


EM+Kmeans En el archivo `instrumentos.zip` encontrará audios de diferentes instrumentos musicales. Se desea clasificar entre las 5 clases de instrumentos presentes en la base de datos.

(a) *Exploración de datos:*

- Utilizando `load` (librosa), construir 5 conjuntos de datos de entrenamiento y 5 de testeo (uno por instrumento). Reservar el último archivo de cada instrumento para el conjunto de testeo, y utilizar el resto para entrenamiento.
- Graficar la señal temporal de la señal de testeo de la guitarra en función del tiempo (en segundos).
- Reproducir el audio de testeo de la guitarra utilizando `Audio` (IPython).
- Utilizando `ShortTimeFFT` (scipy.signal). Extraer un espectrograma de cada señal. : Una buena selección de criterios es utilizar una fft de 64 puntos, un solapamiento del 50 %, descartar la parte simétrica del espectro y utilizar ventana de *hamming* de la misma cantidad de puntos de la fft.


(b) *Entrenamiento:*

- Entrenar 5 algoritmos KMeans (uno por cada instrumento). El código debe estar estructurado de la siguiente manera:


```
class Kmeans:
    # Inicializar atributos y declarar hiperparámetros
    def __init__(self,...

    # Etapa de entrenamiento
    def fit(self,X):

    # Etapa de testeo
    def predict(self,X):
```

- Utilizando `GaussianMixture` (sklearn), entrenar 5 mezclas de 6 gaussianas diagonales cada una (una por cada instrumento) utilizando el algoritmo EM. Inicializar dicho algoritmo utilizando el Kmeans desarrollado en el ítem anterior. : Si bien es evidente que los centroides representan las medias de las gaussianas, no es tan claro pensar como inicializar los pesos y varianzas. Justificar su criterio de inicialización.

(c) *Testeo:*

- Para todas las combinaciones de desea evaluar que tan verosímil es que las muestras de la clase i -ésima correspondan al modelo j -ésimo. Indicar las *log-verosimilitud* correspondientes en un cuadro de doble entrada del algoritmo EM.
- Cada algoritmo j -ésimo define una distribución $p(x|j)$. Asumiendo una probabilidad de cada clase $P(j)$ proporcional a la cantidad de muestras de entrenamiento de cada instrumento, calcular las probabilidades a posteriori a partir de la regla de Bayes. Es decir, indicar en un cuadro de doble entrada la probabilidad de la clase j -ésima para las muestras correspondientes al instrumento i -ésimo $P(j|\mathcal{D}_i)$. : La función `softmax` (scipy) puede ser útil.
- Sea $x(t)$ la señal correspondiente al audio de testeo de la guitarra. Graficar $\log P(j|x(t))$ en función del tiempo, para cada uno de los 5 instrumentos (indexados por j).