

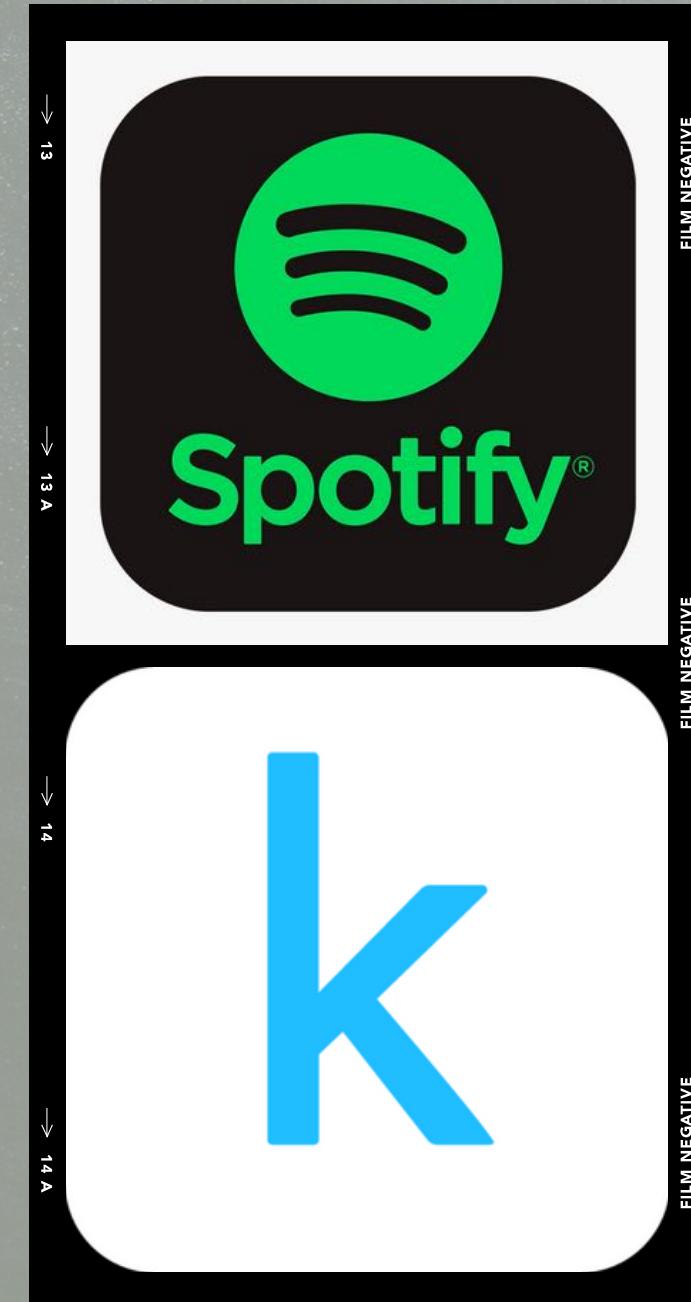


Spotify®

Machine Learning para clasificación de géneros musicales

INDICE

- 1. INTRODUCCIÓN Y OBJETIVO**
- 2. EDA**
- 3. PREPROCESADO DATASET 1**
- 4. MACHINE LEARNING**
- 5. PREPROCESADO DATASET 2**
- 6. APLICACION MODELO ML**
- 7. RESULTADOS**
- 8. CONCLUSIONES**
- 9. REFERENCIAS**



1-INTRODUCCION Y OBJETIVO

En la era de las plataformas de música en línea, como Spotify, la clasificación precisa de géneros musicales se ha vuelto crucial para mejorar la experiencia del usuario y proporcionar recomendaciones más afinadas. Este proyecto se propone utilizar algoritmos de aprendizaje automático para llevar a cabo esta clasificación, aprovechando las diversas características que ofrecen las canciones en estas plataformas.

Contamos con un conjunto de datos robusto proveniente de la API de Spotify, que abarca ocho géneros principales. A través de la aplicación de un modelo de Machine Learning, buscamos crear un modelo de clasificación preciso.

Nuestro objetivo final es aplicar el modelo entrenado a un extenso conjunto de datos, compuesto por más de 1 millón de canciones, donde no se incluyen etiquetas de género. Esta fase permitirá que nuestro modelo realice predicciones de género, abriendo nuevas posibilidades para la personalización de recomendaciones musicales.

Al aprovechar eficientemente las variadas características proporcionadas por Spotify, junto con la implementación de modelos de aprendizaje automático, aspiramos a lograr una clasificación más óptima, contribuyendo así a la mejora continua de la experiencia musical de los usuarios.

2-EDA

Dataset 1: 9.198 filas / 24 columnas

	track_id	playlist_url	playlist_name	track_name	track_popularity	artist_name	album	album_cover	artist_genres	artist_popularity	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature	genre
0	4Gia17DzXBnYFbYUj6SyW	https://open.spotify.com/playlist/7qACZGMjyo64...	The Sound of Blues	Working Man	51	Otis Rush	Mourning In The Morning	https://i.scdn.co/image/ab67616d0000b273fea221...	['blues', 'blues rock', 'chicago blues', 'elec...	41	...	1	0.04	0.48	0.00	0.20	0.84	103.36	147800	4 blues
1	1BjYNhg7jhVQdxqETHBwn	https://open.spotify.com/playlist/7qACZGMjyo64...	The Sound of Blues	Long Way Home	38	Clarence "Gatemouth" Brown	Long Way Home	https://i.scdn.co/image/ab67616d0000b2730eff13...	['blues', 'blues rock', 'memphis blues', 'mode...	33	...	0	0.04	0.91	0.05	0.12	0.42	78.03	338333	4 blues
2	2Cg3GUkjX96nO4p2WRlls	https://open.spotify.com/playlist/7qACZGMjyo64...	The Sound of Blues	She's A Sweet One	49	Junior Wells	Calling All Blues - The Chief, Profile & USA R...	https://i.scdn.co/image/ab67616d0000b27399b18c...	['blues', 'blues rock', 'chicago blues', 'elec...	41	...	1	0.05	0.15	0.03	0.20	0.71	122.86	181786	4 blues
3	5bC6ONDsL88snGN6QasjZH	https://open.spotify.com/playlist/7qACZGMjyo64...	The Sound of Blues	Help Me	59	Sonny Boy Williamson II	More Real Folk Blues	https://i.scdn.co/image/ab67616d0000b273b48c81...	['acoustic blues', 'blues', 'blues rock', 'chi...	46	...	0	0.04	0.60	0.02	0.61	0.77	114.22	188200	4 blues
4	2TKykeHeVKsBq2C8M3SKcN	https://open.spotify.com/playlist/7qACZGMjyo64...	The Sound of Blues	Take Out Some Insurance	51	Jimmy Reed	Rockin' With Reed	https://i.scdn.co/image/ab67616d0000b2739b7573...	['blues', 'blues rock', 'chicago blues', 'elec...	42	...	1	0.05	0.66	0.00	0.12	0.57	111.33	143332	4 blues
...	
9193	2oGYxgu2ztDa64of4edwv	https://open.spotify.com/playlist/50kZecUV5pY2...	The Sound of Techno	Walking with Clouds	17	Transilusion	The Opening of the Cerebral Gate	https://i.scdn.co/image/ab67616d0000b2730c430e...	['electro', 'techno']	14	...	1	0.05	0.00	0.76	0.11	0.23	135.34	212160	4 electronic
9194	2TbbgHlwZVjErsxTm63Lh	https://open.spotify.com/playlist/50kZecUV5pY2...	The Sound of Techno	Start To Move	23	Oscar Mulero	Mannequin	https://i.scdn.co/image/ab67616d0000b273abfdcb...	['minimal dub', 'minimal techno', 'spanish tec...	27	...	1	0.11	0.01	0.94	0.12	0.42	133.99	317483	3 electronic
9195	0Rwl08UX8INW6Cn8eb068P	https://open.spotify.com/playlist/50kZecUV5pY2...	The Sound of Techno	Take Me Away - Truncate Remix	7	DJ 3000	Take Me Away	https://i.scdn.co/image/ab67616d0000b273371d0e...	['detroit techno', 'techno']	7	...	0	0.05	0.00	0.94	0.12	0.18	130.01	355596	4 electronic
9196	1M33B7EKrhx3xEYnoKz	https://open.spotify.com/playlist/50kZecUV5pY2...	The Sound of Techno	Funny	24	Mr. De'	Follow the Leader 4	https://i.scdn.co/image/ab67616d0000b273b581d...	['electro', 'ghettotech', 'techno']	27	...	0	0.07	0.03	0.86	0.11	0.69	148.00	236333	4 electronic
9197	71be9NpmFl3zj0qTXQJf	https://open.spotify.com/playlist/50kZecUV5pY2...	The Sound of Techno	Liquid Slow - Mixed	35	Chris Liebing	Mixmag Presents Charlotte de Witte (DJ Mix)	https://i.scdn.co/image/ab67616d0000b273e23812...	['frankfurt electronic', 'german techno', 'min...	37	...	1	0.04	0.01	0.90	0.12	0.41	134.77	335063	4 electronic

9198 rows x 24 columns

Dataset 2: 1.140.952 filas / 24 columnas

	id	name	album	album_id	artists	artist_ids	track_number	disc_number	explicit	danceability	...	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature	year	release_date
0	7imeHLHBe4nmXzuXc0HDjk	Testify	The Battle Of Los Angeles	2eia0myWFgoHuttJytCxgX	['Rage Against The Machine']	[2d0hyoQ5ynDBrnkvAbJKORj]	1.00	1.00	False	0.47	...	0.07	0.03	0.00	0.36	0.50	117.91	210133.00	4.00	1999.00	1999-11-02
1	1wsRitRRIWYEarl0q22o8	Guerrilla Radio	The Battle Of Los Angeles	2eia0myWFgoHuttJytCxgX	['Rage Against The Machine']	[2d0hyoQ5ynDBrnkvAbJKORj]	2.00	1.00	True	0.60	...	0.19	0.01	0.00	0.15	0.49	103.68	206200.00	4.00	1999.00	1999-11-02
2	1hR0ffK2qRG3RF70pb7	Calm Like a Bomb	The Battle Of Los Angeles	2eia0myWFgoHuttJytCxgX	['Rage Against The Machine']	[2d0hyoQ5ynDBrnkvAbJKORj]	3.00	1.00	False	0.32	...	0.48	0.02	0.00	0.12	0.37	149.75	298893.00	4.00	1999.00	1999-11-02
3	2lbASgTS0D07MTuLAXITW0	Mic Check	The Battle Of Los Angeles	2eia0myWFgoHuttJytCxgX	['Rage Against The Machine']	[2d0hyoQ5ynDBrnkvAbJKORj]	4.00	1.00	True	0.44	...	0.24	0.16	0.00	0.12	0.57	96.75	213640.00	4.00	1999.00	1999-11-02
4	1MQTmpYOZfcMqc56Hdo7T	Sleep Now In the Fire	The Battle Of Los Angeles	2eia0myWFgoHuttJytCxgX	['Rage Against The Machine']	[2d0hyoQ5ynDBrnkvAbJKORj]	5.00	1.00	False	0.43	...	0.07	0.00	0.10	0.08	0.54	127.06	205600.00	4.00	1999.00	1999-11-02
...	
1140947	78pDVGo1Cjj1e81z8JJoK6	All or Nothing - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpShoQ	['Whitesnake']	[3UbYyvNvNIT5DFXU4WgiGpP]	8.00	2.00	False	0.62	...	0.06	0.02	0.14	0.24	0.39	132.53	220646.00	4.00	2019.00	2019-03-08
1140948	2ZWrIVLD6mnFSw0Cqyl6WK	Split It Out - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpShoQ	['Whitesnake']	[3UbYyvNvNIT5DFXU4WgiGpP]	9.00	2.00	False	0.47	...	0.05	0.01	0.00	0.12	0.72	132.31	254125.00	4.00	2019.00	2019-03-08
1140949	5xe8x2Jct1GdM2AylD1PB	Guilty of Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpShoQ	['Whitesnake']	[3UbYyvNvNIT5DFXU4WgiGpP]	10.00	2.00	False	0.44	...	0.08	0.04	0.02	0.33	0.65	174.23	203799.00	4.00	2019.00	2019-03-08
1140950	4pXFNBpsQatvDv2YqKoQGr	Need Your Love so Bad - 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpShoQ	['Whitesnake']	[3UbYyvNvNIT5DFXU4WgiGpP]	11.00	2.00	False	0.26	...	0.03	0.40	0.00	0.51	0.06	107.88	193766.00	3.00	2019.00	2019-03-08
1140951	6EM56vqZv8T8UWfBqciSU	Gambler - Eddie Kramer Mix, 1983; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpShoQ	['Whitesnake']	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

1140952 rows x 24 columns

Tipo datos - Dataset 1

```
track_id          object
playlist_url      object
playlist_name      object
track_name          object
track_popularity    int64
artist_name          object
album              object
album_cover          object
artist_genres        object
artist_popularity    int64
danceability        float64
energy              float64
key                  int64
loudness            float64
mode                  int64
speechiness         float64
acousticness         float64
instrumentalness    float64
liveness             float64
valence              float64
tempo                  int64
duration_ms         int64
time_signature       int64
genre                  object
dtype: object
```

Tipo datos - Dataset 2

```
id                object
name              object
album             object
album_id           object
artists            object
artist_ids         object
track_number       float64
disc_number       float64
explicit           object
danceability      float64
energy              float64
key                  int64
loudness            float64
mode                  int64
speechiness         float64
acousticness         float64
instrumentalness    float64
liveness             float64
valence              float64
tempo                  int64
duration_ms         float64
time_signature      float64
year                  int64
release_date        object
dtype: object
```

DESCRIPCION DE LAS VARIABLES (1/2)

- ACOUSTICNESS: indica la probabilidad de que una canción sea acústica, de 0 a 1. Géneros como el jazz pueden tener valores más altos.
- DANCEABILITY: indica que tan bailable es una canción en rango de 0 a 1. Las canciones pop y la música electrónica tienden a tener valores más altos, ya que suelen ser más rítmicas y bailables.
- DURATION_MS: duración de la pista en milisegundos.
- ENERGY: representa la intensidad y actividad percibida en una canción de 0 a 1. Géneros como el rock y la música electrónica tienden a tener valores de energía más altos, mientras que géneros más relajados como el jazz o la música clásica pueden tener valores más bajos. Normalmente, las pistas energéticas son más rápidas y ruidosas.
- INSTRUMENTALNESS: la probabilidad de que una pista sea completamente instrumental, con un rango de 0 a 1. Géneros como la música clásica o la electrónica pueden tener valores más altos. Por lo contrario, géneros como el rap, suelen ser más vocales.
- LIVENESS: detecta la presencia de público en una grabación, de 0 a 1. Valores altos de liveness representan una alta probabilidad de que la pista provenga de una actuación en vivo. Un valor por encima de 0.8 indicaría una alta posibilidad de que así sea.
- LOUDNESS: El volumen general de una pista en decibelios (dB). Los valores de sonoridad se promedian en toda la pista y son útiles para comparar el volumen relativo de las pistas. Los valores suelen oscilar entre -60 y 0 db.

DESCRIPCION DE LAS VARIABLES (2/2)

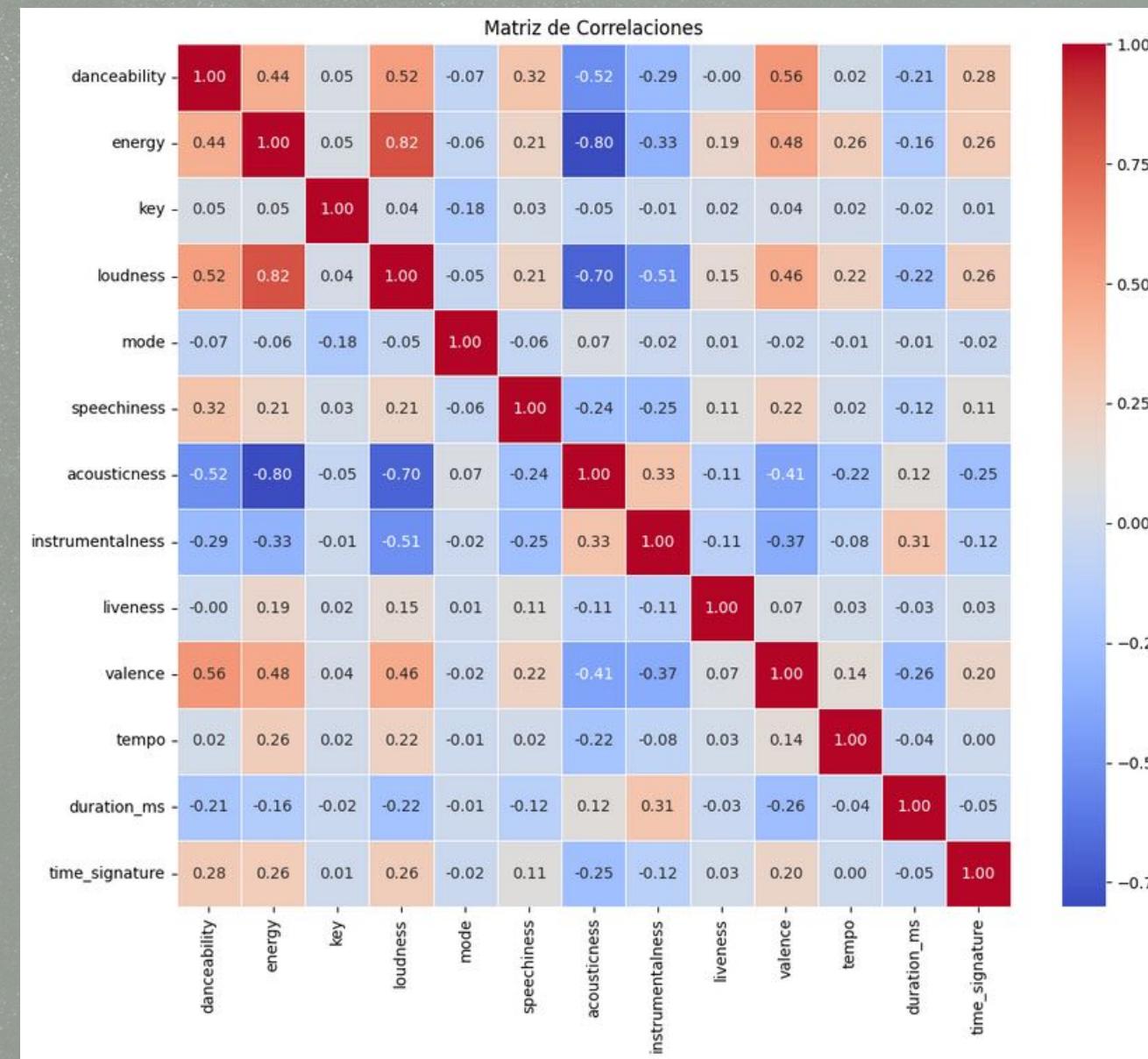
- MODE: indica la modalidad (mayor o menor) de una pista, el tipo de escala de la que se deriva su contenido melódico. El mayor está representado por 1 y el menor es 0.
- SPEECHINESS: detecta la presencia de palabras habladas en una pista. Cuanto más exclusivamente hablada sea la grabación, más cercano a 1,0 será el valor del atributo. Los valores superiores a 0,66 describen pistas que probablemente estén compuestas exclusivamente de palabras habladas. Los valores entre 0,33 y 0,66 describen pistas que pueden contener música y voz, ya sea en secciones o en capas, incluidos casos como la música rap. Los valores inferiores a 0,33 probablemente representen música y otras pistas que no sean de voz.
- TEMPO: el tempo general estimado de una pista en pulsaciones por minuto (BPM). En terminología musical, el tempo es la velocidad o el ritmo de una pieza determinada y se deriva directamente de la duración promedio del tiempo.
- TIME_SIGNATURE: un compás estimado. El tipo de compás es una convención de notación para especificar cuántos tiempos hay en cada compás. El tipo de compás varía de 3 a 7, la mayor parte de las pistas sigue el compás 4/4.
- VALENCE: una medida de 0,0 a 1,0 que describe la positividad musical que transmite una pista. Las pistas con valencia alta suenan más positivas (por ejemplo, felices, alegres, eufóricas), mientras que las pistas con valencia baja suenan más negativas (por ejemplo, tristes, deprimidas, enojadas).
- KEY: la tonalidad de la canción, indicada por un número que representa la clave tonal. El rango es de -1 a 11.

MATRIZ DE CORRELACIONES CON HEAT MAP

Energy y Loudness: Una correlación positiva fuerte (0.82), indicando que a medida que la energía de una canción aumenta, también lo hace su volumen. Esto tiene sentido, ya que las canciones más energéticas suelen ser más ruidosas y dinámicas.

Acousticness y Loudness: También hay una correlación negativa significativa (-0.70) entre las dos, indicando que las canciones más acústicas suelen ser más silenciosas. Esto se puede explicar por el hecho de que las canciones acústicas se graban con menos instrumentos y efectos que las que usan amplificación o sintetización.

Acousticness y Energy: Una correlación negativa fuerte (-0.80), lo que sugiere que las canciones más acústicas tienden a tener menos energía. Esto también es lógico, ya que las canciones acústicas suelen ser más suaves y tranquilas que las eléctricas o electrónicas.



Danceability y Valence: Una correlación positiva moderada (0.56), lo que indica que las canciones más bailables suelen tener una valencia más alta. La valencia es una medida de la positividad o negatividad que transmite una canción, por lo que se puede inferir que las canciones más bailables son más alegres o festivas.

Tempo y Energy: Una correlación positiva débil (0.21), lo que significa que las canciones con un tempo más rápido suelen tener una energía más alta. El tempo es una medida de la velocidad o ritmo de una canción, expresado en pulsaciones por minuto (BPM), mientras que la energía es una medida de la intensidad o actividad de una canción. Esto se puede explicar por el hecho de que las canciones más rápidas suelen ser más dinámicas y estimulantes que las más lentas.

Instrumentalness y Acousticness: Una correlación positiva moderada (0.33), lo que sugiere que las canciones más instrumentales suelen ser más acústicas.

La instrumentalidad es una medida de la presencia de voces humanas en una canción, mientras que la acústica es una medida de la ausencia de sonidos eléctricos o sintetizados. Esto implica que las canciones más instrumentales se basan más en instrumentos naturales o tradicionales que en efectos o procesamientos digitales.

Para hacer el train test, descartaremos 'mode' y 'key' por su baja correlación con el resto de variables, y su posible inferencia negativa en nuestro proceso de clasificación de géneros musicales

3- PREPROCESADO DATASET 1

Eliminamos las variables que no necesitamos para nuestro modelo de ML. Destacar que entre ellas se encuentran 'track_poularity' y 'artist_popularity', que no tienen ninguna influencia alguna en la clasificación de géneros musicales:

```
df_spotify_cleaned = df_spotify.drop(['track_id', 'playlist_url', 'playlist_name', 'track_poularity',
                                         'album_cover', 'artist_genres', 'artist_popularity'], axis=1)
df_spotify_cleaned
```

Convertimos la variable objetivo (target) en categórica:

```
df_spotify_cleaned['genre'] = df_spotify_cleaned['genre'].astype('category')
```

Comprobamos que no hay valores nulos:

```
[ ] df_spotify_cleaned.isnull().sum()

track_name      0
artist_name     0
album           0
danceability    0
energy          0
key             0
loudness        0
mode            0
speechiness     0
acousticness    0
instrumentalness 0
liveness        0
valence         0
tempo           0
duration_ms     0
time_signature   0
genre           0
dtype: int64
```

Tenemos 538 duplicados:

```
duplicated_count = df_spotify_cleaned['track_name'].value_counts()
# Filtramos solo las que tienen más de una ocurrencia (duplicados)
duplicates_count = duplicated_count[duplicated_count > 1]

print("Conteo de TrackName Duplicados:")
duplicates_count

Conteo de TrackName Duplicados:
Stardust              7
Summertime             5
Body And Soul          5
Sweet Home Chicago     5
There Is No Greater Love 5
..
September In The Rain  2
April In Paris          2
Rollin' And Tumblin'   2
Night And Day           2
Bang Bang               2
Name: track_name, Length: 538, dtype: int64
```

Eliminamos los duplicados:

```
df_spotify_final = df_spotify_cleaned.drop_duplicates(subset=['track_name', 'artist_name'], keep='first')

print("DataFrame final sin duplicados:")
df_spotify_final
```

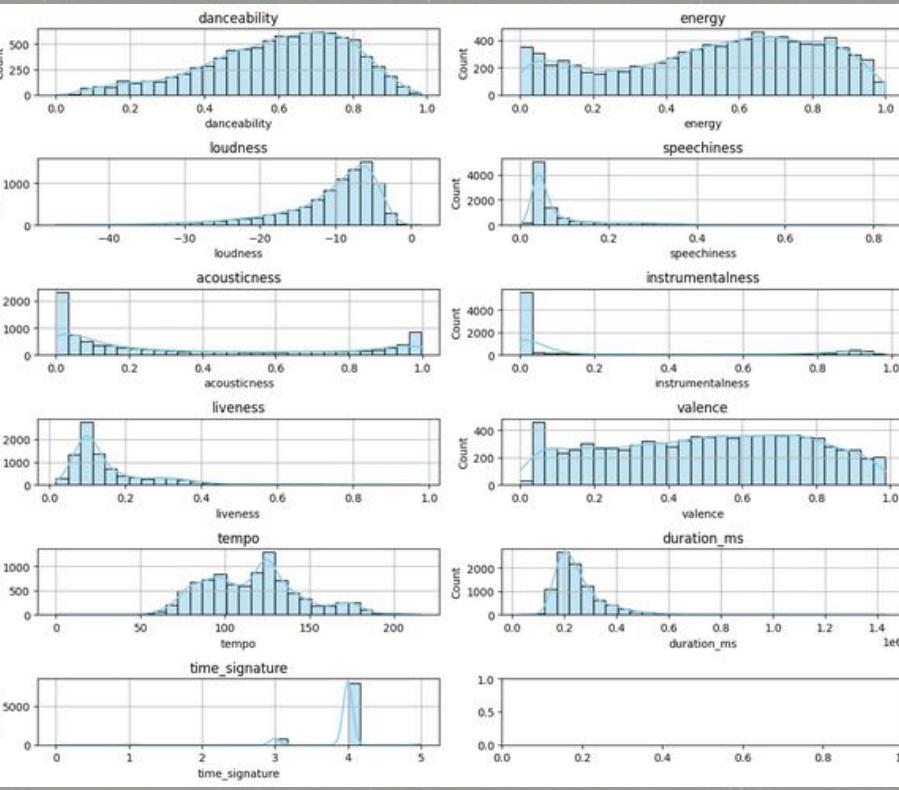
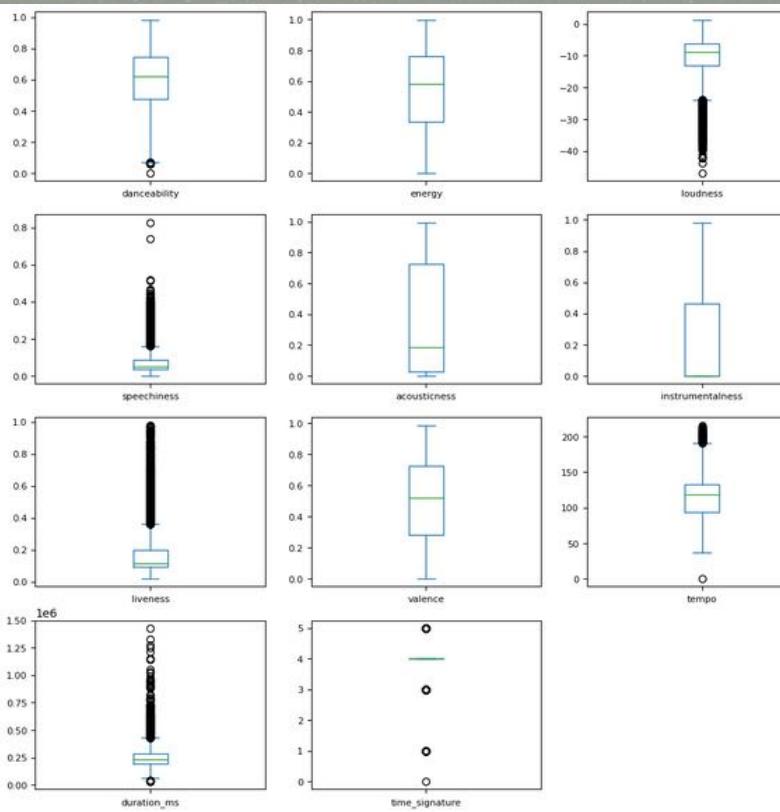
Obtenemos el primer dataframe sin duplicados:

	track_name	artist_name	album	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature	genre
0	Working Man	Otis Rush	Mourning In The Morning	0.63	0.62	0	-10.95	1	0.04	0.49	0.00	0.20	0.84	103.36	147800	4	blues
1	Long Way Home	Clarence "Gatemouth" Brown	Long Way Home	0.73	0.05	11	-22.56	0	0.04	0.91	0.05	0.12	0.42	78.03	338333	4	blues
2	She's A Sweet One	Junior Wells	Calling All Blues - The Chief, Profile & USA R...	0.70	0.48	1	-12.21	1	0.05	0.15	0.03	0.20	0.71	122.86	181786	4	blues
3	Help Me	Sonny Boy Williamson II	More Real Folk Blues	0.74	0.44	5	-9.62	0	0.04	0.60	0.02	0.61	0.77	114.22	188200	4	blues
4	Take Out Some Insurance	Jimmy Reed	Rockin' With Reed	0.75	0.29	9	-14.44	1	0.05	0.66	0.00	0.12	0.57	111.33	143332	4	blues
...	
8977	Walking with Clouds	Transllusion	The Opening of the Cerebral Gate	0.78	0.64	1	-9.45	1	0.05	0.00	0.76	0.11	0.23	135.34	212160	4	electronic
8978	Start To Move	Oscar Mulero	Mannequin	0.72	0.83	1	-9.15	1	0.11	0.01	0.94	0.12	0.42	133.99	317483	3	electronic
8979	Take Me Away - Truncate Remix	DJ 3000	Take Me Away	0.72	0.65	9	-11.22	0	0.05	0.00	0.94	0.12	0.18	130.01	355586	4	electronic
8980	Funny	Mr. De'	Follow the Leader 4	0.76	0.45	1	-10.23	0	0.07	0.03	0.86	0.11	0.69	148.00	236333	4	electronic
8981	Liquid Slow - Mixed	Chris Liebing	Mixmag Presents Charlotte de Witte (DJ Mix)	0.68	0.57	1	-15.71	1	0.04	0.01	0.90	0.12	0.41	134.77	335063	4	electronic

Creamos un nuevo dataframe, con las variables / features seleccionadas para el train-test. Hacemos drop de las variables categóricas (a excepción de 'genre' que es nuestro target), así como las variables 'key' y 'mode' que, como hemos considerado previamente, quedarán fuera del proceso.

```
[ ] df_features = df_spotify_final.drop(['track_name', 'artist_name', 'album', 'mode', 'key'], axis=1)
df_features
```

	danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature	genre
0	0.63	0.62	-10.95	0.04	0.49	0.00	0.20	0.84	103.36	147800	4	blues
1	0.73	0.05	-22.56	0.04	0.91	0.05	0.12	0.42	78.03	338333	4	blues
2	0.70	0.48	-12.21	0.05	0.15	0.03	0.20	0.71	122.86	181786	4	blues
3	0.74	0.44	-9.62	0.04	0.60	0.02	0.61	0.77	114.22	188200	4	blues
4	0.75	0.29	-14.44	0.05	0.66	0.00	0.12	0.57	111.33	143332	4	blues
...
8977	0.78	0.64	-9.45	0.05	0.00	0.76	0.11	0.23	135.34	212160	4	electronic
8978	0.72	0.83	-9.15	0.11	0.01	0.94	0.12	0.42	133.99	317483	3	electronic
8979	0.72	0.65	-11.22	0.05	0.00	0.94	0.12	0.18	130.01	355586	4	electronic
8980	0.76	0.45	-10.23	0.07	0.03	0.86	0.11	0.69	148.00	236333	4	electronic
8981	0.68	0.57	-15.71	0.04	0.01	0.90	0.12	0.41	134.77	335063	4	electronic
8982 rows × 12 columns												



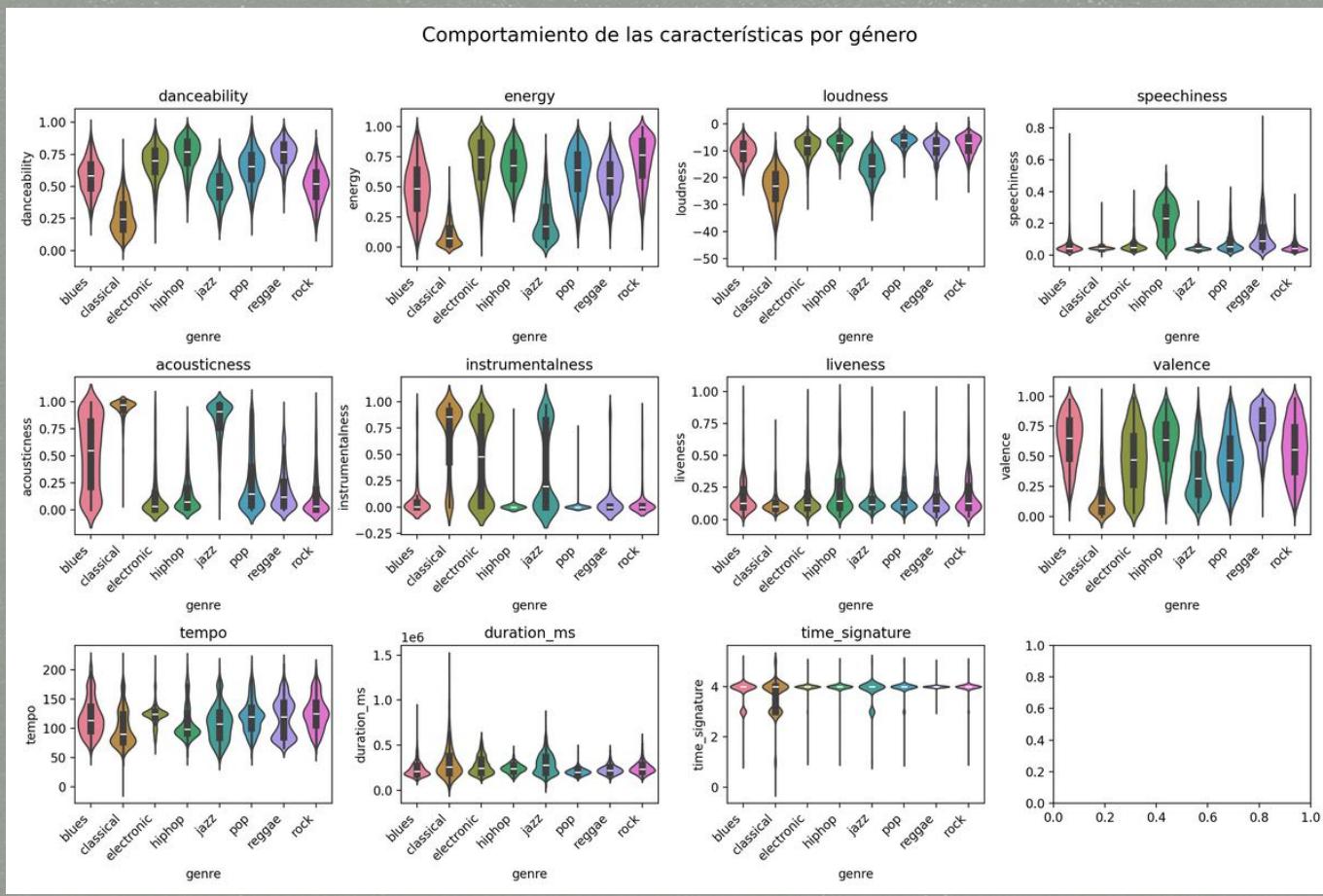
Tanto los boxplots como los histogramas indican que las variables no siguen una distribución normal, y algunas de ellas tienen outliers. Esto nos lo confirma también el test de Shapiro-Wilk. Un p-value < 0.05 (obtenemos un valor generalizado de 0.000), indica que podemos rechazar la hipótesis nula de que los datos siguen una distribución normal o gaussiana.

```
columns = ['danceability', 'energy', 'loudness', 'speechiness',
           'acousticness', 'instrumentalness', 'liveness', 'valence', 'tempo',
           'duration_ms', 'time_signature']

#Realizamos el test de Shapiro-Wilk para cada columna
for column in columns:
    stat, p = shapiro(df_features[column])
    print(f'Columna: {column}, Estadística: {stat:.3f}, p-valor: {p:.3f}')

Columna: danceability, Estadística: 0.969, p-valor: 0.000
Columna: energy, Estadística: 0.950, p-valor: 0.000
Columna: loudness, Estadística: 0.861, p-valor: 0.000
Columna: speechiness, Estadística: 0.668, p-valor: 0.000
Columna: acousticness, Estadística: 0.818, p-valor: 0.000
Columna: instrumentalness, Estadística: 0.653, p-valor: 0.000
Columna: liveness, Estadística: 0.725, p-valor: 0.000
Columna: valence, Estadística: 0.963, p-valor: 0.000
Columna: tempo, Estadística: 0.980, p-valor: 0.000
Columna: duration_ms, Estadística: 0.807, p-valor: 0.000
Columna: time_signature, Estadística: 0.377, p-valor: 0.000
```

Comportamiento de las características por género



Bailabilidad (danceability): Los géneros pop, hip hop y reggae tienden a tener una alta bailabilidad. Por otro lado, los géneros classical y blues muestran una bailabilidad más baja.

Energía (energy): Los géneros rock, pop y hip hop muestran una alta energía, lo que podría indicar que las canciones de estos géneros son más intensas y rápidas. Por su parte, géneros como classical y jazz tienen valores más bajos, lo que podría indicar que las canciones de estos géneros son más tranquilas y lentas.

Volumen (loudness): Los géneros rock y pop tienden a tener un volumen más alto, lo que podría indicar que las canciones de estos géneros son más ruidosas. En cambio, el género classical muestra un volumen más bajo.

Habla (speechiness): El género hip hop muestra un alto speechiness, lo que indica que las canciones de este género contienen más palabras habladas. Por otro lado, los géneros classical y electronic muestran valores más bajos.

Acústica (acousticness): Los géneros classical y jazz muestran una alta acústica. Por otro lado, los géneros rock y electronic muestran una acústica más baja, esto indica que las canciones de estos géneros son menos propensas a contener sonidos acústicos.

Instrumentalidad (instrumentalness): Los géneros classical y electronic muestran una alta instrumentalidad. Por otro lado, los géneros pop y hip hop muestran una instrumentalidad más baja.

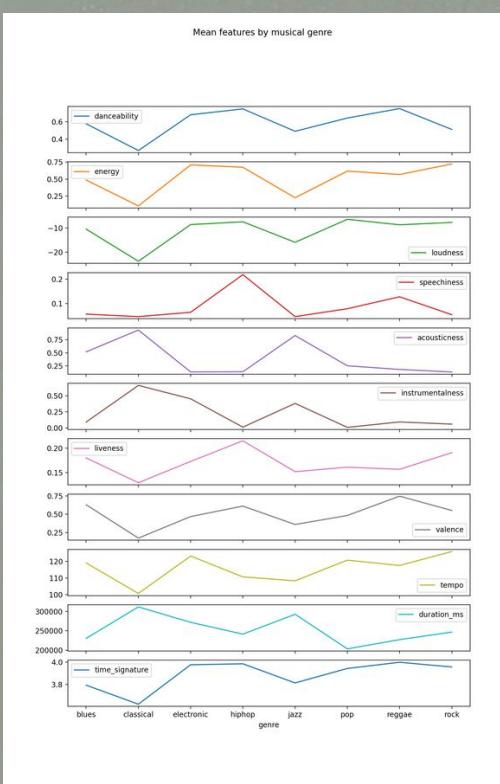
Vivacidad (liveness): Los géneros rock y jazz muestran una alta vivacidad, lo que podría indicar que las canciones de estos géneros son más propensas a contener sonidos de una audiencia en vivo. A diferencia de esto, los géneros pop y electronic muestran una vivacidad más baja.

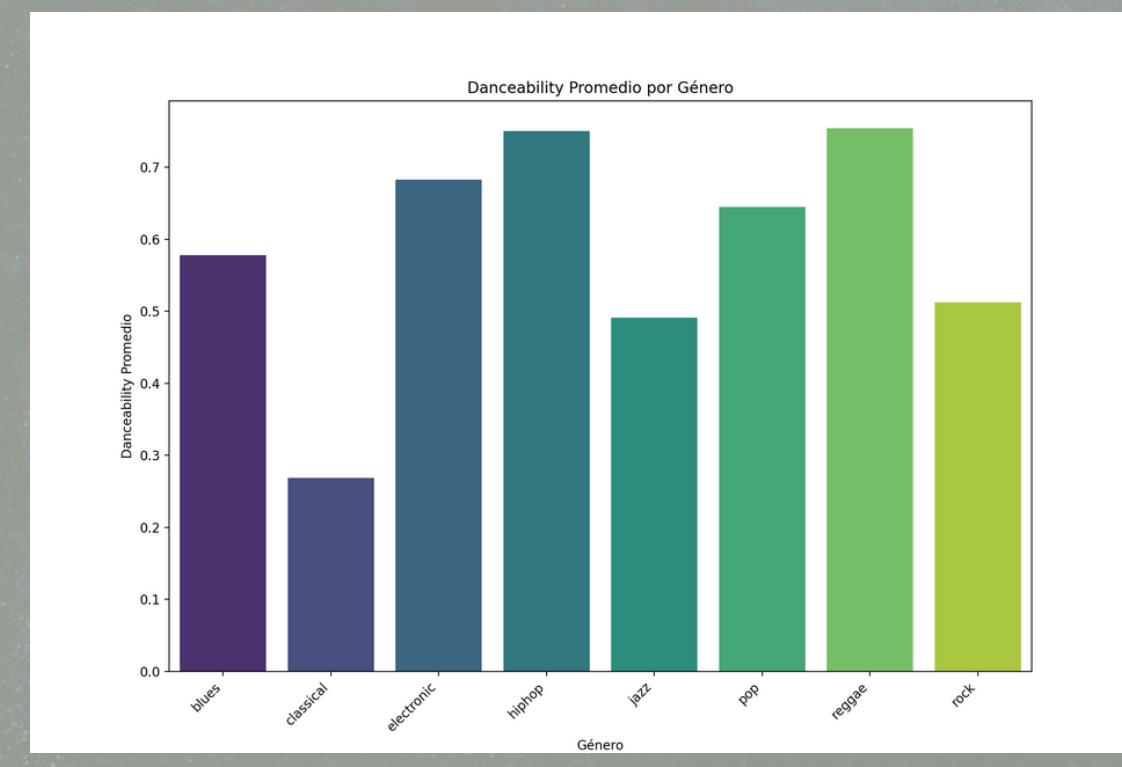
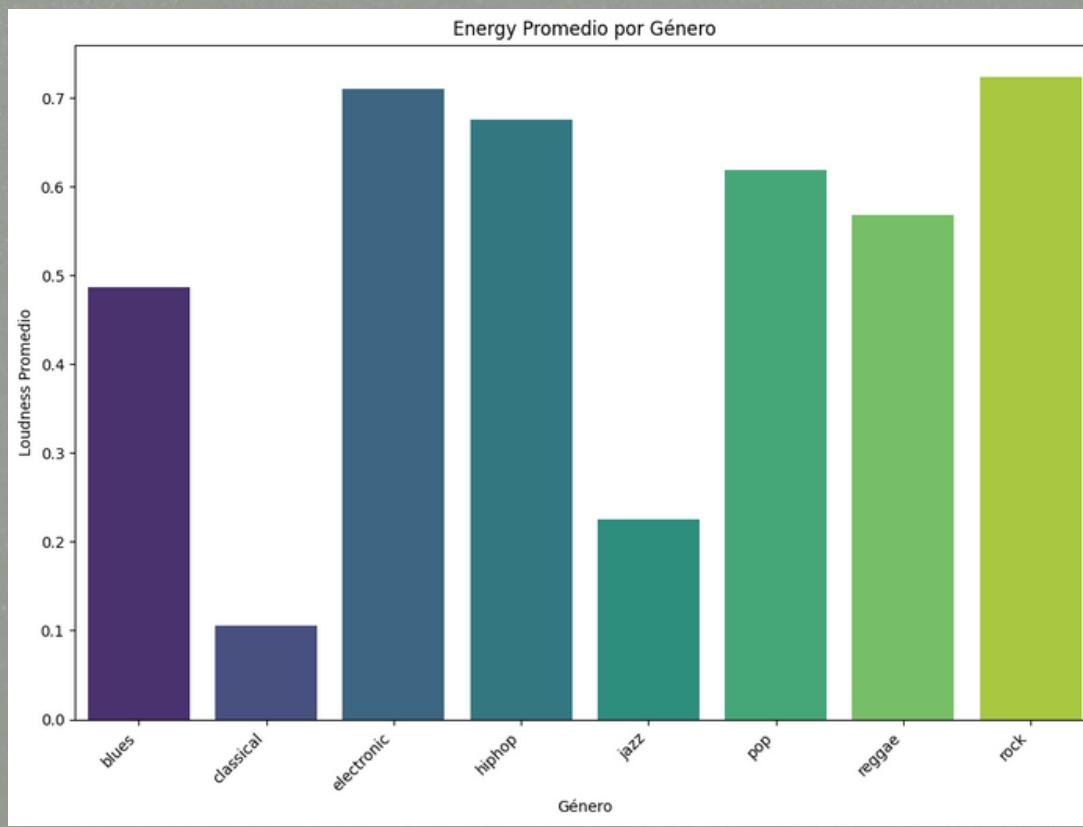
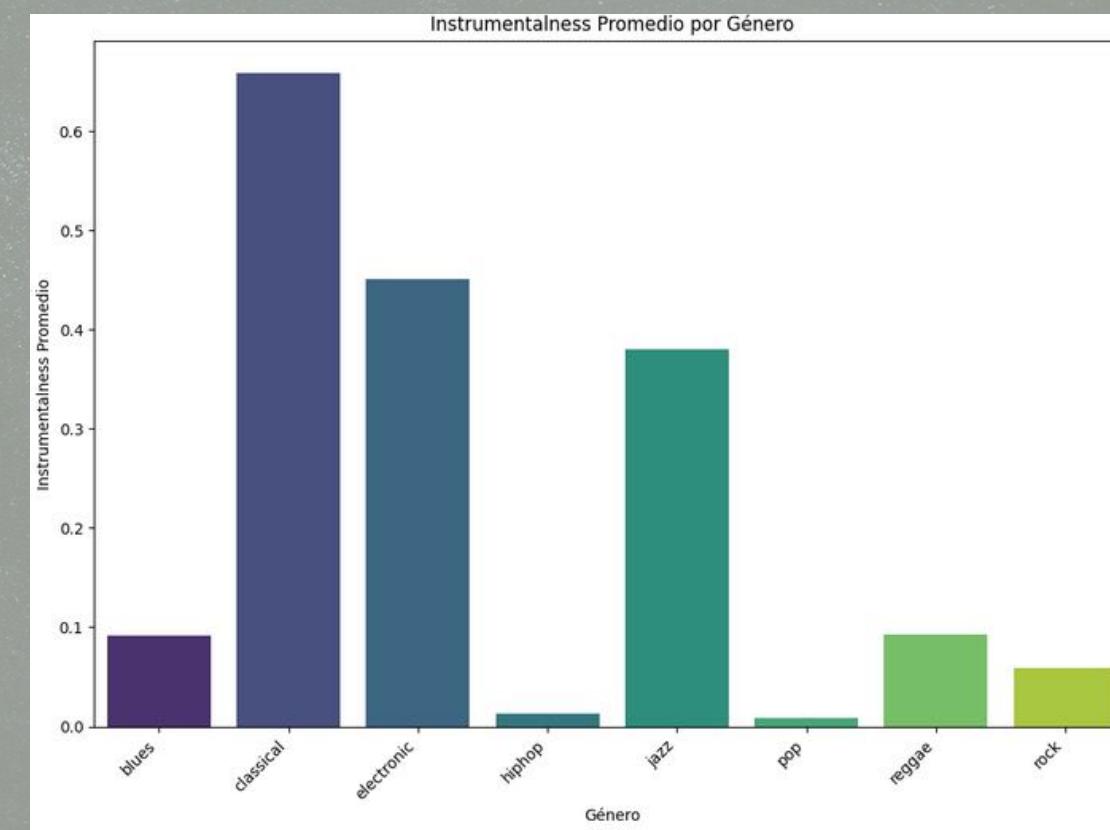
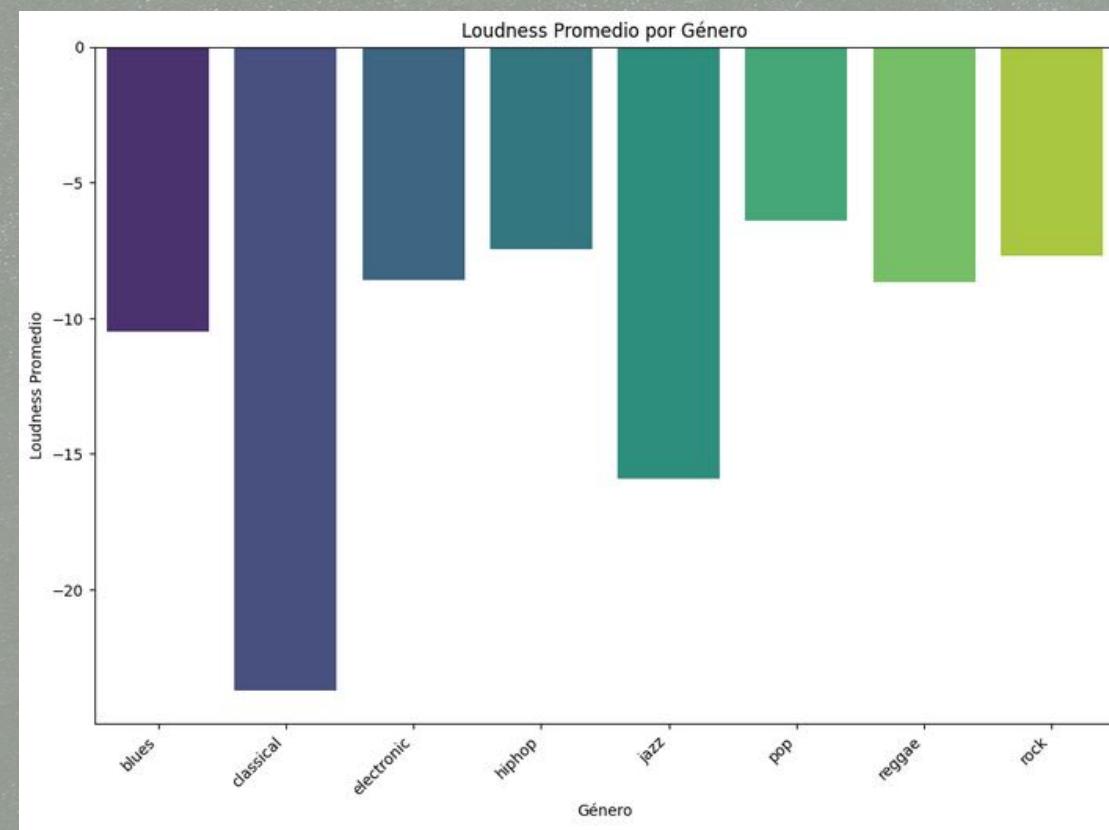
Valencia (valence): Los géneros pop y reggae muestran una alta valencia, lo que indica que las canciones de estos géneros son más propensas a transmitir una sensación positiva. Géneros blues y classical muestran una valencia más baja, da a entender que las canciones de estos géneros son menos propensas a transmitir una sensación positiva.

Tempo: Los géneros electronic y rock muestran un tempo más rápido. Si lo comparamos con el blues y classical, vemos que éstos tienen un tempo más lento.

Firma de tiempo (time signature): La mayoría de los géneros tienden a tener una firma de tiempo similar, con un patrón rítmico común, como 4/4 (cuatro tiempos por compás), que es muy común en la música popular.

Duración (duration_ms): La duración de una canción puede variar significativamente dependiendo del género. Los géneros classical y jazz podrían tener una duración más larga en promedio, lo que tiene sentido ya que estas canciones a menudo incluyen solos instrumentales largos.





4 - MACHINE LEARNING

Aplicamos Label Encoder a la variable target y hacemos la división train test

```
X = df_features.drop('genre', axis=1)
y = df_features['genre']

# Aplicamos LabelEncoder a la variable objetivo
le = LabelEncoder()
y = le.fit_transform(y)

# Dividimos los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Aplicamos Robust Scaler

```
# Inicializamos el RobustScaler
robust_scaler = RobustScaler()

# Ajustamos y transformamos los datos de entrenamiento
X_train_scaled = robust_scaler.fit_transform(X_train)

# transformamos los datos de prueba
X_test_scaled = robust_scaler.transform(X_test)
```

Aplicamos Random Forest

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
RandomForestClassifier	0.71	0.70	None	0.71	2.43
XGBClassifier	0.70	0.69	None	0.69	4.55
ExtraTreesClassifier	0.70	0.69	None	0.70	1.32
LGBMClassifier	0.69	0.69	None	0.69	1.81
SVC	0.68	0.68	None	0.68	2.07
NuSVC	0.66	0.65	None	0.66	4.39
BaggingClassifier	0.66	0.65	None	0.66	0.75
QuadraticDiscriminantAnalysis	0.64	0.64	None	0.64	0.03
LogisticRegression	0.64	0.63	None	0.64	0.59
KNeighborsClassifier	0.63	0.62	None	0.63	0.28
LinearDiscriminantAnalysis	0.62	0.62	None	0.63	0.07
CalibratedClassifierCV	0.63	0.61	None	0.62	9.47
LinearSVC	0.62	0.61	None	0.62	2.84
NearestCentroid	0.58	0.59	None	0.59	0.05
LabelSpreading	0.59	0.58	None	0.59	4.96
LabelPropagation	0.59	0.58	None	0.59	2.40
GaussianNB	0.58	0.58	None	0.58	0.03
SGDClassifier	0.58	0.57	None	0.58	0.26
RidgeClassifier	0.58	0.56	None	0.57	0.05
RidgeClassifierCV	0.58	0.56	None	0.56	0.05
DecisionTreeClassifier	0.57	0.56	None	0.57	0.16
BernoulliNB	0.51	0.50	None	0.51	0.04

```
Inicializamos el modelo Random Forest Classifier:

# clasificador de Random Forest
rf_clf = RandomForestClassifier(n_estimators=100, random_state=42)

[ ] # Entrenamos el modelo
rf_clf.fit(X_train, y_train)

RandomForestClassifier(random_state=42)

[ ] # Realizamos predicciones en el conjunto de prueba
y_pred_rf = rf_clf.predict(X_test)
# informe de clasificación
print(classification_report(y_test, y_pred_rf))

precision    recall   f1-score   support
          0       0.58      0.54      0.56      193
          1       0.90      0.85      0.87      200
          2       0.80      0.78      0.79      358
          3       0.71      0.78      0.75      198
          4       0.67      0.71      0.69      177
          5       0.64      0.68      0.66      239
          6       0.68      0.63      0.66      209
          7       0.64      0.65      0.64      223

accuracy                           0.71      1797
macro avg       0.70      0.70      0.70      1797
weighted avg    0.71      0.71      0.71      1797

Se confirman los resultados ya obtenidos con el Lazy Classifier. Obtenemos hiperparámetros:

[ ] rf_clf.get_params()

{'bootstrap': True,
 'ccp_alpha': 0.0,
 'class_weight': None,
 'criterion': 'gini',
 'max_depth': None,
 'max_features': 'sqrt',
 'max_leaf_nodes': None,
 'max_samples': None,
 'min_impurity_decrease': 0.0,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 100,
 'n_jobs': None,
 'oob_score': False,
 'random_state': 42,
 'verbose': 0,
 'warm_start': False}
```

```

Ajustamos hiperparámetros:

param_dist_rf = {
    'n_estimators': [int(x) for x in np.linspace(start=100, stop=500, num=5)],
    'max_features': ['auto', 'sqrt'],
    'max_depth': [int(x) for x in np.linspace(10, 50, num=5)] + [None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2],
    'bootstrap': [True, False]
}

Hacemos Randomized Search. Dada la complejidad del modelo y alto número de datos a procesar, la opción de GridSearch tiene un alto costo computacional, y optamos por tanto por Randomized Search, el cual ofrece resultados similares con un costo computacional no tan elevado.

random_search_rf = RandomizedSearchCV(rf_clf, param_dist_rf, n_iter=50, cv=5, verbose = 2, scoring='accuracy', n_jobs=-1)

random_search_rf.fit(X_train, y_train)

```

Fitting 5 folds for each of 50 candidates, totaling 250 fits

- RandomizedSearchCV
- estimator: RandomForestClassifier
- RandomForestClassifier

Obtenemos los mejores hiperparámetros:

```

[1]: # Mejores parámetros encontrados
best_params_rf = random_search_rf.best_params_
print("Mejores hiperparámetros encontrados:", best_params_rf)
Mejores hiperparámetros encontrados: {'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': 30, 'bootstrap': False}

```

Tras ajustar hiperparámetros, el rendimiento del modelo Random Forest, no experimenta mejoras.

```

best_model_rf = random_search_rf.best_estimator_
y_pred_rf = best_model_rf.predict(X_test)

print("Informe de clasificación:\n", classification_report(y_test, y_pred_rf))

Informe de clasificación:
      precision    recall  f1-score   support
          0       0.59     0.58     0.58     193
          1       0.90     0.84     0.87     200
          2       0.79     0.78     0.78     358
          3       0.72     0.73     0.72     198
          4       0.67     0.70     0.69     177
          5       0.64     0.69     0.66     239
          6       0.66     0.62     0.64     209
          7       0.64     0.66     0.65     223

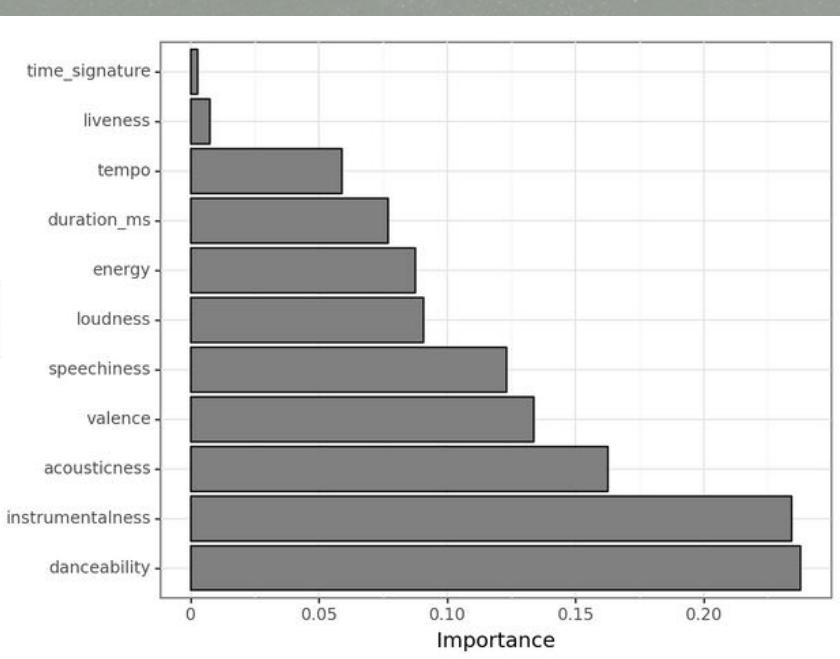
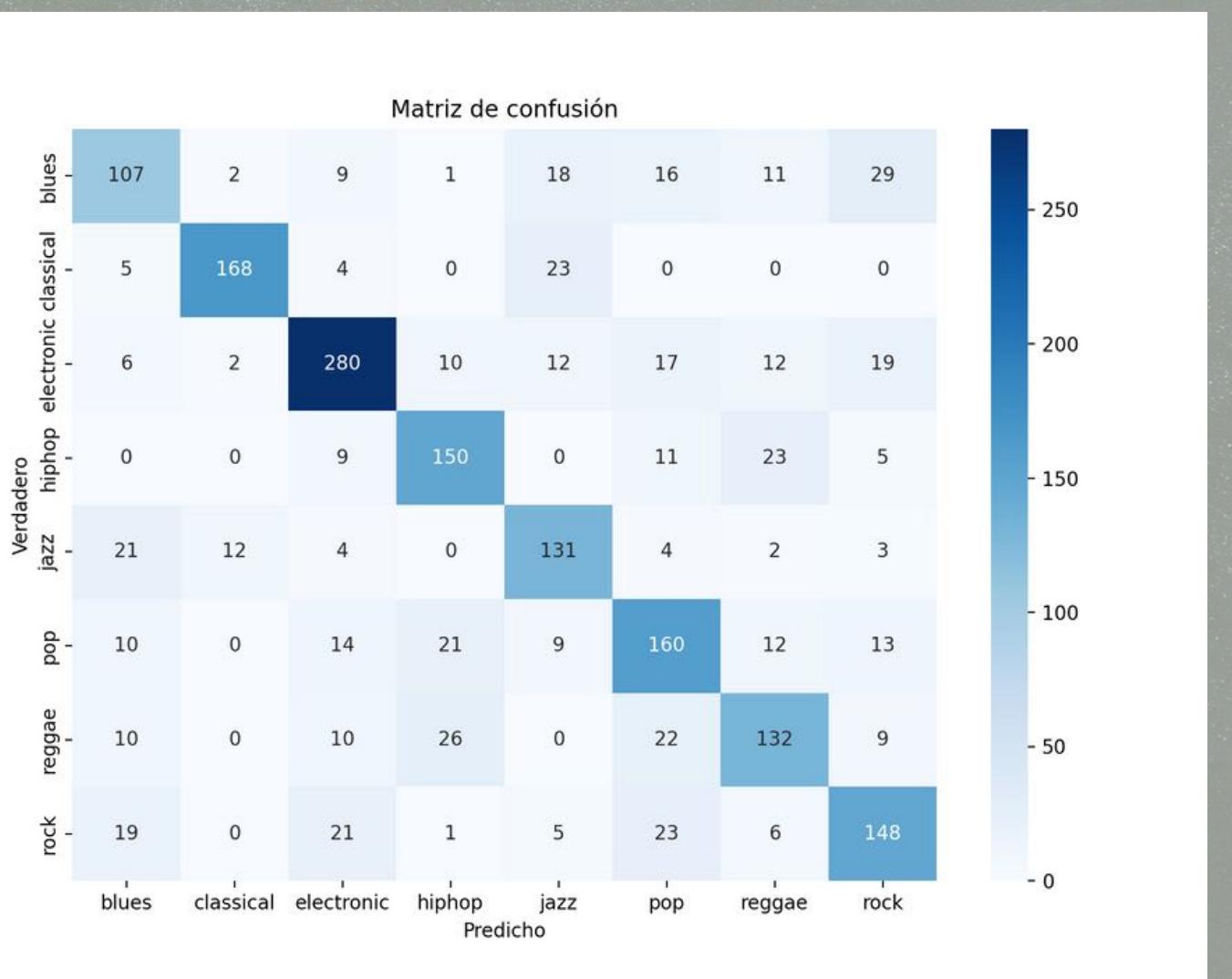
    accuracy                           0.70      1797
   macro avg       0.70     0.70     0.70      1797
weighted avg       0.71     0.71     0.71      1797

```

Cross Validation

Resultados de Validación Cruzada:
Precisión en cada pliego: [0.69589422 0.68475992 0.71607516 0.71189979 0.6986778]
Precisión media: 0.7014613778705636
Desviación estándar de la precisión: 0.011315505427517265

Importancia/influencia de las características en la clasificación de géneros



Al observar la diagonal de la matriz de confusión, podemos verificar un grado de acierto bastante aceptable en la predicción de géneros, alcanzando un 71% de precisión según nuestro modelo. Sin embargo, también notamos un porcentaje de error en la asignación:

Blues: Clasifica ocasionalmente erróneamente géneros como jazz, pop, reggae y rock. Esto se debe a que el rock y el jazz tienen raíces en el blues, lo que puede llevar a confusiones en la clasificación.

Classical: La clasificación es altamente precisa, con solo algunas pistas de jazz y blues clasificadas erróneamente. Esto podría deberse a la influencia directa de la música clásica en estos géneros.

Electronic: Puede clasificar como electrónica a la música rock, pop y hip hop, ya que estos géneros a menudo incorporan elementos electrónicos en sus composiciones.

Hip Hop: En ocasiones, puede clasificar géneros como reggae, pop y electrónico. Esto refleja la naturaleza fusionada del hip hop, que incorpora elementos de diversos géneros musicales.

Jazz: Proviene del blues y la música clásica, lo que puede resultar en clasificaciones erróneas como estos estilos en algunas ocasiones.

Pop: A excepción de la música clásica, el pop tiende a adoptar aspectos de varios géneros, lo que contribuye al número de clasificaciones no acertadas.

Reggae: Nace del ska y se mezcla con influencias como el country, soul, blues y rock, lo que se refleja en la matriz de confusión.

Rock: La dificultad en su clasificación radica en que el rock deriva del blues y puede incorporar elementos de la electrónica, como sintetizadores y samplers. Esto contribuye a las dificultades observadas en la matriz de confusión.

5 - PREPROCESADO DATASET 2

0	7meHLHBe4nmXzuXc0HDjk	Testify	The Battle Of Los Angeles	2ela0myWFgoHuttJytCxgX	[Rage Against The Machine]	[2d0hyoQ5ynDBnvAbjKORi]	1.00	1.00	False	0.47	...	0.07	0.03	0.00	0.36	0.50	117.91	210133.00	4.00	1999.00	1999-11-02
1	1wsRifRRtWyEapi0q2zob8	Guerrilla Radio	The Battle Of Los Angeles	2ela0myWFgoHuttJytCxgX	[Rage Against The Machine]	[2d0hyoQ5ynDBnvAbjKORi]	2.00	1.00	True	0.60	...	0.19	0.01	0.00	0.15	0.49	103.68	206200.00	4.00	1999.00	1999-11-02
2	1hr0flFK2qRG3fRF70pb7	Calm Like a Bomb	The Battle Of Los Angeles	2ela0myWFgoHuttJytCxgX	[Rage Against The Machine]	[2d0hyoQ5ynDBnvAbjKORi]	3.00	1.00	False	0.32	...	0.48	0.02	0.00	0.12	0.37	149.75	298893.00	4.00	1999.00	1999-11-02
3	2lbASgTSOD07MtLAXITW0	Mic Check	The Battle Of Los Angeles	2ela0myWFgoHuttJytCxgX	[Rage Against The Machine]	[2d0hyoQ5ynDBnvAbjKORi]	4.00	1.00	True	0.44	...	0.24	0.16	0.00	0.12	0.57	96.75	213640.00	4.00	1999.00	1999-11-02
4	1MQT1mpYQZ6l6Mqc56Hd07T	Sleep Now In the Fire	The Battle Of Los Angeles	2ela0myWFgoHuttJytCxgX	[Rage Against The Machine]	[2d0hyoQ5ynDBnvAbjKORi]	5.00	1.00	False	0.43	...	0.07	0.00	0.10	0.08	0.54	127.06	205600.00	4.00	1999.00	1999-11-02
...	
1140947	78pDVGo1Cjje81z&JJoK8	All or Nothing - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpSHoQ	[Whitesnake]	[3UbyYnVNT5DFXU4WgGpP]	8.00	2.00	False	0.62	...	0.06	0.02	0.14	0.24	0.39	132.53	220646.00	4.00	2019.00	2019-03-08
1140948	2ZWrlVLD6mnFSw0Cqy6WK	Spit It Out - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpSHoQ	[Whitesnake]	[3UbyYnVNT5DFXU4WgGpP]	9.00	2.00	False	0.47	...	0.05	0.01	0.00	0.12	0.72	132.31	254125.00	4.00	2019.00	2019-03-08
1140949	5xs8x2Jcth1GdM2yD1PB	Guilty of Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpSHoQ	[Whitesnake]	[3UbyYnVNT5DFXU4WgGpP]	10.00	2.00	False	0.44	...	0.08	0.04	0.02	0.33	0.65	174.23	203799.00	4.00	2019.00	2019-03-08
1140950	4pXFNBpsQatvDv2YqKoGr	Need Your Love so Bad - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpSHoQ	[Whitesnake]	[3UbyYnVNT5DFXU4WgGpP]	11.00	2.00	False	0.26	...	0.03	0.40	0.00	0.51	0.06	107.88	193766.00	3.00	2019.00	2019-03-08
1140951	8EM56vqZveT8UW8BqSu	Gambler - Eddie Kramer Mix; 1983; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	4tbNTP0YzfD13oC1UpSHoQ	[Whitesnake]	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

df_predictions_final.reset_index(drop=True, inplace=True)
df_predictions_final
name album artists danceability energy key loudness mode speechiness acousticness instrumentalness liveliness valence tempo duration_ms time_signature year
0 Testify The Battle Of Los Angeles ['Rage Against The Machine'] 0.47 0.98 7.00 -5.40 1.00 0.07 0.03 0.00 0.36 0.50 117.91 210133.00 4.00 1999
1 Guerrilla Radio The Battle Of Los Angeles ['Rage Against The Machine'] 0.60 0.96 11.00 -5.76 1.00 0.19 0.01 0.00 0.15 0.49 103.68 206200.00 4.00 1999
2 Calm Like a Bomb The Battle Of Los Angeles ['Rage Against The Machine'] 0.32 0.97 7.00 -5.42 1.00 0.48 0.02 0.00 0.12 0.37 149.75 298893.00 4.00 1999
3 Mic Check The Battle Of Los Angeles ['Rage Against The Machine'] 0.44 0.97 11.00 -5.83 0.00 0.24 0.16 0.00 0.12 0.57 96.75 213640.00 4.00 1999
4 Sleep Now In the Fire The Battle Of Los Angeles ['Rage Against The Machine'] 0.43 0.93 2.00 -6.73 1.00 0.07 0.00 0.10 0.08 0.54 127.06 205600.00 4.00 1999
...
1140946 Hungry for Love - UK Mix; 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] 0.63 0.95 2.00 -6.43 1.00 0.05 0.02 0.00 0.16 0.51 134.74 237761.00 4.00 2019
1140947 All or Nothing - UK Mix; 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] 0.62 0.94 2.00 -7.32 0.00 0.06 0.02 0.14 0.24 0.39 132.53 220646.00 4.00 2019
1140948 Spit It Out - UK Mix; 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] 0.47 0.93 9.00 -7.06 1.00 0.05 0.01 0.00 0.12 0.72 132.31 254125.00 4.00 2019
1140949 Guilty of Love - UK Mix; 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] 0.44 0.97 6.00 -5.49 0.00 0.08 0.04 0.02 0.33 0.65 174.23 203799.00 4.00 2019
1140950 Need Your Love so Bad - 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] 0.26 0.30 4.00 -10.92 1.00 0.03 0.40 0.00 0.51 0.06 107.88 193766.00 3.00 2019

Eliminamos las columnas que no necesitamos. El dataset pasa a llamarse df_predictions_cleaned:

```
[ ] df_predictions_cleaned = df_predictions.drop(['id', 'album_id', 'artist_ids', 'track_number', 'disc_number', 'explicit', 'release_date'], axis=1)
```

```
df_predictions_cleaned.isnull().sum()
```

Eliminaremos los NaNs

```
filas_con_nulos = df_predictions_cleaned.isnull().any(axis=1)

print(f'Filas con al menos un valor nulo: {filas_con_nulos}')
df_predictions_cleaned[filas_con_nulos]

Filas con al menos un valor nulo:
name album artists danceability energy key loudness mode speechiness acousticness instrumentalness liveliness valence tempo duration_ms time_signature year
1140951 Gambler - Eddie Kramer Mix; 1983; 2019 Remaster Slide It In (The Ultimate Edition; 2019 Remaster) ['Whitesnake'] NaN NaN
```

Obtenemos el dataframe sin NaNs

Convertimos year a integer (int64):

```
df_predictions_final['year'] = df_predictions_final['year'].astype('int64')
```

Eliminamos los corchetes y las comillas de los strings de la columna 'artists'

```
df_predictions_final['artists'] = df_predictions_final['artists'].str.replace("[", "")
df_predictions_final['artists'] = df_predictions_final['artists'].str.replace("]", "")
df_predictions_final['artists'] = df_predictions_final['artists'].str.replace("'", "")
```

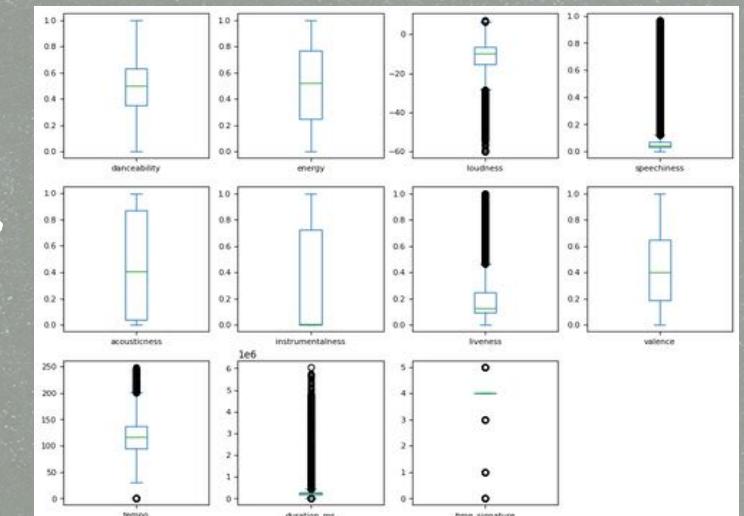
6- APLICACION DEL MODELO AL SEGUNDO DATAFRAME

Vamos a aplicar el mejor modelo obtenido con Random Forest en el segundo dataset, para predecir los géneros.

. Para ello, debemos tener las mismas variables/columnas que hemos entrenado en el primer dataset:

	danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature
0	0.47	0.98	-5.40	0.07	0.03	0.00	0.36	0.50	117.91	210133.00	4.00
1	0.60	0.96	-5.76	0.19	0.01	0.00	0.15	0.49	103.68	206200.00	4.00
2	0.32	0.97	-5.42	0.48	0.02	0.00	0.12	0.37	149.75	298893.00	4.00
3	0.44	0.97	-5.83	0.24	0.16	0.00	0.12	0.57	96.75	213640.00	4.00
4	0.43	0.93	-6.73	0.07	0.00	0.10	0.08	0.54	127.06	205600.00	4.00
...
1140946	0.63	0.95	-6.43	0.05	0.02	0.00	0.16	0.51	134.74	237761.00	4.00
1140947	0.62	0.94	-7.32	0.06	0.02	0.14	0.24	0.39	132.53	220646.00	4.00
1140948	0.47	0.93	-7.06	0.05	0.01	0.00	0.12	0.72	132.31	254125.00	4.00
1140949	0.44	0.97	-5.49	0.08	0.04	0.02	0.33	0.65	174.23	203799.00	4.00
1140950	0.26	0.30	-10.92	0.03	0.40	0.00	0.51	0.06	107.88	193766.00	3.00
1140951 rows x 11 columns											

Hacemos un BoxPlot: Confirmamos que no siguen una distribución normal, y algunas tienen outliers



Hacemos el mismo escalado que para el primer dataset.
Utilizamos Robust Scaler

# Inicializar el RobustScaler robust_scaler = RobustScaler()	# Ajustar y transformar todos los datos df_features_2_scaled = robust_scaler.fit_transform(df_features_2)	# Crear un nuevo DataFrame con los datos escalados df_features_2_scaled = pd.DataFrame(df_features_2_scaled, columns=df_features_2.columns)	# Mostrar el DataFrame escalado df_features_2_scaled
danceability energy loudness speechiness acousticness instrumentalness liveness valence tempo duration_ms time_signature	0 -0.11 0.88 0.51 0.79 -0.45 -0.01 1.56 0.22 0.03 -0.13 0.00	1 0.36 0.84 0.47 3.99 -0.47 -0.01 0.20 0.19 -0.30 -0.17 0.00	2 -0.67 0.87 0.51 12.18 -0.45 -0.01 -0.02 -0.07 0.77 0.66 0.00

	name	album	artists	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	duration_ms	time_signature	year	predicted_genre
0	Testify	The Battle Of Los Angeles	Rage Against The Machine	0.47	0.98	7.00	-5.40	1.00	0.07	0.03	0.00	0.36	0.50	117.91	210133.00	4.00	1999	7
1	Guerrilla Radio	The Battle Of Los Angeles	Rage Against The Machine	0.60	0.96	11.00	-5.76	1.00	0.19	0.01	0.00	0.15	0.49	103.68	206200.00	4.00	1995	7
2	Calm Like a Bomb	The Battle Of Los Angeles	Rage Against The Machine	0.32	0.97	7.00	-5.42	1.00	0.48	0.02	0.00	0.12	0.37	149.75	298893.00	4.00	1999	7
3	Mic Check	The Battle Of Los Angeles	Rage Against The Machine	0.44	0.97	11.00	-5.83	0.00	0.24	0.16	0.00	0.12	0.57	96.75	213640.00	4.00	1999	3
4	Sleep Now In the Fire	The Battle Of Los Angeles	Rage Against The Machine	0.43	0.93	2.00	-6.73	1.00	0.07	0.00	0.10	0.08	0.54	127.06	205600.00	4.00	1999	7
...	
1140946	Hungry for Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	0.63	0.95	2.00	-6.43	1.00	0.05	0.02	0.00	0.16	0.51	134.74	237761.00	4.00	2019	7
1140947	All or Nothing - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	0.62	0.94	2.00	-7.32	0.00	0.06	0.02	0.14	0.24	0.39	132.53	220646.00	4.00	2019	2
1140948	Spit It Out - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	0.47	0.93	9.00	-7.06	1.00	0.05	0.01	0.00	0.12	0.72	132.31	254125.00	4.00	2019	7
1140949	Guilty of Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	0.44	0.97	6.00	-5.49	0.00	0.08	0.04	0.02	0.33	0.65	174.23	203799.00	4.00	2019	7
1140950	Need Your Love so Bad - 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	0.26	0.30	4.00	-10.92	1.00	0.03	0.40	0.00	0.51	0.06	107.88	193766.00	3.00	2019	7
1140951 rows x 18 columns																		



Ahora aplicamos nuestro mejor modelo obtenido con Random Forest al conjunto de datos:

```
[ ] # Realizamos predicciones en este segundo conjunto de datos
predicted_genres = best_model_rf.predict(df_features_2)
```

```
[ ] predicted_genres
array([7, 7, 7, ..., 7, 5, 5])
```

Una vez obtenido el array con los géneros predichos (predicted_genres), los adjuntamos en una nueva columna en nuestro dataframe 'df_predictions_final'. Podemos ver la asignación de género a cada una de las canciones:

```
[ ] df_predictions_final['predicted_genre'] = predicted_genres
```

Hemos conseguido nuestro objetivo de predecir los géneros musicales del segundo dataset

7 - RESULTADOS

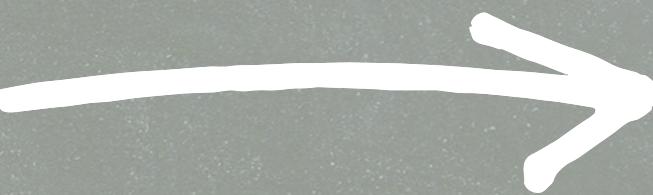
```
df_final_genre_predictions = df_predictions_final.drop(['danceability', 'energy', 'key', 'loudness',
                                                       'mode', 'speechiness', 'acousticness', 'instrumentalness', 'liveness',
                                                       'valence', 'tempo', 'duration_ms', 'time_signature'], axis=1)

df_final_genre_predictions
```

	name	album	artists	year	predicted_genre
0	Testify	The Battle Of Los Angeles	Rage Against The Machine	1999	7
1	Guerrilla Radio	The Battle Of Los Angeles	Rage Against The Machine	1999	7
2	Calm Like a Bomb	The Battle Of Los Angeles	Rage Against The Machine	1999	7
3	Mic Check	The Battle Of Los Angeles	Rage Against The Machine	1999	3
4	Sleep Now In the Fire	The Battle Of Los Angeles	Rage Against The Machine	1999	7
...
1140946	Hungry for Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	7
1140947	All or Nothing - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	2
1140948	Spit It Out - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	7
1140949	Guilty of Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	7
1140950	Need Your Love so Bad - 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	7

1140951 rows × 5 columns

Hacemos un mapping para obtener los nombres de los géneros



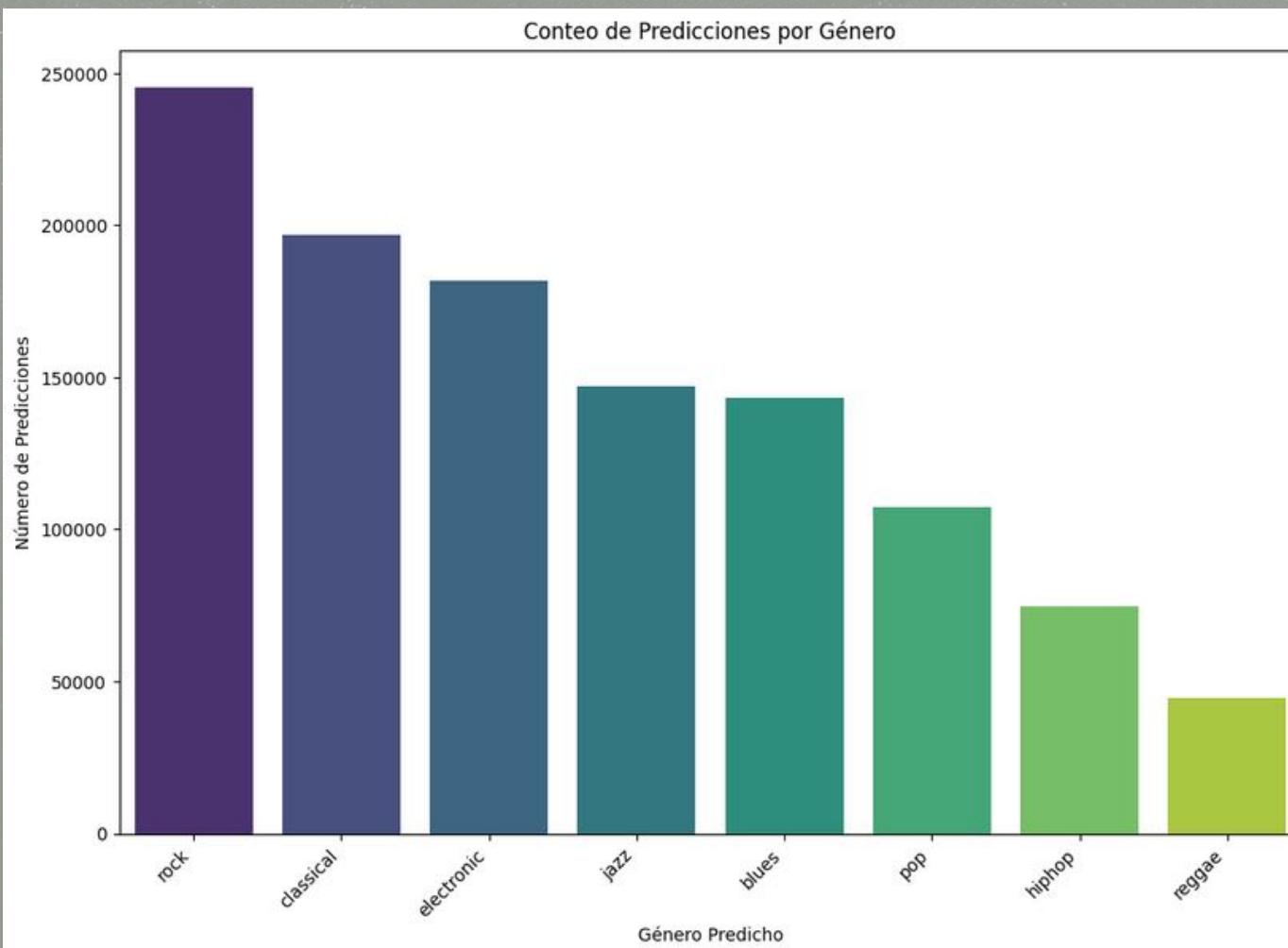
```
#mapeo de géneros
genre_mapping = dict(zip(codigos_asignados, equivalentes_originales))

# reemplazamos los códigos en el DataFrame
df_final_genre_predictions['predicted_genre'] = df_final_genre_predictions['predicted_genre'].map(genre_mapping)

df_final_genre_predictions
```

	name	album	artists	year	predicted_genre
0	Testify	The Battle Of Los Angeles	Rage Against The Machine	1999	rock
1	Guerrilla Radio	The Battle Of Los Angeles	Rage Against The Machine	1999	rock
2	Calm Like a Bomb	The Battle Of Los Angeles	Rage Against The Machine	1999	rock
3	Mic Check	The Battle Of Los Angeles	Rage Against The Machine	1999	hiphop
4	Sleep Now In the Fire	The Battle Of Los Angeles	Rage Against The Machine	1999	rock
...
1140946	Hungry for Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	rock
1140947	All or Nothing - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	electronic
1140948	Spit It Out - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	rock
1140949	Guilty of Love - UK Mix; 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	rock
1140950	Need Your Love so Bad - 2019 Remaster	Slide It In (The Ultimate Edition; 2019 Remaster)	Whitesnake	2019	rock

1140951 rows × 5 columns



rock	245381
classical	196929
electronic	181582
jazz	147130
blues	143209
pop	107482
hiphop	74738
reggae	44500

Adicionalmente podemos filtrar los géneros en dataframes individualizados:

Classical

	name	album	artists	year	predicted_genre	
834860	Shadowfall	Field Report	Android Invasion	2018	classical	
464318	Sem-i Ruhuna Cismimi Pervane Durendum - My Bod...	The Music of Islam, Vol. 14: Mystic Music Thro...	Galata Mevlevi Music and Sema Ensemble	1997	classical	
533964	Piano Sonata No. 16 in F Minor, DLR VI:1:16 (a...	Granados, E.: Piano Music, Vol. 9 - Transcri...	Enrique Granados, Douglas Riva	2007	classical	
722019	Serenade in G Major (original version): I. Pro...	Moeran: Cello Concerto - Serenade	Ernest John Moeran, Ulster Orchestra, Joann Fa...	2013	classical	
611588	Rooftop	Vespertine	This Will Destroy You	2020	classical	
153591	Don Giovanni, K. 527: In quali eccessi (Rec.)	Mozart	Wolfgang Amadeus Mozart, Annette Dasch, Marc P...	2008	classical	
1128623	Motet in C Major, Op. 3, No. 5, "Per ogni fest..."	Bonporti, F.A.: Motets, Op. 3, Nos. 1-6	Francesco Antonio Bonporti, Ellen Hargis, Ense...	1996	classical	
340300	Klavierkonzert Nr. 21 C-Dur, K. 467, "Elvira M..."	Mozart (The Best Of)	Wolfgang Amadeus Mozart, Jenó Jandó, Concentus...	1997	classical	
284643	Majestic Peace	Where Eagles Soar	"Dan Gibson's Solitudes"	2013	classical	
859075	Ich bin eine Stimme	Leander, Zarah: Centenary Edition - The Comple...	Peter Igelhoff, Zarah Leander, Odeon-Künstler...	2007	classical	

Rock

	name	album	artists	year	predicted_genre
507083	Dry As A Bone	Feels Like Family	Lauren Ellis	2005	rock
376785	L.F.D.Y.	A Beat Missing or a Silence Added	The Vacancies	2005	rock
1155399	Australia's Pride	Save The World: Earth Lives Or The Next Mars?	Eddie Florano	2007	rock
1154126	Sunday - 1987; Live on Wers	If Only You Were Dead	The Lemonheads	2014	rock
696118	Inspiration Points	Let It Beard	Boston Spaceships	2011	rock
618922	Just Right	Nostalgia	July For Kings	2005	rock
48970	The Silent Creature	Up the Tombstones!!! Live 2000	Deceased...	2001	rock
17526	Love's a Loaded Gun	Hey Stopid	Alice Cooper	1991	rock
722210	Drawing Down	Fiber	Dead Register	2016	rock
236417	Can't Get Enough - En Directo	Bailaré sobre tu tumba	Sinistro Total	1985	rock

Jazz

	name	album	artists	year	predicted_genre	
1054099	Three Clay Pots	The Peach Orchard	William Parker, Cooper-Moore, Rob Brown, Susie...	1994	jazz	
694165	Ohne dich	Comedian Harmonists	Comedian Harmonists	2006	jazz	
459849	Academy Awards Medley (Songs Nominated for an ...	Jule Styne in Hollywood	Jason Daniele, Marin Mazzie	2006	jazz	
923869	Agrippina, HWV 6: Aria: Ogni vento	Berlioz: Les nuits d'été, Op. 7 - Handel: Aria...	George Frideric Handel, Lorraine Hunt Lieberson...	2016	jazz	
614713	Welfare Wednesday	The Man Who Married Music: The Best Of Stephen...	Stephen Fearing	2009	jazz	
235176	Moonlight Sonata	Piano Love Songs	Bradley Joseph	2006	jazz	
783013	Showboat Medley	Evening With Earl Hines, An	Earl Hines	1973	jazz	
825080	Jacques Matherin	Cool Water	Mark Austin	2005	jazz	
1035585	A Dream in Paradise	A Dream in Paradise	Tjits Ven	2019	jazz	
421391	Somewhere Different Now	Everything's Easy	Girlyman	2009	jazz	

Electronic

	name	album	artists	year	predicted_genre
1154584	Tides	Transmitter	Jeffrey Koepper	2017	electronic
203989	Educating Me	Be My Sailor	Zeebee	2010	electronic
1123579	Venice	Orchards	Muddyoush	2020	electronic
347004	Stroke It - PT's Wet Dream Club Mix	TraXXX 123 RemiXed	Clint Crisher	2009	electronic
953152	Swamp Opera - Live	Bootleg Series Vol. 1 Too Slim and the Talldraggers, Tim Langford	2001	electronic	
886676	Devil	Music Library, Vol. 1	DontDolBeats	2019	electronic
585546	Hymn to the Muse	Invoking Aphrodite	Layne Redmond	2009	electronic
945271	Drunk Groove - Kolya Funk & Mephisto Remix	Drunk Groove (Remixes, Pt.1)	MARUV, Boosin	2018	electronic
937656	CLOSE combined (Artificial Technology) - Live	CLOSE COMBINED (Live, GLASGOW, LONDON, TOKYO)	Rosper, Richie Hawtin	2019	electronic
729340	Pretty Words Don't Mean a Thing (Lie to Me)	Greatest Funk Classics	The New Birth	2001	electronic

Blues

	name	album	artists	year	predicted_genre	
477845	Don't You Leave Me Here	Blues for Harlem	Larry Johnson	2000	blues	
341529	Perfect Tonight	Looking Back From Space	Cary Judd	2006	blues	
1102731	He'll Be Right There	God Gets The Glory	Mississippi Mass Choir	1991	blues	
305697	Sink Lateral	Pistachio Island	Ilkae	2001	blues	
358724	Knowledge of Self (feat. Random Thought)	Greatest Hits	Foundation Movement	2006	blues	
175973	Rachel's Bulgar	Fenci's Blues	Shtreiml & Ismail Fencioiglu	2006	blues	
169476	My Only Inspiration	Seconds	The Electric Pop Group	2010	blues	
892448	You Can Take Away My Woman (But Please Don't T...	The Promise Highway	Mark Cook	2002	blues	
512528	Ads And Smiles, Hey You-Buy This!, Innovation....	Negativland Presents Over The Edge Vol. 7: Tim...	Negativland	1994	blues	
401818	Flesh and Blood	The Essential Johnny Cash	Johnny Cash	2002	blues	

Hip Hop

	name	album	artists	year	predicted_genre
16107	Bye Bye Baby	They Never Saw Me Coming	TQ	1998	hiphop
252843	Patriot Act - Feat. I-Self Devine	Thee Adventures Of A B-Boy D-Boy	Muja Messiah	2009	hiphop
540265	Pour vrai	Gesamtkunstwerk	Dead Obies	2016	hiphop
993098	Hibernation	Twenty Nine	K.A.A.N.	2020	hiphop
522411	Bong Hits	Three Loco EP	Three Loco	2012	hiphop
834214	L.O.V.E.	L.O.V.E.	RVBY	2018	hiphop
36932	Cella Dwellas	RealmsN'Reality	Cella Dwellas	1996	hiphop
794396	Tango (Go)	Light Of Day	Preme	2018	hiphop
879142	Dip (feat. Nicki Minaj)	Dip (feat. Nicki Minaj)	Tyga, Nicki Minaj	2018	hiphop
506920	So Tired	Christopher (Bonus Track Version)	Sleep Of Oldominion	2009	hiphop

Reggae

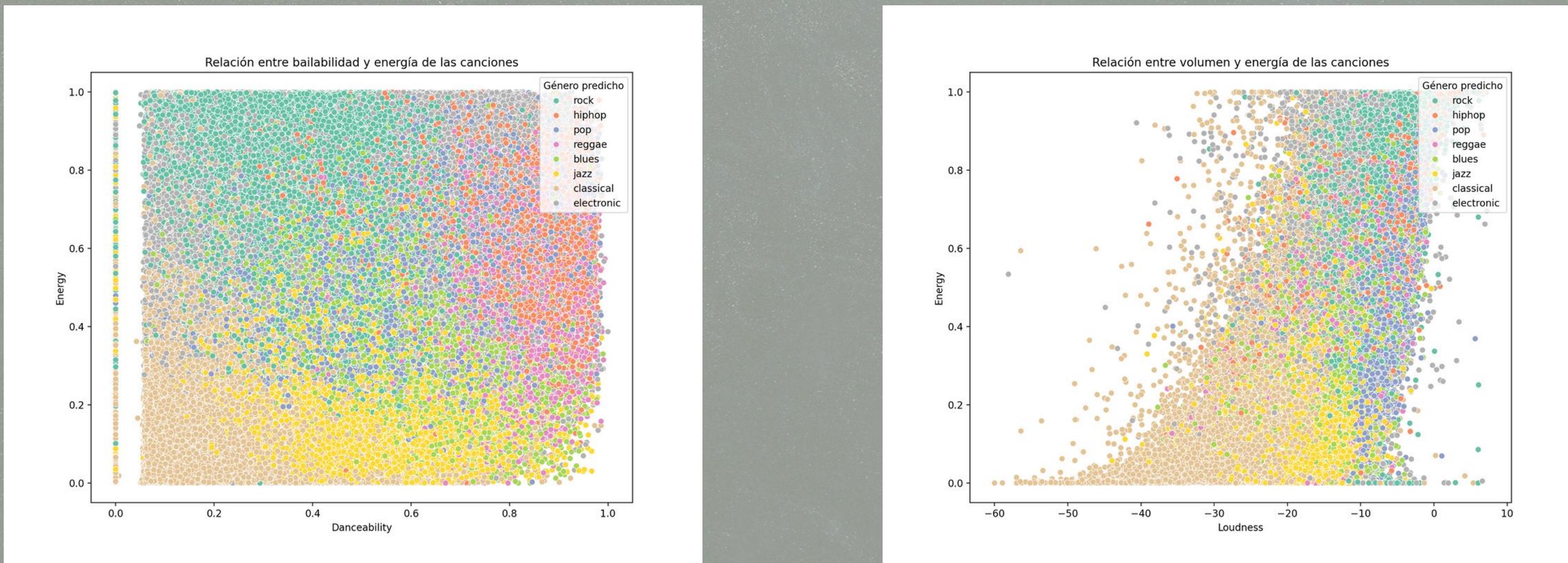
	name	album	artists	year	predicted_genre	
1057185	Death Row	Rich Slave	Young Dolph	2020	reggae	
587687	Cabaretera	15 Exitos	Aniceto Molina	2015	reggae	
1172368	Play the Time Away	The Definitive Ernest Ranglin	Ernest Ranglin	2012	reggae	
1101435	Look at You	Look at You	Fwea-Go Jit	2019	reggae	
1124162	Soul Mate	Spiritual Pop	Michael Franks	2019	reggae	
26517	Walk in London	A Million Different Moments	Null Device	2004	reggae	
411546	Nowhere Fast	Movin' On	Celia Slattery	2001	reggae	
187411	Et moi, et moi, et moi	Le meilleur de ...	Jacques Dutronc	1999	reggae	
169374	Peace	Second Impressions: Don't Stop	Gershon Veroba	2009	reggae	
465243	Where Do We Go?	Cutting Season 3	Eric Bellinger	2019	reggae	

Pop

	name	album	artists	year	predicted_genre

<tbl_r cells="6" ix="5" maxcspan="1" maxrspan

Gráficos de dispersión de las canciones



Se puede observar una distribución variada, indicando que hay una amplia gama de energía y bailabilidad dentro de cada género. Aunque no hay clusters definidos para los géneros específicos, se puede observar cierta agrupación basada en el color/género.

La densidad de los puntos varía a lo largo del gráfico; hay una concentración particularmente alta en el rango medio tanto para la energía como para la bailabilidad. Esto podría sugerir que muchas canciones tienden a tener niveles medios de energía y bailabilidad, independientemente del género.

Cabe destacar, como era de esperar, que las canciones con más energía y bailabilidad son las de música electrónica, seguidas del hip hop.

Existe una tendencia general donde a medida que aumenta el volumen, también lo hace la energía, aunque hay variaciones significativas dentro de cada género. Se observa que las canciones de rock y pop pueden tener un volumen y una energía más altos en comparación con los géneros como el jazz y la música clásica.

Estos 2 gráficos también nos ayudan a visualizar y entender la dificultad en la clasificación de géneros musicales. Podemos observar puntos aislados en zonas predominadas por un género en concreto, que vienen a ser principalmente aquellas pistas que han sido erróneamente clasificadas.

DEFINICION DE FUNCIONES PARA BUSQUEDA POR CANCION Y ARTISTA (EJECUTAREMOS PRUEBA EN PYTHON)

Función para búsqueda por canción:

```
def obtener_info_cancion(dataframe, nombre_cancion):
    # Filtrar el DataFrame para encontrar exactamente la canción y obtener su información
    resultados = dataframe[dataframe['name'].str.lower() == nombre_cancion.lower()].drop_duplicates(subset=['name', 'artists', 'album', 'year'])

    # Verificar si se encontró información y mostrar el resultado
    if not resultados.empty:
        for index, row in resultados.iterrows():
            genero = row['predicted_genre']
            artista = row['artists']
            cancion = row['name']
            album = row['album']
            año = row['year']

            print(f"Información para la canción '{cancion}':")
            print(f"  Artista: {artista}")
            print(f"  Género: {genero}")
            print(f"  Álbum: {album}")
            print(f"  Año: {año}")
    else:
        print(f"No se encontró información para la canción '{nombre_cancion}'.")

# Ejemplo de uso
nombre_cancion = input("Introduce el nombre de la canción: ")
obtener_info_cancion(df_final_genre_predictions, nombre_cancion)

Género: jazz
Álbum: Lullaby Renditions of AC/DC
Año: 2008
Información para la canción 'Thunderstruck':
Artista: Santa Claws and the Naughty But Nice Orchestra
Género: rock
Álbum: Hell's Bells of Christmas: The Holiday Tribute to AC/DC
Año: 2007
Información para la canción 'Thunderstruck':
Artista: "Meredith d'Ambrosio"
Género: jazz
Álbum: The Cove
Año: 1989
Información para la canción 'Thunderstruck':
Artista: AC/DC
Género: rock
Álbum: The Razors Edge
Año: 1990
```

Función para búsqueda por artista:

```
def obtener_info_artista(dataframe, nombre_artista):
    # Filtrar el DataFrame para encontrar todas las canciones del artista y obtener su información
    resultados = dataframe[dataframe['artists'].str.lower() == nombre_artista.lower()].drop_duplicates(subset=['name', 'artists', 'album', 'year'])

    # Verificar si se encontró información y mostrar el resultado
    if not resultados.empty:
        print(f"Información para el artista '{nombre_artista}':")
        for index, row in resultados.iterrows():
            genero = row['predicted_genre']
            artista = row['artists']
            cancion = row['name']
            album = row['album']
            año = row['year']

            print(f"  Canción: {cancion}")
            print(f"  Género: {genero}")
            print(f"  Álbum: {album}")
            print(f"  Año: {año}")
    else:
        print(f"No se encontró información para el artista '{nombre_artista}'.")

# Ejemplo de uso
nombre_artista = input("Introduce el nombre del artista: ")
obtener_info_artista(df_final_genre_predictions, nombre_artista)

...
Canción: Doing the Unstuck - Live in Detroit
Género: rock
Álbum: Show
Año: 1993
Canción: Friday I'm in Love - Live Detroit Version
Género: rock
Álbum: Show
Año: 1993
Canción: In Between Days - Live in Detroit
Género: rock
Álbum: Show
Año: 1993
Canción: From the Edge of the Deep Green Sea - Live in Detroit
Género: rock
Álbum: Show
Año: 1993
Canción: Never Enough - Live in Detroit
Género: rock
Álbum: Show
Año: 1993
```

8 - CONCLUSIONES

A pesar de no obtener una precisión en la clasificación superior al 71%, el modelo demuestra una capacidad prometedora, consiguiendo unos resultados bastante satisfactorios. Con futuros ajustes e inclusión de más datos, se espera mejorar aún más la predicción para la clasificación de géneros musicales.

Delgada línea de separación entre géneros:

Reconocer la delgada línea que separa a un género de otro y la dificultad inherente en la clasificación, considerando la variabilidad y fusiones que existen en la música.

Possible categorización errónea de géneros por parte de Spotify:

Es bastante factible que no todos los géneros estén correctamente definidos por Spotify, pudiendo nuestro modelo confundir géneros aún teniendo métricas y patrones similares.

Aspectos a mejorar:

Explorar la posibilidad de ampliar el abanico de géneros y subgéneros musicales para lograr un modelo más robusto y representativo de la diversidad musical. Para este estudio, hemos clasificado las pistas en solo 8 géneros principales, obviando géneros tan importantes como podrían ser, entre otros, el soul, r&b, country, disco, metal, indie, folk, etc.

Refinar y optimizar la selección de características y parámetros del modelo.

9 - REFERENCIAS

Barreda, L. (2023). Spotify Tracks by Genre: 8 Genres Classification. Kaggle. A dataset containing 9199 tracks belonging to Spotify playlists. The tracks are classified with the genre of the playlist, posted by Spotify or relevant agencies from the music industry.
<https://www.kaggle.com/datasets/laurabarreda/spotify-tracks-by-genre-8-genres-classification>

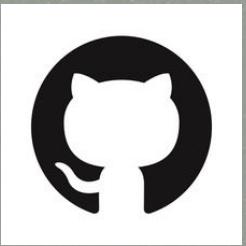
Figueroa, R. (2021). Spotify 1.2M Songs. Kaggle. The file tracks_features.csv contains audio features for over 1.2 million songs, obtained with the Spotify API.
<https://www.kaggle.com/datasets/rodolfofigueroa/spotify-12m-songs>

Spotify. (2024). Spotify for Developers.
<https://developer.spotify.com/>

Rozanec, M., & Merlino, H. (2022). Comparación de técnicas de aprendizaje automático aplicadas a la clasificación de géneros musicales. Conferencia Latinoamericana, Buenos Aires (Argentina).

¡¡¡MUCHAS GRACIAS!!!

Alex Vallés Gutiérrez



<https://github.com/AlexV0611/Final-Project---Data-Science>



valles.alex76@gmail.com