retrieved 18 November 2021, from http://www.cs.us.es/~fsancho/?e=77

Aprendizaje Supervisado: Introducción a la Clasificación y Principales Algoritmos | by Victor Roman | Ciencia y Datos | Medium. (2021). Retrieved 18 November 2021, from https://medium.com/datos-y-ciencia/aprendizaje-supervisado-introducci%C3%B3n-a-la-clasificaci%C3%B3n-y-principales-algoritmos-dadee99c9407

Anexos

i



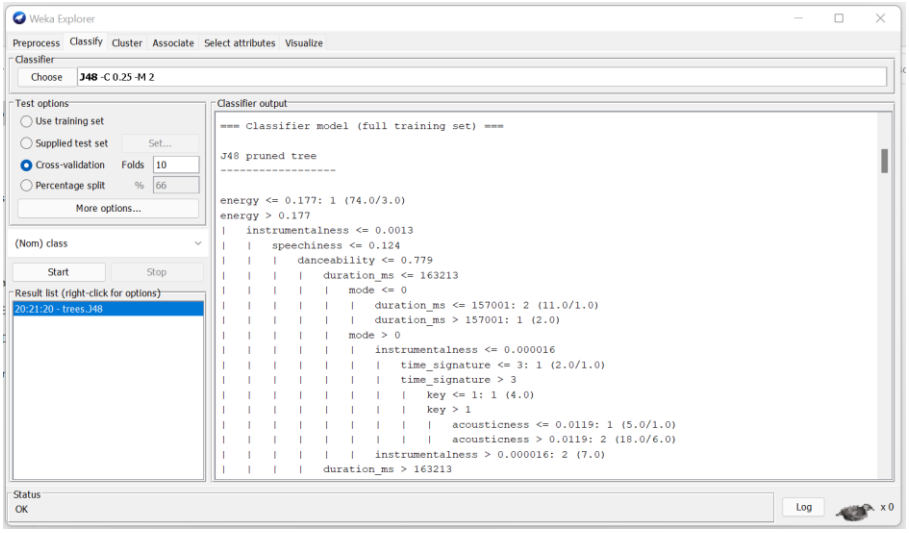
Esquema del proceso de recomendación de Spotify

ii

a. Weka

Weka es un programa contiene una extensa colección de algoritmos de machine learning desarrollados por la universidad de Waikato (Nueva Zelanda) implementados en Java; útiles para ser aplicados sobre datos mediante los interfaces que ofrece o para embeberlos dentro de cualquier aplicación, como es en el caso de Expliclas[ii].

iii



Interfaz del programa Weka (Material original)

iv

Expliclas

Por otra parte, en la segunda fase de la investigación, usaremos el programa Expliclas[iv], un generador de explicaciones en lenguaje natural[iv], de libre acceso a través del cual obtendremos representaciones gráficas y todas las explicaciones relativas a la explicabilidad de los diferentes algoritmos. En un principio, seleccionaremos aquellos tres que obtengan una mayor precisión en Weka, pero que a su vez queden representadas ambos tipos de algoritmos de cajas negras y de cajas blancas.

vi

Comparativa de las métricas en detalle:

Algoritmos							
	Cajas Blancas			Cajas Negras			
Nombre	**J48**	**JRip**	**Furia**	**RandomForest**	**MLP**	**SMO**	**Bayes NET**
TP Rate	0,709	0,711	0,728	0,771	0,709	0,661	0,693
a	0,711	0,679	0,716	**0,777**	0,733	0,734	0,694
b	0,707	0,742	0,740	**0,766**	0,685	0,590	0,691
FP Rate	0,291	0,290	0,272	**0,228**	0,290	0,337	0,307
a	0,293	0,258	0,260	**0,234**	0,315	0,410	0,309
b	0,289	0,321	0,284	**0,223**	0,267	0,266	0,306
Precision	0,709	0,711	0,728	**0,772**	0,710	0,666	0,693
a	0,703	0,720	0,729	**0,764**	0,695	0,637	0,687
b	0,715	0,703	0,727	**0,779**	0,724	0,694	0,698
Recall	0,709	0,711	0,728	**0,771**	0,709	0,661	0,693
a	0,711	0,679	0,716	**0,777**	0,733	0,734	0,694
b	0,707	0,742	0,740	**0,766**	0,685	0,590	0,691
F-Measure	0,709	0,711	0,728	**0,771**	0,709	0,660	0,693
a	0,707	0,699	0,723	**0,771**	0,714	0,682	0,691
b	0,711	0,722	0,734	**0,772**	0,704	0,638	0,695

A partir de estos resultados los organizamos para llevar a cabo las comparativas entre ellos:

Algoritmos							
	Cajas Blancas			Cajas Negras			
Nombre	**J48**	**JRip**	**Furia**	**RandomForest**	**MLP**	**SMO**	**Bayes NET**

Tiempo necesario	0,17	0,43	1,16	0,61	1,65	0,16	0,06
Correct Cl.	70,897 %	71,096 %	72,831 %	**77,144%**	70,897%	66,138 %	69,261%
Incorrect Cl.	29,103 %	28,904 %	27,169 %	**22,856%**	29,103%	33,862 %	30,739%
Relative absolute error	65,907 %	75,850 %	55,285 %	**66,194%**	69,109%	67,733 %	75,860%
Ranking	**4**	**3**	**2**	**1**	**4**	**7**	**6**
TP Rate	0,709	0,711	0,728	**0,771**	0,709	0,661	0,693
Ranking	**5**	**3**	**2**	**1**	**3**	**7**	**6**
FP Rate	0,291	0,290	0,272	**0,228**	0,290	0,337	0,307
Ranking	**5**	**3**	**2**	**1**	**4**	**7**	**6**
Precision	0,709	0,711	0,728	**0,772**	0,710	0,666	0,693
Ranking	**4**	**3**	**2**	**1**	**4**	**7**	**6**
Recall	0,709	0,711	0,728	**0,771**	0,709	0,661	0,693
Ranking	**4**	**3**	**2**	**1**	**4**	**7**	**6**
F-Measure	0,709	0,711	0,728	**0,771**	0,709	0,660	0,693
	4	**5**	**2**	**1**	**3**	**7**	**6**
a	0,707	0,699	0,723	**0,771**	0,714	0,682	0,691
	4	**3**	**2**	**1**	**5**	**7**	**6**
b	0,711	0,722	0,734	**0,772**	0,704	0,638	0,695

	J48	**JRip**	**Furia**	**RandomForest**	**MLP**	**SMO**	**Bayes NET**
Ranking medio	4,286	4,286	2,000	1,000	3,857	7,000	6,000
Ranking STD	0,488	0,756	0,000	0,000	0,690	0,000	0,000
Pre/ Recall/ F Ranking	4,333	3,000	2,000	1,000	4,000	7,000	6,000
RANKING FINAL DE PRECISIÓN	***5º***	***3º***	***2º***	***1º***	***4º***	***7º***	***6º***

vii

i. Técnicas y métodos complementarios

Una de las técnicas que utilizaremos durante el proceso de determinación de la precisión será por una parte la validación cruzada, y por otra parte las matrices de confusión.:

a. Validación cruzada

La validación cruzada es una técnica para evaluar modelos de ML mediante el entrenamiento de varios modelos de ML en subconjuntos de los datos de entrada disponibles y evaluarlos con el subconjunto complementario de los datos. Consiste en dividir los datos de forma aleatoria en k grupos de aproximadamente el mismo tamaño, k-1 grupos se emplean para entrenar el modelo y uno de los grupos se emplea como validación. Este proceso se repite k veces utilizando un grupo distinto como validación en cada iteración. Este método ayuda a conseguir resultados con mayor verosimilitud y certeros.

b. *Matriz de confusión*

La matriz de confusión es una herramienta muy útil para valorar cómo de bueno es un modelo clasificación basado en aprendizaje automático. En particular, sirve para mostrar de forma explícita cuándo una clase es confundida con otra, lo cual nos, permite trabajar de forma separada con distintos tipos de error.

<table>
<tr><td colspan="2" rowspan="2">Matriz de confusión</td><td colspan="2">Estimado por el modelo</td><td colspan="2" rowspan="2"></td></tr>
<tr><td>Negativo (N)</td><td>Positivo (P)</td></tr>
<tr><td rowspan="2">Real</td><td>Negativo</td><td>a: (TN)</td><td>b: (FP)</td><td></td><td></td></tr>
<tr><td>Positivo</td><td>c: (FN)</td><td>d: (TP)</td><td>Precisión
(“precision”)
Porcentaje predicciones positivas correctas:</td><td>d/(b+d)</td></tr>
<tr><td colspan="2"></td><td>Sensibilidad, exhaustividad
(“Recall”)
Porcentaje casos positivos detectados</td><td>Especifidad
(Specifity)
Porcentaje casos negativos detectados</td><td colspan="2">Exactitud
(“accuracy”)
Porcentaje de predicciones correctas
(No sirve en datasets poco equilibrados)</td></tr>
<tr><td colspan="2"></td><td>d/(d+c)</td><td>a/(a+b)</td><td colspan="2">(a+d)/(a+b+c+d)</td></tr>
</table>

viii

Variables del audio en bruto

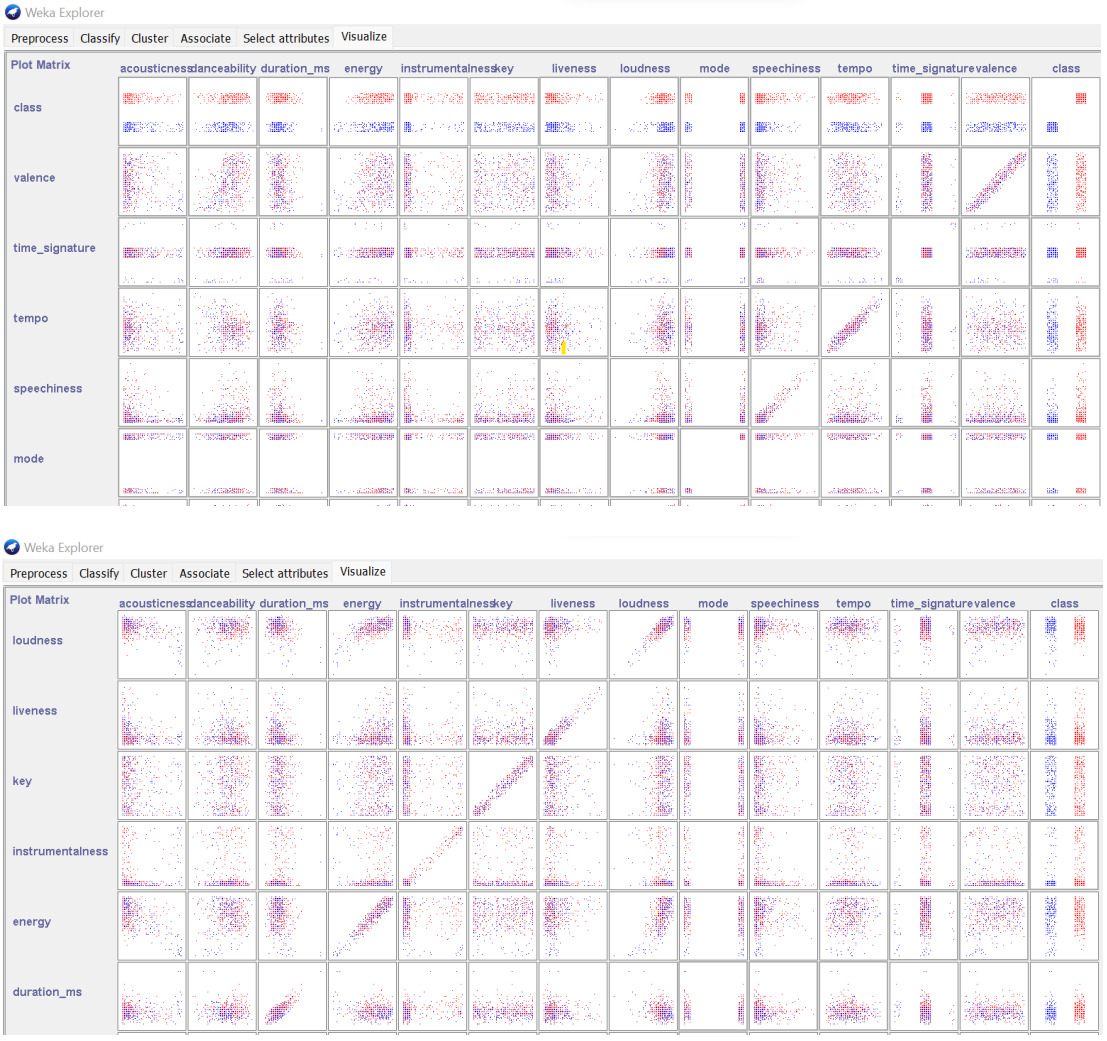
Las variables que son atribuidas a cada una de las canciones se definen de la siguiente manera:[viii]

Nombre	Definición
Acousticness (acústica)	Una medida de confianza de 0,0 a 1,0 de si la pista es acústica. 1,0 representa una alta confianza en que la pista es acústica. [0,1]
Danceability (bailabilidad)	La bailabilidad describe lo adecuada que es una pista para bailar basándose en una combinación de elementos musicales que incluyen el tempo, la estabilidad del ritmo, la fuerza del compás y la regularidad general. Un valor de 0.0 es el menos bailable y 1.0 es el más bailable.
Duration_ms (duración)	La duración de la pista en milisegundos.
Energy (energía)	La energía es una medida de 0,0 a 1,0 y representa una medida perceptiva de intensidad y actividad. Típicamente, las pistas energéticas se sienten rápidas, fuertes y ruidosas. Por ejemplo, el death metal tiene mucha energía, mientras que un preludio de Bach tiene una puntuación baja en la escala. Entre las características perceptivas que contribuyen a este atributo se encuentran el rango dinámico, el volumen percibido, el timbre, la velocidad de aparición de la

	entropía general.
Key (clave)	La tonalidad de la pista. Los números enteros se asignan a los tonos utilizando la notación estándar de Pitch Class. Por ejemplo, 0 = C, 1 = C♯/D♭, 2 = D, etc. Si no se detecta ninguna clave, el valor es [-1, 1]
Liveness (vivacidad)	Detecta la presencia de público en la grabación. Los valores de *liveness* más altos representan una mayor probabilidad de que la pista haya sido interpretada en directo. Un valor superior a 0,8 proporciona una fuerte probabilidad de que la pista sea en vivo.
loudness (sonoridad)	La sonoridad general de una pista en decibelios (dB). Los valores de sonoridad se promedian en toda la pista y son útiles para comparar la sonoridad relativa de las pistas. La sonoridad es la cualidad de un sonido que es el principal correlato psicológico de la fuerza física (amplitud). Los valores suelen oscilar entre -60 y 0 dB.
mode (modo)	El modo indica la modalidad (mayor o menor) de una pista, el tipo de escala del que se deriva su contenido melódico. La mayor se representa con 1 y la menor con 0.
speechiness	La locuacidad detecta la presencia de palabras habladas en una pista. Cuanto más exclusivamente hablada sea la grabación (por ejemplo, un programa de entrevistas, un audiolibro o una poesía), más se acercará a 1,0 el valor del atributo. Los valores superiores a 0,66 describen pistas que probablemente estén compuestas exclusivamente por palabras habladas. Los valores entre 0,33 y 0,66 describen pistas que pueden contener tanto música como voz, ya sea en secciones o en capas, incluyendo casos como la música rap. Los valores por debajo de 0,33 representan probablemente música y otras pistas no habladas.
tempo	El tempo global estimado de una pista en pulsaciones por minuto (BPM). En la terminología musical, el tempo es la velocidad o el ritmo de una pieza determinada y se deriva directamente de la duración media de los tiempos.
time_signature (tiempo musical)	Una firma de tiempo estimada. La signatura de tiempo (metro) es una convención de notación para especificar cuántos tiempos hay en cada compás (o medida). La signatura de tiempo va de 3 a 7, indicando signaturas de tiempo de "3/4", a "7/4". Rango de valores: [3,7]
valence (valencia)	Una medida de 0,0 a 1,0 que describe la positividad musical que transmite una pista. Las pistas con alta valencia suenan más positivas (por ejemplo, felices, alegres, eufóricas), mientras que las pistas con baja valencia suenan más negativas (por ejemplo, tristes, deprimidas, enfadadas).Rango de valores: [0,1]
Instrumentalness (instrumentalidad)	Predice si una pista no contiene voces. Los sonidos "Ooh" y "aah" se tratan como instrumentales en este contexto. Las pistas de rap o de palabras habladas son claramente "vocales". Cuanto más se acerque el valor de instrumentalización a 1,0, mayor será la probabilidad de que la pista no tenga contenido vocal. Los valores superiores a 0,5 representan pistas instrumentales, pero la confianza es mayor a medida que el valor se acerca a 1,0.

	A	B	C	D	E	F	G	H	I	J	K	L
1	id,acousticness,danceability,duration_ms,energy,instrumentalness,key,liveness,loudness,mode,speechiness,tempo,time_signature,valence,target,song_title,artist											
2	0,0.0102,0.833,204600,0.434,0.0219,2,0.165,-8.795,1,0.431,150.062,4.0,0.286,1,Mask Off,Future											
3	1,0.199,0.743,326933,0.359,0.00611,1,0.137,-10.401,1,0.0794,160.083,4.0,0.588,1,Redbone,Childish Gambino											
4	2,0.0344,0.838,185707,0.412,0.000234,2,0.159,-7.148,1,0.289,75.044,4.0,0.173,1,Xanny Family,Future											
5	3,0.604,0.494,199413,0.338,0.51,5,0.0922,-15.236,1,0.0261,86.468,4.0,0.23,1,Master Of None,Beach House											
6	4,0.18,0.678,392893,0.561,0.512,5,0.439,-11.648,0,0.0694,174.004,4.0,0.904,1,Parallel Lines,Junior Boys											
7	5,0.00479,0.804,251333,0.56,0.0,8,0.164,-6.682,1,0.185,85.023,4.0,0.264,1,Sneakinâ€™,Drake											
8	6,0.0145,0.739,241400,0.472,7.27e-06,1,0.207,-11.204,1,0.156,80.03,4.0,0.308,1,Childs Play,Drake											
9	7,0.0202,0.266,349667,0.348,0.664,10,0.16,-11.609,0,0.0371,144.154,4.0,0.393,1,GyÃ¶ngyhajÃº lÃ¡ny,Omega											
10	8,0.0481,0.603,202853,0.944,0.0,11,0.342,-3.626,0,0.347,130.035,4.0,0.398,1,I've Seen Footage,Death Grips											
11	9,0.00208,0.836,226840,0.603,0.0,7,0.571,-7.792,1,0.237,99.994,4.0,0.386,1,Digital Animal,Honey Claws											
12	10,0.0572,0.525,358187,0.855,0.0143,5,0.649,-7.372,0,0.0548,111.951,3.0,0.524,1,Subways - In Flagranti Extended Edit,The Avalanches											
13	11,0.0915,0.753,324880,0.748,0.00348,10,0.212,-8.62,1,0.0494,104.322,4.0,0.642,1,Donme Dolap - Baris K Edit,Modern Folk ÃœÃ§lÃ¼sÃ¼											
14	12,0.253,0.603,356973,0.434,0.0619,0,0.108,-11.062,1,0.0342,127.681,4.0,0.381,1,Cemalim,Erkin Koray											
15	13,0.366,0.762,243270,0.476,0.0,0,0.103,-12.686,1,0.114,130.007,4.0,0.367,1,One Night,Lil Yachty											
16	14,0.44,0.662,247288,0.603,0.0,9,0.0972,-8.317,0,0.0793,125.011,4.0,0.351,1,Oh lala,PNL											
17	15,0.019,0.637,188333,0.832,0.0563,6,0.316,-6.637,1,0.163,99.988,4.0,0.317,1,Char,Crystal Castles											
18	16,0.0239,0.603,270827,0.955,0.0451,1,0.119,-4.111,1,0.0458,123.922,4.0,0.773,1,World In Motion,New Order											
19	17,0.233,0.789,447907,0.659,0.00049,4,0.184,-12.654,0,0.0429,122.415,4.0,0.842,1,One Nation Under a Groove,Funkadelic											

Muestra en Excel de los valores de las variables para algunas canciones (Material Original)



Resultados gráficos de la comparativa de las diferentes variables entre sí en el programa Weka (Material Original)

ix

a. Unforgattable, Frech Montana:

Algorithm	Global explanation
J48	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 70,85%. There is confusion related to all types of example.
FURIA	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 72,83%. There is confusion related to all types of example.
RandomForest	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 77,14%. There is confusion related to all types of example.

Algorithm	Local explanation
J48	Example is 1 because acousticness, duration_ms, instrumentalness, loudness and mode are low and energy and tempo are medium.
FURIA	We have a high confidence in the classification result. It is very likely that this example is 2 because loudness and liveness are low and instrumentalness and energy are medium. However, this is wrong because the type should be 1 instead of 2 according to the information in the dataset.
RandomForest	This instance is classified as 1

b. *Get Lucky, Daft Punk* [ix]

Algorithm	Global explanation
J48	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 70,85%. There is confusion related to all types of example.
FURIA	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 72,83%. There is confusion related to all types of example.
RandomForest	There are 2 types of example: 1 and 2. This classifier is quite confusing because correctly classified instances represent a 77,14%. There is confusion related to all types of example.

Algorithm	Local explanation
J48	Example is 1 because danceability and valence are high, duration_ms, instrumentalness, speechiness and tempo are low and energy is medium. However, this classification is wrong because type should be 2 instead of 1 according to the information in the dataset.
FURIA	Classifier performs stretching to determine that example is 1. However, this is wrong because the type should be 2 instead of 1 according to the information in the dataset.
RandomForest	This instance is classified as 2

[xi] *Valores del audio en bruto para la canción Uforgettable:*

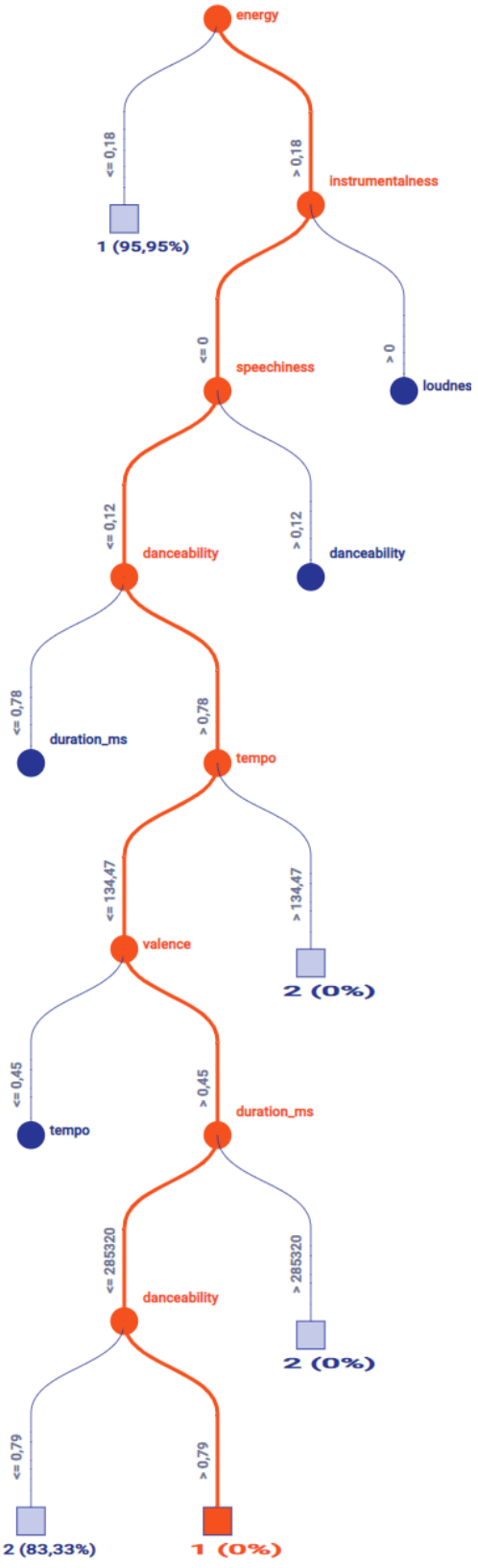
Name	Value	Property name	Property value
acousticness	0.0293	Low	0.33 , 0
		Medium	0.66 , 0.33
		High	0.99 , 0.66
danceability	0.726	Low	0.41 , 0.12
		Medium	0.7 , 0.41
		High	0.98 , 0.7
duration_ms	233833	Low	345570.33 , 16042
		Medium	675098.67 , 345570.33
		High	1004627 , 675098.67
energy	0.769	Low	0.34 , 0.01
		Medium	0.67 , 0.34
		High	1 , 0.67
instrumentalness	0.0101	Low	0.33 , 0
		Medium	0.65 , 0.33
		High	0.98 , 0.65
key	6	Low	3.67 , 0
		Medium	7.33 , 3.67
		High	11 , 7.33
liveness	0.104	Low	0.34 , 0.02
		Medium	0.65 , 0.34
		High	0.97 , 0.65
loudness	-5.043	Low	-22.17 , -33.1
		Medium	-11.24 , -22.17
		High	-0.31 , -11.24
mode	1	Low	0.33 , 0
		Medium	0.67 , 0.33
		High	1 , 0.67
speechiness	0.123	Low	0.29 , 0.02
		Medium	0.55 , 0.29
		High	0.82 , 0.55
tempo	97.985	Low	105.02 , 47.86
		Medium	162.17 , 105.02
		High	219.33 , 162.17
time_signature	4	Low	2.33 , 1
		Medium	3.67 , 2.33
		High	5 , 3.67
valence	0.75	Low	0.35 , 0.03
		Medium	0.67 , 0.35
		High	0.99 , 0.67

[xi] *Valores del audio en bruto para la canción Get Lucky:*

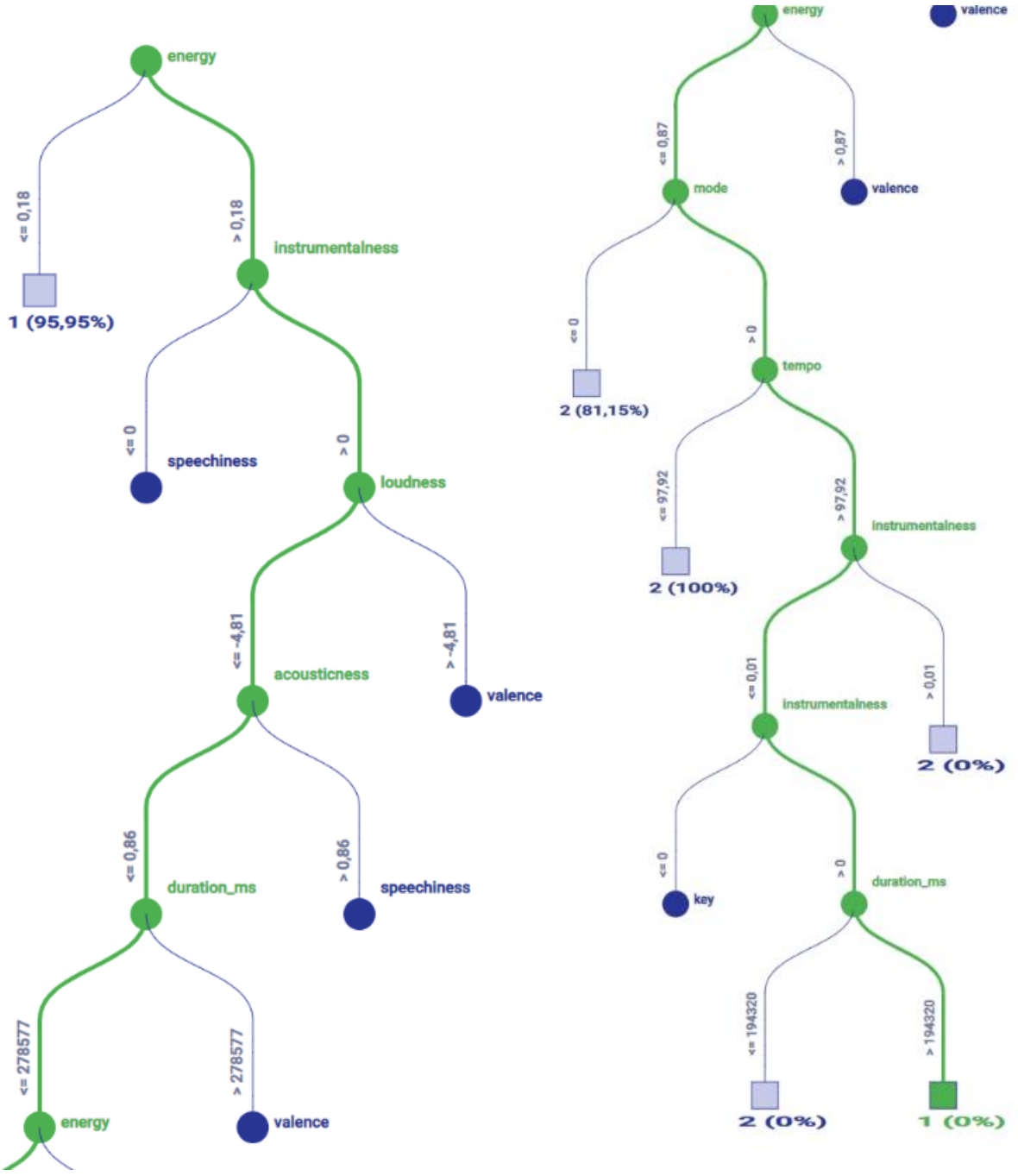
mode	0	Low	0.33 , 0
		Medium	0.67 , 0.33
		High	1 , 0.67
speechiness	0.038	Low	0.29 , 0.02
		Medium	0.55 , 0.29
		High	0.82 , 0.55
tempo	116.047	Low	105.02 , 47.86
		Medium	162.17 , 105.02
		High	219.33 , 162.17
time_signature	4	Low	2.33 , 1
		Medium	3.67 , 2.33
		High	5 , 3.67
valence	0.865	Low	0.35 , 0.03
		Medium	0.67 , 0.35
		High	0.99 , 0.67

Name	Value	Property name	Property value
acousticness	0.0426	Low	0.33 , 0
		Medium	0.66 , 0.33
		High	0.99 , 0.66
danceability	0.794	Low	0.41 , 0.12
		Medium	0.7 , 0.41
		High	0.98 , 0.7
duration_ms	248413	Low	345570.33 , 16042
		Medium	675098.67 , 345570.33
		High	1004627 , 675098.67
energy	0.811	Low	0.34 , 0.01
		Medium	0.67 , 0.34
		High	1 , 0.67
instrumentalness	0.000001	Low	0.33 , 0
		Medium	0.65 , 0.33
		High	0.98 , 0.65
key	6	Low	3.67 , 0
		Medium	7.33 , 3.67
		High	11 , 7.33
liveness	0.101	Low	0.34 , 0.02
		Medium	0.65 , 0.34
		High	0.97 , 0.65
loudness	-8.966	Low	-22.17 , -33.1
		Medium	-11.24 , -22.17
		High	-0.31 , -11.24

xii Árboles de decisión de J48:



Árbol de decisión de la canción Get Lucky, Daft Punk (J48)



Árbol de decisión de la canción Unforgettable, Frech Montana (J48)