Estimación de la cantidad de mentas en una caja plástica mediante análisis de audio

Maria Jose Sanchez, Oscar Rodriguez, Mateo Sanchez
Escuela de Física, Universidad Industrial de Santander
Bucaramanga, Colombia
3 de mayo de 2025
{maria2221385, oscar2221532, sebastian2221784}@correo.uis.edu.co

***Abstract*—Esto al final**

I. Introducción

En entornos donde la observación directa no es posible o resulta costosa, los métodos indirectos de estimación se han convertido en herramientas esenciales. Poder determinar la cantidad de elementos contenidos dentro de un recipiente cerrado —sin necesidad de abrirlo o intervenirlo físicamente— representa un desafío con implicaciones relevantes para industrias como la farmacéutica, alimentaria y logística, donde la verificación rápida y no invasiva puede traducirse en mejoras sustanciales en eficiencia, seguridad y control de calidad.

Este proyecto aborda el problema de estimar la cantidad de mentas contenidas en una caja plástica a partir del análisis del sonido generado al agitarla. Aunque el escenario puede parecer simple, el enfoque utilizado refleja una tendencia cada vez más relevante en ingeniería y ciencia de datos: el uso de señales acústicas como fuente de información cuantitativa, procesadas mediante técnicas de machine learning.

A partir de un conjunto de grabaciones sonoras etiquetadas, se realizó un proceso de extracción de características acústicas, incluyendo coeficientes MFCCs, energía (RMS), Zero Crossing Rate y Spectral Cen- troid, con el fin de capturar la estructura espectral y dinámica de cada señal. Estas características fueron empleadas como entradas para un modelo de clasificación basado en el algoritmo Random Forest, el cual ofrece ventajas como alta precisión, manejo de datos no lineales y análisis de importancia de variables.

El presente trabajo demuestra cómo el aprendizaje automático, en conjunto con el procesamiento digital de señales, puede emplearse para resolver problemas reales de estimación sin recurrir a sensores complejos o técnicas invasivas. La capacidad de entrenar modelos que aprendan a identificar patrones en los datos acústicos abre nuevas posibilidades para la implementación de sistemas inteligentes en escenarios industriales donde la automatización del conteo, monitoreo o control de contenido es una necesidad creciente.

II. Objetivos

A. Objetivo general

Diseñar e implementar un sistema basado en técnicas de aprendizaje automático que permita estimar la cantidad de mentas contenidas en una caja, a partir del análisis del sonido generado al agitarla.

B. Objetivo especifico

- Recolectar grabaciones de audio suficientes para distintas cantidades de mentas
- Extraer características relevantes de las señales de audio (MFCCs, RMS, Zero Crossing Rate, Spectral Centroid), las cuales permitan representar cuantitativamente el sonido y su relación con la cantidad de mentas.
- Entrenar un modelo de Random Forest el cual sera el encargado de realizar las predicciones en base a las características extraídas.
- Evaluar la viabilidad y el desempeño del modelo.
- Desarrollar un sistema funcional que permita grabar audio en tiempo real y estimar automáticamente la cantidad de mentas contenidas en la caja.

III. Marco Teórico

A. Coeficientes Cepstrales en las Frecuencias de Mel (MFCCs)

Los *MFCCs* (Mel-Frequency Cepstral Coefficients) son una representación del contenido espectral de una señal de audio que busca imitar la percepción auditiva humana [1]. Se utilizan ampliamente en reconocimiento de voz, clasificación de sonidos y tareas donde se requiere extraer información estructurada de señales acústicas.

El cálculo de los MFCCs se basa en los siguientes pasos:

1) Preénfasis y enventanado: La señal de audio $x[n]$ es dividida en ventanas cortas (típicamente de 20 a 40 ms) donde se asume que el contenido estadístico es estacionario. Cada ventana se multiplica por una ventana de tipo Hamming para reducir los efectos de discontinuidad en los bordes.
2) Transformada de Fourier: Se calcula la *Transformada Rápida de Fourier (FFT)* de cada ventana para obtener el espectro de magnitudes $|X[k]|$. Este paso convierte la señal del dominio temporal al dominio frecuencial:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi kn/N}$$

3) Espectrograma en la escala Mel: Se aplica un banco de filtros triangulares en escala *Mel* sobre el espectro

de potencia. La escala Mel está diseñada para reflejar cómo los humanos perciben las frecuencias, siendo más sensible a las bajas frecuencias que a las altas. La conversión de una frecuencia f en Hz a la escala Mel está dada por:

$$m = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right)$$

4) Logaritmo de energía: A continuación, se toma el logaritmo natural de las energías filtradas para simular la percepción logarítmica del volumen por parte del oído humano:

$$\log(E_i), \quad i = 1, \ldots, M$$

donde E_i es la energía en el filtro i y M es el número total de filtros en la banca Mel.

5) Transformada Discreta del Coseno (DCT): Finalmente, se aplica la *Transformada Discreta del Coseno* a los log-energías obtenidos, lo que reduce la correlación entre coeficientes y deja los primeros valores como los más representativos del contenido del sonido:

$$c_n = \sum_{i=1}^{M} \log(E_i) \cdot \cos\left[\frac{\pi n}{M}(i - 0.5)\right], \quad n = 1, \ldots, N$$

Los coeficientes c_n obtenidos son los MFCCs. Normalmente se conservan los primeros 12 o 13 coeficientes, que contienen la mayor parte de la información relevante.

En este proyecto, los MFCCs permiten representar cada grabación como un vector numérico que refleja la estructura espectral del sonido producido por una determinada cantidad de mentas al ser agitadas. Al promediar los MFCCs a lo largo del tiempo, se obtiene una única representación fija por archivo, adecuada para su uso en algoritmos de aprendizaje automático.

B. Valor Cuadrático Medio (RMS)

El valor cuadrático medio (*Root Mean Square*, RMS) es una medida estadística de la magnitud de una señal que permite estimar su energía promedio a lo largo del tiempo. En Física, el RMS se interpreta como el valor eficaz de una señal oscilatoria, lo cual es útil para analizar, por ejemplo, tensiones y corrientes alternas, o el movimiento oscilatorio de un sistema.

Para una señal discreta $x[n]$ compuesta por N muestras, el valor RMS se define como:

$$x_{\text{rms}} = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x[n]^2}$$

Este valor representa una medida de la potencia acústica promedio de una señal de audio. En el contexto del proyecto, al agitar una caja con mentas, la energía de la señal registrada depende de factores como la cantidad de colisiones internas y su intensidad, que a su vez están influenciadas por el número de mentas dentro de la caja.

Cada colisión entre mentas produce una perturbación mecánica que se propaga como una onda de presión en el aire, la cual es captada por el micrófono. El RMS proporciona una estimación de la energía promedio asociada a esas oscilaciones acústicas.

C. Zero Crossing Rate, ZCR

La Tasa de Cruces por Cero (Zero Crossing Rate, ZCR) es una medida que describe la cantidad de veces que una señal cambia de signo dentro de un marco de tiempo [2]. En otras palabras, contabiliza cuántas veces la señal pasa de ser positiva a negativa o viceversa, normalizado por la longitud de la ventana analizada. Esta medida se utiliza frecuentemente para caracterizar la textura temporal de una señal de audio.

Matemáticamente, el ZCR se puede expresar como:

$$Z(i) = \frac{1}{2W_l}\sum_{n=1}^{W_l} \left|\text{sgn}(x_i(n)) - \text{sgn}(x_i(n-1))\right|,$$

donde $x_i(n)$ es la muestra n-ésima de la i-ésima ventana, W_l es la longitud de la ventana, y la función signo se define como:

$$\text{sgn}(x_i(n)) = \begin{cases} 1, & x_i(n) \geq 0, \\ -1, & x_i(n) < 0. \end{cases}$$

Una mayor tasa de cruces por cero suele estar asociada con frecuencias más altas en la señal, ya que implica una mayor cantidad de oscilaciones por unidad de tiempo.

D. Spectral Centroid

El centroide espectral (*Spectral Centroid*) es una medida del centro de "gravedad" espectral de una señal, útil para describir la posición espectral promedio de la energía contenida en un marco de audio. [3]

Matemáticamente, el centroide espectral del i-ésimo marco de señal, denotado por C_i, se define como:

$$C_i = \frac{\sum_{k=1}^{W_i} k \cdot X_i(k)}{\sum_{k=1}^{W_i} X_i(k)},$$

donde:

- $X_i(k)$ es la magnitud del espectro en la frecuencia correspondiente al índice k del frame i,
- W_i es el número total de bins espectrales del frame i.

Desde una perspectiva computacional, el centroide espectral se obtiene a partir de la transformada de Fourier de la señal y sus respectivas magnitudes. Conceptualmente, el centroide espectral indica hacia qué zona del espectro (bajas o altas frecuencias) se concentra mayor energía acústica. Valores más altos del centroide están asociados a sonidos con mayor contenido de altas frecuencias, comúnmente percibidos como más "brillantes", mientras que valores más bajos indican sonidos más "oscuros" o graves.

Además, esta medida es robusta al ruido y se usa ampliamente en aplicaciones como clasificación de timbre, análisis de textura sonora y segmentación de eventos acústicos [4]. En este proyecto, se espera que el centroide espectral refleje variaciones debidas al número de mentas en la caja, ya que colisiones múltiples tienden a generar componentes de mayor frecuencia en el espectro.

E. Random Forest

El algoritmo *Random Forest* es un método de aprendizaje automático supervisado que se utiliza tanto para clasificación como para regresión. Se basa en la construcción de múltiples árboles de decisión entrenados sobre subconjuntos aleatorios de los datos de entrenamiento. Cada árbol produce una predicción, y la predicción final del modelo se obtiene por votación mayoritaria (en tareas de clasificación) o por promedio (en tareas de regresión) (ver diagrama 1).

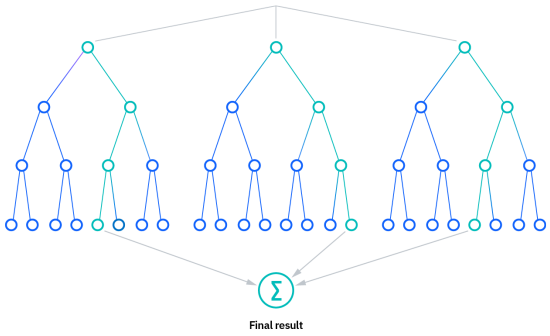

Fig. 1. Diagrama del funcionamiento del algoritmo Random Forest

El nombre *Random Forest* proviene de dos aspectos clave del modelo: por un lado, el uso de *bagging* (bootstrap aggregating), que implica entrenar cada árbol con un subconjunto diferente de datos seleccionados al azar con reemplazo; y por otro lado, la selección aleatoria de un subconjunto de características para dividir en cada nodo durante la construcción del árbol. Esta aleatoriedad introducida reduce la correlación entre los árboles y mejora la generalización del modelo [5].

Las principales ventajas del algoritmo Random Forest son:

- Tiene alta precisión y generalización.
- Es robusto frente al sobreajuste, especialmente cuando se entrena con suficientes árboles.
- Puede manejar eficientemente grandes conjuntos de datos y características.
- Proporciona medidas de importancia para cada característica.

De acuerdo con IBM [6], Random Forest se ha convertido en una herramienta fundamental para tareas de minería de datos, detección de fraudes, diagnóstico médico, y procesamiento de señales, debido a su balance entre precisión, interpretabilidad y eficiencia computacional.

Cada árbol del modelo realiza una predicción independiente, y el resultado final se obtiene por votación:

$$\hat{y} = \text{mode}\left(T_1(x), T_2(x), \ldots, T_B(x)\right)$$

donde $T_i(x)$ representa la predicción del i-ésimo árbol para la entrada x, y B es el número total de árboles del bosque.

F. Grid Search

En modelos de aprendizaje automático, los hiperparámetros son valores configurables que controlan el proceso de entrenamiento (por ejemplo, la profundidad de un árbol o el número de árboles en un Random Forest). A diferencia de los parámetros internos del modelo, los hiperparámetros no se aprenden directamente a partir de los datos, sino que deben seleccionarse cuidadosamente para maximizar el rendimiento del modelo.

Grid Search es una técnica sistemática para la optimización de hiperparámetros. Consiste en definir una malla de posibles valores para cada hiperparámetro y luego entrenar y validar el modelo para todas las combinaciones posibles de estos valores. La evaluación se realiza generalmente usando validación cruzada, lo cual permite obtener un estimado más confiable del rendimiento del modelo.

El procedimiento se puede resumir en los siguientes pasos:

1) Se define un conjunto de valores posibles para cada hiperparámetro.
2) Se entrena el modelo con cada combinación de valores.
3) Se evalúa cada combinación mediante validación cruzada.
4) Se selecciona la combinación con mejor desempeño según una métrica específica (como la precisión o el *F1-score*).

Esta técnica garantiza que se explore exhaustivamente el espacio de búsqueda, aunque su desventaja es que puede ser computacionalmente costosa si el número de combinaciones es elevado. No obstante, se trata de un método robusto y ampliamente utilizado en tareas de clasificación y regresión cuando se dispone de recursos computacionales suficientes.

IV. Metodología

A. Diseño y Materiales

Para el desarrollo de este proyecto se utilizaron caramelos Tic Tac®, un producto fabricado por la empresa italiana Ferrero. Estos caramelos se caracterizan por su consistencia sólida, su forma elipsoidal y su tamaño uniforme. Además, vienen contenidos en una caja rígida de plástico, la cual resulta ideal para generar sonidos reproducibles al ser agitada. Estas propiedades físicas —tanto del contenido como del envase— fueron fundamentales para establecer un sistema experimental controlado en el cual el sonido generado dependiera principalmente de la cantidad de caramelos en el interior.

(AÑADIR PARTE DEL MICRÓFONO)

B. Experimental

La fase experimental comenzó con la recolección de datos mediante grabaciones de audio. Se realizaron un total de 60 grabaciones, correspondientes a cantidades de mentas que iban desde 1 hasta 29 unidades. Cada grabación fue realizada bajo condiciones controladas, manteniendo constante el tipo de recipiente (caja original de Tic Tac®) y el método de agitación

(5 agitadas en aproximadamente 3 segundos), con el objetivo de minimizar fuentes externas de variabilidad.

Posteriormente, se desarrolló un script en `Python` que permitió la extracción de características relevantes del audio. En particular, se calcularon los coeficientes cepstrales en las frecuencias de Mel (MFCCs), la energía cuadrática media (RMS), la tasa de cruces por cero (Zero Crossing Rate) y el centroide espectral (Spectral Centroid). Estas características fueron seleccionadas debido a su capacidad para capturar diferentes aspectos del contenido sonoro: los MFCCs codifican la envolvente espectral, el RMS representa la intensidad del sonido, el ZCR indica la tasa de cambio de polaridad, y el centroide espectral proporciona una medida del brillo o prominencia de altas frecuencias.

Los valores extraídos fueron almacenados en un `DataFrame` y exportados a un archivo `.csv`, el cual sirvió como base para el entrenamiento del modelo de aprendizaje automático.

Todos los datos, scripts de procesamiento y entrenamientos pueden encontrarse en el siguiente repositorio de GitHub: https://github.com/Matt22vL/RETOS-.

V. Modelo de predicción (Random Forest)

Para abordar el problema de estimar la cantidad de mentas contenidas en una caja a partir del análisis de sus características acústicas, se optó por el uso del algoritmo de *Random Forest*, una técnica de aprendizaje supervisado ampliamente utilizada por su robustez, precisión y facilidad de interpretación.

Random Forest es un método de ensamblado que construye múltiples árboles de decisión durante la fase de entrenamiento y, para la clasificación, emite como predicción final el voto mayoritario entre dichos árboles. Esta estrategia permite reducir el sobreajuste típico de los árboles individuales, y al mismo tiempo mejora la capacidad del modelo para generalizar a nuevos datos. En este proyecto, dicha característica resulta fundamental, dado que las señales de audio pueden contener ruido y variabilidad asociada a la forma en la que se agita la caja o la posición del micrófono.

El conjunto de datos fue dividido en un 80% para entrenamiento y un 20% para prueba utilizando la función `train_test_split` de la biblioteca `scikit-learn`. Las características extraídas de los audios (MFCCs, RMS, *Zero Crossing Rate*, *Spectral Centroid*, entre otras) fueron utilizadas como variables predictoras, mientras que la etiqueta objetivo corresponde al número de mentas presentes en la caja.

El modelo fue implementado con `RandomForestClassifier` de `scikit-learn`, y posteriormente optimizado mediante una búsqueda de hiperparámetros, cuyo proceso se describe con mayor detalle en la siguiente subsección.

Finalmente, el modelo ajustado fue evaluado utilizando métricas clásicas de clasificación y visualizaciones como la matriz de confusión y la importancia relativa de las características extraídas.

A. Búsqueda de parámetros con Grid Search

Para optimizar el rendimiento del modelo Random Forest, se utilizó el método de Grid Search. Este método permite explorar de manera sistemática una combinación de hiperparámetros definidos previamente, con el fin de encontrar la configuración que proporcione el mejor desempeño del modelo.

En particular, se consideraron los siguientes hiperparámetros:

- `n_estimators`: número de árboles en el bosque.
- `max_depth`: profundidad máxima de cada árbol.
- `min_samples_split`: número mínimo de muestras requeridas para dividir un nodo.
- `min_samples_leaf`: número mínimo de muestras que debe tener una hoja.
- `max_features`: número de características a considerar para la mejor división.
- `bootstrap`: indica si se usan muestras con reemplazo.

La malla de búsqueda empleada fue la siguiente:

```
param_grid = {
    'n_estimators': [200, 300],
    'max_depth': [20, None],
    'min_samples_split': [2, 4],
    'min_samples_leaf': [1, 2],
    'max_features': ['sqrt'],
    'bootstrap': [True]
}
```

Listing 1. Definición de la malla de hiperparámetros para Grid Search

El procedimiento se llevó a cabo utilizando `GridSearchCV` de la biblioteca `scikit-learn`, con validación cruzada de 5 pliegues (`cv=5`) y la métrica de precisión como criterio de evaluación. De esta manera, se garantizó una evaluación robusta del rendimiento del modelo en distintos subconjuntos del conjunto de entrenamiento.

Una vez finalizada la búsqueda, se seleccionó el conjunto de hiperparámetros que obtuvo la mayor precisión promedio, y se utilizó para entrenar el modelo final. Esta optimización permitió mejorar significativamente la capacidad del clasificador para predecir correctamente la cantidad de mentas en nuevos datos.

B. Métricas de precisión

Una vez entrenado el modelo con los hiperparámetros óptimos hallados mediante *GridSearch*, se procedió a evaluar su desempeño sobre el conjunto de prueba utilizando diversas métricas de precisión. Los mejores parámetros encontrados fueron:

- `bootstrap = True`
- `max_depth = none`
- `max_features = sqrt`
- `min_samples_leaf = 1`
- `min_samples_split = 4`
- `n_estimators = 200`

El modelo optimizado alcanzó una precisión de validación cruzada de **0.813**, y una precisión final sobre el conjunto de prueba de **0.836**.

a) Importancia de características: Antes de analizar los resultados cuantitativos, se evaluó la importancia relativa de cada una de las variables de entrada utilizando la propiedad `feature_importances_` del clasificador basado en árboles. En la Figura 2 se muestra el ranking de las características.

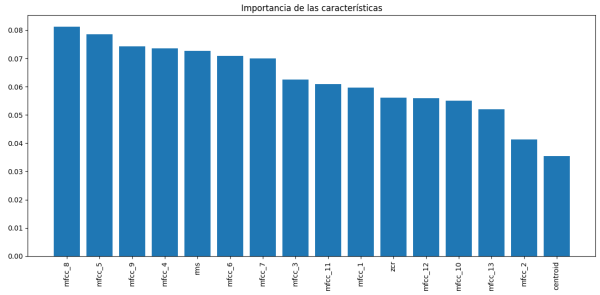

Fig. 2. Importancia relativa de las características según el modelo entrenado. Las más relevantes son los coeficientes MFCC, especialmente `mfcc_8`, `mfcc_5` y `mfcc_9`.

Este análisis revela que las características relacionadas con los coeficientes MFCC son las más informativas para la clasificación, lo cual es coherente con la literatura en análisis de señales de audio, donde los MFCC suelen capturar patrones acústicos distintivos.

b) Matriz de confusión: En la Figura 3 se presenta la matriz de confusión del modelo final. Se observa un alto grado de precisión, con una fuerte presencia de valores en la diagonal principal, lo cual indica una correcta identificación de la mayoría de las clases.

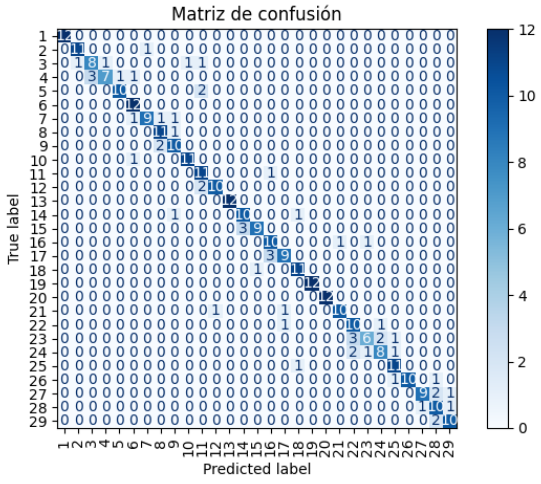

Fig. 3. Matriz de confusión normalizada. Cada celda muestra la cantidad de predicciones para cada par (real, predicho).

En general, se observa un bajo nivel de confusión entre clases distintas, lo cual sugiere que el modelo ha capturado patrones discriminativos efectivos. No obstante, algunas clases como la 23 y la 4 presentan menor rendimiento, lo cual podría deberse a similitudes acústicas con otras clases.

1) Precisión por clase: En la gráfica 4 se muestra la precisión del modelo para cada conjunto de numero de mentas, es posible notar que aproximadamente cada 5 mentas la precisión del modelo mejora.

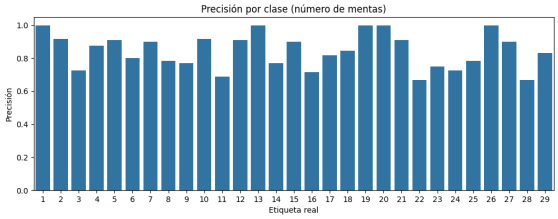

Fig. 4. Precisión del modelo Random Forest por numero de menta

VI. Puesta en practica

Debido al rendimiento de nuestro modelo para cierto numero de mentas, se probo el modelo(con distintos grabaciones de audios a las usadas en el entrenamiento) para 5,10,15,20,25 mentas con el argumento de que son las cantidades que en general tienen una tendencia a mejor comportamiento.

Así entonces se obtuvo la Matriz de confusión 5 en donde el comportamiento no es el esperado.

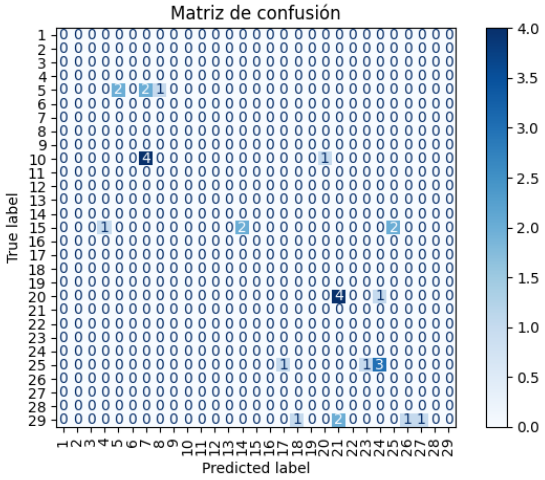

Fig. 5. Matriz de confusión para grabaciones de 5,10,15,20,25 mentas

Así entonces, es posible notar que se puede aprovechar el echo del peso estadístico que tiene cada estimación para realizar un sistema que funcione para el conjunto de mentas $\{5,10,15,20,25\}$. De esta manera, para mejorar la estimación del numero de mentas, se uso 5 grabaciones de un mismo numero de mentas, posteriormente se le realizo la estimación individual, y finalmente se realizo un promedio ponderado teniendo en cuenta la probabilidad de cada estimación.

VII. Implementación del sistema

Teniendo en cuenta todas las consideraciones anteriormente mencionadas.

VIII. Discusion

IX. Conclusiones

References

[1] B. Logan, "Mel frequency cepstral coefficients for music modeling," *Proc. 1st Int. Symposium Music Information Retrieval*, 11 2000.

[2] T. Giannakopoulos and A. Pikrakis, "Chapter 4 - audio features," in *Introduction to Audio Analysis*, T. Giannakopoulos and A. Pikrakis, Eds. Oxford: Academic Press, 2014, pp. 59–103. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780080993881000042

[3] V. K. Harpale and V. K. Bairagi, "Chapter 3 - seizure detection methods and analysis," in *Brain Seizure Detection and Classification Using EEG Signals*, V. K. Harpale and V. K. Bairagi, Eds. Academic Press, 2022, pp. 51–100. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780323911207000086

[4] M. Massar, M. Fickus, E. Bryan, D. Petkie, and A. Terzuoli, "Fast computation of spectral centroids," *Adv. Comput. Math.*, vol. 35, pp. 83–97, 07 2011.

[5] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[6] IBM, "¿qué es un bosque aleatorio?" https://www.ibm.com/mx-es/think/topics/random-forest, 2023, consultado el 3 de mayo de 2025.

[7] B. McFee *et al.*, "librosa: Audio and music signal analysis in python," 2023. [Online]. Available: https://librosa.org/doc/main/generated/librosa.feature.mfcc.html

[8] C. Panagiotakis and G. Tziritas, "A speech/music discriminator based on rms and zero-crossings," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 155–166, 2005.