

# Inteligencia artificial

Alejandro López Gómez

Emilio Macías Do Santos

Cristina Del Pilar Mallaupoma Cáceres

3º del GIC

# Mejora de imágenes (Parte 1):



Imagen con ruido

Eliminación de ruido  
usando redes GAN



Imagen esperada

# Mejora de imágenes (Parte 2):

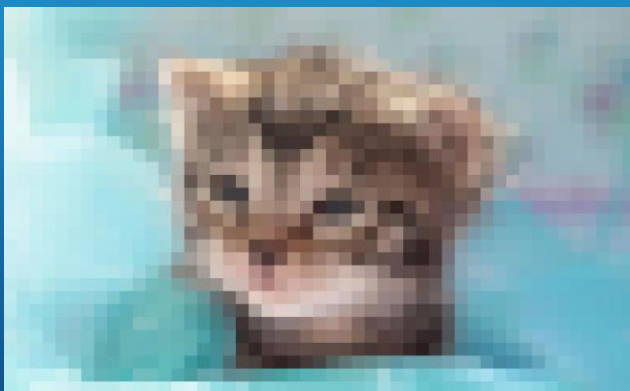


Imagen de peor  
calidad



Imagen a menor  
resolución/escala

Mejora de calidad usando  
redes GAN

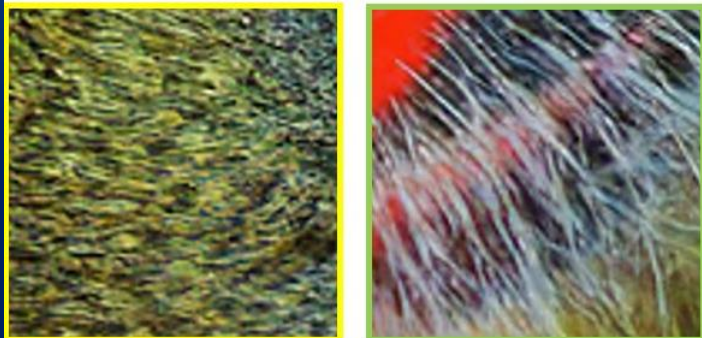
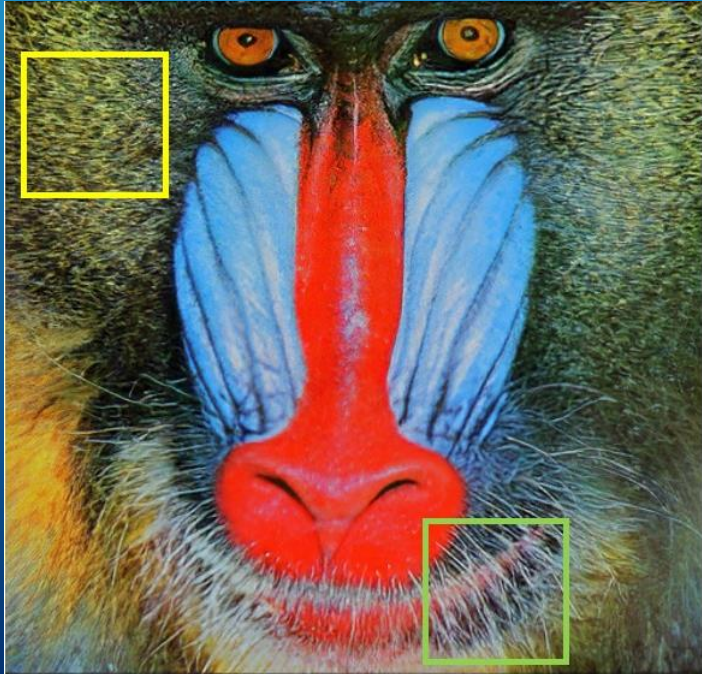
Mejora de escalado  
usando redes GAN



Imagen con  
mayor calidad



# Aplicación ESRGAN:

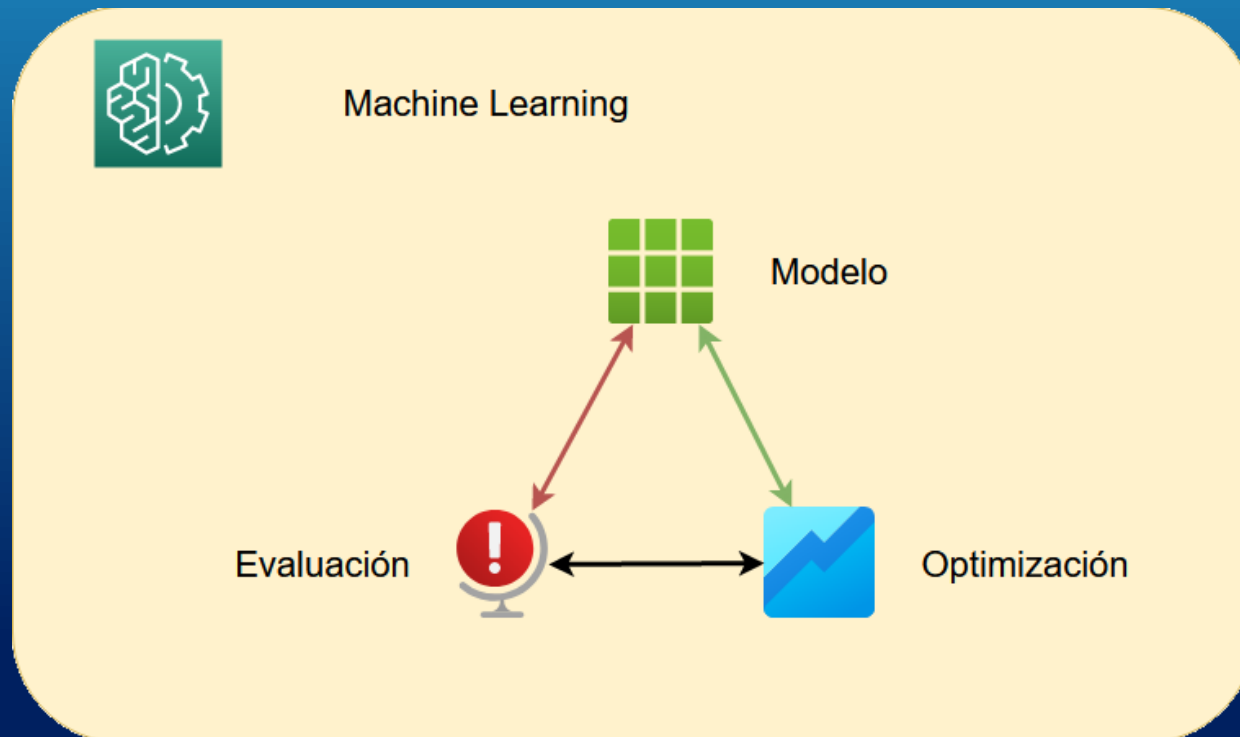


ESRGAN



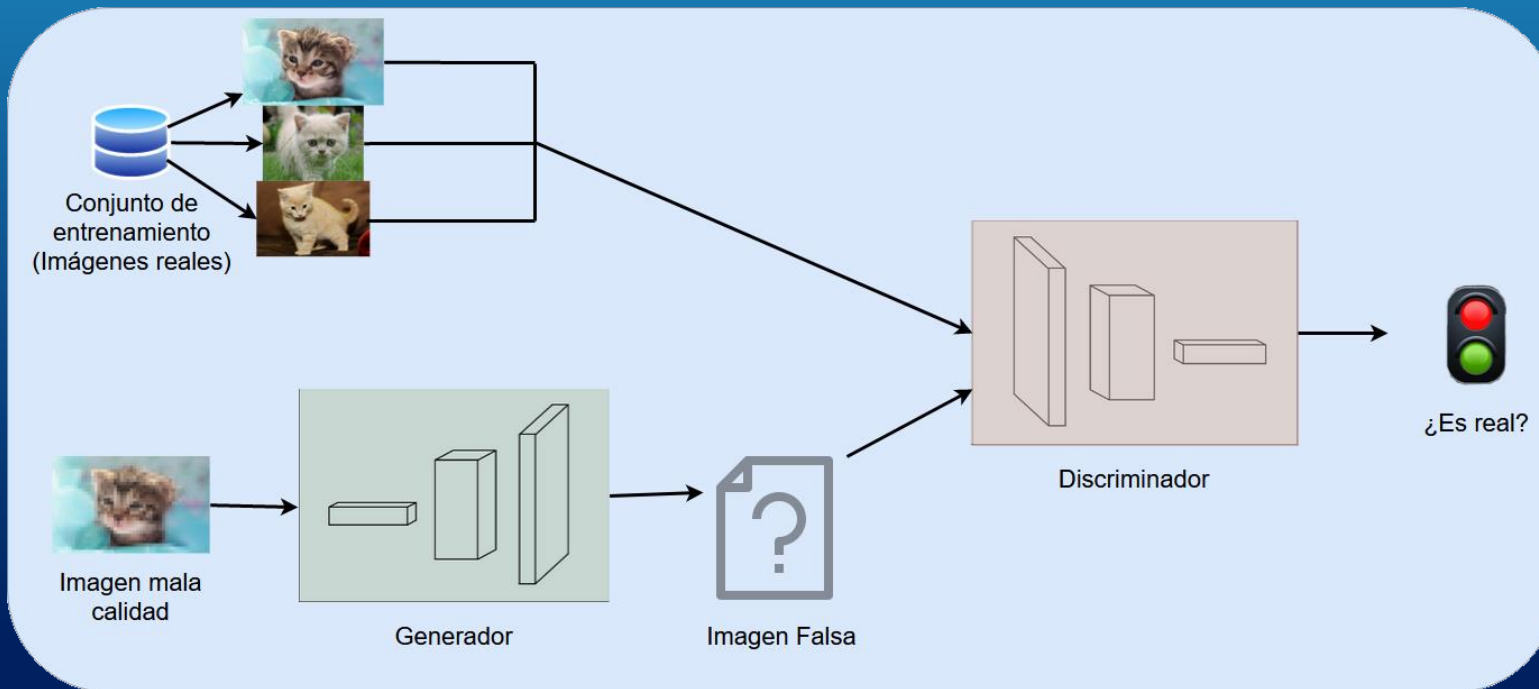
Utilizaremos la aplicación ESRGAN para el tratamiento de estas imágenes.

# Estructura de la presentación:



1. Representación (Modelos usados).
2. Evaluación (Sobreajuste, infraajuste, etc).
3. Optimización (Funciones de pérdida usadas, etc).

# Explicación del modelo GAN (Parte 1):



Se componen de dos elementos:

- El discriminador detecta si una imagen es real o falsa.
- El generador produce imágenes lo más fieles a la realidad que sea posible.

# Explicación del modelo GAN (Parte 2):



Imagen

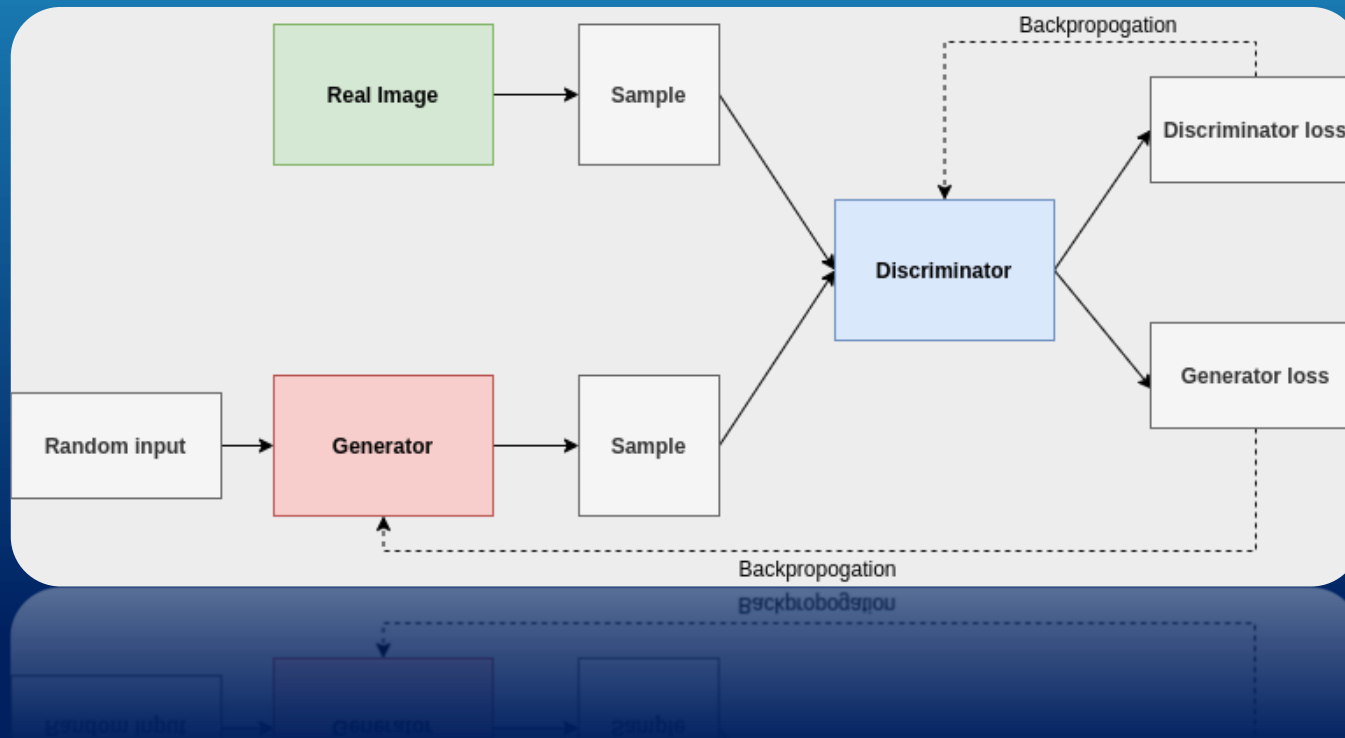
Red GAN

No

¿Es falsa?

- Se encargará de sintetizar los detalles de la imagen con la información que le da la red convolucional.
- En el Generador y Discriminador se utilizarán redes CNN para interpretar la imagen.

# Modelo GAN vs ESRGAN:

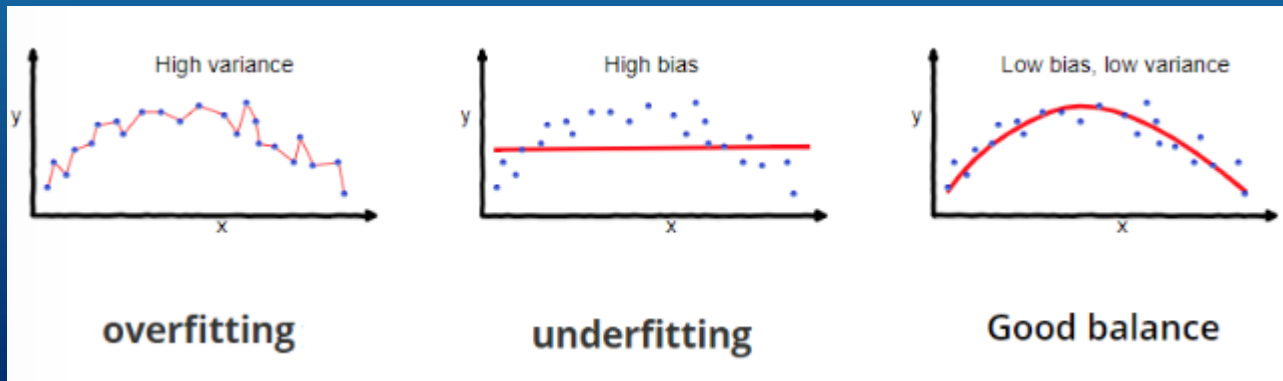


El modelo ESRGAN se basa en GAN, pero añade los siguientes elementos:

- Una técnica de pérdida de perfección perceptual (Perceptual Loss).
- Una técnica de regularización de estilo (Style-based regularization).



# Evaluación – Problemas Generales (Parte 1):



Ejemplo de sobreajuste de datos.

Problema:

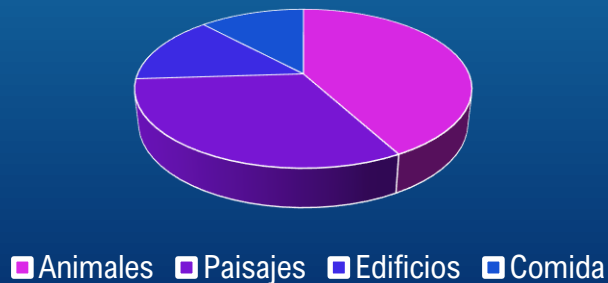
- Sobreajuste del modelo

Soluciones:

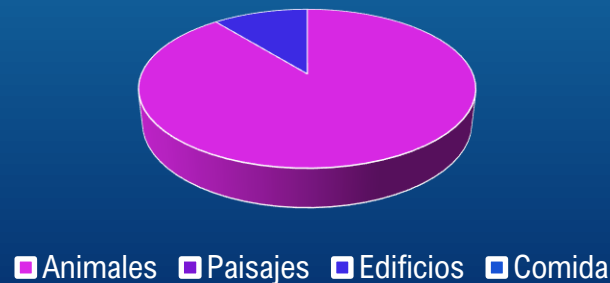
- Aumentos de datos.
- Regularización.

# Evaluación – Problemas Generales (Parte 2):

Datos balanceados



Datos desbalanceados



Problema:

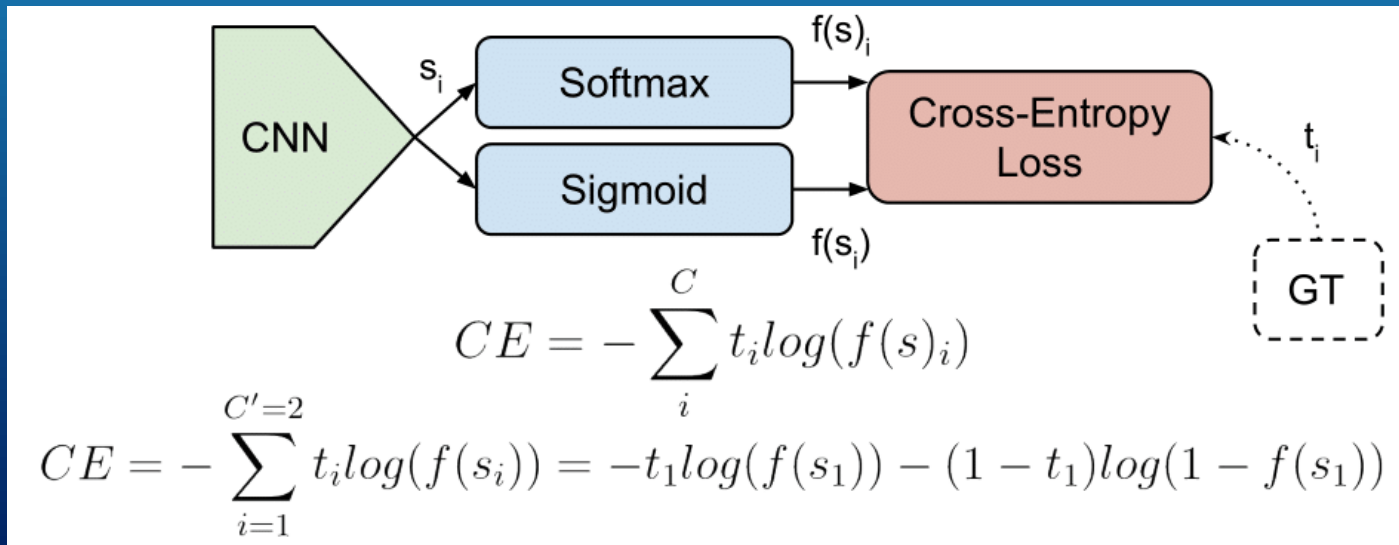
- Desbalanceo de datos.

Soluciones:

- Reducir el número de ejemplos de la clase predominante.
- Utilizar otra función de pérdida adecuada.

Ejemplo de desbalanceo de datos.

# Evaluación – Problemas Generales (Parte 3):



Ejemplo de entropía cruzada binaria

Funciones de pérdida.

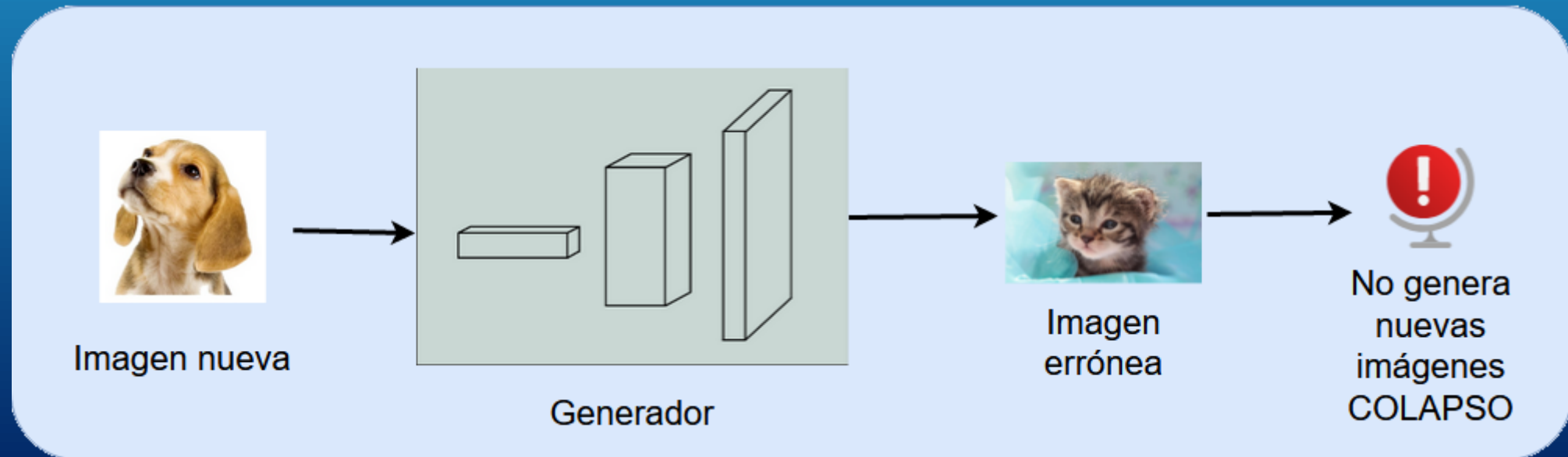
- Entropía cruzada binaria (binary cross-entropy).
- Una para el generador y otra para el discriminador.

Solución propuesta:

- Usar la función Wasserstein Loss (Lo utiliza ESRGAN).

# Evaluación – Problemas GAN (Parte 1):

Colapso de modo:



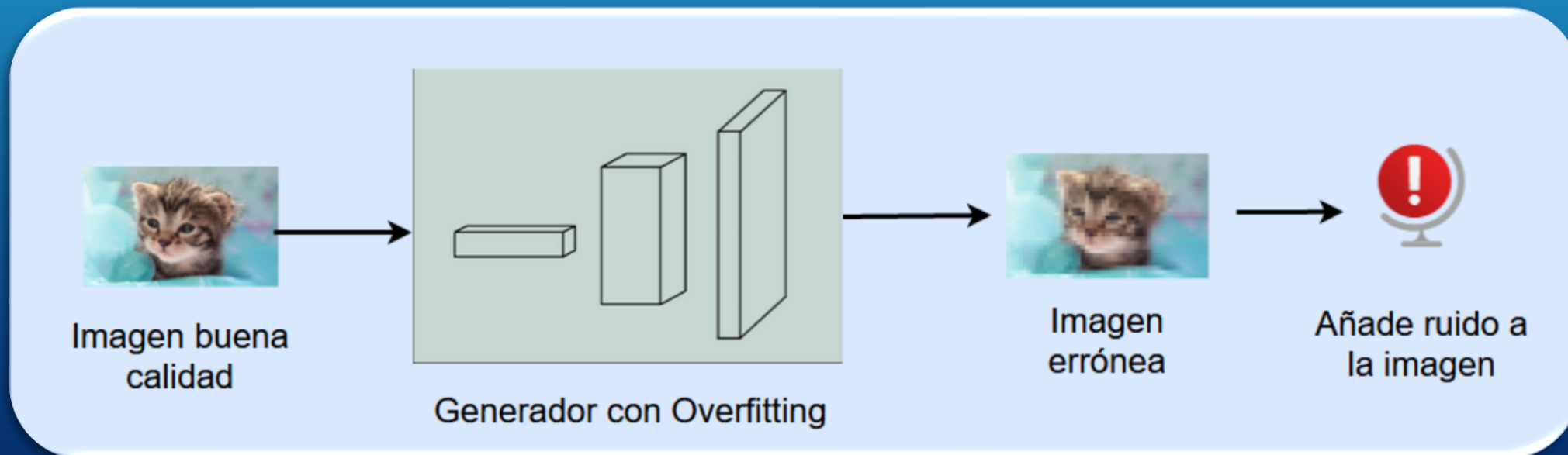
El generador colapsa, produciendo muestras limitadas, solo puede generar cosas que están dentro del set de entrenamiento.

Solución:

- Utilizar un discriminador más robusto y un aumento de datos.

# Evaluación – Problemas GAN (Parte 2):

Valores de pérdida relativos:



Se ha realizado un mal entrenamiento del modelo, por lo que generará imágenes erróneas añadiendo ruido.

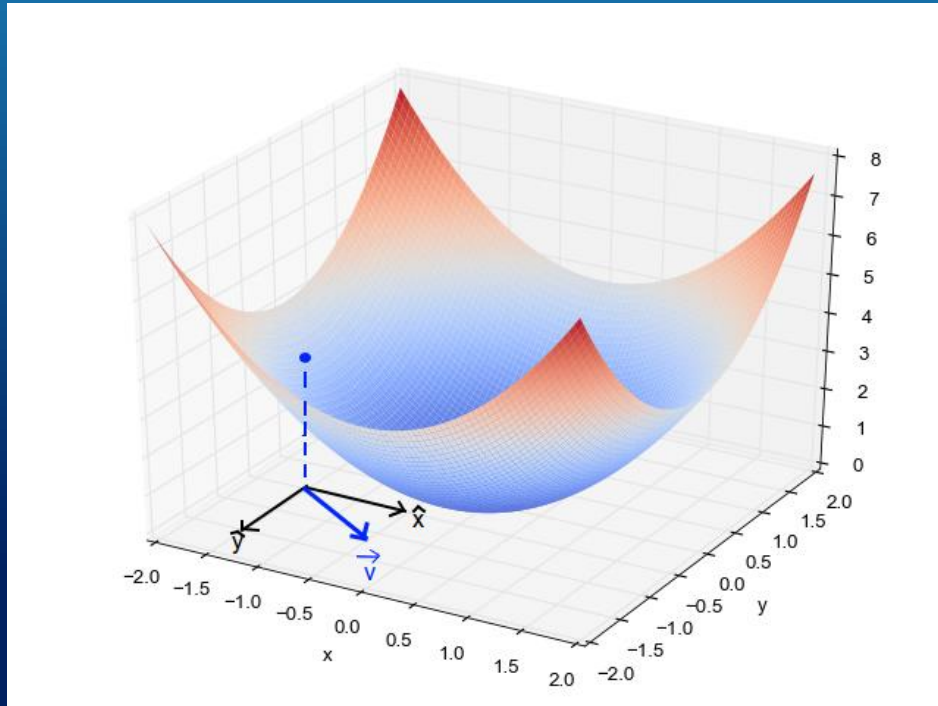
Solución:

- Aumentar el tamaño de la muestra de entrenamiento y utilizar técnicas de early stopping.



# Evaluación – Problemas GAN (Parte 3):

Gradiente disminuido:



El cambio en la función de pérdida entre las épocas de entrenamiento se está volviendo cada vez más pequeño.

- Empeora el aprendizaje de una de las redes.

Soluciones:

- Cambiar el algoritmo de optimización utilizado.
- Modificar las capas del modelo o cambiarlo.

# Evaluación - Métricas:



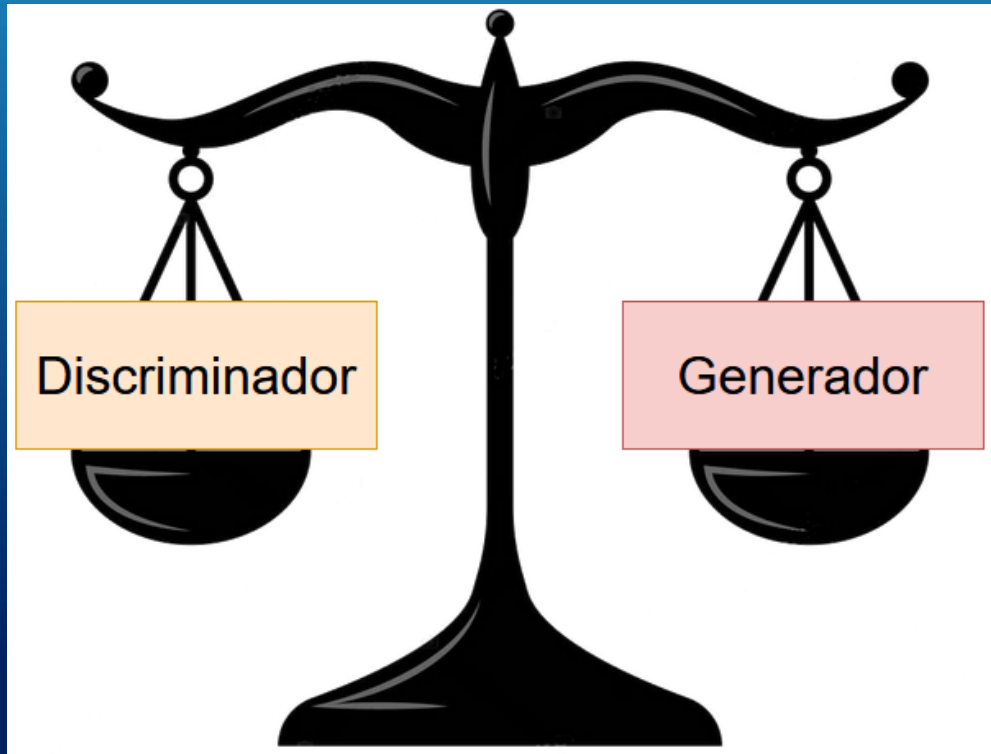
Medidas usadas para calcular la precisión:

- PSNR
- MSE

Medidas usadas para calcular el recall:

- Entropía

# Optimización (Parte 1):



Tenemos los siguientes problemas a tratar:

- Balance de la pérdida.
- Estabilidad en el entrenamiento.
- Diseño de la arquitectura.
- Elección del algoritmo de optimización.

# Optimización (Parte 2):

## Algoritmos posibles:

### Algoritmo ADAM:

#### Ventajas:

- Combina el gradiente descendiente estocástico con el movimiento de momento.
- Mejor capacidad de manejo de problemas a escala diferente.

#### Inconvenientes:

- Sensible a la elección de los parámetros iniciales.
- Sensible a las condiciones de ruido en los datos.

### Algoritmo RMSprop:

#### Ventajas:

- Tasa de aprendizaje adaptativa para cada parámetro. -> Evitar problemas de estabilidad y convergencia.
- Memoria a corto plazo de los gradientes. -> Evitar problemas de oscilación.

#### Inconvenientes:

- No tiene una memoria a largo plazo de los gradientes como Adam.

# Mejoras ESRGAN:

## Mejoras generales:

- Función de pérdida más avanzada, Relativistic Average GAN (RaGAN).
- Utilizar una arquitectura de red más compleja, con más capas y más parámetros.

## Mejoras del proceso de entrenamiento:

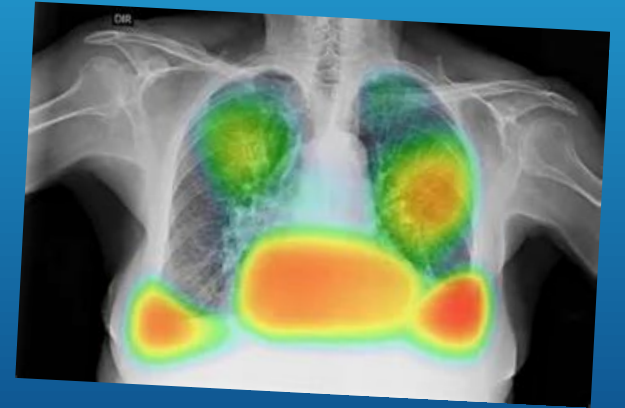
- Utilizar imágenes de alta resolución y baja resolución.
- Utilizar "Path Length Regularization" para regularizar la complejidad de las funciones del generador.



# Conclusiones:

Puntos de aprendizaje:

- Hemos fortalecidos y ampliado los conocimientos vistos en teoría.
- Diferentes aplicaciones de un modelo GAN.



# Fin

Alejandro López Gómez

Emilio Macías Do Santos

Cristina Del Pilar Mallaupoma Cáceres