



Clasificación de Generos de Musica

TP1 - Analisis Predictivo Avanzado



Indice



Introducción 01

Caso de negocio 02

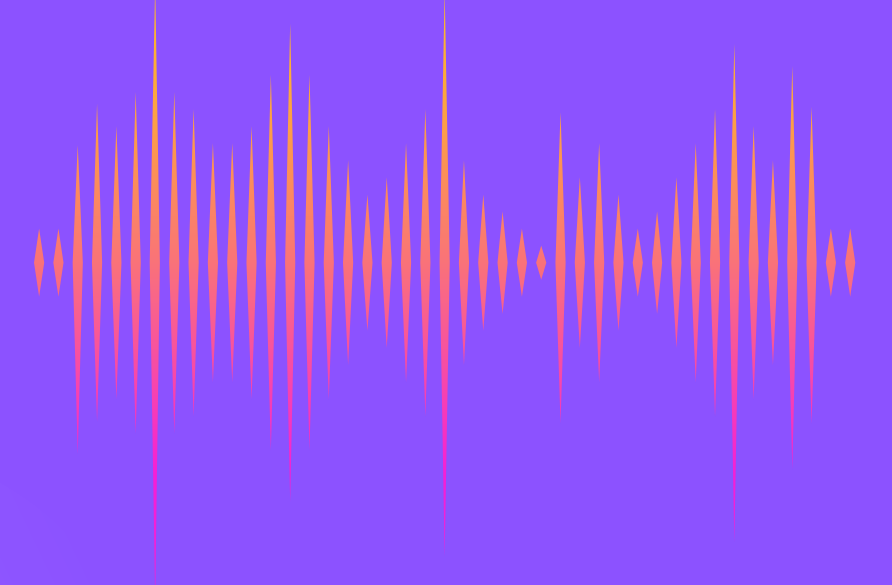
EDA 03

Preprocesamiento 04

Modelos de ML y Ajustes 05

**Conclusiones y acciones de
acciones de negocio** 06

Introducción



En la siguiente exposición se buscara desarrollar un modelo de clasificación automática de género musical basado en las características de audio de las samples de canciones de 10 generos distintos.



Caso de negocio

El propósito del algoritmo es optimizar las recomendaciones de música nueva en nuestra plataforma, similar a Spotify, mejorando así la experiencia de usuario en la selección de canciones recién añadidas.

Clasificación Automática de Nuevas Canciones



Géneros musicales:

Rock

Pop

Jazz

Disco

Country

Hiphop

Classical

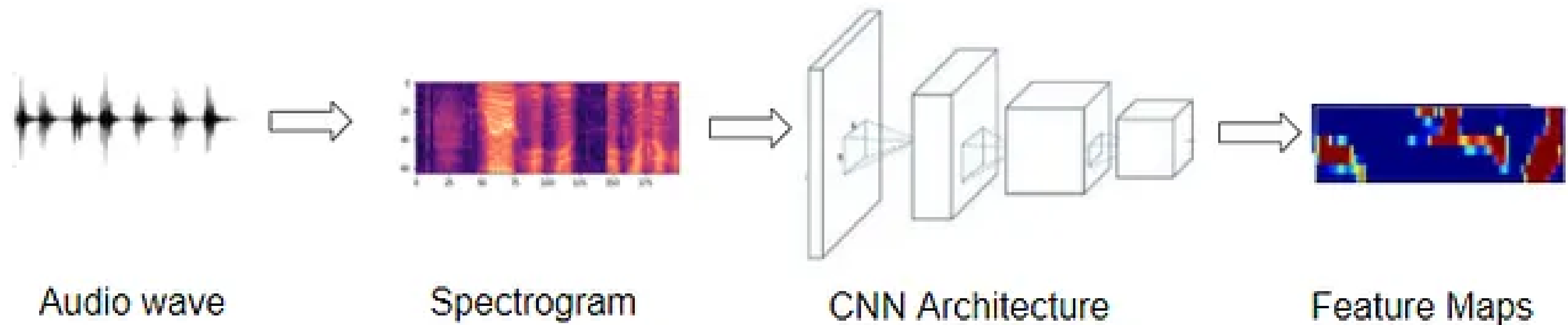
Metal

Blues

Reggae



CONVERSION DE AUDIO A FEATURES



EDA

filename	length	chroma_stft_mean	chroma_stft_var	rms_mean	rms_var	spectral_centroid_mean	spectral_centroid_var
blues.00000.wav	661794	0.350088	0.088757	0.130228	0.002827	1784.165850	129774.064525
blues.00001.wav	661794	0.340914	0.094980	0.095948	0.002373	1530.176679	375850.073649
blues.00002.wav	661794	0.363637	0.085275	0.175570	0.002746	1552.811865	156467.643368
blues.00003.wav	661794	0.404785	0.093999	0.141093	0.006346	1070.106615	184355.942417
blues.00004.wav	661794	0.308526	0.087841	0.091529	0.002303	1835.004266	343399.939274

59

Features

950

Observaciones

01

Variable
objetivo: label

Características del dataset

01

Contamos con 10 categorías de géneros musicales

02

Los datos se encuentran desbalanceados

03

La longitud de los samples es de 30 segundos

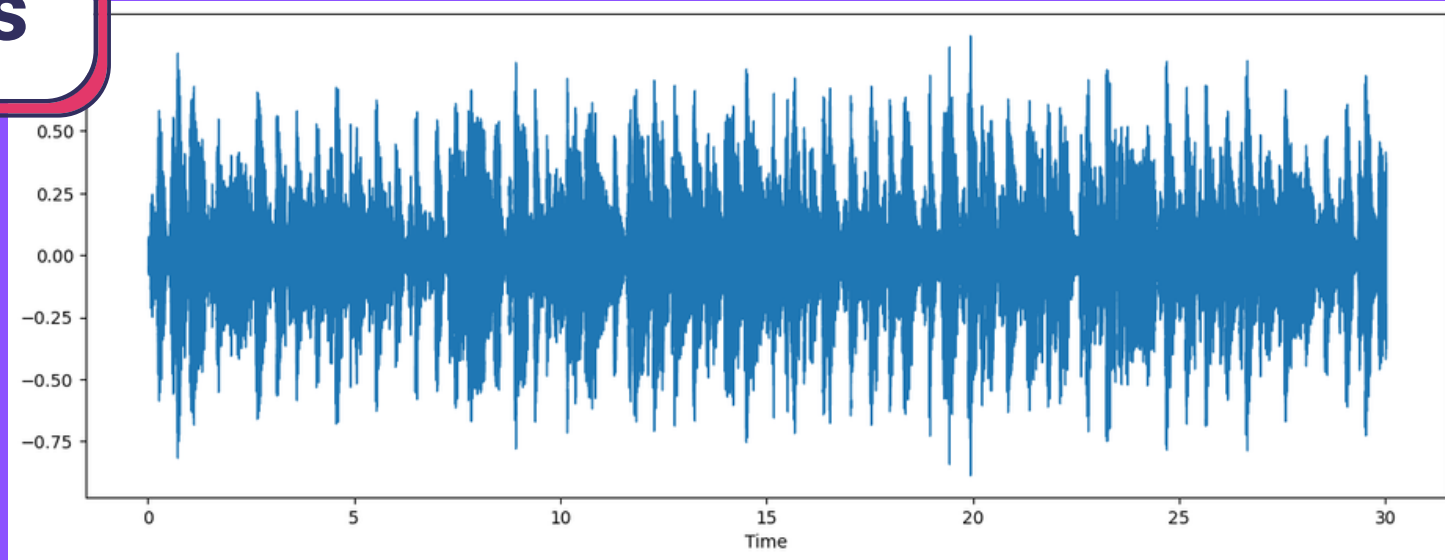
04

El dataset cuenta con los siguientes features: 'filename', 'length', 'chroma_stft_mean', 'rms_mean', 'spectral_centroid_mean', 'spectral_bandwidth_mean', 'rolloff_mean', 'zero_crossing_rate_mean', 'harmony_mean', 'perceptr_mean', 'tempo', 'mfcc1_mean', 'label'

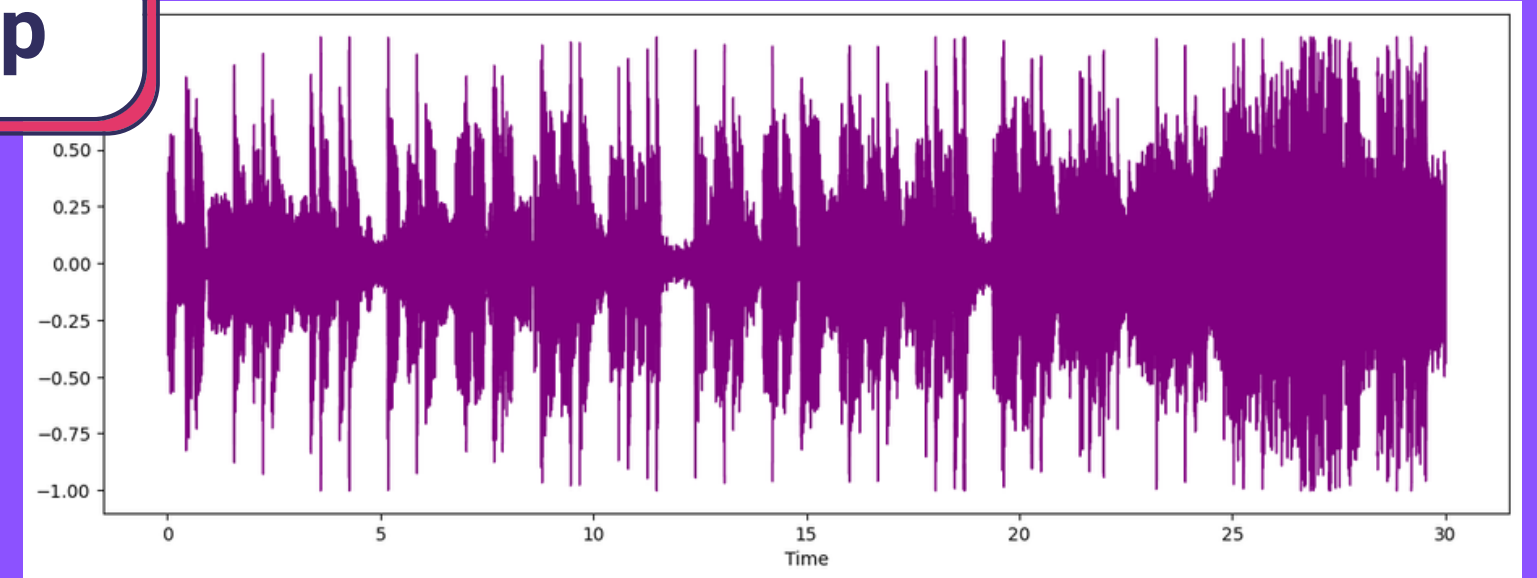
blues	100
country	100
hiphop	100
jazz	100
metal	100
pop	100
reggae	100
rock	100
classical	80
disco	70

Amplitud de algunos géneros musicales

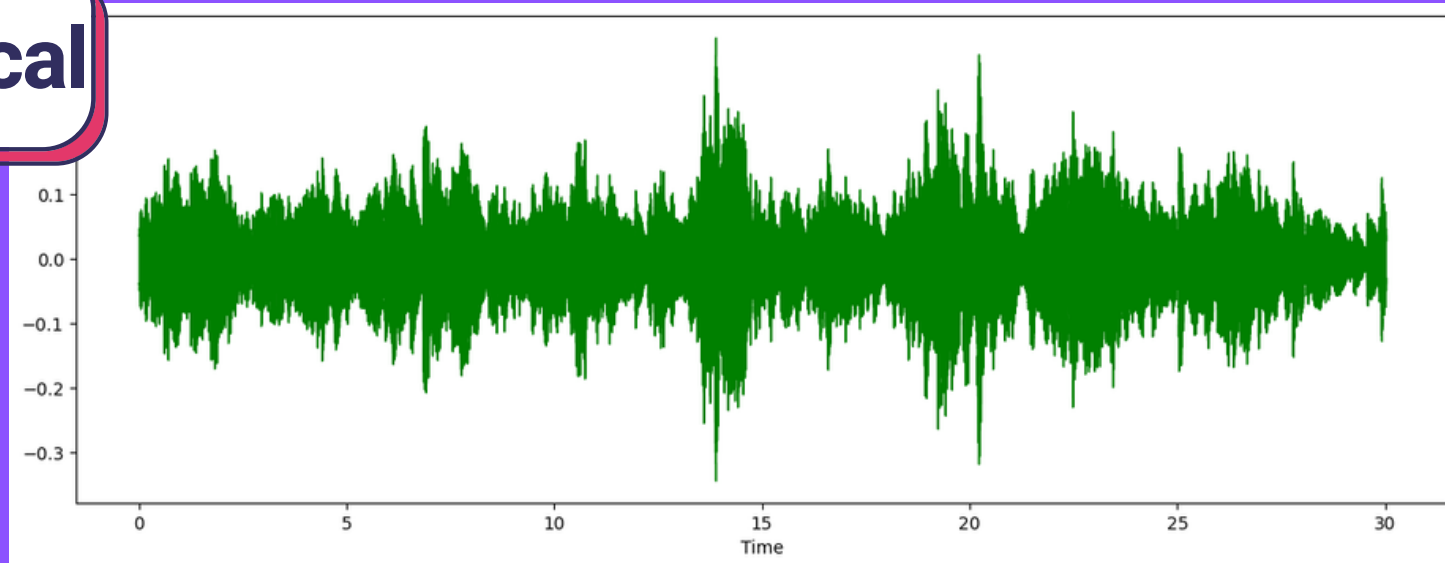
Blues



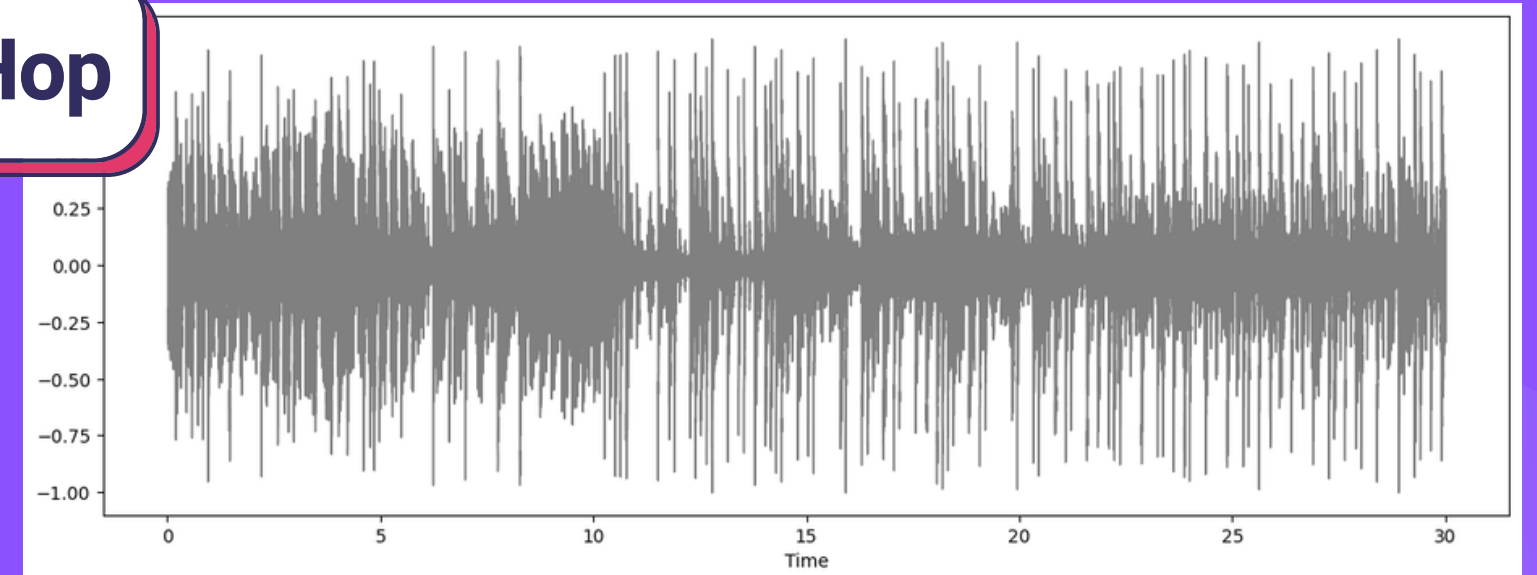
Pop



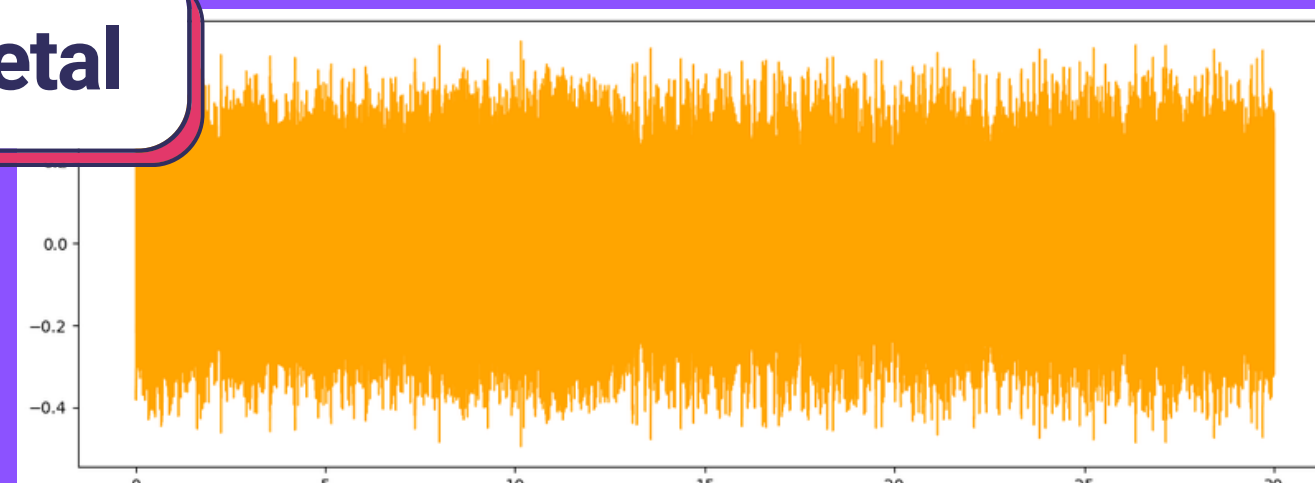
Classical



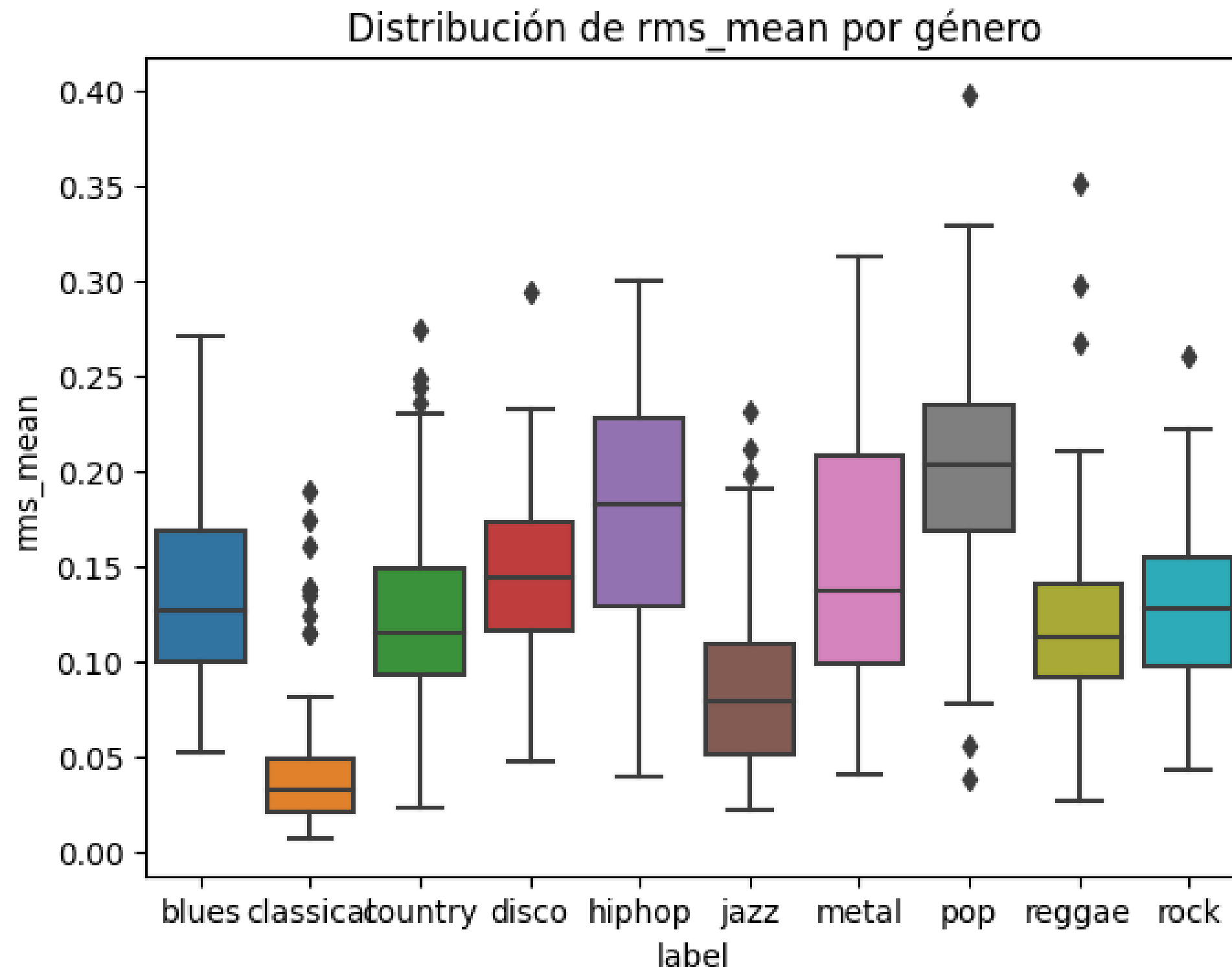
HipHop



Metal



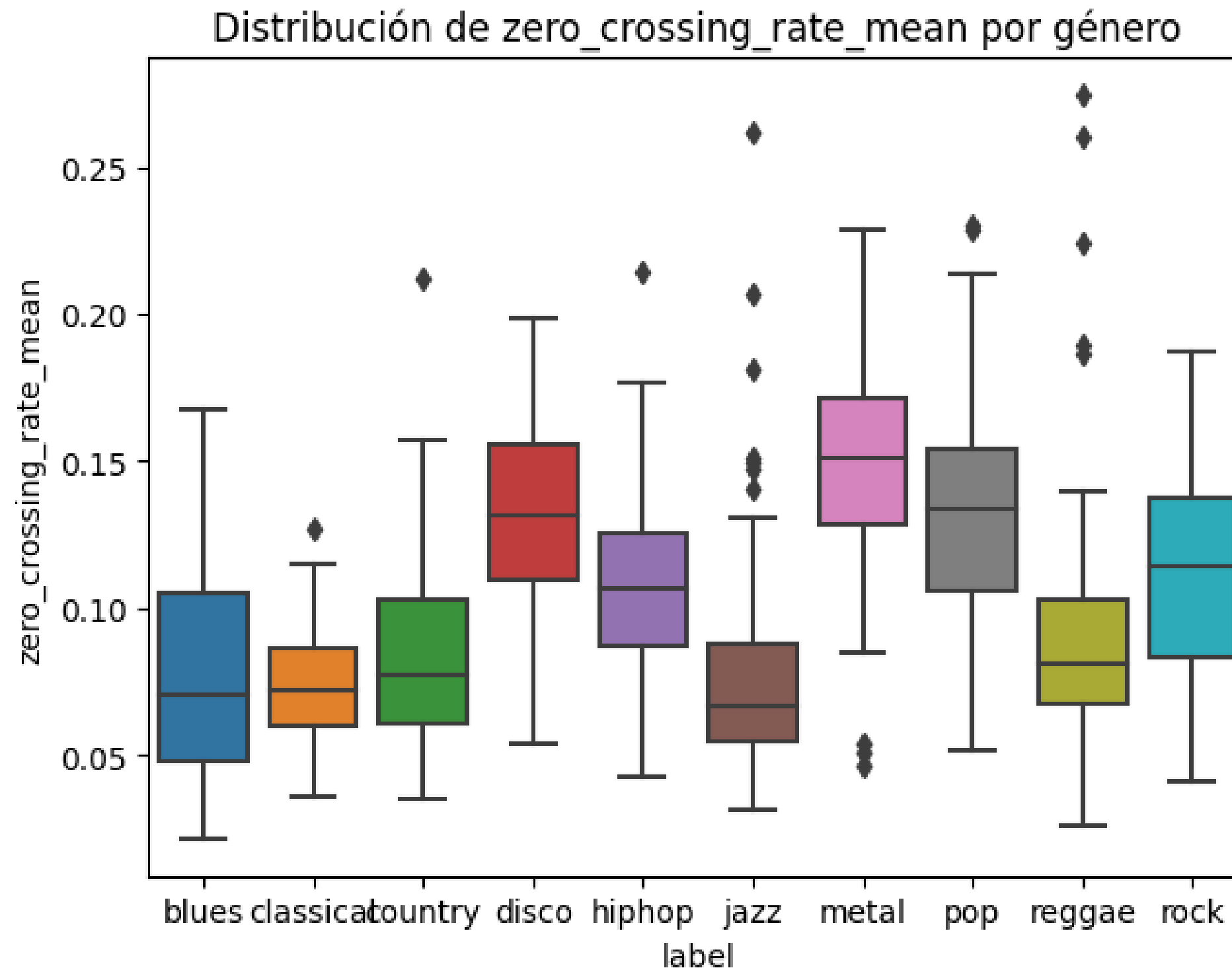
Root Mean Square (RMS)



Conceptos Básicos: RMS es una medida de la magnitud de la señal de audio. rms_mean es el promedio de la energía RMS, mientras que rms_var es la variabilidad de la energía RMS.

Relación con Música: rms_mean puede indicar la intensidad o volumen percibido de la música. Géneros más fuertes y energéticos tendrán un rms_mean más alto.

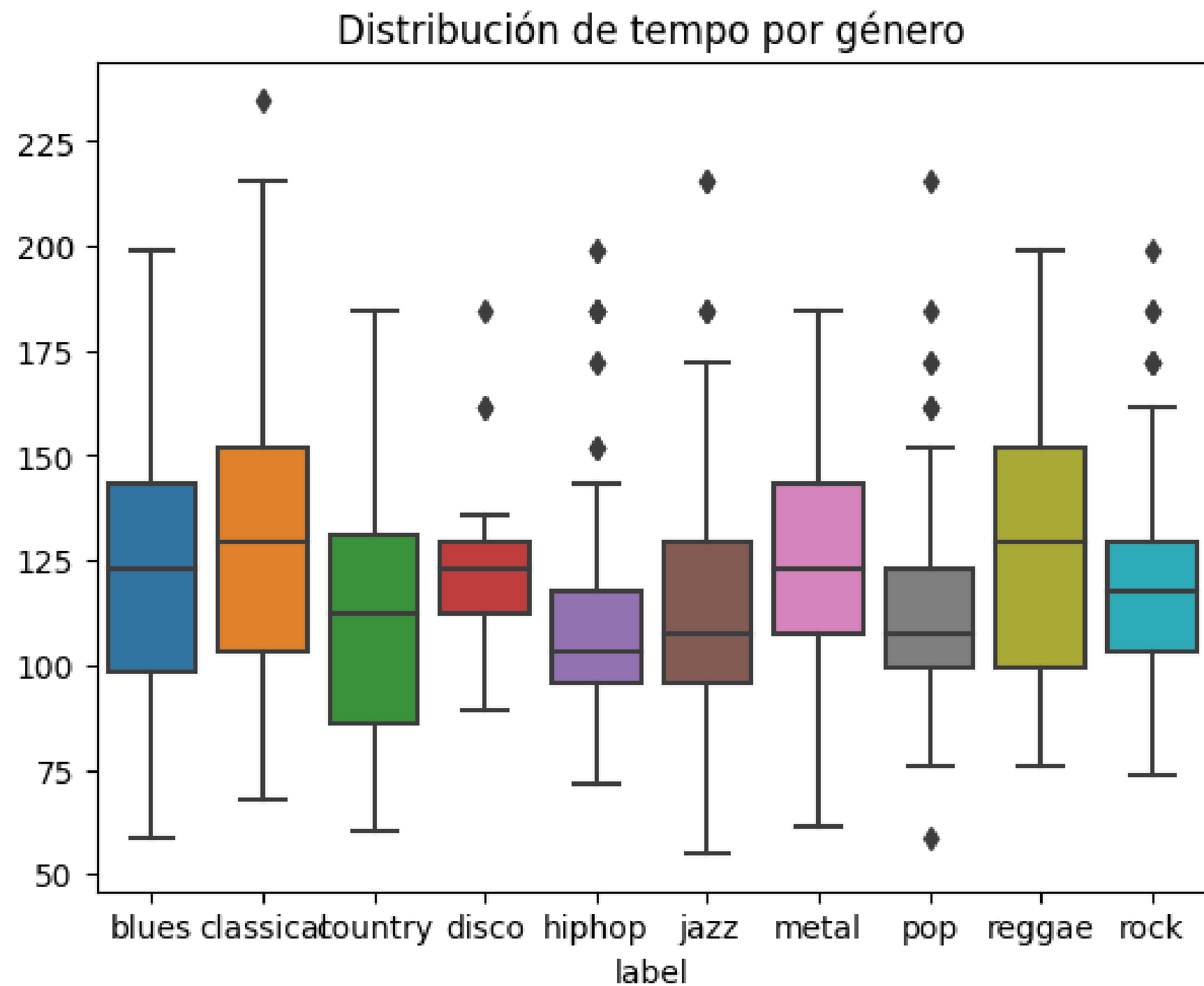
Zero Crossing Rate



Conceptos Básicos: Es la tasa a la que la señal cambia de positivo a negativo o viceversa.

Relacion con la musica: Géneros con más elementos percusivos o ruidosos, como el rock o el pop, podrían tener un ZCR más alto en comparación con géneros más melódicos y armónicos.

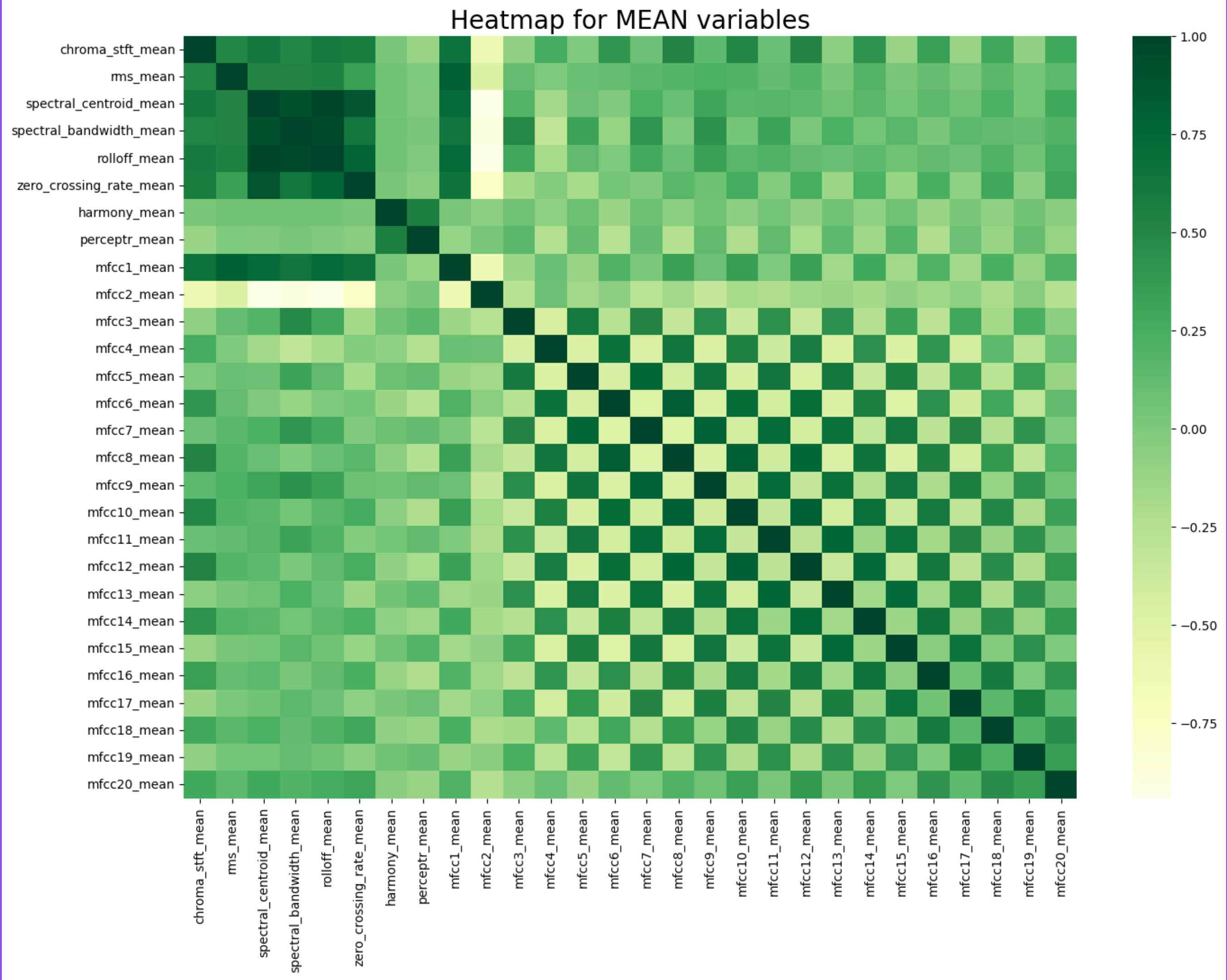
Tempo



Conceptos Básicos: El tempo se refiere a la velocidad o el ritmo de una pieza musical.

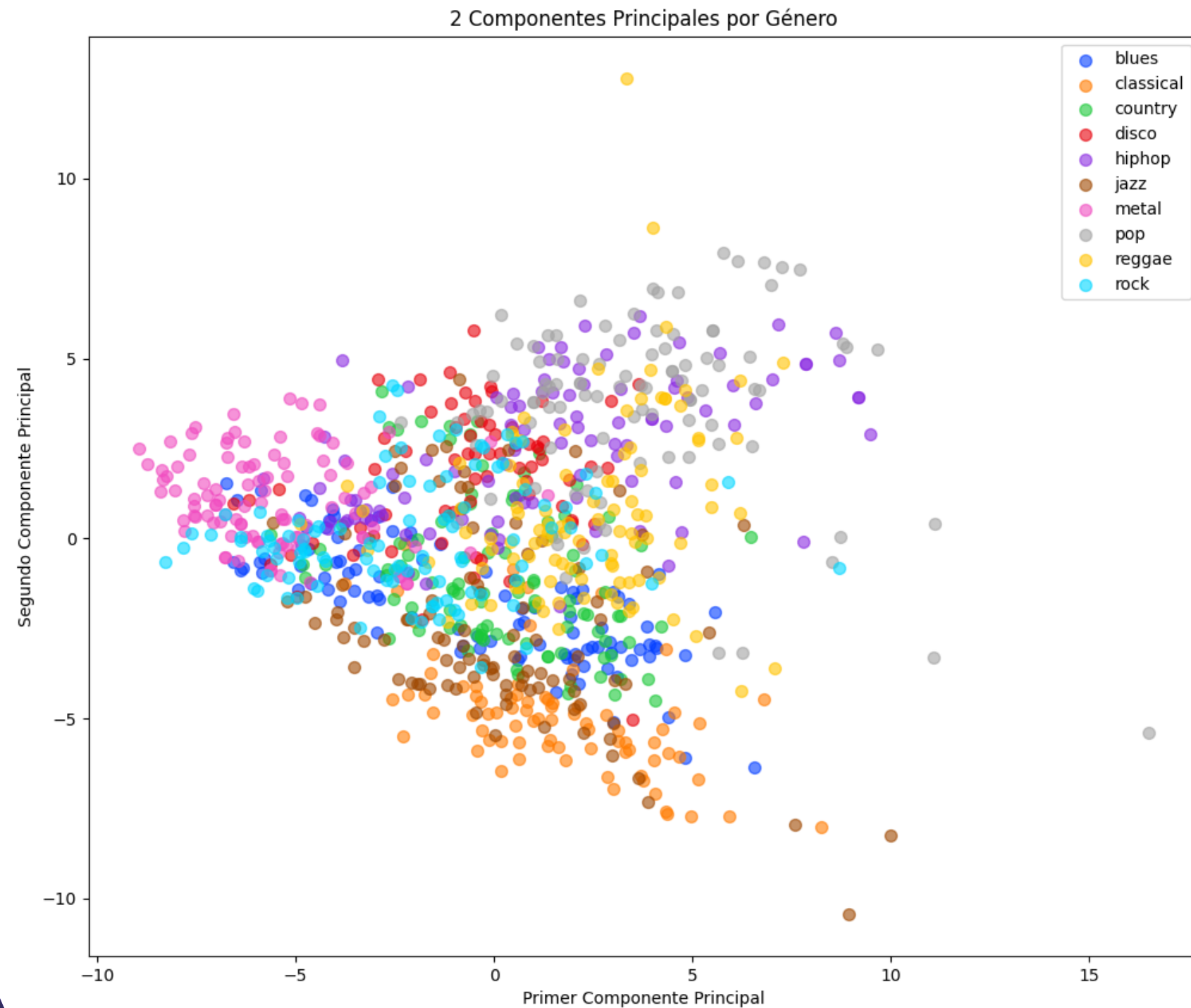
Relación con Música: El tempo es fundamental para el carácter y el estilo de una pieza musical. Géneros como el metal y el reggae tienen tempos más rápidos, mientras que generos como el country y el hip hop presentan tempos más lentos.

Corr Matrix

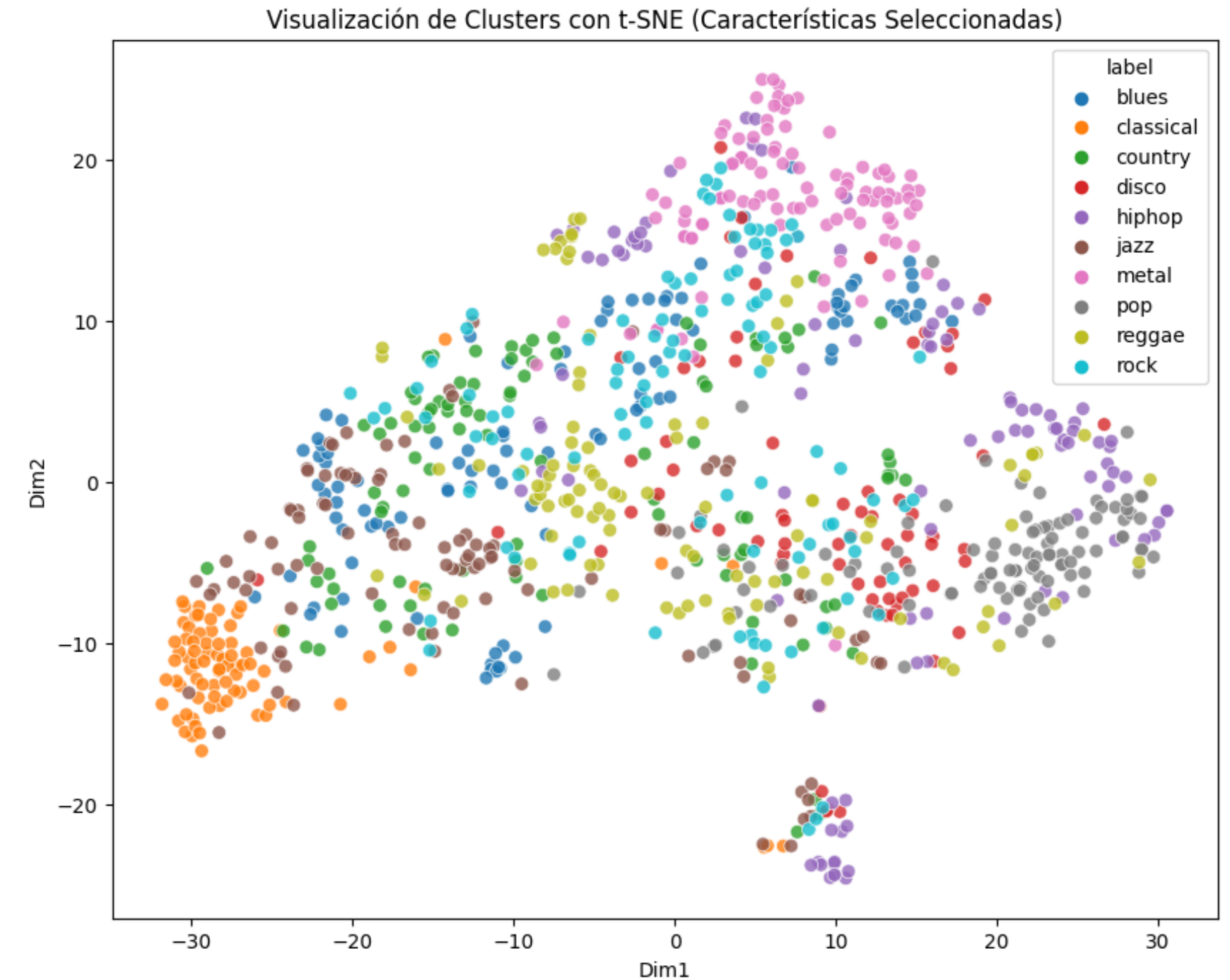


Analisis descriptivo Clusters

PCA



T-SNE



Modelado de la solución: Pipeline

Metrica elegida: Average weighted F1-Score
Optimizadores Utilizados: Bayes Search, Random Search

01

Preproscecamiento:
StandardScale()

02

Balanceo:
SMOTE()
RandomUnderSampler()

03

Modelos:
RandomForestClassifier()
XGBClassifier()
ExtraTreesClassifier()

```
# Preprocesamiento de datos
X = music_data.drop(columns=['filename', 'label'])
y = music_data['label']
label_encoder = LabelEncoder()
y = label_encoder.fit_transform(y)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Definir una métrica personalizada para F1-score con average='weighted'
f1_scorer = make_scorer(f1_score, average='weighted')

# Definir modelos, transformers y técnicas de balanceo
models = [RandomForestClassifier(), XGBClassifier(), ExtraTreesClassifier()]
samplers = [SMOTE(), RandomUnderSampler()]
scalers = [StandardScaler()]

best_f1 = 0
best_pipe = None

# Bucle a través de todas las combinaciones
for model in models:
    for sampler in samplers:
        for scaler in scalers:
            # Mensaje para mostrar la combinación actual
            print(f"Probando: Modelo - {model.__class__.__name__}, Sampler - {sampler.__class__.__name__}, Scaler - {scaler.__class__.__name__}")

            pipeline = Pipeline([
                ('scaler', scaler),
                ('sampler', sampler),
                ('model', model)
            ])

```



```
# Definir rangos de hiperparámetros para la búsqueda bayesiana
param_space = {
    'model__n_estimators': (100, 2000),
    'model__max_depth': (5, 50),
    'model__min_samples_split': (2, 20),
    'model__min_samples_leaf': (1, 20),
}

# Realizar la búsqueda de hiperparámetros utilizando el método bayesiano
bayes_search = BayesSearchCV(
    pipeline,
    param_space,
    scoring=f1_scorer,
    cv=5,
    n_iter=60, # Número de iteraciones de optimización
    random_state=42,
    n_jobs=-1 # Utiliza todos los núcleos disponibles
)
```

```
bayes_search.fit(X_train, y_train)

# Mejor modelo seleccionado
best_model = bayes_search.best_estimator_

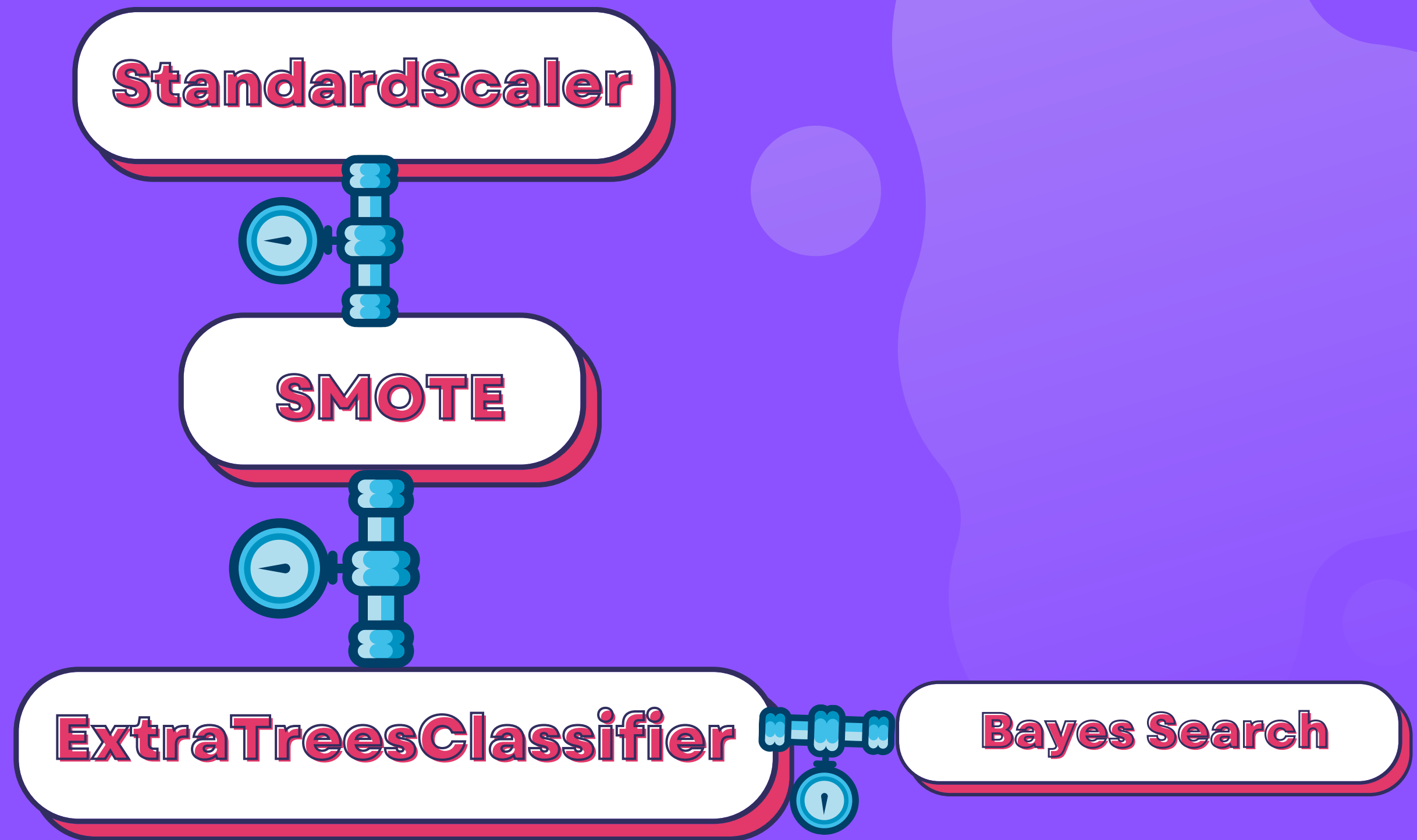
# Evaluar el modelo en el conjunto de prueba
y_pred = best_model.predict(X_test)
f1 = f1_score(y_test, y_pred, average='weighted')

# Actualizar el mejor modelo si es necesario
if f1 > best_f1:
    best_f1 = f1
    best_pipe = best_model

print(f"Resultado F1 Score: {f1}")
```

```
print(f"Mejor F1 Score: {best_f1}")
print(f"Mejor pipeline: {best_pipe}")
print(f"Mejores Hiperparámetros: {bayes_search.best_params_}")
```

Pipeline final



Resultados Modelo

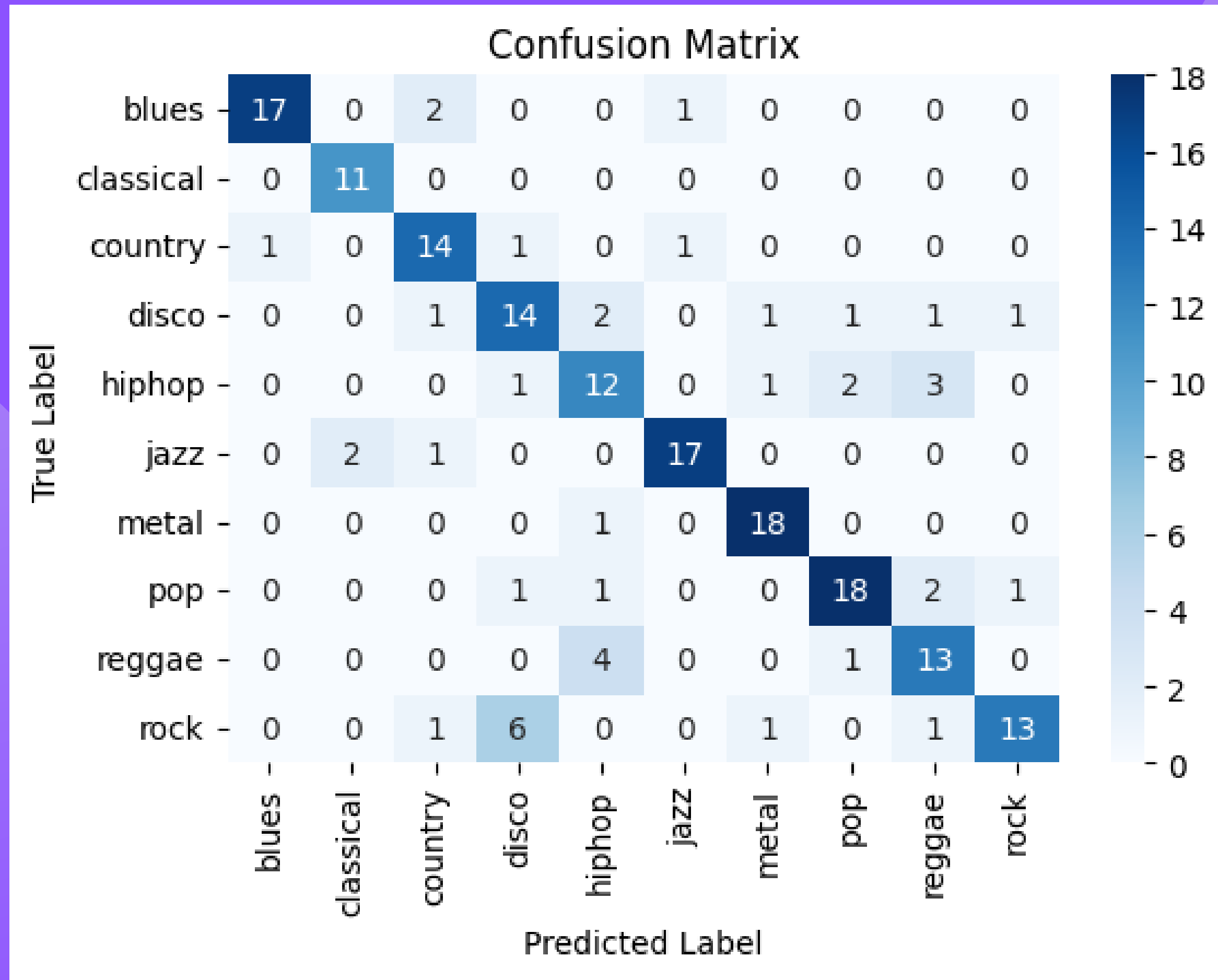
Average F1-Score

0.792

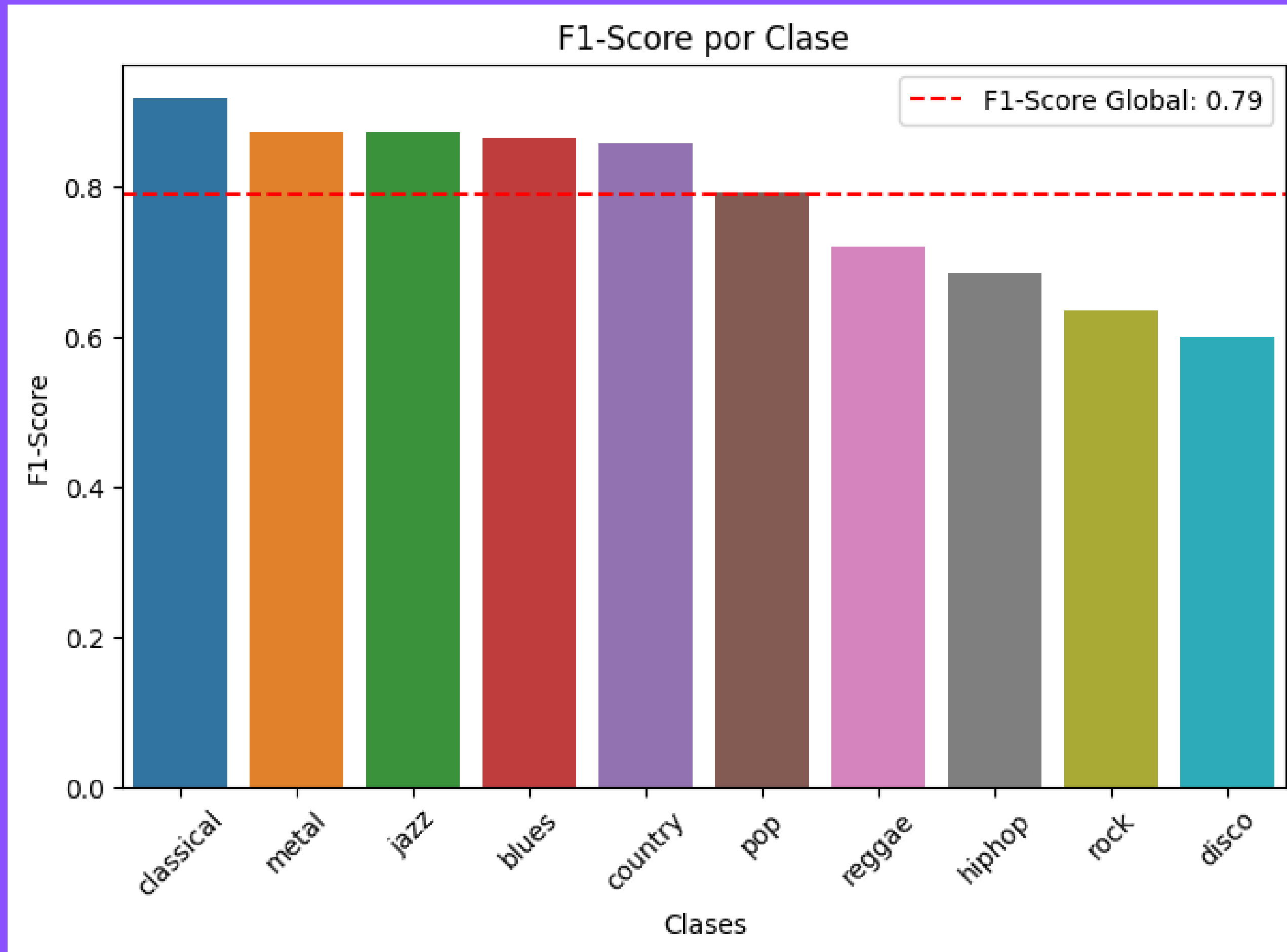
Mejores hiperparámetros

- model_max_depth: 39,
- modelmin_samples_leaf: 1,
- modelmin_samples_split: 2,
- model_n_estimators: 834

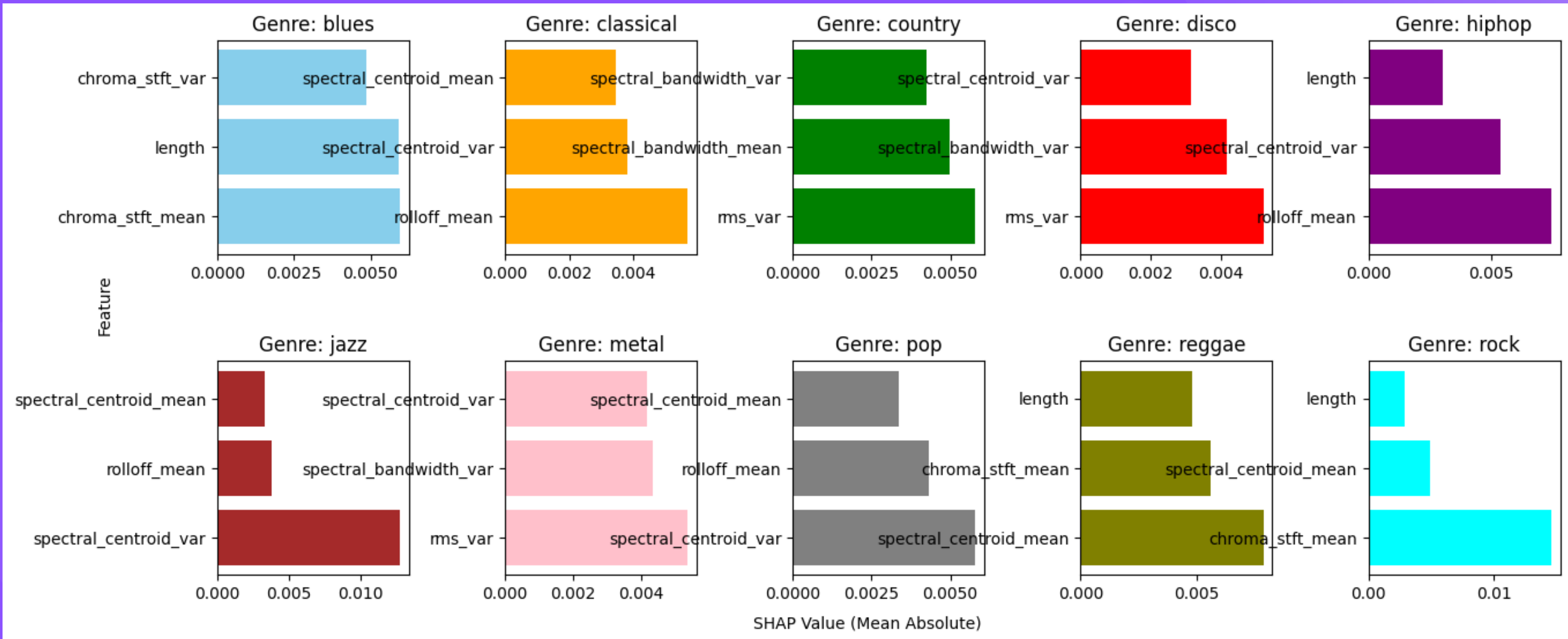
Matriz de confusión del modelo



Analisis de f1-Score



Feature Importance: Shap values



CONCLUSIONES Y ACCIONES DE NEGOCIO

