# *DEFENSA DE UNA RED NEURONAL ANTE EJEMPLOS ADVERSARIOS*

**LLampi Aliaga Elias Josue, Miranda Bailón Edmundo Manuel, Rios Yamamoto Jose Manuel Yoichi**

**Alumno 1, Facultad de Ciencias, Universidad Nacional de Ingeniería, email: elias.llampi.a@uni.pe**

**Alumno 2, Facultad de Ciencias, Universidad Nacional de Ingeniería, email: emirandab@uni.pe**

**Alumno 2, Facultad de Ciencias, Universidad Nacional de Ingeniería, email: jriosy@uni.pe**

***Palabras clave:*** *Inteligencia artificial / Redes neuronales / Ejemplos Adversarios / Modelo de aprendizaje / Python*

## 1. Objetivos

- Encontrar la falla en una red neuronal en base al ataque de un adversario.
- Mejorar una red neuronal en base a un ataque de adversario
- Construir una red neuronal que sea resistente al ataque de un adversario.

## 2. Introducción

En la actualidad, en el desarrollo continuo de métodos de desarrollo en Machine Learning, uno de sus modelos, las Redes neuronales, las cuales su desarrollo ha dado buenos frutos como, por ejemplo, su aplicación en el procesamiento de datos, el reconocimiento de voz, la robótica o su uso para los engañosos Deepfake; sin embargo, este modelo posee una vulnerabilidad y esa debilidad son los ejemplos adversarios.

Un ejemplo adversario es una entrada a la red neuronal que deriva en una salida incorrecta, en otras palabras, clasifica mal la entrada, y los ejemplos adversarios su propósito es el engañar o confundir a una red neuronal.

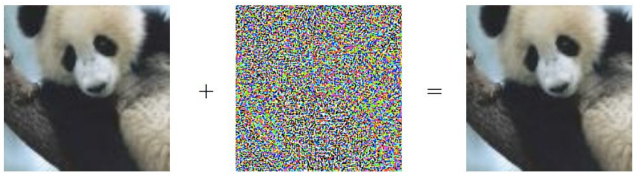


*Figura 1: Ejemplo adversario para engañar una red neuronal y tome un gibón por un panda*

Aquí podemos observar que mediante un ligero agregado de *ruido* a una imagen que la red neuronal ya reconocía, ahora obtenemos una respuesta diferente.

En los objetivos de los ejemplos adversarios están las redes neuronales aplicadas a [2]:

- Visión por computadora:clasificación de imágenes, detección de objetos.
- Procesamiento de lenguaje natural: traducción automática, generación de texto.
- Seguridad del ciberespacio: servicios en la nube, detección de malware, intrusiones en la red.
- Físico: reconocimiento de señales de trafico, cámaras de suplantación, reconocimiento de rostros.

En los ejemplos adversarios nos encontramos con 2 tipos de ejemplos, aquellos que simplemente buscan cualquier entrada para que la red neuronal se pueda confundir y se obtenga la salida con la cual se diseña el ejemplo adversario, estos son llamados ejemplos **no dirigidos**, pronto veremos que estos son en su mayoría entradas sin sentido; y el otro tipo de ejemplo, los **dirigidos**, los cuales son construidos para parecerse a una entrada determinada pero logran que la red neuronal se confunda y de como resultado una salida diferente [3].

En este trabajo analizaremos una red neuronal programada en el lenguaje Python entrenada bajo una base de datos MNIST de forma que clasifique números de 28x28 pixeles, luego diseñaremos los ejemplos adversarios para identificar las vulnerabilidades de la red neuronal [1], estudiaremos posibles defensas y de acuerdo a eso poder diseñar una red neuronal que sea resistente a ejemplos adversarios.

## 3. Librerías

Para el trabajo se harán uso de las siguientes librerías tanto para el entrenamiento de la red neuronal como de la ejecución de los adversarios.

- *network*
- *pickle*
- *numpy*
- *matplotlib*

La librería *network* nos ayudará para el diseño de la red neuronal junto a la librería *pickle* que nos permitirá la carga de una red neuronal pre-entrenada.

La librería *numpy* nos facilitará el uso de matrices para poder desarrollar los modelos matemáticos para el diseño de adversarios en forma de funciones a optimizar. Finalmente la librería *matplotlib* nos permitirá la creación de gráficas comparativas y tener un mejor enfoque de los resultados.

## Referencias


[1] Ian J. Goodfellow, J. S. . C. S. *Explaining and Harnessing Adversarial Examples*. ICLR, 2015.

[2] Shilin Qiu, Qihe Liu, S. Z. . C. W. *Review of Artificial Intelligence Adversarial Attack and Defense Technologies*. MDPI, 2019.

[3] Veerapaneni, D. G. . R. *Tricking Neural Networks: Create your own Adversarial Examples*. Machine learning at Berkeley, 2018.

**Índice de figuras**