

# Auto-encoders y variational auto-encoders

---

Auto-encoders y variational auto-encoders

---

# Introducción

# Introducción

## Aprendizaje no supervisado

Hasta ahora hemos estado hablando siempre de **aprendizaje profundo supervisado**, pero también podemos resolver problemas **no supervisados**.

### Objetivo:

- Extraer patrones directamente de los datos “sin etiquetar”, solo tenemos  $x$  y no  $y$ .

### Tareas comunes:

- Modelos generativos: Entender la distribución de  $x$  y generar nuevas muestras.
- Autoencoders: “Comprimir”  $x$  proyectándolo en un espacio de menor dimensión.

# Introducción

## Aprendizaje no supervisado



*The brain has about  $10^{14}$  synapses and we only live for about  $10^9$  seconds. So we have a lot more parameters than data. This motivates the idea that we must do a lot of **unsupervised learning** since the perceptual input (including proprioception) is the only place we can get  $10^5$  dimensions of constraint per second.*

Geoffrey Hinton, 2014

# Introducción

## Aprendizaje no supervisado



*We need tremendous amount of information to build machines that have common sense and generalize.*

Yann LeCun, 2016

### ■ "Pure" Reinforcement Learning (cherry)

- ▶ The machine predicts a scalar reward given once in a while.

▶ **A few bits for some samples**

### ■ Supervised Learning (icing)

- ▶ The machine predicts a category or a few numbers for each input
- ▶ Predicting human-supplied data
- ▶ **10→10,000 bits per sample**

### ■ Unsupervised/Predictive Learning (cake)

- ▶ The machine predicts any part of its input for any observed part.
- ▶ Predicts future frames in videos
- ▶ **Millions of bits per sample**

■ (Yes, I know, this picture is slightly offensive to RL folks. But I'll make it up)



# Introducción

## Modelos generativos

Un **modelo generativo** es un modelo probabilístico  $p$  que puede ser utilizado como un “simulador de datos”.

Su propósito es generar datos sintéticos pero realistas de alta dimensión

$$\mathbf{x} \sim p_{\theta}(\mathbf{x}),$$

que se asemejen lo más posible a la distribución desconocida de datos  $p(\mathbf{x})$ .

# Introducción

## Modelos generativos

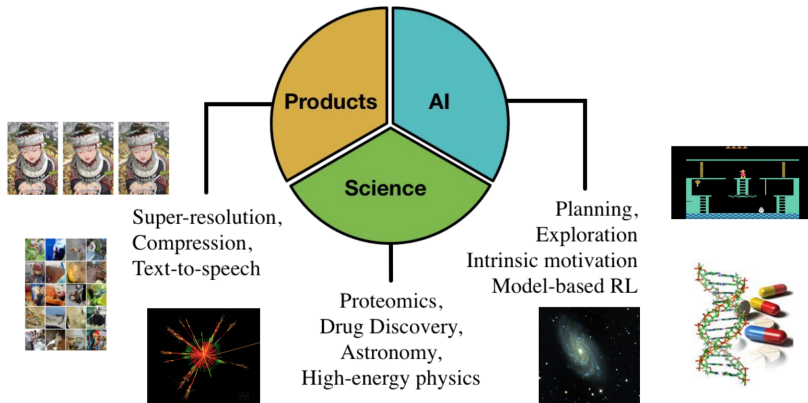


Ley de Moore de los modelos generativos de imágenes

# Introducción

## Modelos generativos

Algunas aplicaciones:



Los modelos generativos tienen un rol muy importante en muchos problemas actuales

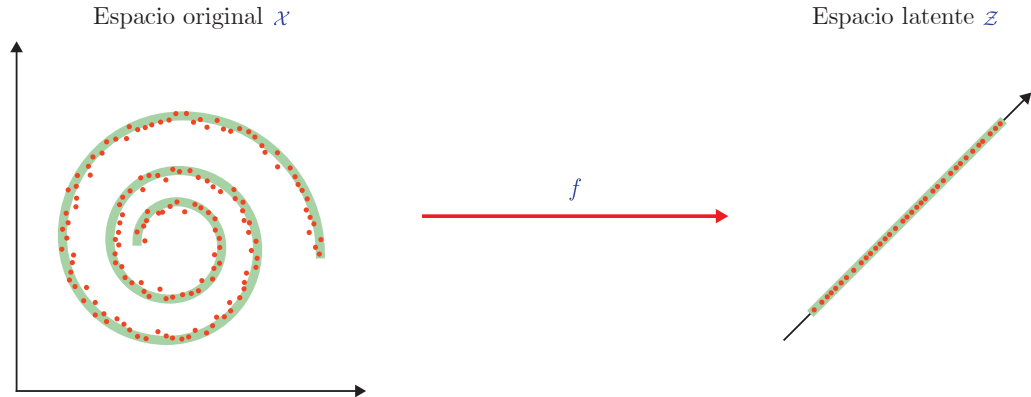


Auto-encoders y variational auto-encoders

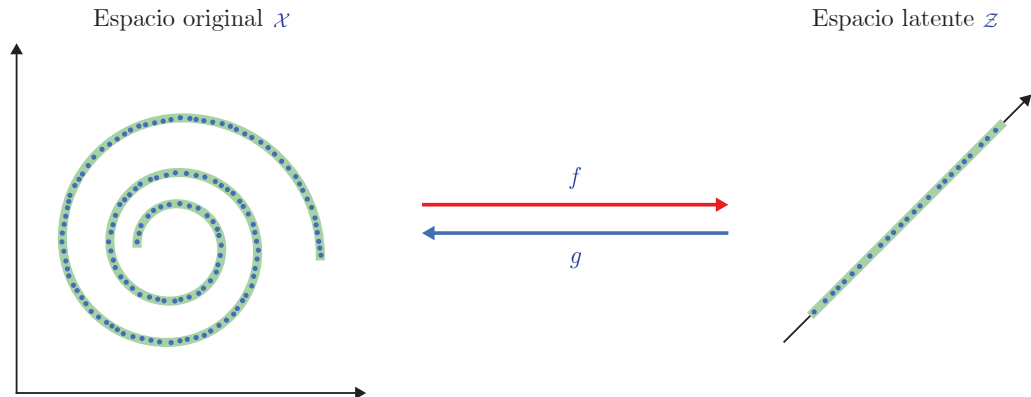
---

# Auto-encoders

# Auto-encoders



# Auto-encoders

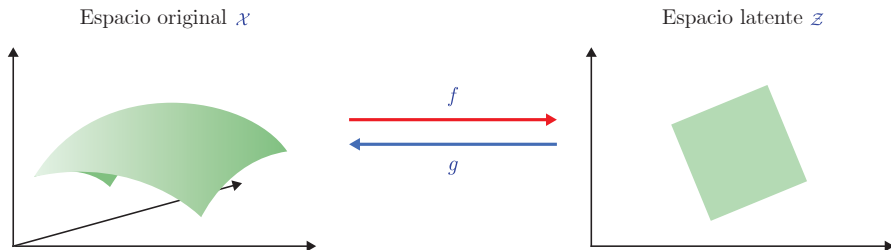


# Auto-encoders

Un **auto-encoder** es una función compuesta a partir de:

- Un **encoder**  $f$  que proyecta del espacio original  $\mathcal{X}$  al espacio latente  $\mathcal{Z}$ .
- Un **decoder**  $g$  que proyecta de vuelta al espacio original.

El objetivo es que  $g \circ f$ , es decir, que la composición de funciones se aproxime lo máximo posible a los datos originales o función identidad.



# Auto-encoders

Siendo  $p(\mathbf{x})$  la distribución de los datos en  $\mathcal{X}$ , un buen auto-encoder puede caracterizarse con la *reconstruction loss*:

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} [\|\mathbf{x} - g \circ f(\mathbf{x})\|^2] \approx 0.$$

Esta función de pérdida mide como de bien el auto-encoder puede reconstruir los datos originales.

Dadas dos funciones de proyección con parámetros  $f(\cdot; \theta_f)$  and  $g(\cdot; \theta_g)$ , el entrenamiento consiste aprender los parámetros que minimicen dicha loss:

$$\theta_f, \theta_g = \arg \min_{\theta_f, \theta_g} \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - g(f(\mathbf{x}_i, \theta_f), \theta_g)\|^2.$$

# Auto-encoders

## Ejemplo

Imaginemos, por ejemplo, un auto-encoder lineal con

$$f : \mathbf{z} = \mathbf{U}^T \mathbf{x}$$

$$g : \hat{\mathbf{x}} = \mathbf{U} \mathbf{z},$$

con  $\mathbf{U} \in \mathbb{R}^{p \times d}$ , el *reconstruction loss* se reduce a

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[ \|\mathbf{x} - \mathbf{U} \mathbf{U}^T \mathbf{x}\|^2 \right].$$

Auto-encoders y variational auto-encoders

---

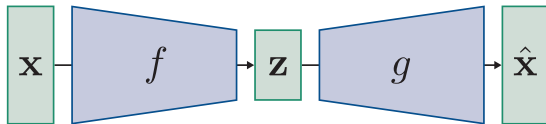
## Deep Auto-encoders

## Mayor profundidad

Para obtener mejores resultados, en vez de proyecciones lineales se suelen utilizar redes neuronales profundas en  $f$  y  $g$ .

Algunos ejemplos:

- Combinando un MLP encoder  $f : \mathbb{R}^p \rightarrow \mathbb{R}^d$  con un MLP decoder  $g : \mathbb{R}^d \rightarrow \mathbb{R}^p$ .
- Combinando un convolutional network encoder  $f : \mathbb{R}^{w \times h \times c} \rightarrow \mathbb{R}^d$  con un decoder decoder  $g : \mathbb{R}^d \rightarrow \mathbb{R}^{w \times h \times c}$  compuesto de capas convolucionales reciprocas.

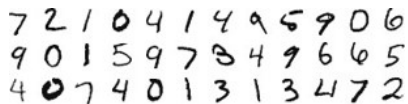




# Deep Auto-encoders

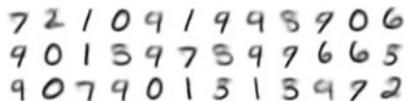
## Ejemplo MNIST

Datos originales  $\mathbf{x}$  con  $d = 784$ .



7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de CNN con  $d = 2$ .



7	2	1	0	9	1	9	9	8	9	0	6
9	0	1	3	9	7	3	9	9	6	6	5
9	0	7	9	0	1	3	1	3	9	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de PCA con  $d = 2$ .

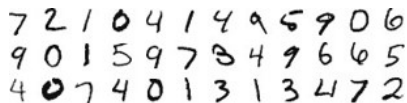


9	2	1	0	9	1	9	9	0	9	0	0
9	0	1	3	9	9	3	9	9	0	9	0
9	0	9	9	0	1	3	1	3	0	9	0

# Deep Auto-encoders

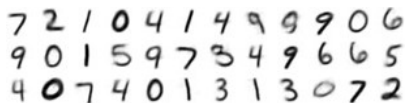
## Ejemplo MNIST

Datos originales  $\mathbf{x}$  con  $d = 784$ .



7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de CNN con  $d = 4$ .



7	2	1	0	4	1	4	9	9	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	0	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de PCA con  $d = 4$ .

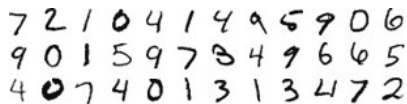


7	2	1	0	4	1	9	9	0	9	0	6
9	0	1	3	9	9	0	9	9	0	6	5
9	0	9	9	0	1	3	1	3	4	9	0

# Deep Auto-encoders

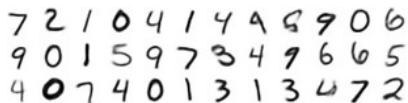
## Ejemplo MNIST

Datos originales  $\mathbf{x}$  con  $d = 784$ .



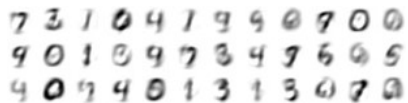
7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de CNN con  $d = 8$ .



7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de PCA con  $d = 8$ .

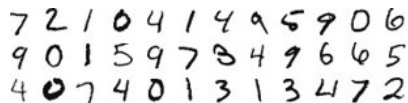


7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

# Deep Auto-encoders

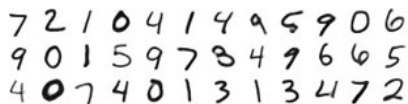
## Ejemplo MNIST

Datos originales  $\mathbf{x}$  con  $d = 784$ .



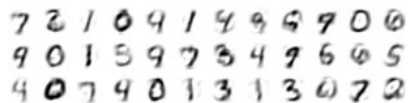
7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	8	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de CNN con  $d = 16$ .



7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	8	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de PCA con  $d = 16$ .

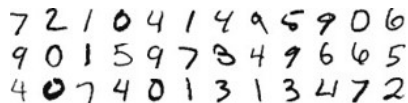


7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	8	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

# Deep Auto-encoders

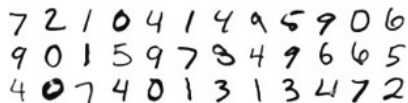
## Ejemplo MNIST

Datos originales  $\mathbf{x}$  con  $d = 784$ .



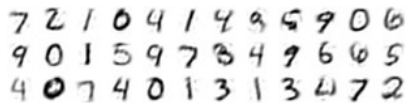
7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de CNN con  $d = 32$ .



7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

Resultado de auto-encoder  $g \circ f$  creado a partir de PCA con  $d = 32$ .



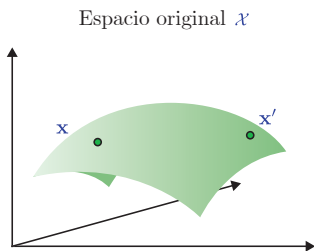
7	2	1	0	4	1	4	9	5	9	0	6
9	0	1	5	9	7	3	4	9	6	6	5
4	0	7	4	0	1	3	1	3	4	7	2

# Deep Auto-encoders

## Interpolación

### Espacio latente

Para comprender la representación latente aprendida, podemos elegir dos muestras  $\mathbf{x}$  y  $\mathbf{x}'$  al azar e interpolar muestras a lo largo de la línea en el espacio latente.

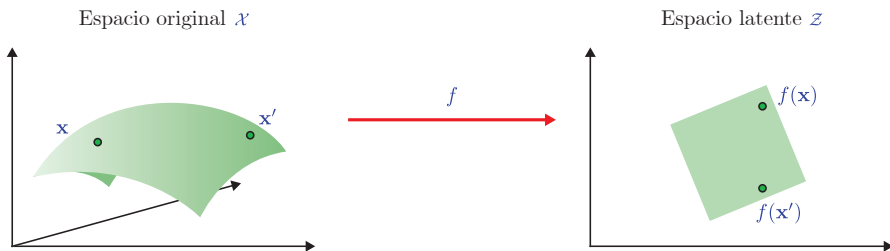


# Deep Auto-encoders

## Interpolación

### Espacio latente

Para comprender la representación latente aprendida, podemos elegir dos muestras  $\mathbf{x}$  y  $\mathbf{x}'$  al azar e interpolar muestras a lo largo de la línea en el espacio latente.

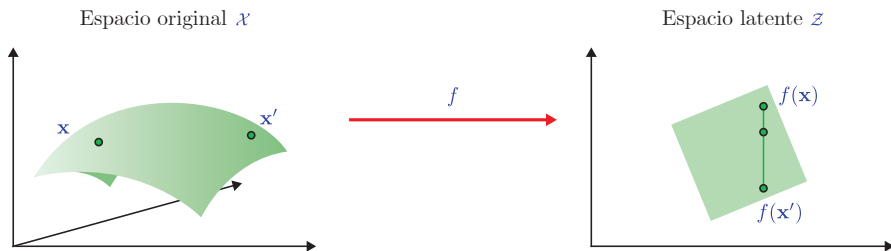


# Deep Auto-encoders

## Interpolación

### Espacio latente

Para comprender la representación latente aprendida, podemos elegir dos muestras  $\mathbf{x}$  y  $\mathbf{x}'$  al azar e interpolar muestras a lo largo de la línea en el espacio latente.



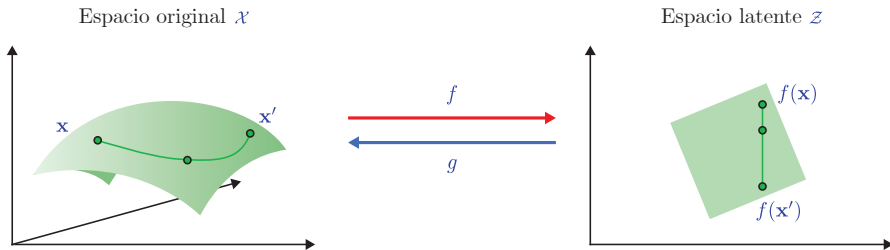


# Deep Auto-encoders

## Interpolación

### Espacio latente

Para comprender la representación latente aprendida, podemos elegir dos muestras  $\mathbf{x}$  y  $\mathbf{x}'$  al azar e interpolar muestras a lo largo de la línea en el espacio latente.



Auto-encoders y variational auto-encoders

---

## Referencias

- 1 **Lecture 11: Auto-encoders and variational auto-encoders**
- 2 **Deep Learning Course**