Ciencia de Datos
Tarea 5

Para entregar el 27 de mayo de 2020

1. Considera los datos `diabetes.csv`, que contiene diferentes características médicas de mujeres de la tribu de Indios Pima. La variable de respuesta es `Diabetes`, con valores 1 y 0. Separa un conjunto de entrenamiento y prueba de forma aleatoria con una proporción 70 y 30 %, respectívamente.

   a) Implementa el modelo clásico de perceptrón (versión que trabaja en línea) y pruébalo con los datos de diabetes. Reporta los errores de entrenamiento y prueba ¿Qué puedes decir sobre su desempeño?

   b) Implementa clasificadores basados Regresión Logística, LDA y QDA. Compara su desempeño.

   c) Las curvas ROC (Receiver Operating Characteristics) es un método muy común para comparar algoritmos de clasificación binarios basado en la tabla de errores (falsos positivos y falsos negativos) que se cometen. Usa los resultados de los incisos anteriores para comparar los clasificadores usando este criterio. ¿Cuál método elegirias? Usa el criterio del área bajo la curva (AUC).[1]

2. Considera los datos que se encuentran en el archivo `my_all_tracks.csv`. Estos corresponden a un extracto del Free Music Archive (FMA) [1], que es una base de datos muy extensa de archivos de audio usada para diversas tareas de Music Information Retrieval (MIR), como lo mencionamos en clase[2].

   Los datos describen diferentes características de los audios. En forma general, podemos mencionar tres partes (ve Nota 1):

   - **Tracks** (columnas 1 a 23): Información correspondiente a la canción, álbum y artista. De aquí destacamos `track.genre1`, que indica el género dado a cada canción (ve Nota 2)
   - **Características de audio** (columnas 24 a 31): Características de audio en forma de *indicadores*, extraídos con la API de Spotify (antes Echonest).
   - **Características de la señal** (columnas 32 a 549): Características de audio

[1] Hay bastante literatura sobre ROC, como referencia, puedes consultar el paper de T. Fawcett, [2]. Puedes usar `roc_curve`, del módulo `sklearn_metrics`, y demás clases relacionadas. Ve la documentación y los ejemplos que aparecen ahí.

[2] Este subconjunto de datos lo construí basado en los archivos originales y haciendo mucho preproceso de los mismos. En general, traté de incluir la información relevante para éste ejercicio, quitando valores nulos que no podían estimarse, entre otras cosas. Los datos originales son considerablemente mas grandes.

obtenidas mediante diferentes análisis de la señal correspondiente según lo platicamos en la clase. Estas se reportan mediante estadísticos (promedio, desviación estándar, mediana, mínimos, máximos, kurtosis, entre otros) calculados en ciertas porciones de la señal. En clase hablamos un poco sobre esto, pero si quieres profundizar puedes puedes consultar las referencias de [1], el `GitHub` correspondiente o preguntarme.

a) Realiza un análisis no supervisado para encontrar patrones "interesantes" en los datos. Para esto, usa las variables que corresponden a las características de audio y otras que consideres adecuadas (por ejemplo, `track.duration`).

¿Puedes identificar los géneros, o al menos, algún subconjunto de ellos?

Obten visualizaciones apropiadas de baja dimensión. Puedes usar los métodos de visualización y clustering que consideres apropiados, pero incluye spectral clustering. Documenta el procedimiento y hallazgos que encuentres.

b) Considera `audio_features.danceability` y `audio_features.energy` como variables de respuesta. Selecciona un conjunto de entrenamiento y otro de prueba. Para cada variable de respuesta:

- Construye una variable con 3 categorias (baja, media, alta) a partir de sus valores numéricos
- Ajusta clasificadores basados en LDA, QDA, Multilogit y Redes Neuronales, para estimar el nivel de "bailabilidad" y "energía" usando el bloque de características de la señal como covariables.
  Compara los resultados para cada clasificador y documenta tus hallazgos.
- Estima el nivel de "bailabilidad" y "energía" en algunos audios del conjunto de validación que se encuentra en el archivo `my_all_tracks_No_genre_2019.csv`. Verifica `cualitativamente` tus resultados. ¿Te parecen adecuados? (ver Nota 5)

Notas.

1. Los archivos de datos se encuentran en `http://201.116.172.100/clases_v/`, ya que son demasiado grandes para subirlos al Moodle.

2. El género que contiene la columna `track.genre1` fue obtenido (cuando no era proporcionado originalmente en `track.genre_top`), siguiendo el esquema jerárquico descrito en `genres.csv`. Se muestran los primeros dos niveles de la jerarquía. Puedes recurrir a la documentación original para más detalles.

3. Puedes usar un subconjunto aleatorio de los datos, en caso de que los modelos que uses tengan problemas con el tiempo de computo.

4. Hay algunas variables en las características de las señales que tienen valores constantes (o casi constantes), por ejemplo, algunos mínimos... Te recomiendo quitar éstas variables, ya que puden ocasionar problemas para los modelos de clasificación.

5. Puedes oir un extracto de 30 segundos de los archivos de audio del conjunto de datos. Estos se encuentran en el servidor `http://201.116.172.100/clases_v/` para acceder fuera de Cimat y `http://10.14.40.22/clases_v/` para acceder dentro. Los pongo de ésta forma porque son muchos Gb de información para ponerlo en la página del curso. ¡Muchas gracias a Héctor! por hacer el HTML con el codec adecuado para reproducir los archivos.

Bibliografía

[1] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. Fma: A dataset for music analysis. In *18th International Society for Music Information Retrieval Conference*, 2017.

[2] Tom Fawcett. An introduction to roc analysis. *Pattern Recognit. Lett.*, 27(8):861–874, 2006.