

# Práctica – Semana 02

Generar un cuaderno de Colab con el siguiente formato NombreApellido\_Practica02\_ML

Ejemplo: JuanPerez\_Practica02\_ML. Explica los resultados finales de cada ejercicio.

## 1. Explora el dataset CIFAR-10

- Carga cifar10 desde keras.datasets.
- Muestra 5 imágenes de ejemplo con sus etiquetas.
- Investiga la forma y el rango de valores de los píxeles.
- ¿Cuántas clases tiene el dataset?

## 2. Normaliza y codifica etiquetas

- Convierte las imágenes de CIFAR-10 a flotantes y normalízalas a [0,1].
- Aplica to\_categorical a las etiquetas para una red multicategoría.

## 3. Crea tu primera CNN para CIFAR-10

- Crea una CNN con:
  - 1 capa Conv2D
  - 1 MaxPooling2D
  - 1 Flatten
  - 1 Dense (oculta)
  - 1 Dense de salida con softmax
- Entrénala 5 épocas y evalúala.

## 4. Clasificación de ropa con Fashion MNIST

- Usa fashion\_mnist desde keras.datasets.
- Muestra 10 imágenes aleatorias con sus nombres (ej: "sneaker", "shirt").
- Entrena una CNN básica para clasificar las prendas.

## 5. Cambia la arquitectura y compara

- Toma el modelo de CIFAR-10 y agrega una segunda capa convolucional.
- Compara la precisión antes y después del cambio.

## 6. Evalúa el modelo con una matriz de confusión

- Usa sklearn.metrics.confusion\_matrix.

- Aplica argmax a las salidas y etiquetas.
- Visualiza la matriz con `seaborn.heatmap`.

### **7. Aumenta el número de épocas**

- Usa el modelo de Fashion MNIST.
- Entrena con 20 épocas.
- Guarda en una lista el accuracy de cada época y gráficalo.

### **8. Agrega Dropout a tu modelo (Investiga qué es Dropout, explícalo con tus palabras y luego resuelve)**

- Inserta una capa `Dropout(0.3)` entre capas densas.
- Compara el sobreajuste antes y después con gráficos.

### **9. Cambia el tamaño del batch**

- Entrena el mismo modelo de CIFAR-10 con `batch_size = 16` y luego con `batch_size = 64`.
- Compara tiempo de entrenamiento y precisión.

### **10. Clasificador con aumento de datos (Data Augmentation)**

#### **Clasificación de rostros con LFW (Labeled Faces in the Wild)**

Aplicar una red convolucional básica para reconocer rostros usando el dataset lfw.

#### **Instrucciones:**

- Carga el dataset lfw desde `tensorflow.keras.datasets.lfw`.
- Redimensiona las imágenes si es necesario.
- Utiliza `ImageDataGenerator` para optimizar el aprendizaje de tu red.
- Diseña una CNN pequeña con 2 capas convolucionales cada una con su capa `Maxpooling`.
- Entrena con una fracción de clases (por ejemplo, 5 personas distintas).
- Evalúa precisión y discute limitaciones.