

## Regresión Logística

Se desea desarrollar un clasificador de imágenes de gatos y perros utilizando regresión logística. Para ello descargar la base de datos `dog-and-cat-classification-dataset` de kaggle. : La siguiente instrucción puede ayudar con la descarga:  
`kagglehub.dataset_download("bhavikjikadara/dog-and-cat-classification-dataset")`

### (a) Pre-Procesamiento:

1. Las imágenes poseen diferentes tamaños. Convertirlas todas a  $16 \times 16$  utilizando `resize` (PIL).
2. Las imágenes también poseen diferentes formatos. Convertirlas todas a escala de grises.
3. Mostrar 5 imágenes.
4. Utilice el comando `train_test_split` (sklearn) para definir dos conjuntos de datos. El conjunto de entrenamiento debe contener 20000 muestras, el resto serán de testeo.

### (b) Clasificación:

1. Utilizando `LogisticRegression` (sklearn), realizar una regresión logística sin regularización.
2. Implementar una función que permita calcular la *accuracy* a partir de la salida de `predict`. Reportar el *accuracy* de entrenamiento y testeo.
3. Implementar una función que permita calcular la *cross-entropy* a partir de la salida de `predict_log_proba`. Reportar la *cross-entropy* de entrenamiento y testeo. : Por un tema de continuidad asuma que  $p \log(q) = 0$  si  $p = q = 0$ .
4. Crear una imagen de  $16 \times 16$  a partir del consejero de la guía  y clasificarlo. : En el repositorio encontrará una imagen del consejero.

(c) *Regularización*: Utilice un mapa polinómico de orden 2 y un término de penalización para regularizar el problema. Indicar el *accuracy* tanto para el entrenamiento como el testeo. : Una regresión de muchos parámetros puede tardar un tiempo. Un buen tip es configurar `max_iter` en un valor muy bajo para probar el código y una vez funcionando darle un valor razonable.