

Universidad del Valle De Guatemala

Facultad de Ingeniería

Deep Learning

Javier Fong



Laboratorio 5: Generative Adversarial Network

Javier Mombiela 20067

Roberto Ríos 20979

Guatemala, 29 de septiembre 2023

Estructura G(x)

El generador toma un vector de ruido aleatorio como entrada y genera una imagen que se asemeja a las imágenes del conjunto de datos MNIST. La estructura del generador utilizada para este laboratorio fue la siguiente:

- Input Layer: La entrada es un vector de ruido de dimensión 100.
- Capa Densa (Fully Connected): Una capa densa con 256 neuronas y activación LeakyReLU.
- Capa Densa: Otra capa densa con 512 neuronas y activación LeakyReLU.
- Capa Densa: Una capa densa con 1024 neuronas y activación LeakyReLU.
- Capa de Salida: La capa de salida es una capa densa con 784 neuronas (28x28 píxeles) y utiliza la activación 'tanh'. Esto escala los valores generados a un rango entre -1 y 1, que es el rango de píxeles válido para imágenes en escala de grises.

El generador crea imágenes a partir del ruido aleatorio proporcionado como entrada, y la activación 'tanh' asegura que los píxeles generados estén en el rango correcto.

Estructura D(x)

El discriminador es una red neuronal que tiene como objetivo distinguir entre imágenes reales (del conjunto de datos MNIST) y falsas (generadas por el generador). La estructura del discriminador utilizado para este laboratorio fue el siguiente:

- Input Layer: La entrada de la red es una imagen de 28x28 píxeles en escala de grises.
- Flatten Layer: La primera capa aplanar la imagen en un vector unidimensional para que pueda ser procesado por capas densas.
- Dropout Layer: Se usa una capa de dropout para regularizar la red y evitar el sobreajuste.
- Capa Densa (Fully Connected): Una capa densa con 1024 neuronas y activación LeakyReLU. Se utilizó LeakyReLU, ya que permite un pequeño gradiente negativo, lo que puede ayudar en el entrenamiento.
- Dropout Layer: Otra capa de dropout para regularización.
- Capa Densa: Una capa densa con 512 neuronas y activación LeakyReLU.
- Dropout Layer: Capa de dropout adicional.
- Capa Densa: Otra capa densa con 512 neuronas y activación LeakyReLU.
- Capa de Salida: Una capa densa con 1 neurona y activación sigmoide. Esta neurona de salida produce una probabilidad que indica si la imagen de entrada es real o falsa.

Este discriminador tiene una estructura profunda con capas densas intercaladas con capas de dropout para evitar el sobreajuste. La activación LeakyReLU se utiliza para permitir gradientes negativos. La capa de salida utiliza la activación sigmoide para producir una probabilidad de clasificación binaria, ya que solo estamos lidiando con true y false.

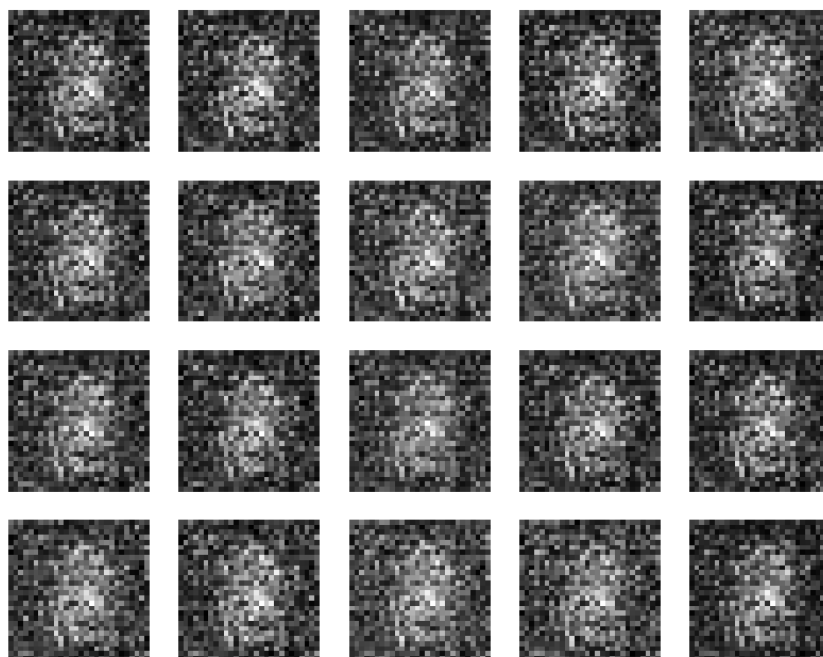
Pruebas del entrenamiento

Abajo, se muestran 5 imágenes del proceso de entrenamiento, en donde se pueden observar imágenes después de 1, 25, 50, 75 y 100 épocas completadas, respectivamente. Como se puede observar en las imágenes, el modelo definido por las estructuras mencionadas anteriormente, sí es exitoso, ya que al completar todo el entrenamiento, se logra ver una gran diferencia clara a la primera imagen.

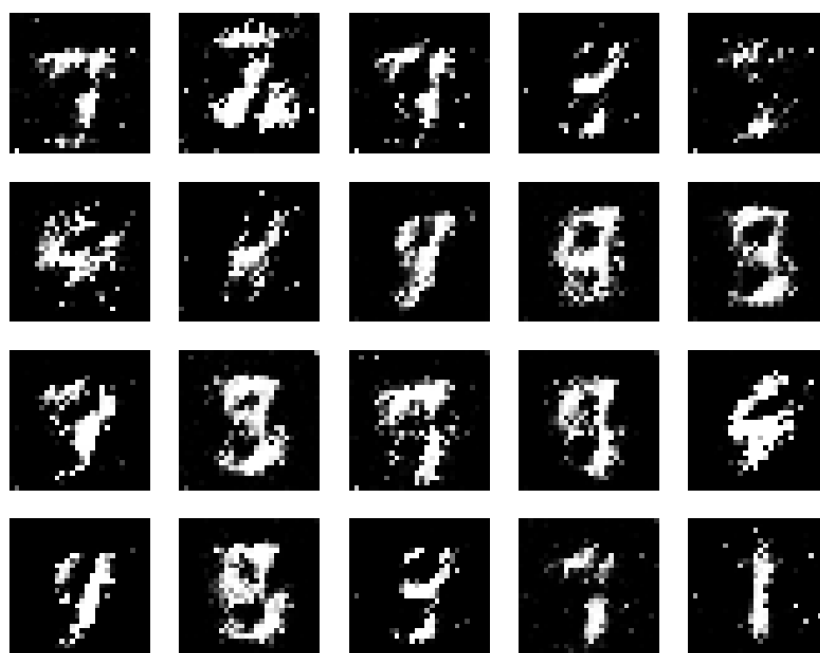
En la primera época, las imágenes son ruido aleatorio, ya que el generador no ha aprendido patrones significativos. Tras 25 épocas, las imágenes comienzan a mostrar formas vagamente similares a los dígitos MNIST, aunque carecen de detalles. Con 50 épocas, la calidad mejora notablemente, con formas más definidas pero posibles imperfecciones. Luego, después de 75 épocas, las imágenes se asemejan más a los dígitos, y se observa un cambio significativo, pero aún no son perfectas. Finalmente, después de 100 épocas, se observa una pequeña mejora, aunque no es tanto la diferencia como la que hubo entre las últimas dos imágenes.

Esto significa que al completar las 100 épocas, el modelo fue exitoso, ya que este sí cumplió con la tarea de poder definir las imágenes, aunque no lo hizo de una manera perfecta. Esto nos lleva a pensar, que se pudo haber agregado unas cuantas épocas más al entrenamiento, para que el resultado fuera aún mejor. Pero esto no se hizo por dos razones. La primera: el entrenamiento de las 100 épocas duro varias horas, y aumentar más épocas significaba un tiempo de espera mayor y no tanta diferencia. La segunda razón, y como se mencionó en la primera, esto no tendría tanta diferencia, ya que entre la época 75 y 100, no se vio ningún cambio tan significativo, lo que significa que haber agregado más épocas no aseguraba un resultado perfecto, y es más, esto podría también causar un sobre ajuste.

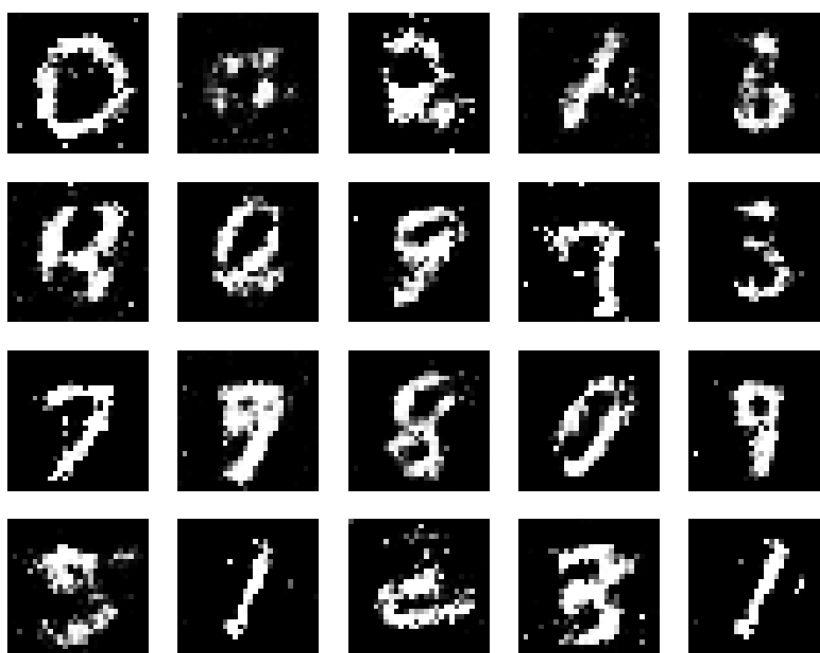
Entrenamiento con 1 época completada



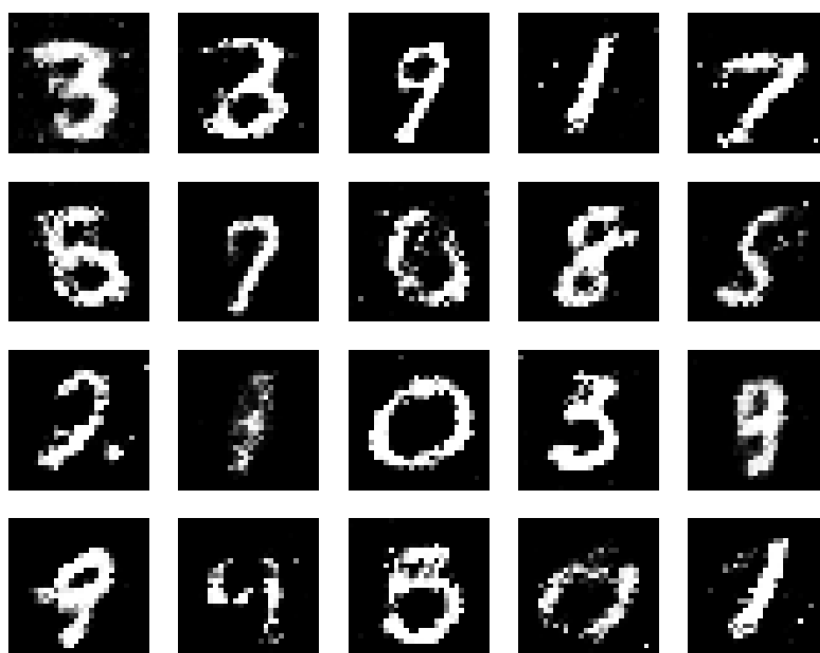
Entrenamiento con 25 épocas completadas



Entrenamiento con 50 épocas completadas



Entrenamiento con 75 épocas completadas



Entrenamiento con 100 épocas completadas

