

## Laboratorio – GANs

CC3092 – Deep Learning y Sistemas Inteligentes En grupos de

3 personas:

### Práctica:

1. Utilice el código compartido en la actividad para **crear su propia GAN**.
2. Su objetivo es **crear un sistema de redes más liviano**  
(Generador/Discriminador). No importa si la red pierde precisión.
3. **Describa los cambios** que haya realizado en el generador y el discriminador.
  - Explique **por qué** estos cambios hacen su red más eficiente en términos computacionales.

Nuestras primeras dos redes modificadas sí tardan menos tiempo. Esto es porque en ambas, estamos eliminando filtros de uno de nuestros dos componentes. Esto implica que reducimos no solo el número de scans y mapeos que se le hacen a cada imagen, sino que también reducimos las convoluciones de la red. Y además de todo esto, reducimos el uso de memoria.

Por otro lado, nuestras segundas modificaciones estaban ampliando el tiempo. Consideramos que lo que pasa es que le está costando significativamente más converger, por lo que está tardando más tiempo

4. **Muestre los resultados** de su GAN cada 10 epochs.
5. **Describa** si la red parece aprender a generar imágenes de números o no.

Algunas. Veamos que 2 de nuestras redes, parecen aprender a hacer imágenes más claras, pero no necesariamente números. Sin embargo, nuestra primera modificación, y la última sí parece estar aprendiendo a hacer números. Veamos que inclusive hay un 2 en el resultado, un 9, un 7 y un 1. Esto nos indica que al final, es más importante tener un buen discriminador que un buen generador.

### Teoría:

1. **Modifique las funciones de pérdida** del generador y del discriminador para monitorear su avance durante el entrenamiento. (Grafíquelos)

- ¿Qué comportamiento **se espera** de las funciones de pérdida durante el entrenamiento?  
Bastante estable y que se vaya reduciendo con el tiempo
- ¿Qué comportamiento **observa** en la práctica?  
Oscilante y muchas veces se mantiene estable durante toda la corrida, es decir, no baja ni sube la función de pérdida

2. Tenga en cuenta la función de pérdida de una GAN y **disecte sus elementos**.

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$$

$\min_G \max_D$  Representa al Discriminador y al Generador. Veamos que el Discriminador intenta maximizar la función para distinguir los elementos reales de los generados, mientras el generador la intenta minimizar para confundir al discriminador.

$\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})]$  Este término representa el valor esperado de la distribución de los datos.

$\mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))]$ . Este término simplemente representa el ruido que puede tener la distribución

3. Explique con sus palabras, qué es un **problema de minimax** y por qué la función de pérdida de una GAN se considera uno.

Un problema Min Max es cuando tenemos dos “unidades” compitiendo por cosas opuestas, una quiere maximizar la función, y otra la quiere minimizar la función. En este caso es muy claro el por qué es un problema min max, como se explicó en la disección de la función. El generador quiere minimizar la función y el discriminador maximizar la función