📄 **Ficha Técnica del Proyecto de Machine Learning**

**1. Información General**

| **Campo** | **Descripción** |
| --- | --- |
| **Nombre del Proyecto** | Clasificador Automático de Géneros Musicales |
| **Equipo de Trabajo** | Eylin Alejandra Mora Arboleda  Yulitza Tatiana Caicedo Mosquera  Juan Esteban Gómez Londoño  Geronimo Martinez Higuita |
| **Fechas** | Inicio: [17/06/2025] - Finalización: [DD/06/2025] |
| **Instructor/a** | Luz Eliana Martinez Ramos |
| **Descripción del proyecto.** | Desarrollar un modelo de Machine Learning capaz de clasificar canciones en su género musical correcto (ej. Rock, Pop, Jazz, Clásica) a partir del análisis de sus características de audio. El objetivo es crear una herramienta funcional que, a partir de un archivo de audio, pueda predecir su género de forma automática. |
| **Módulos Proyecto:** | * Extracción de Características: Módulo para procesar archivos de audio **.wav** y extraer métricas relevantes (**tempo, croma, MFCCs**). * Entrenamiento de Modelo: Script para entrenar y comparar diferentes algoritmos de clasificación y seleccionar el de mejor rendimiento. * API de Predicción: Interfaz simple para recibir datos de una canción y devolver el género predicho. * Interfaz Web (Streamlit): Aplicación interactiva para que un usuario pueda probar el modelo. |

**2. Descripción de los Datos**

**Origen y Estructura**

* **Fuente**: El proyecto utilizará el dataset "GTZAN Genre Collection", disponible públicamente en Kaggle: [https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification](https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification" \t "_blank).
* **Formato**: El dataset principal es un archivo features\_3\_sec.csv (59 MB) que contiene las características extraídas. Adicionalmente, se cuenta con los 1000 archivos de audio originales en formato .wav.
* **Registros**: El archivo CSV contiene **9990 filas**, donde cada fila representa un segmento de 3 segundos de una canción.
* **Variables**: El archivo CSV tiene **59 columnas**, incluyendo el nombre del archivo, la duración, el género (variable objetivo) y 57 características numéricas del audio.

**3. Modelo**

* **Red Neuronal**: \_\_\_\_\_ SI \_\_\_X\_\_\_NO **Tipo red Neuronal**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* **Tipo de aprendizaje**: \_\_X\_\_ Supervisado \_\_\_\_\_\_\_ No Supervisado \_\_\_\_\_ por Refuerzo
* **Algoritmos**:
  + **Random Forest (Bosque Aleatorio):** Es robusto y generalmente ofrece un gran rendimiento en problemas de clasificación.
  + **Máquinas de Vectores de Soporte (SVM):** Eficaz para encontrar el mejor hiperplano que separe las clases.
  + **K-Vecinos más Cercanos (K-NN):** Un algoritmo simple basado en la similitud entre los datos.
* **Librerías**:
  + **Pandas:** Para la manipulación y carga de los datos del archivo CSV.
  + **Scikit-learn:** Para implementar, entrenar y evaluar los modelos de clasificación.
  + **Librosa:** (Opcional, si se desea hacer extracción de características desde cero) Para analizar los archivos de audio.
  + **Matplotlib / Seaborn:** Para crear las visualizaciones de datos y resultados.
* **Selección del Modelo**: Se entrenarán los tres algoritmos mencionados (Random Forest, SVM, K-NN) utilizando el mismo conjunto de datos. El modelo final será aquel que obtenga el mayor **porcentaje de precisión (accuracy)** en el conjunto de prueba.

**4. Entrenamiento y Evaluación**

* **Describa el tipo de entrenamiento realizado al modelo:** El conjunto de datos se dividirá en dos partes: un **80% para entrenamiento** y un **20% para prueba**. Los modelos serán entrenados únicamente con el conjunto de entrenamiento, aprendiendo a asociar las características del audio con su respectivo género musical. Se utilizará una técnica de validación cruzada para asegurar la robustez del modelo.
* **Describa evaluación y resultados de la evaluación del modelo. (Porcentaje de precisión): *La evaluación se realizará con el conjunto de prueba (el 20% de los datos que el modelo nunca ha visto). Se medirá el rendimiento de cada algoritmo calculando su porcentaje de precisión, que indica la proporción de canciones clasificadas correctamente. El objetivo es alcanzar una precisión superior al 75%. El resultado final se presentará en una matriz de confusión para visualizar qué géneros son más fáciles o difíciles de predecir para el modelo.***

**5. Diseño de streamlit**

***(Aquí describirías la captura de pantalla que vas a insertar en el documento)***

La interfaz web tendrá un diseño limpio y sencillo. Contará con:

1. Un título claro: "Clasificador de Géneros Musicales".
2. Una sección para que el usuario seleccione una canción de muestra de una lista desplegable.
3. Un botón grande con el texto "Clasificar Género".
4. Un área de resultados donde se mostrará el género predicho (ej. "Género: Rock") y una visualización gráfica de algunas de las características del audio de la canción.

**5. Anexos**

* **Código**: <https://github.com/MHGeronimo/Proyecto_ML_SteticSoft_Clasificador_de_Musica.git>
* **Visualizaciones**: [Ej: "Figura 1: Matriz de confusión", "Figura 2: Curva ROC"]