

# Tema 7.1

## Modelos generativos

Miguel Ángel Martínez del Amor

Deep Learning

Departamento Ciencias de la Computación e Inteligencia Artificial

Universidad de Sevilla

# Contenido

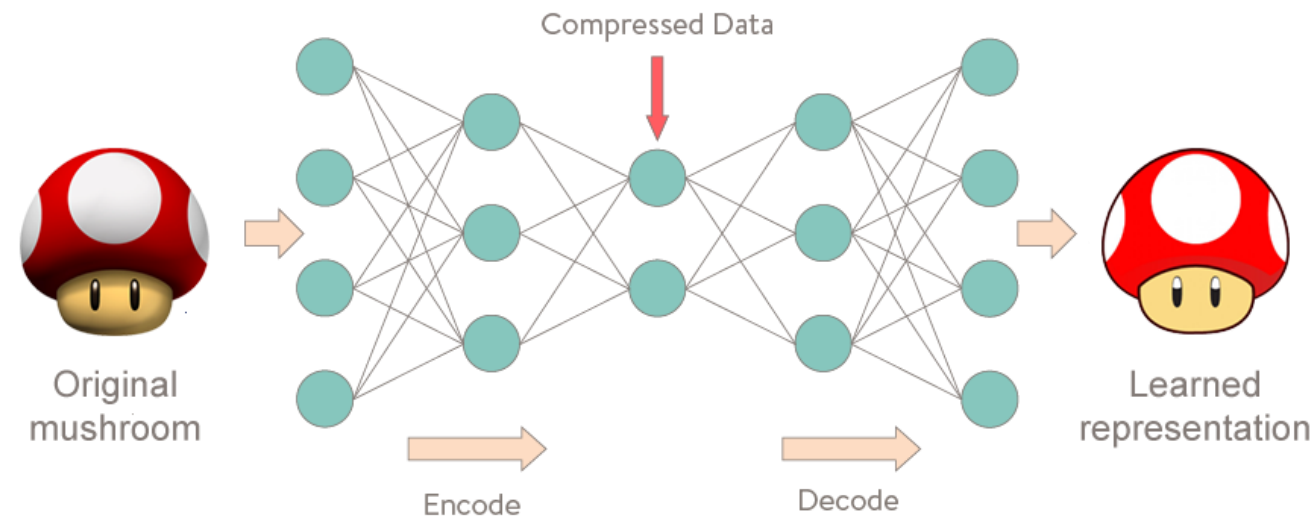
- Aprendizaje no supervisado
- Autoencoders
- Variational Autoencoders (VAE)
- Generative Adversarial Networks (GAN)

# Aprendizaje no supervisado

- Hasta ahora nos hemos enfocado en Deep Learning con aprendizaje supervisado.
- Aprendizaje **supervisado**:
  - Datos: los ejemplos son del tipo  $(X,Y)$ , hay etiquetas!
  - Objetivo: aprender una función que mapee  $X$  a  $Y$ .
- Aprendizaje **no supervisado**:
  - Datos: los ejemplos son del tipo  $(X)$ . Más fácil de conseguir!
  - Objetivo: descubrir alguna estructura en los datos.
- Veremos a continuación ejemplos de no supervisado.

# Autoencoders

- Analicemos qué pasa si buscamos que la salida de una red sea exacta a la entrada (sobre un conjunto de datos limitado).
- Y ahora además pongamos varias capas en medio, con distintos tamaños: aprender la función identidad
- Si se consigue calcular la salida correctamente:
  - ¿qué ha pasado en medio de la red?
  - ¿qué información dan los valores calculados en las capas intermedias?



# Autoencoders

- **Encoder:**

- CNN con ReLU
- Reducción de dimensionalidad:  $Z$  es menor que  $X$ .
- $Z$  = **vector/representación latente**

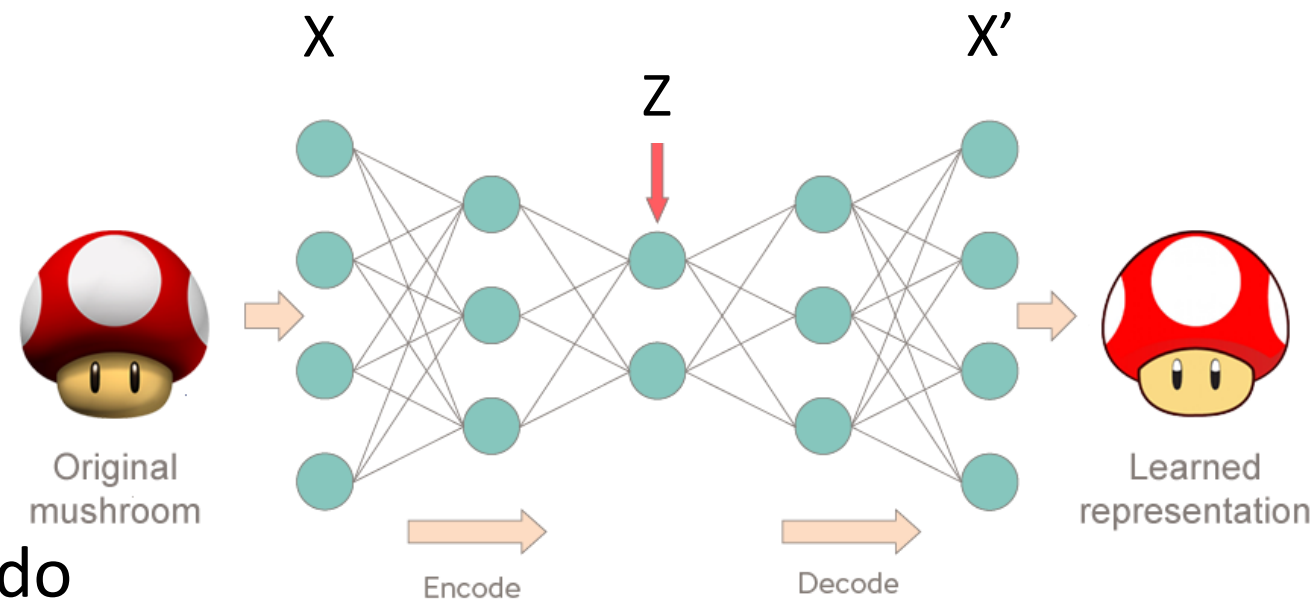
- **Decoder:**

- **CNN transpuesta** con ReLU
- Obtiene  $X'$

- Función de pérdida:

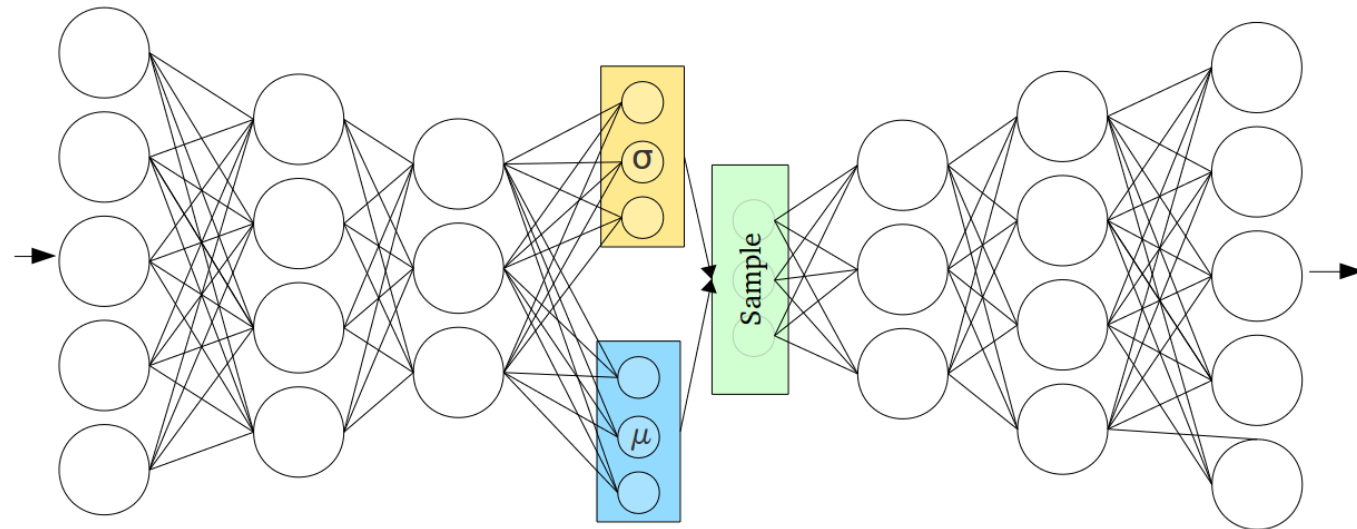
- L2-norm:  $\|X - X'\|^2$

- Uso: inicializar modelo supervisado



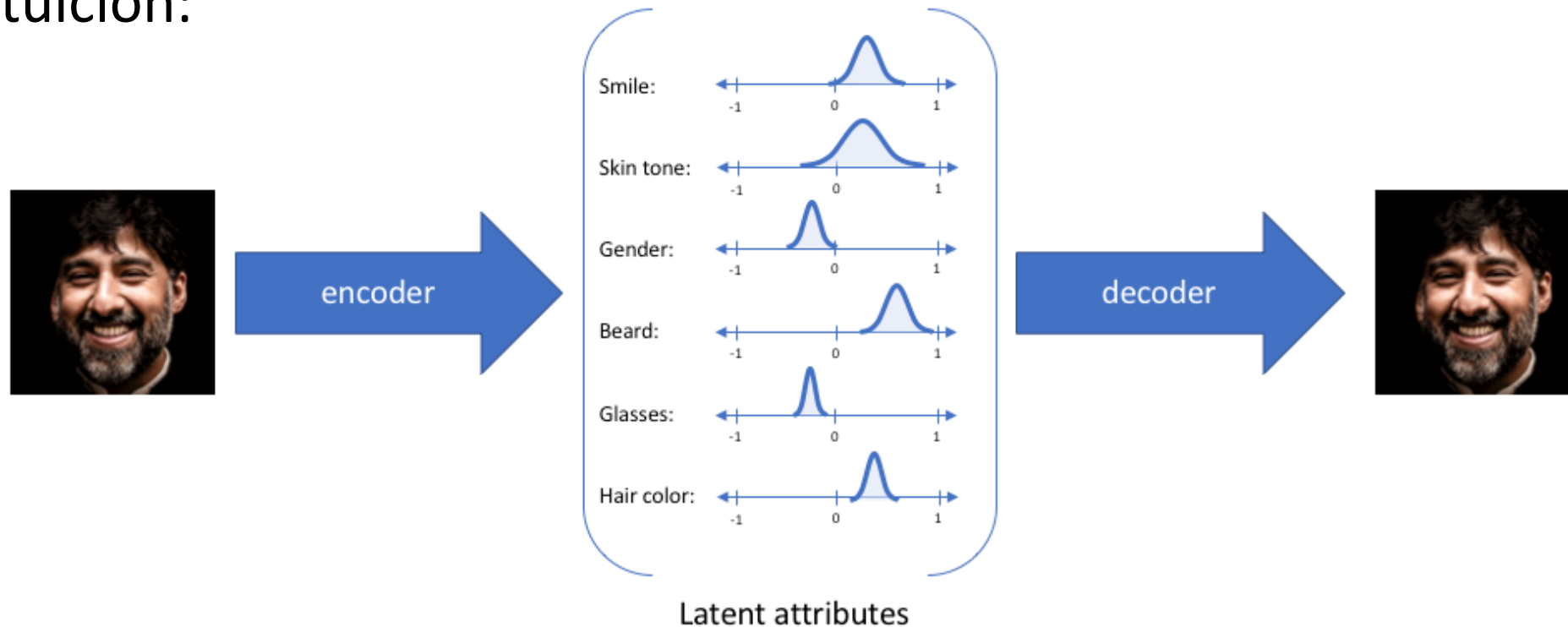
# Variational Autoencoders

- En autoencoders, si muestreamos el espacio representado en Z conseguimos una salida!
  - Problema: Z no es continuo, y reconstruiríamos lo mismo que lo aprendido
- **VAE**: el espacio latente es continuo, y se muestrea aleatoriamente
- El espacio latente son dos vectores: [[Kingma et al 2013](#)] [[Rezende et al 2014](#)]
  - $\mu$ : media de distribución
  - $\sigma$ : desviación estándar
- A continuación se muestrea aleatoriamente (*Sample*) de una normal  $N(\mu, \sigma^2)$



# Variational Autoencoders

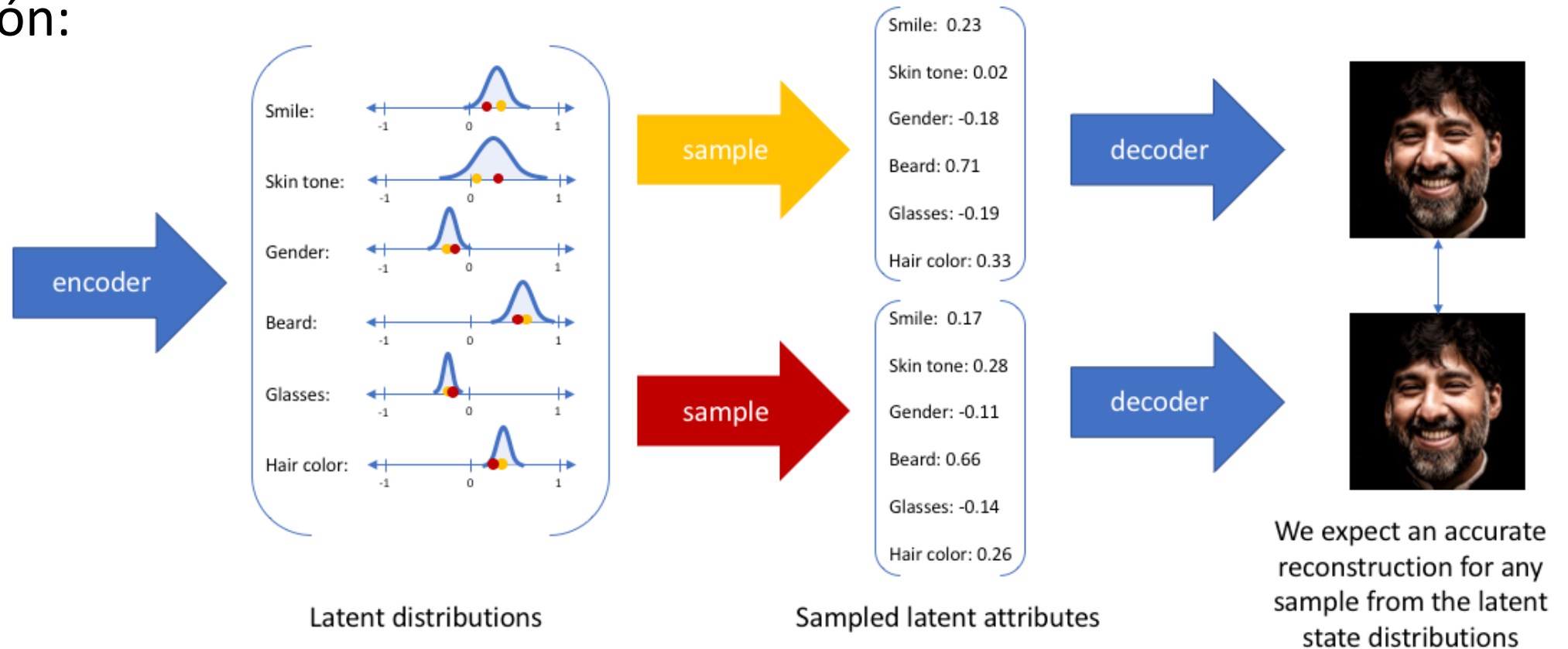
- Intuición:



<https://www.jeremyjordan.me/variational-autoencoders/>

# Variational Autoencoders

- Intuición:



<https://www.jeremyjordan.me/variational-autoencoders/>



# Variational Autoencoders

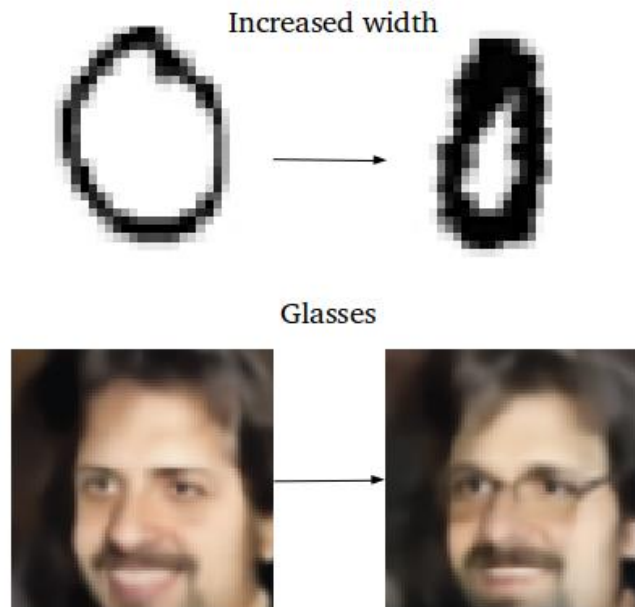
- Ejemplos:



Espacio continuo de caras con VAEs (por T. White)

# Variational Autoencoders

- Ejemplos (variando atributo específico):



Variando el vector de sonrisa

# Generative Adversarial Networks

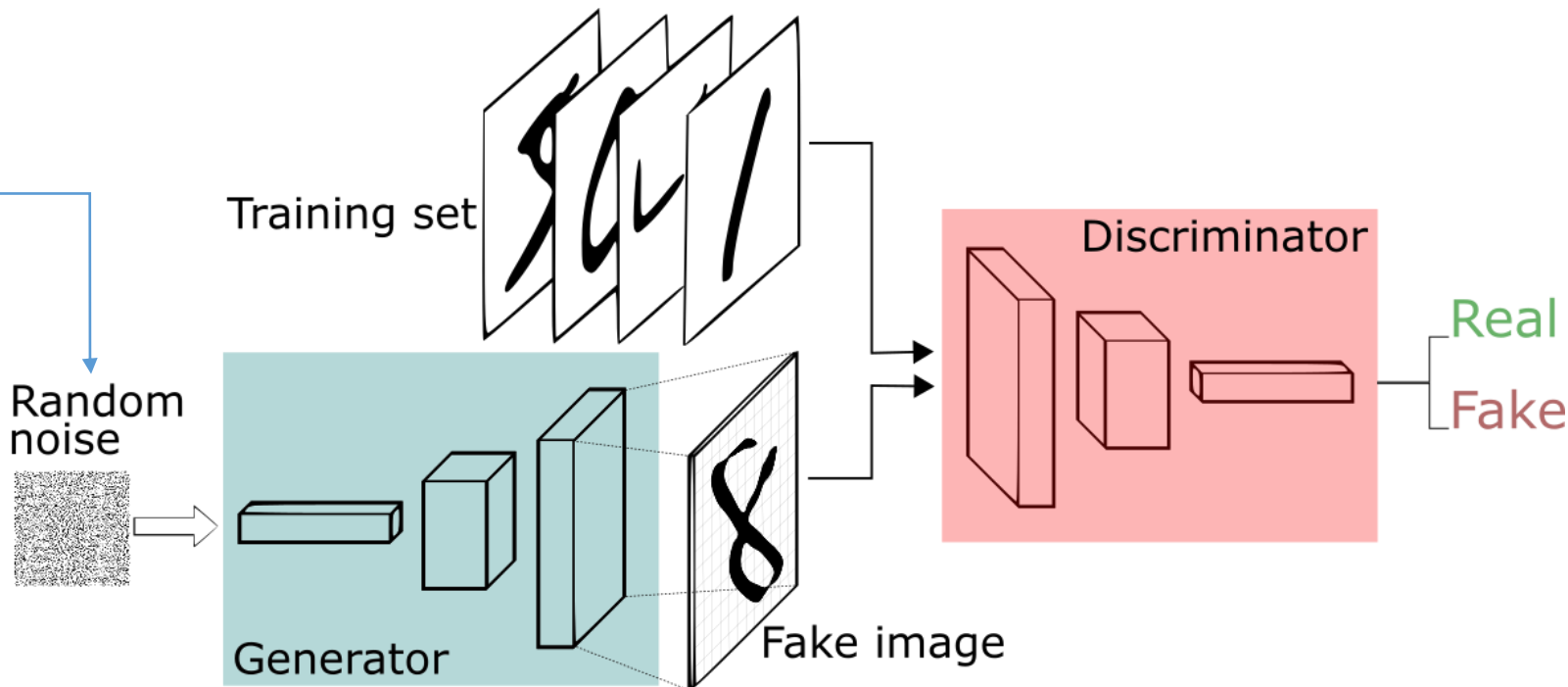
- Enfrentar dos redes:

## Generative Adversarial Networks (GANs)



# Generative Adversarial Networks

- Arquitectura: [[Goodfellow 2014](#)]
  - **Generador:** Genera ejemplos muestreando aleatoriamente vector latente
  - **Discriminador (adversaria):** determina si los ejemplos (mezclando del training set y del generador) recibidos son falsos o reales



# Generative Adversarial Networks

- El generador se adapta a generar imágenes más realistas cada vez porque el discriminador también se hace más bueno encontrándolas
- Problema: el mínimo óptimo no está garantizado
  - En cada paso del descenso por gradiente, el paisaje de montaña cambia un poco
  - Es un sistema dinámico que busca el equilibrio, no el mínimo
  - Muy difíciles de entrenar (tanto que el mismo autor sacó otro artículo demostrando que no funcionaba en la práctica)

# Generative Adversarial Networks

- Guía para Deep Convolutional GANs estables [\[Radford et al 2016\]](#)
  - Reemplazar capas de pooling for strided convolutions (en discriminador) y fractional-strided convolutions (en generador)
  - Usar BatchNorm en discriminador y generador
  - Eliminar capas ocultas totalmente conectadas (FC) para más profundidad
  - En generador, usar activación ReLU para todas las capas menos para la última, que usa tanh
  - En el discriminador, usar LeakyReLU para todas las capas.



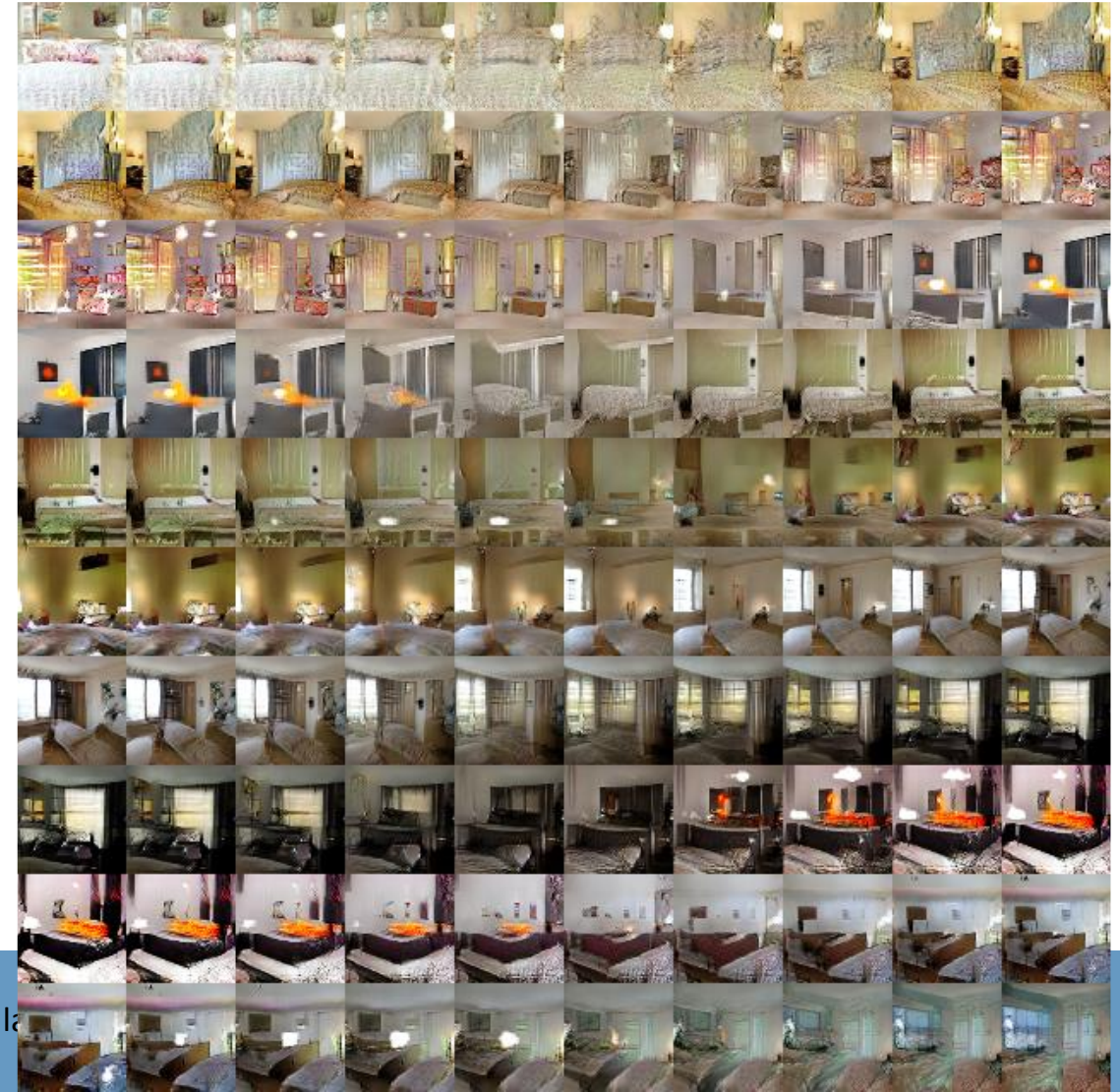
# Generative Adversarial Networks

- Ejemplos: [[Goodfellow 2014](#)]



# Generative Adversarial Networks

- Ejemplos: [Radford et al 2016]





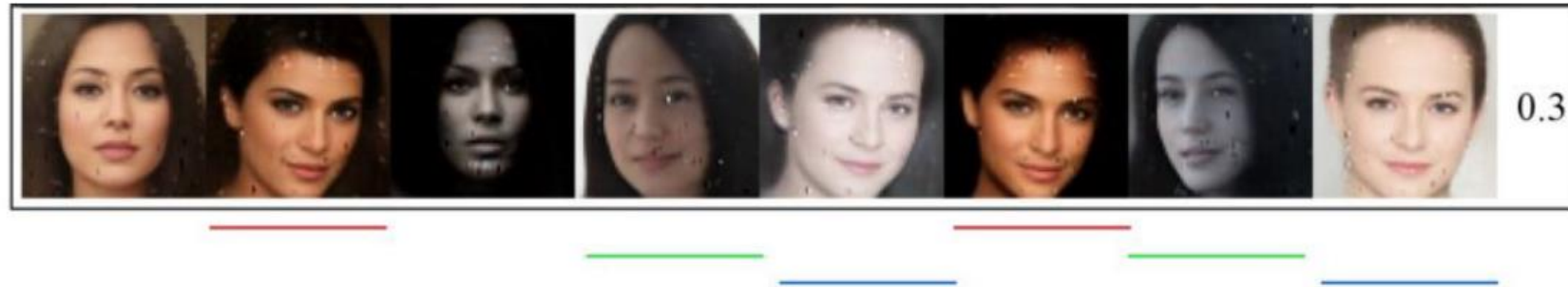
# Generative Adversarial Networks

- Ejemplos de aritmética de vectores para conceptos visuales: [Radford et al 2016]



# Generative Adversarial Networks

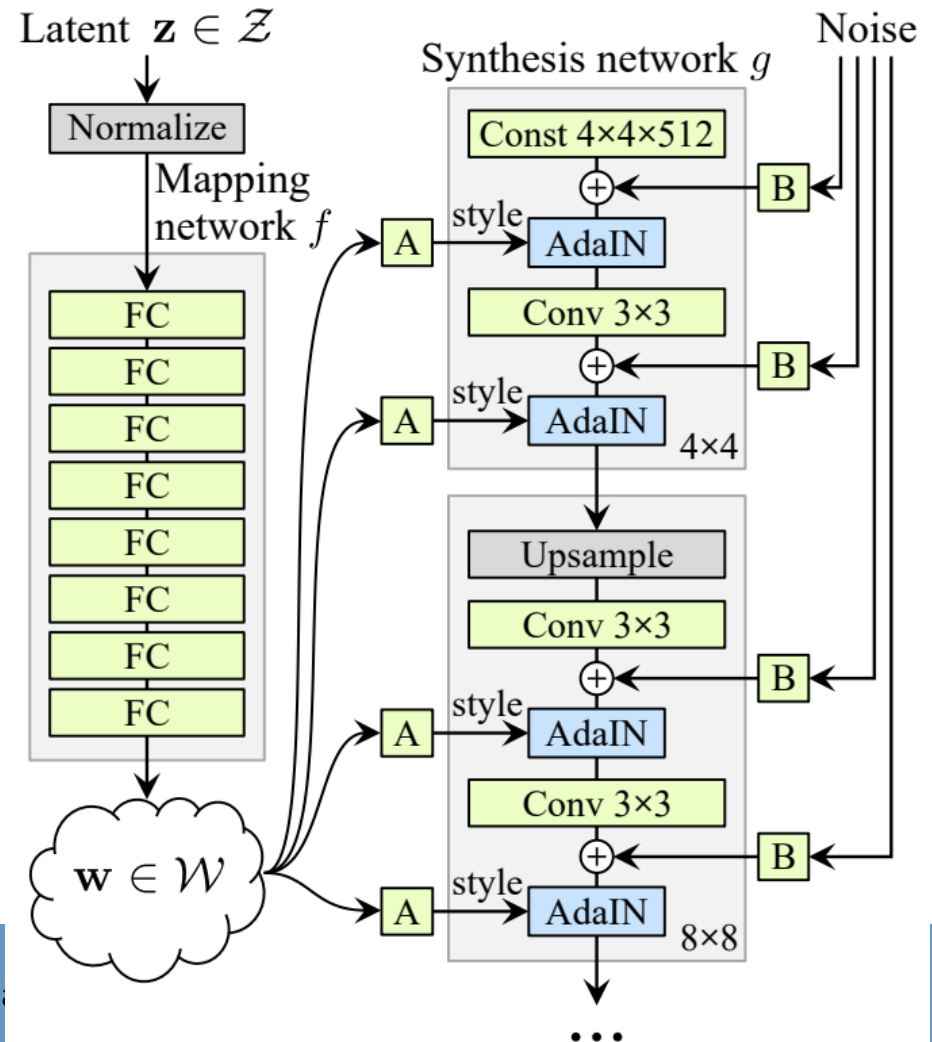
- Problema del colapso del generador



- Solución:
  - Otras funciones de coste, como la Wasserstein (WGAN)
  - Intercalar generadores con resolución creciente

# Generative Adversarial Networks

- StyleGAN [[Karras et al 2018](#)]:
  - Dos fuentes de aleatoriedad para el generador
    - Style (ruido por pixel)
    - Noise
  - El generador por fases de creciente resolución  $(4 \times 4) \rightarrow (8 \times 8) \rightarrow (16 \times 16) \dots \rightarrow (1024 \times 1024)$



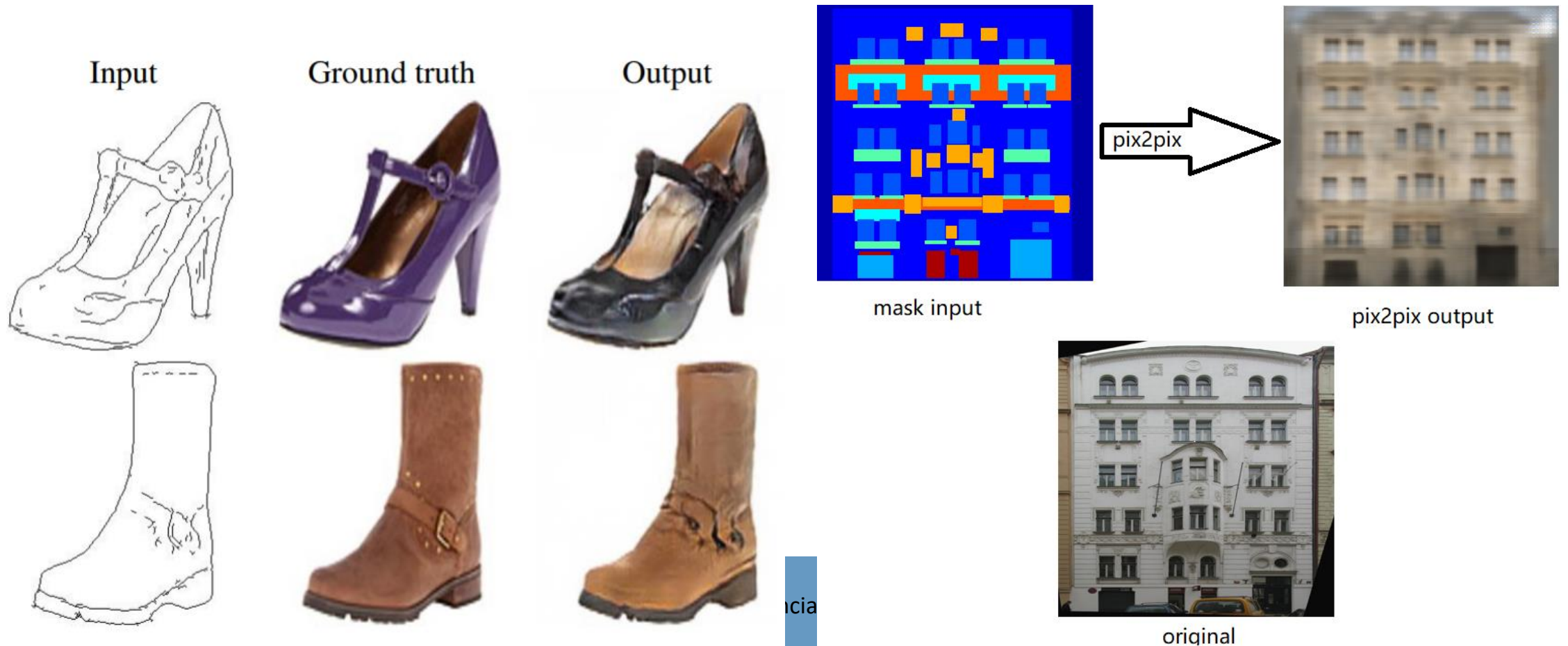
# Generative Adversarial Networks

- StyleGAN:



# Generative Adversarial Networks

- Pix2pix: basado en cGAN (GANs condicionadas) [[Isola et al 2018](#)]





# Recapitulando

- Un ejemplo de **aprendizaje no supervisado** en Deep Learning son los modelos generativos.
- **Autoencoders**, que requieren de un tipo especial de convolución transpuesta
- **VAEs**, aprenden un estado latente como una distribución desde la que muestrear. Permite jugar con las componentes.
- **GANs**, permite entrenar una red generadora frente a una discriminadora para que la generadora cree ejemplos falsos de forma realista. Variantes: **WGAN, StyleGAN, cGAN, Pix2Pix,...**