Python, Tensorflow y Keras

**Requisitos**

Tener una [instalación de python](http://facundoq.github.io/guides/python_install) con las librerías **keras**, **jupyter** y **pandas**. En anaconda ya vienen instalados; si usas un entorno virtual, deberás instalarlos con **pip. Nota:** si la versión de numpy es igual o mayor a 1.20, deberás instalar una versión anterior ya que tensorflow 2.x no es completamente compatible con numpy 1.20 o superior.

En particular el ejercicio 5 no funcionará correctamente sin este ajuste.

**Objetivos**

Estos ejercicios tienen como objetivo repasar y/o aprender conceptos de programación en Python con Numpy, probar el entorno de desarrollo y explorar algunos modelos de redes neuronales.

**Preparación**

Desde una terminal con python en el path, ubicarse en el directorio de la práctica donde se encuentran los archivos con extensión “.ipynb” (notebooks) y escribir: “jupyter notebook”. Desde el entorno web, acceder a la dirección del servidor (generalmente <http://localhost:8888/>) para abrir los archivos. Si lanzás jupyter notebook desde Anaconda Navigator, copiá los archivos de la práctica a la carpeta raíz del notebook.

**Ejercicio 1** Completa el tutorial inicial de Python para recordar conceptos sobre el lenguaje (opcional).

*Archivo*: Tutorial Python.ipynb

**Ejercicio 2** Completa el tutorial inicial de NumPy (opcional).

*Archivo*: Tutorial de Numpy.ipynb

**Ejercicio 3** Este notebook utiliza una red preentrenada llamada MobileNet para clasificar imágenes. Examiná el código, ejecutalo, y probá distintas imágenes y el resultado que tienen. Probá también descargar nuevas imágenes, ubicarlas en la carpeta del notebook, y cargarlas con su nombre para probar el funcionamiento de la red.

*Archivo*: Clasificador de objetos en imágenes con red MobileNet.ipynb

**Ejercicio 4** Este notebook entrena un tipo de red llamada “Red Adversaria Generativa” o *Generative Adversarial Network* (GAN), y luego la utiliza para generar imágenes.

La generación de las imágenes se hace a partir de un vector de dos números que en su combinación codifica la imagen de salida. Probá entrenar la red y observar como mejora la calidad de generación durante el proceso. Al finalizar, podés también elegir una combinación particular de dos números para generar y visualizar la imagen resultante.

Por la manera en que se entrena la red, estos números codifican imágenes de forma apropiada si se utilizan valores entre -1 y 1 aproximadamente (aunque podés probar valores más extremos).

*Archivo*: Generación de imágenes con Generative Adversarial Networks (GAN).ipynb

**Ejercicio 5**  Los modelos de redes recurrentes permiten modelar series temporales o secuenciales. En particular, el texto es una secuencia de carácteres. Este notebook entrena un modelo de generación de texto a nivel de caracteres, en base a un texto (extenso) preexistente. Luego, puede generar texto con un estilo similar al del conjunto de datos con el que fue entrenado.

*Archivo*: Generación de Texto con una Red Recurrente.ipynb