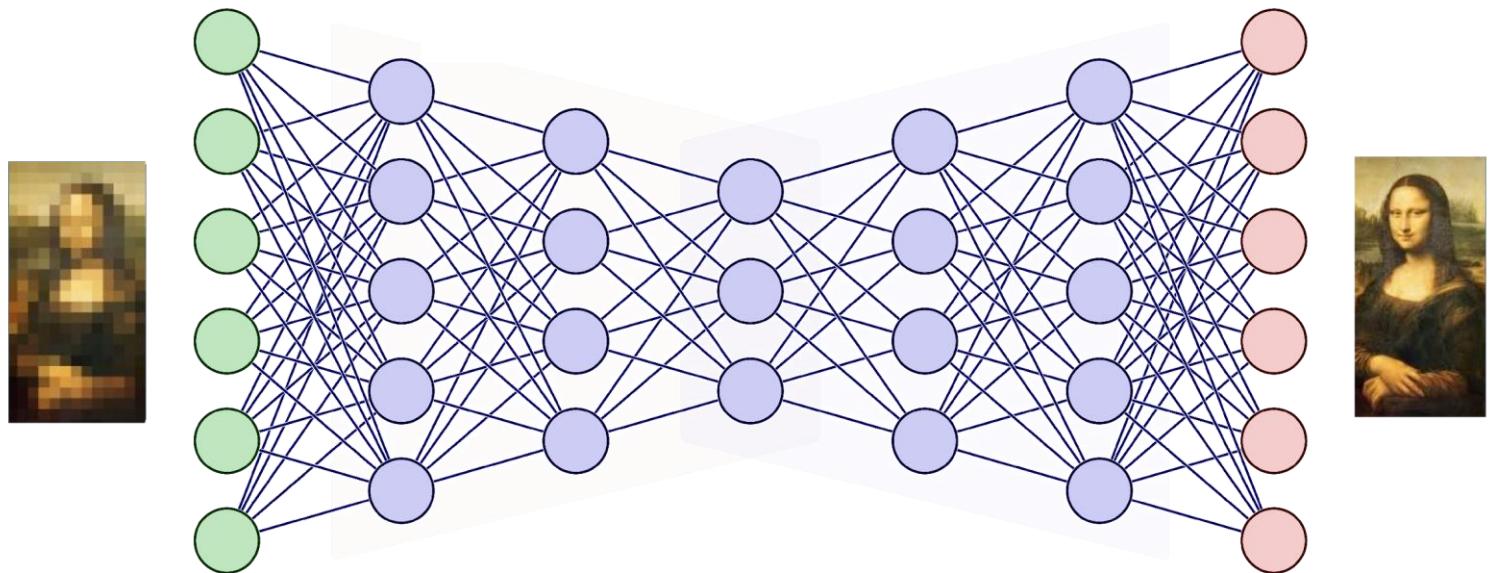


# Práctica 6

## Autoencoders



### Objetivos

El objetivo de esta práctica es comprender el funcionamiento de las redes neuronales basadas en autoencoders.

### Temas

- Redes Neuronales basadas Autoencoders y Autoencoders Variacionales. Capa UpSampling2D, Conv2DTranspose y Resizing.

### Lectura

Material de Lectura: Capítulo 6 del libro Neural Networks and Deep Learning.

### Ejercicio 1

El archivo **ecg.csv** contiene 5,000 latidos cardíacos provenientes del dataset "ECG5000", una colección de datos de electrocardiograma (ECG) de "BIDMC Congestive Heart Failure Database". Cada fila representa un ECG completo de un paciente, compuesto por 140 puntos de datos (lecturas). Cada fila contiene una etiqueta que indica si el ECG es normal (valor 1) o anormal (valor 0). La versión original del archivo se encuentra en: <https://www.kaggle.com/datasets/devavratatripathy/ecg-dataset>



Desarrolle un modelo de autoencoder utilizando redes neuronales densas con TensorFlow/Keras que posibilite diferenciar electrocardiogramas normales de anormales.

### Ejercicio 2

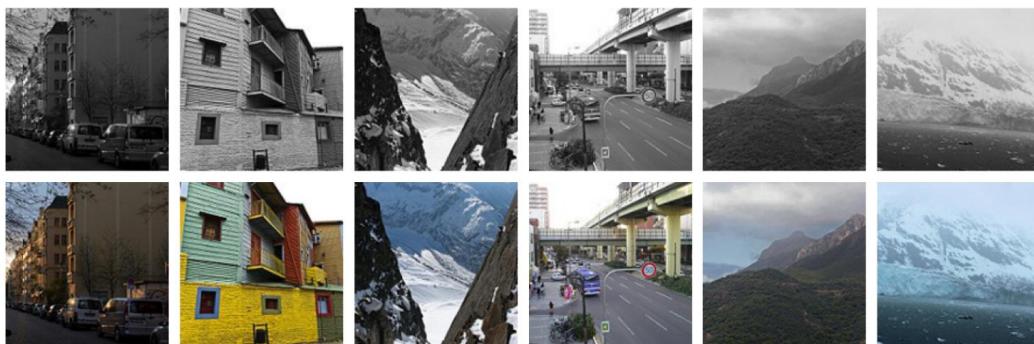
Desarrolle un modelo de autoencoder que permita detectar transacciones realizadas con tarjetas de crédito fraudulentas.

Utilice el dataset **creditcard.csv** provisto en la carpeta de datos del Moodle o descárguelo desde el enlace: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>. Este presenta transacciones que ocurrieron en dos días, donde se detectaron 492 fraudes de un total de 284,807 transacciones. Los datos están altamente desbalanceados, ya que la clase positiva (fraudes) representa menos del 0.2% de todas las transacciones.

### Ejercicio 3

El dataset **Landscape Image Colorization** consta de 7219 imágenes de calles, edificios, montañas, glaciares, y árboles, entre otros. Este contiene dos versiones de cada imagen: una en escala de grises y otra en color, almacenadas en dos carpetas separadas. La versión original de este archivo está disponible en:

<https://www.kaggle.com/datasets/theblackmamba31/landscape-image-colorization/data>



Desarrolle un modelo de autoencoder que permita generar versiones a color de imágenes en escala de grises. Utilice capas convolucionales tanto en el bloque codificador como decodificador.

Dado que en la etapa del decodificador requiere capas para aumentar la dimensión para formar la imagen final, implemente 3 versiones diferentes:

- Utilizando capas Upsampling2D.
- Utilizando capas Reescale.
- Utilizando capas Conv2DTranspose.

Luego de entrenados los distintos modelos compare los resultados de las imágenes coloreadas.

### Ejercicio 4

Replique el ejercicio anterior utilizando el dataset “**Flowers**” compuesto por imágenes de flores. El objetivo es entrenar un autoencoder que coloree imágenes en escala de grises. Obtenga una imagen en escala de grises de cada imagen a color para generar el dataset de entrada del autoencoder.



### Ejercicio 5

El dataset MNIST consiste en un conjunto de 70,000 imágenes en escala de grises de dígitos escritos a mano (0 a 9), con un tamaño de 28x28 píxeles.



Desarrolle un autoencoder variacional para realizar una representación comprimida del dataset MNIST, utilizando TensorFlow/Keras. Use 2 dimensiones para codificar el espacio latente. Luego de entrenar el modelo:

- Grafique en una cuadrícula las imágenes resultantes al variar las dimensiones de control del espacio latente.
- Utilizando 2 controles deslizantes para variar los parámetros, grafique una imagen reconstruida que se actualice automáticamente al mover cualquiera de los controles.