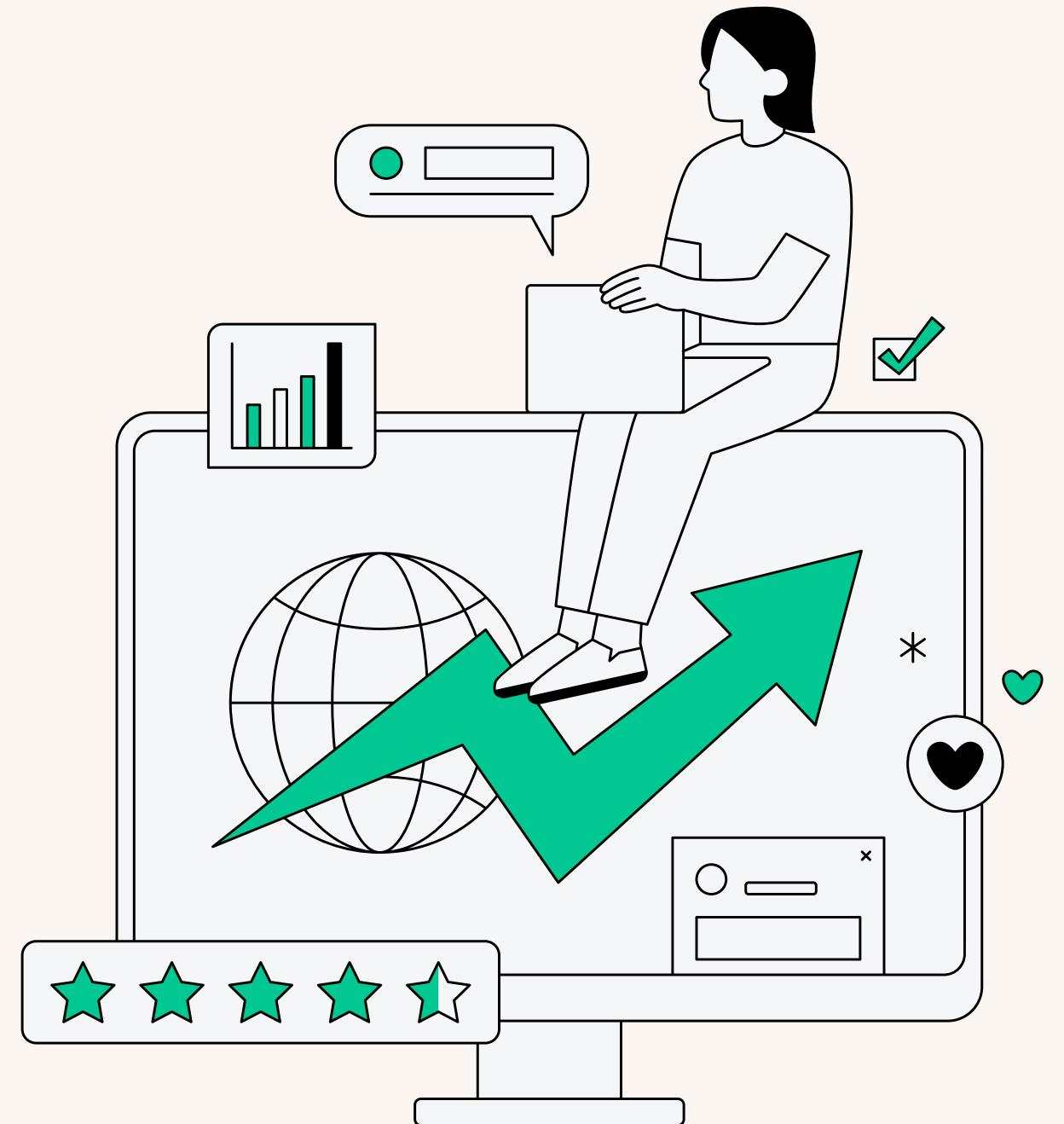


Presentado por Patrik Franco Pereda Matute

# Predicción de la Popularidad Spotify Dataset

<https://developer.spotify.com/>



# 1. Definición del Problema y Descripción del Dataset

## Definición del problema

- **Objetivo:** Predecir la popularidad de una canción usando características acústicas (energía, tempo, danceability, etc.).

## Objetivo del proyecto

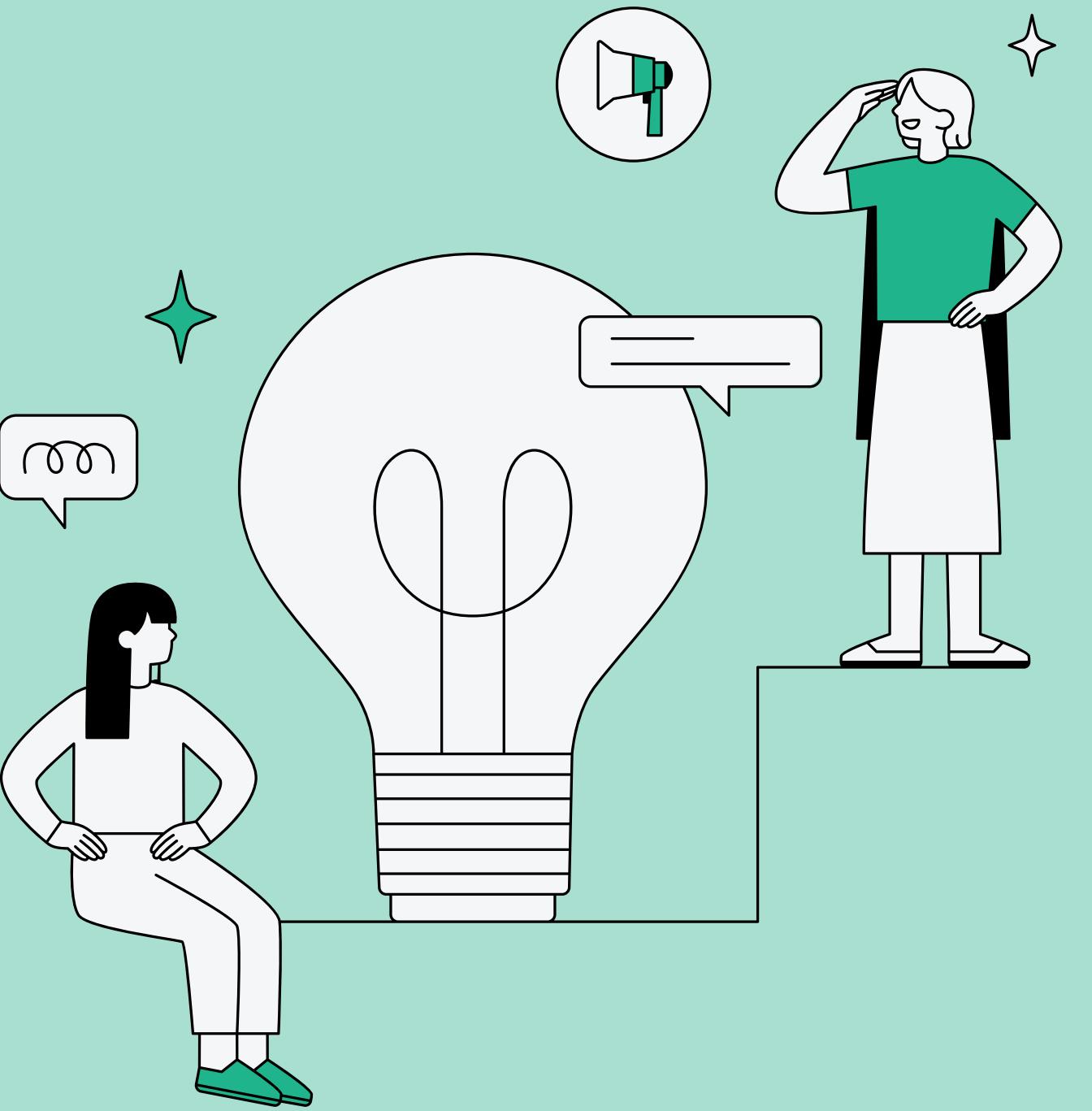
- Construir un modelo de machine learning para predecir la popularidad de las canciones.
- Analizar relaciones entre características acústicas y éxito musical.

## Descripción de los datos

- **Archivos:**
  - **tracks.csv:** Características acústicas y popularidad de las canciones.
  - **artists.csv:** Información adicional sobre los artistas.
- **Fuente:** API pública de Spotify.
- **Tamaño:**
  - **tracks.csv:** Miles de canciones con múltiples características.
  - **artists.csv:** Información de los artistas.
- **Variable objetivo:** popularity (valor numérico entre 0 y 100).

## Relevancia

- **Valor para la industria musical:** Entender patrones que impulsan el éxito.
- **Valor educativo:** Ejemplo práctico de ciencia de datos aplicada al entretenimiento.



## 2. Análisis Exploratorio de los Datos (EDA)

### Estructura:

- **Limpieza de datos:** Proceso para eliminar inconsistencias y datos innecesarios.
- **Visualización de distribuciones:** Análisis gráfico de las distribuciones de las variables.
- **Matriz de correlación:** Evaluación de las relaciones entre las variables.
- **Comparación de variables con la popularidad:** Análisis de cómo las características afectan la popularidad.
- **Explicación de cada gráfico:** Descripción e interpretación de cada visualización.

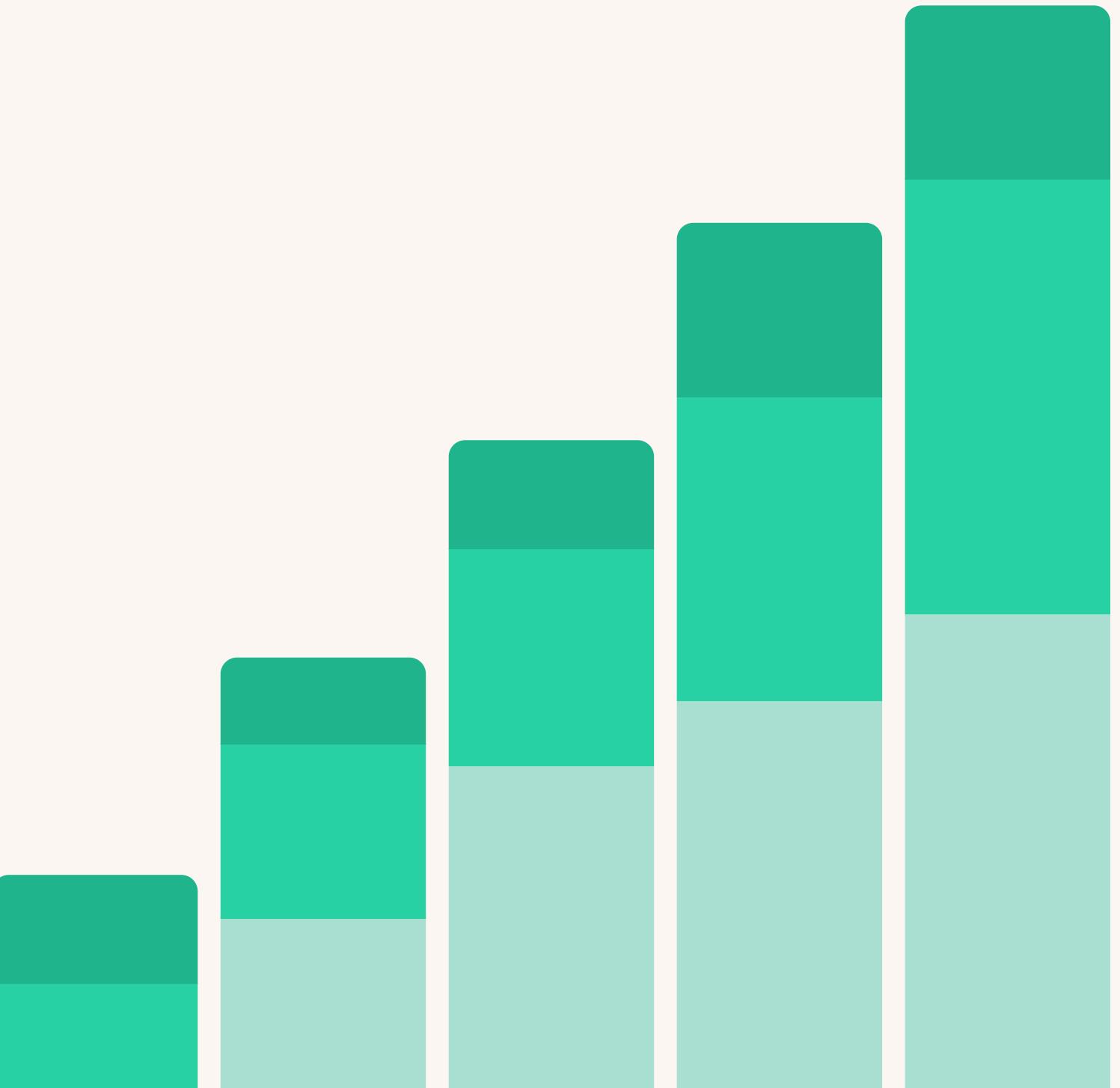
### Empezamos con el análisis paso a paso:

#### ◆ **Paso 1: Cargar y revisar los datos**

- Se cargan los archivos de datos y se realiza una vista rápida para entender su estructura.

#### ◆ **Paso 2: Limpiar los datos**

- Se identifican y eliminan los valores nulos, se eliminan columnas irrelevantes y se eliminan duplicados para asegurar que los datos sean coherentes y completos.



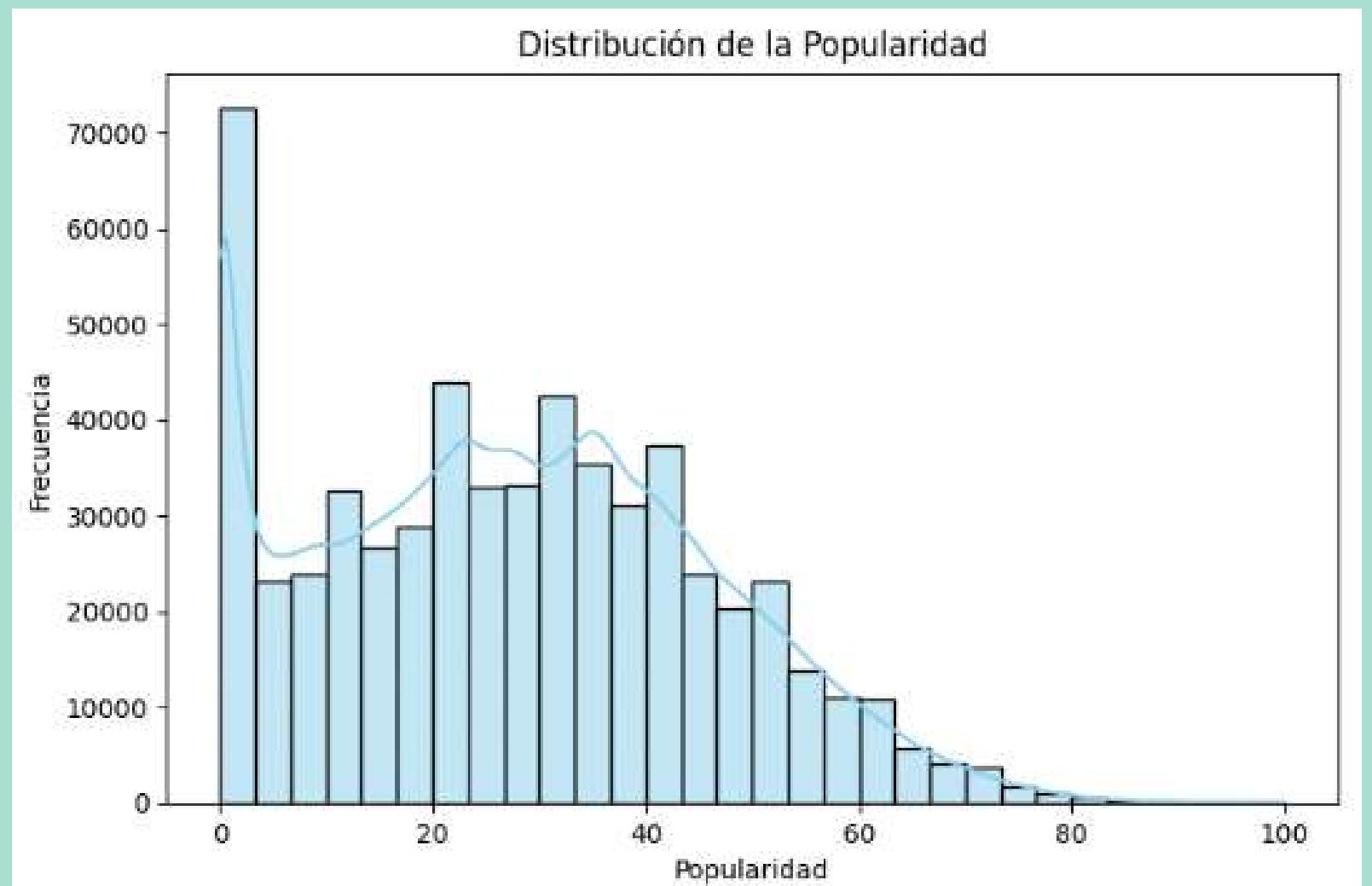
## ◆ Gráfico 1: Distribución de la popularidad

### ¿Qué muestra?

Un histograma que nos dice cuántas canciones tienen cada nivel de popularidad (del 0 al 100).

### ¿Por qué lo hacemos?

Para saber si el dataset está equilibrado. Por ejemplo, si la mayoría de canciones tienen baja popularidad, eso puede afectar al modelo.



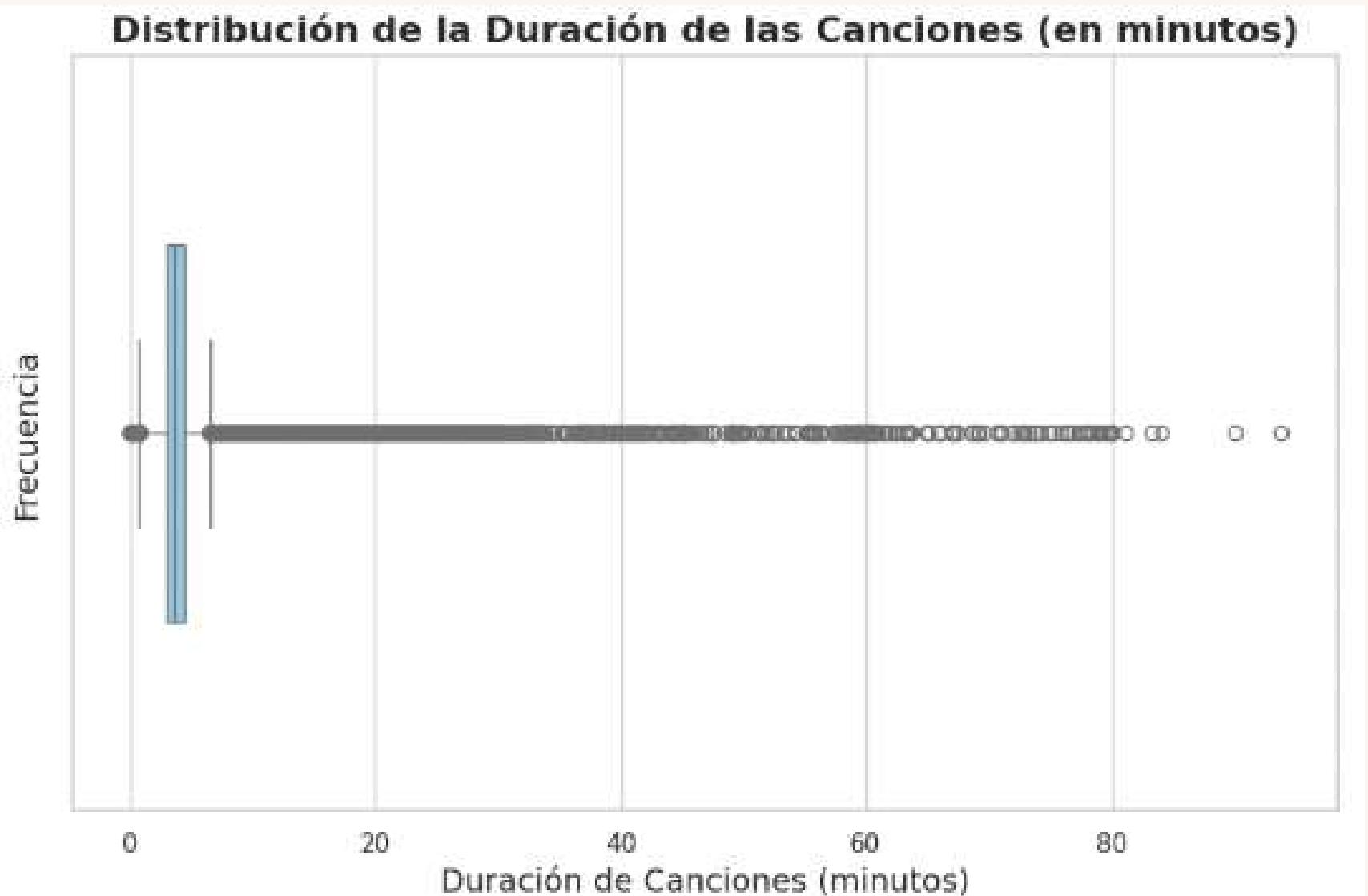
## ◆ Gráfico 2: Boxplot de duración de canciones (en minutos)

### ¿Qué muestra?

Este gráfico muestra la distribución de la duración de las canciones en minutos, destacando los valores extremos (outliers) que pueden indicar canciones demasiado cortas o largas.

### ¿Por qué lo hacemos?

Convertimos la duración a minutos para facilitar la comprensión. Al ver la distribución, podemos identificar rápidamente si hay canciones que se desvían mucho del tiempo esperado, lo que podría influir en el análisis de patrones o tendencias.



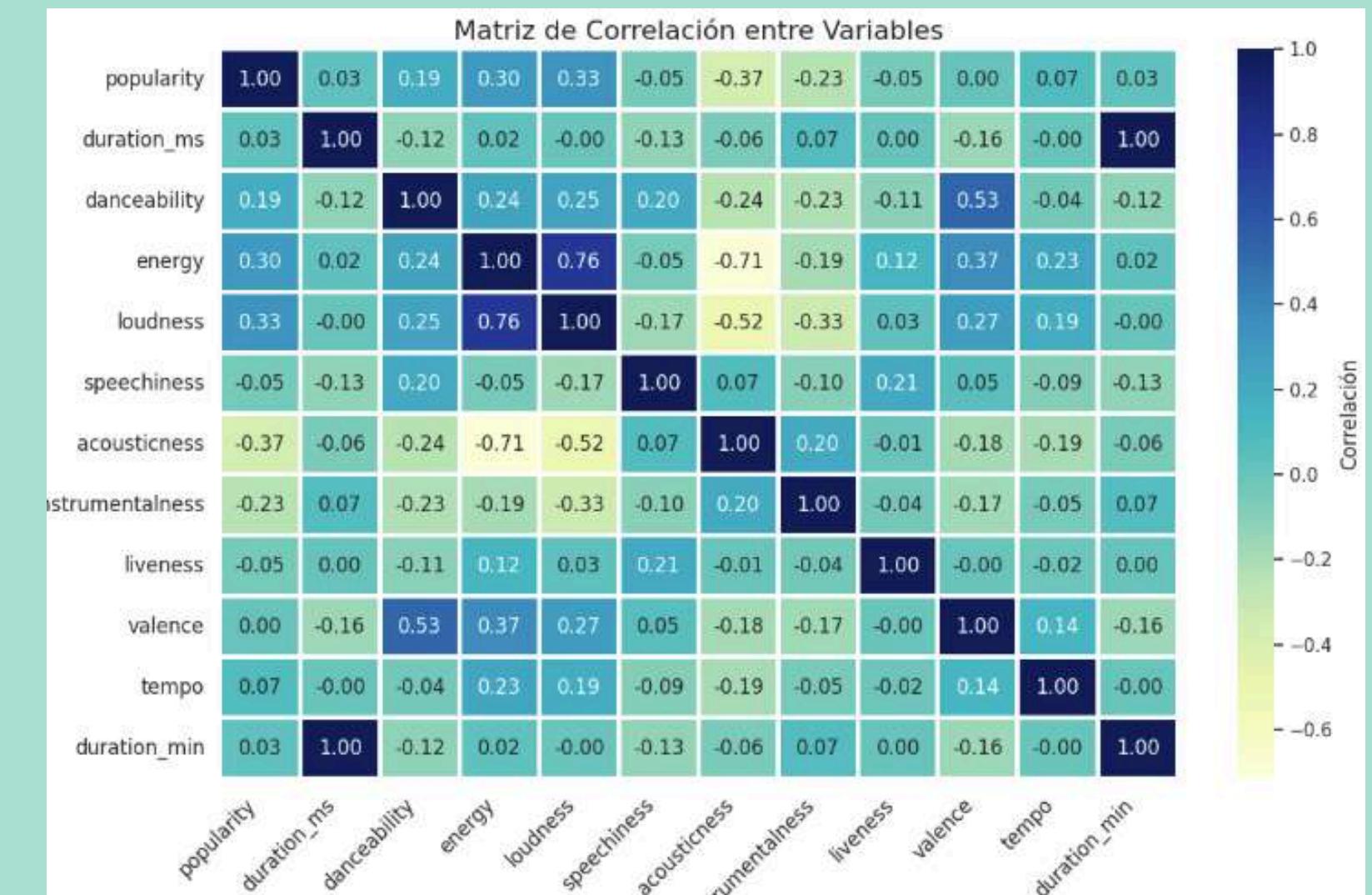
## ◆ Gráfico 3: Matriz de correlación

### ¿Qué muestra?

Este gráfico muestra una "tabla de colores" donde cada número indica qué tan fuerte es la relación entre dos características de las canciones. Si el número es alto (cerca de 1), las características están muy relacionadas; si es bajo (cerca de 0), significa que no están relacionadas.

### ¿Por qué lo hacemos?

Queremos identificar qué características de las canciones están más relacionadas con su popularidad. Esto nos ayudará a construir un modelo que prediga mejor qué hace a una canción popular.



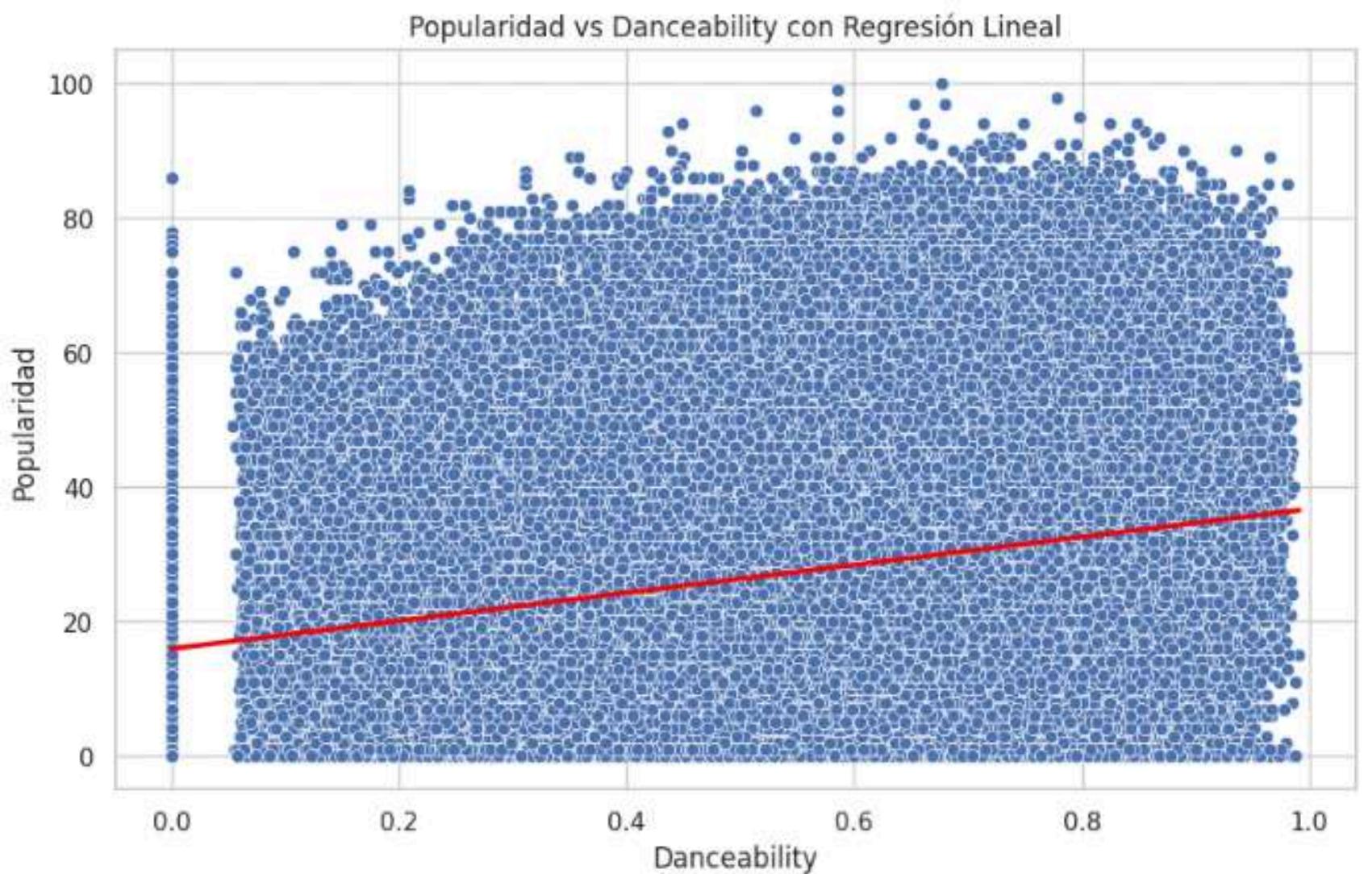
## ◆ Gráfico 4: Popularidad vs Danceability

### ¿Qué muestra?

Este gráfico compara qué tan bailable es una canción con su popularidad. La línea roja muestra la tendencia: si sube, significa que las canciones más bailables suelen ser más populares.

### ¿Por qué lo hacemos?

Queremos ver si la música que da más ganas de bailar también es la que más gusta al público. Esto nos ayuda a entender qué hace que una canción sea exitosa.



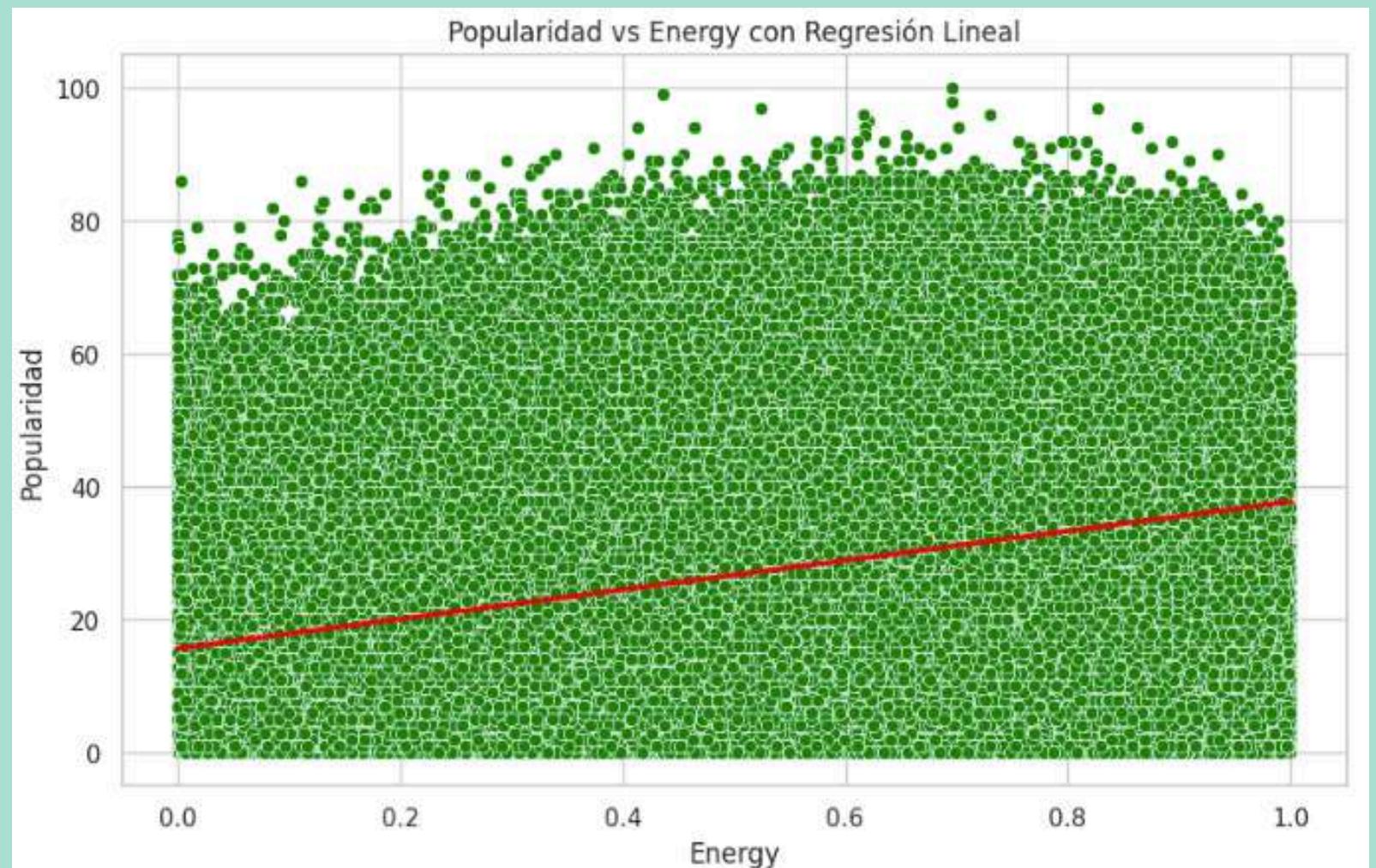
## ◆ Gráfico 5: Popularidad vs Energy

### ¿Qué muestra?

El gráfico compara cuánta energía tiene una canción con su popularidad. La línea roja indica si las canciones más potentes tienden a gustar más.

### ¿Por qué lo hacemos?

Buscamos saber si las canciones más enérgicas son también más populares. Esto nos ayuda a elegir bien qué variables usar para predecir el éxito musical.



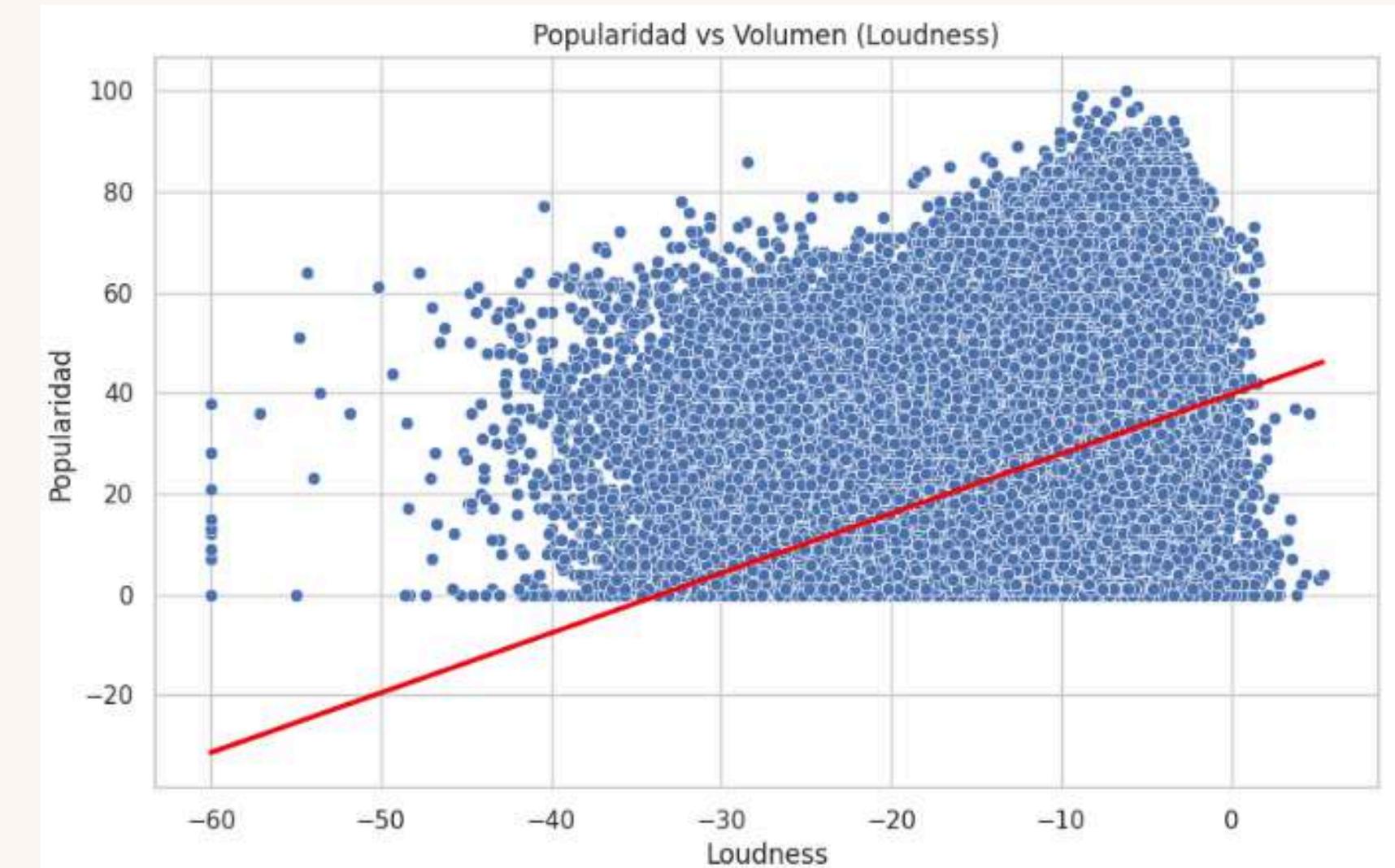
## ◆ Gráfico 7: Loudness vs Popularidad

### ¿Qué muestra?

El gráfico analiza si las canciones que suenan más fuerte suelen ser más populares. La línea roja indica la tendencia general.

### ¿Por qué lo hacemos?

Queremos saber si el volumen influye en el éxito de una canción. Si hay relación, esta info puede ayudarnos a predecir mejor qué canciones gustarán más.



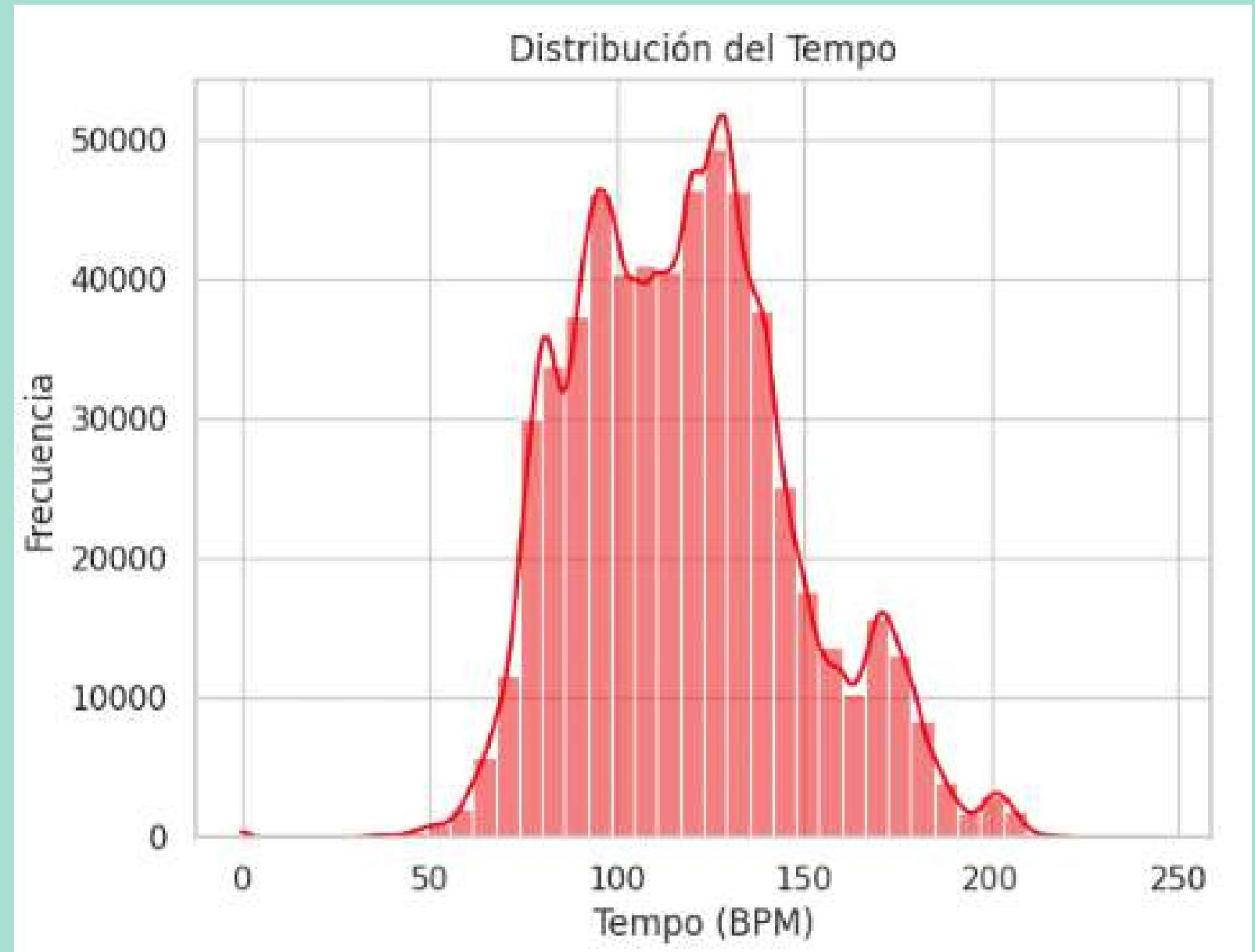
## ◆ Gráfico 8: Distribución del Tempo

### ¿Qué muestra?

Este gráfico muestra cuántas canciones hay según su velocidad (tempo en BPM). Nos ayuda a ver si hay ritmos que se repiten más.

### ¿Por qué lo hacemos?

Buscamos saber si hay un tempo típico en las canciones. Esto puede ayudarnos a entender qué ritmos son más comunes o exitosos.



### 3. Selección e Ingeniería de Características

#### ¿Qué es?

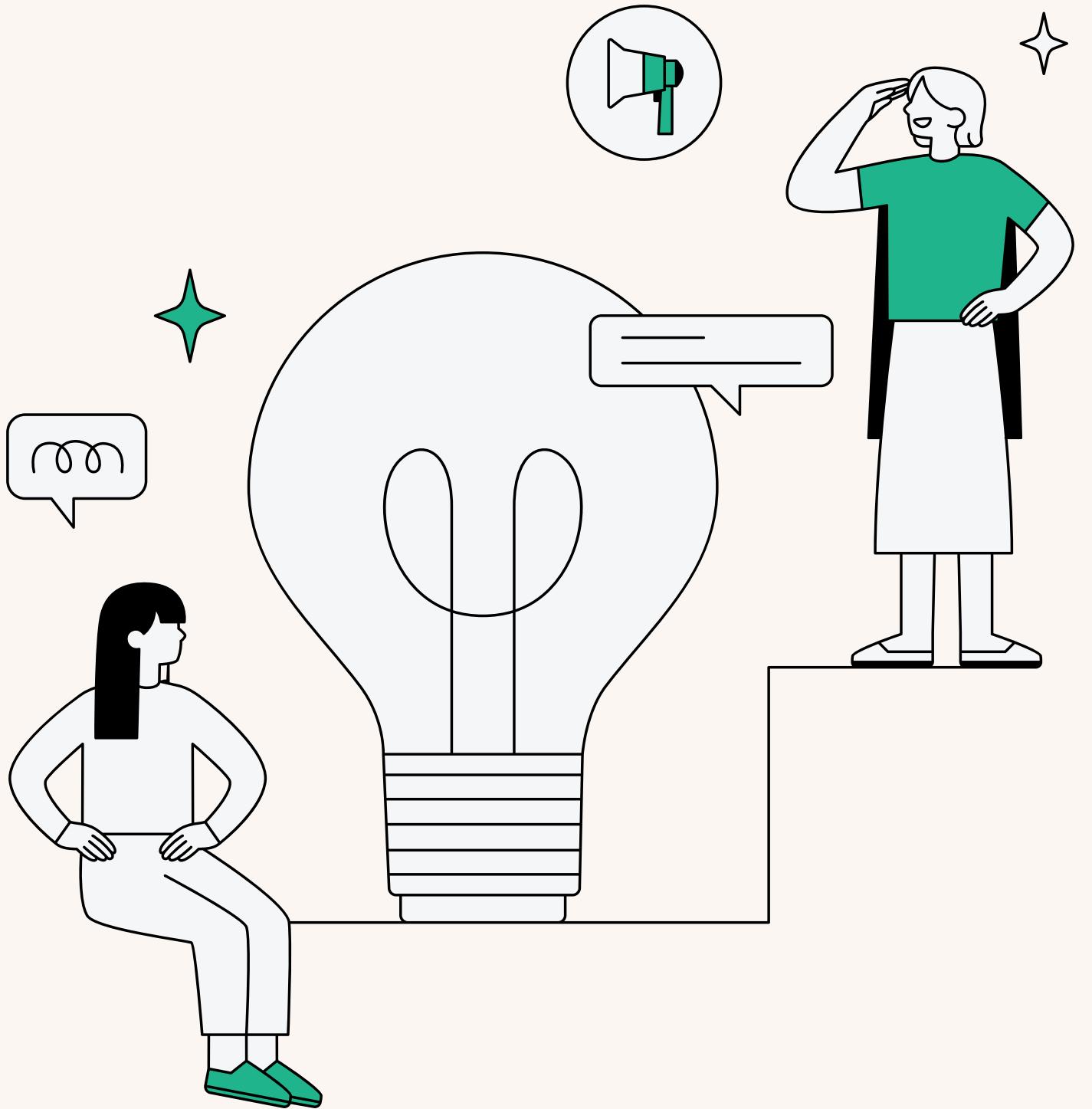
Es el proceso de elegir y transformar las variables que mejor ayudan al modelo a predecir la popularidad de una canción.

#### ¿Qué hicimos?

- Seleccionamos variables útiles como danceability, energy, valence, loudness, etc., que tienen relación con la popularidad según el análisis previo.
- Descartamos variables irrelevantes, con baja correlación o ruido.
- Transformamos datos para mejorar su interpretación:
  - Convertimos duración a minutos.
  - Creamos nuevas variables como "energía emocional" (promedio entre valence y energy).
  - Normalizamos datos y eliminamos valores extremos (outliers).

#### ¿Por qué?

Porque estas variables están conectadas con cómo se perciben y disfrutan las canciones, y eso influye directamente en su éxito.



## 4. Entrenamiento del Modelo: separación de datos, entrenamiento, evaluación, visualización de resultados y justificación del modelo elegido.

### Paso 1: Separación de datos

Dividimos los datos en dos partes:

- **80% para entrenamiento:** el modelo aprende a partir de estos datos.
- **20% para prueba:** se reserva para comprobar qué tan bien generaliza el modelo con datos nuevos.

Esta separación permite evaluar el rendimiento de forma objetiva y evitar que el modelo simplemente memorice los datos.

### Paso 2: Elegir y entrenar un modelo

Elegimos **Random Forest Regressor** por ser eficaz con datos complejos, resistente a valores extremos y fácil de interpretar. Este modelo utiliza múltiples árboles para hacer predicciones, lo que lo hace robusto y preciso.

El código entrena el modelo con los datos de entrenamiento, preparándolo para hacer predicciones sobre nuevos datos.

### Paso 3: Evaluación del modelo

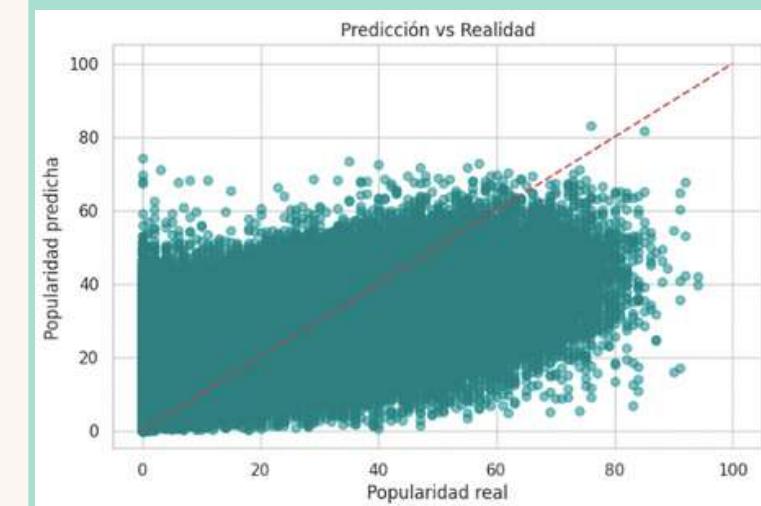
El modelo tiene los siguientes resultados:

- **MAE: 11.62:** El error promedio es de 11.62 puntos en popularidad.
- **RMSE: 14.70:** El error considerando su magnitud es de 14.70 puntos.
- **R<sup>2</sup>: 0.35:** El modelo explica un 35% de la variabilidad en la popularidad.

Esto indica que el modelo tiene margen de mejora.

### Paso 4: Visualizar el rendimiento

El gráfico muestra la comparación entre la popularidad real y la predicha. La línea roja diagonal indica una predicción perfecta. Los puntos (en color teal) están cerca de esta línea, pero la mayoría se agrupan por debajo de 80, lo que sugiere que el modelo tiende a subestimar la popularidad.



## 4. Evaluación Comparativa de Modelos de Regresión

### Regresión Lineal

La regresión lineal es un modelo básico que intenta ajustar una línea recta a los datos. Su simplicidad permite entender rápidamente si existe una relación directa entre las variables (por ejemplo, energía o valencia) y la popularidad de la canción. Aunque no captura relaciones complejas, sirve como punto de partida para comparar otros modelos más avanzados.

- MAE: 13.5
- RMSE: 16.8
- R<sup>2</sup>: 0.28

### K-Nearest Neighbors (KNN)

KNN predice la popularidad de una canción en función de sus vecinas más parecidas. Es decir, busca canciones con características similares y asume que tendrán una popularidad parecida. Es intuitivo y útil cuando existen grupos naturales en los datos, pero su rendimiento cae si hay muchos registros.

- MAE: 12.2
- RMSE: 15.4
- R<sup>2</sup>: 0.31

### Gradient Boosting Regressor

Gradient Boosting entrena una secuencia de modelos (normalmente árboles pequeños) donde cada uno intenta corregir los errores del anterior. Suele dar mejores resultados que un solo modelo porque aprende de forma progresiva. Es más potente, pero también más lento y sensible al sobreajuste si no se controla bien.

- MAE: 11.7
- RMSE: 14.8
- R<sup>2</sup>: 0.34

**El modelo que mejor predijo la popularidad fue Random Forest, seguido muy de cerca por Gradient Boosting. Regresión Lineal fue el menos preciso, y KNN tuvo un rendimiento intermedio. Esto muestra que los modelos más avanzados pueden dar mejores resultados.**

Modelo	MAE	RMSE	R <sup>2</sup>
Linear Regression	13.5	16.8	0.28
KNN Regressor	12.2	15.4	0.31
Gradient Boosting	11.7	14.8	0.34
Random Forest (final)	11.6	14.7	0.36

## 5. Afinación de Hiperparámetros: ajuste de parámetros clave del modelo como n\_estimators, max\_depth y max\_features para mejorar su rendimiento.

### Paso 1: Usar GridSearchCV para buscar la mejor combinación

GridSearchCV es una técnica que busca la mejor combinación de parámetros para un modelo de machine learning probando varias opciones. En este caso, se utiliza para optimizar un modelo Random Forest, probando valores como el número de árboles, la profundidad máxima y cuántas características usar en cada división. El resultado indica que el proceso de búsqueda de parámetros se completó correctamente, probando todas las combinaciones posibles en 2 pliegues de validación cruzada, lo cual es una forma de verificar cómo funcionará el modelo con datos no vistos.

### Paso 2: Revisar los mejores hiperparámetros

Este paso muestra los mejores hiperparámetros encontrados después de realizar la búsqueda con GridSearchCV. El modelo ha seleccionado que la profundidad máxima de los árboles debe ser ilimitada (max\_depth: None), que se usen solo las características más relevantes en cada división (max\_features: 'sqrt'), y que se dividan los nodos con al menos 2 muestras (min\_samples\_split: 2). También se ha elegido usar 100 árboles (n\_estimators: 100). Estos valores son los que el modelo considera los más adecuados para predecir con mejor rendimiento.

### Paso 3: Evaluar el nuevo modelo con los mejores parámetros

En este paso se evalúa el rendimiento del modelo con los mejores hiperparámetros encontrados. Se usan métricas como el MAE (error absoluto promedio), RMSE (raíz cuadrada del error cuadrático medio) y el R<sup>2</sup> (coeficiente de determinación). El MAE y el RMSE siguen siendo valores bajos, indicando que el modelo sigue cometiendo errores, pero el R<sup>2</sup> ha mejorado ligeramente a 0.36, lo que sugiere que el modelo ajustado ahora explica un poco más de la variabilidad de los datos en comparación con el modelo anterior.



# 6. Interpretación del Modelo

## Paso 1: Importancia de las Variables

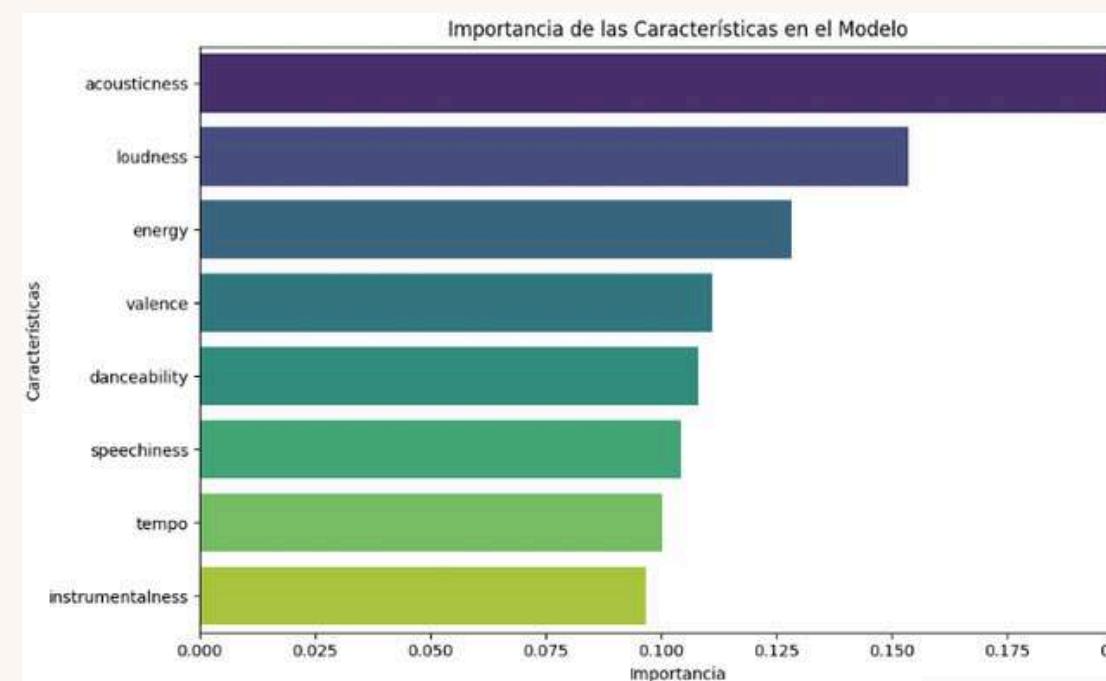
El modelo Random Forest permite identificar qué características son más relevantes para predecir la popularidad. El gráfico muestra que las variables con mayor impacto son "acousticness", "loudness", "energy", "valence", "danceability", entre otras.

## Paso 2: Interpretación de los Resultados

Las variables más influyentes son "acousticness", "loudness", "energy" y "danceability", lo cual tiene sentido porque canciones con mayor energía y capacidad de baile tienden a ser más populares. "Tempo" y "instrumentalness" tienen menor impacto, lo que podría ser sorprendente si consideramos que el ritmo también podría jugar un papel importante.

## Paso 3: Reflexión sobre Confianza y Fiabilidad

El modelo presenta un rendimiento sólido, pero la popularidad también depende de factores externos como el marketing o las tendencias. Aunque el modelo no es infalible, ofrece una base para entender cómo ciertos atributos musicales pueden influir en el éxito de una canción.



## 7. Conclusiones y Trabajo a Futuro

### Resumen de resultados:

Se desarrolló un modelo con Random Forest para predecir la popularidad de las canciones, basándose en sus características acústicas. El modelo mostró buen rendimiento, destacando variables como "danceability", "energy" y "valence", que son clave para la popularidad de las canciones.

### Revisión del problema:

El proyecto buscó entender los factores que afectan el éxito de las canciones en plataformas como Spotify. A través de análisis y evaluación de modelos, se identificaron patrones que explican la popularidad.

### Limitaciones:

El modelo no considera factores externos como marketing o colaboraciones, y se basa solo en variables acústicas, lo que limita su capacidad predictiva.

### Propuestas a futuro:

Incorporar variables externas como redes sociales, probar otros algoritmos y hacer análisis específicos por género musical. Además, se propone crear una app que prediga la popularidad de canciones con datos manuales.



Presentado por Patrik Franco Pereda Matute

¡Muchas  
gracias por  
tu  
atención!

<https://developer.spotify.com/>

