



Autoencoders - TP4

Grupo 11

Francisco Pérez Sammartino
& Gabriel Silvatici



Implementación



¿Cómo se construyó?

Perceptron Multicapa

- Cantidad y tipo de capas simétricos.

Inicialización de pesos (desvío $(layer+1)^{-\frac{1}{2}}$)

$$\frac{1}{(layer+1)^{1/2}} * \text{random}(0, 1)$$

Optimización

- Adam (m, v)
- Adagrad (squared)
- **Gradiente Estocástico (SGD)**
 - Batch de K muestras.



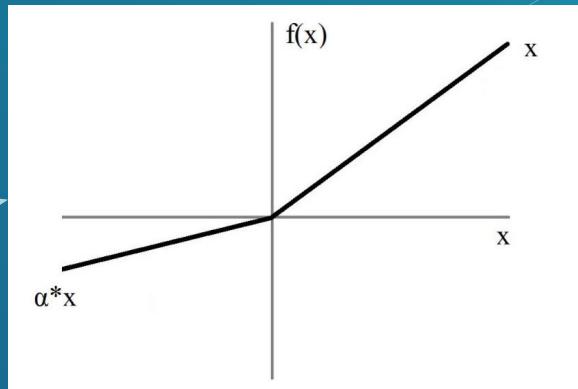
Funciones de Activación

Tangente Hiperbólica

Leaky ReLU:

$$\alpha = 0.2$$

Softmax:

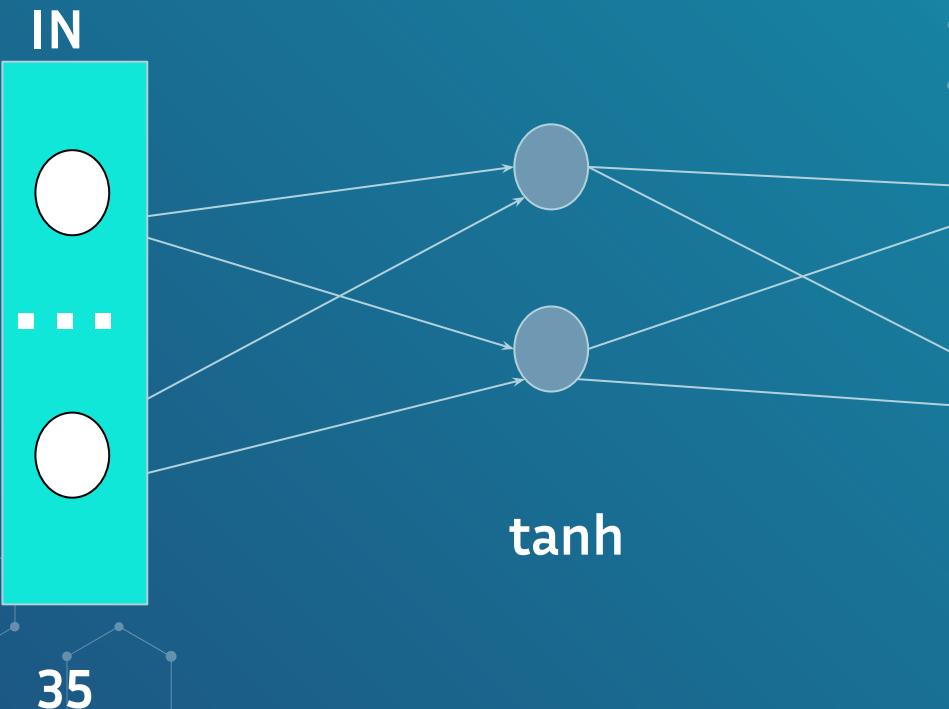


$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

2

Imágenes Binarias - Data set 3

Arquitectura 1



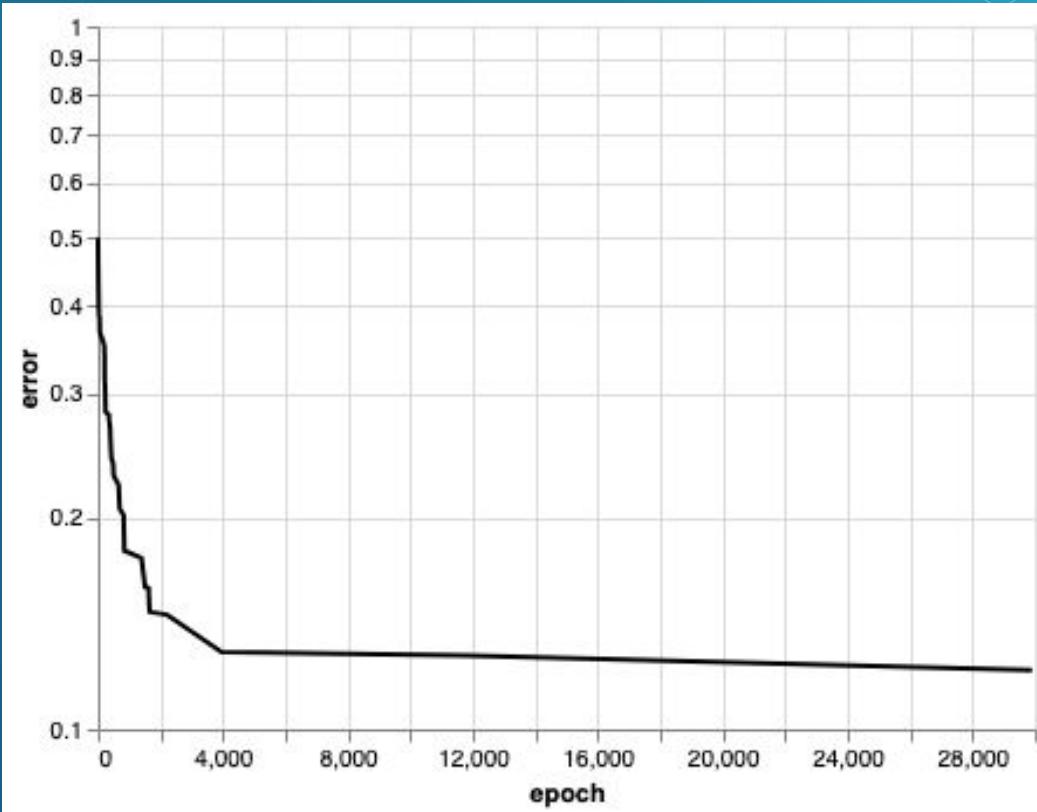
OUT

35

tanh

Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch = 5



Arquitectura 2



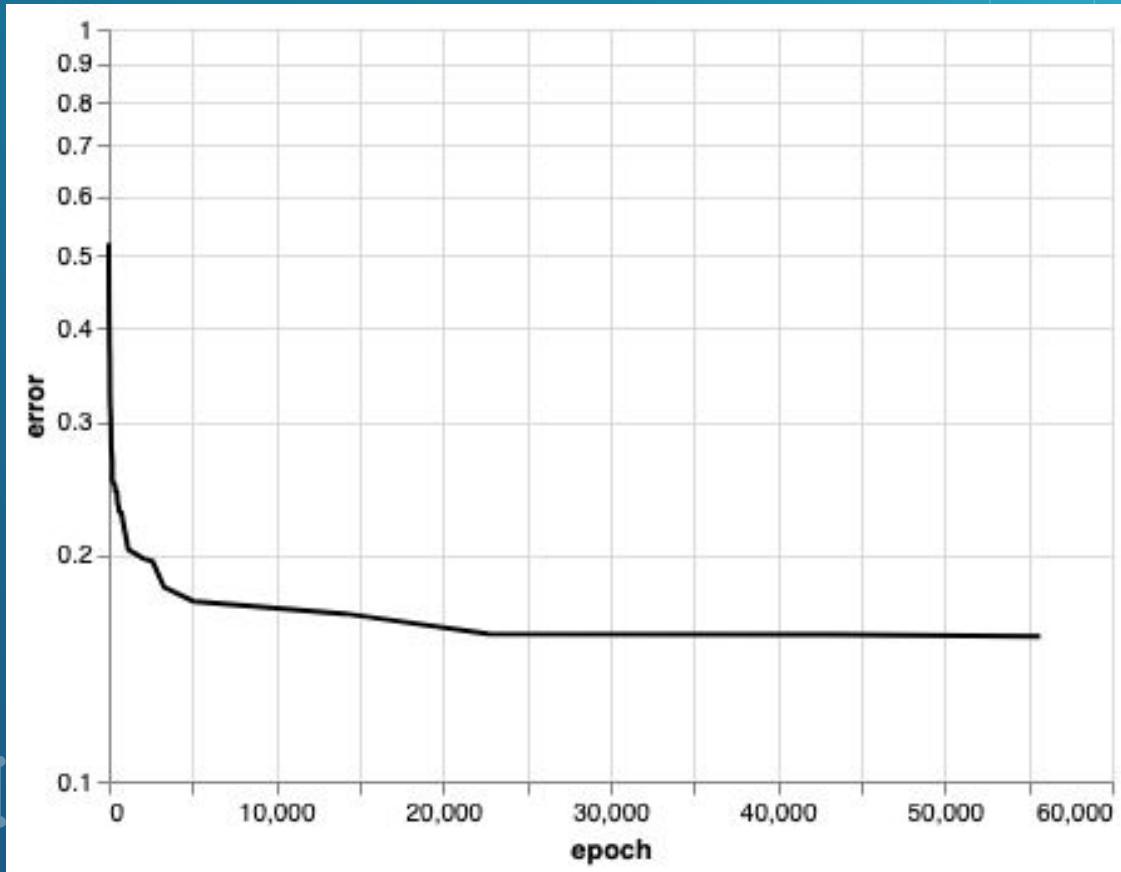
OUT

35

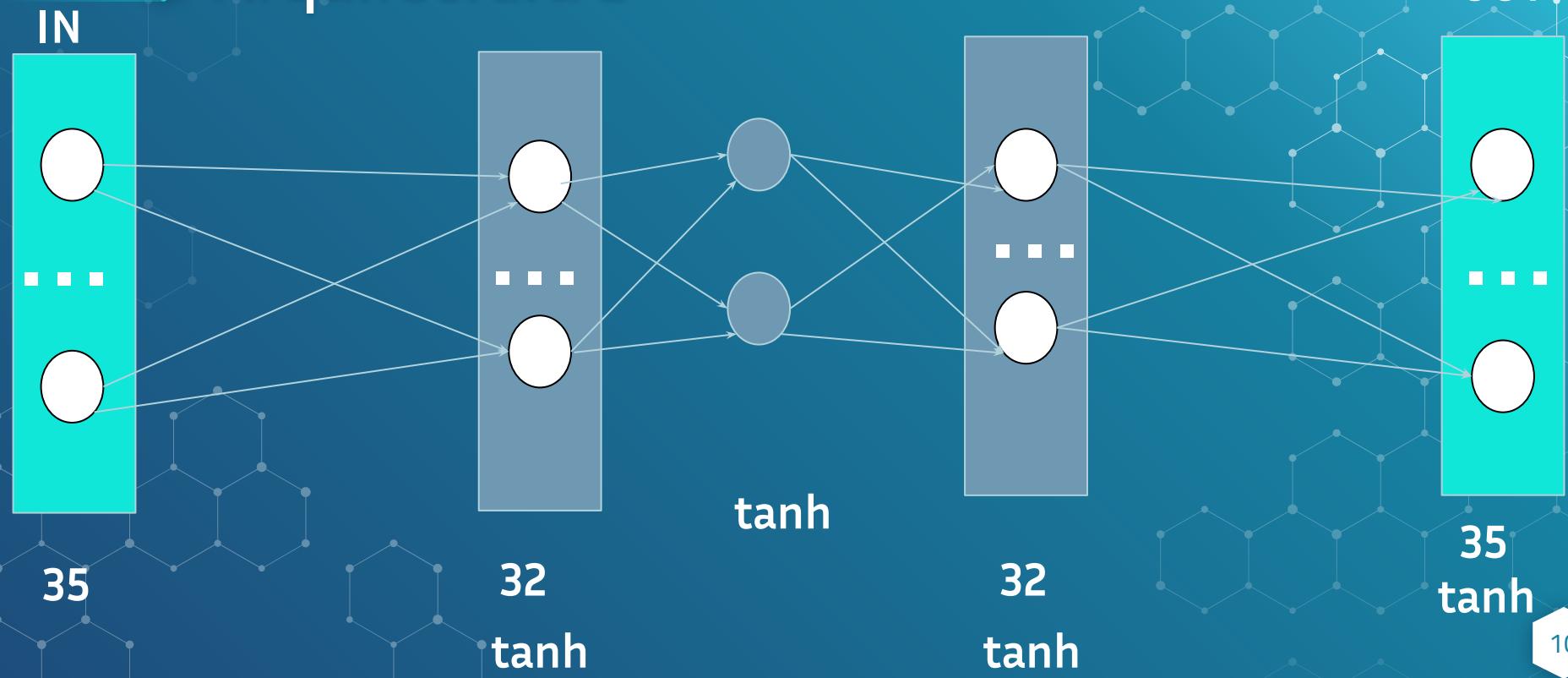
tanh

Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch = 5

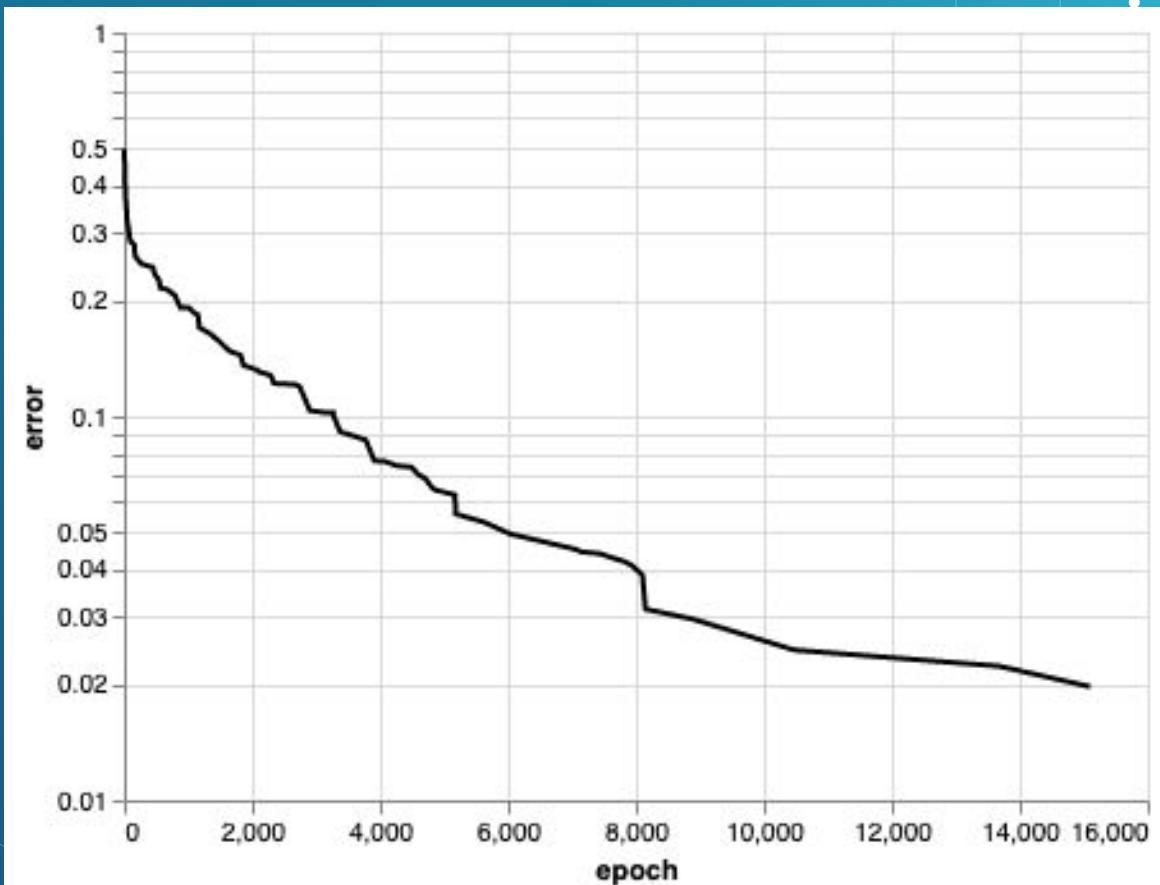


Arquitectura 3

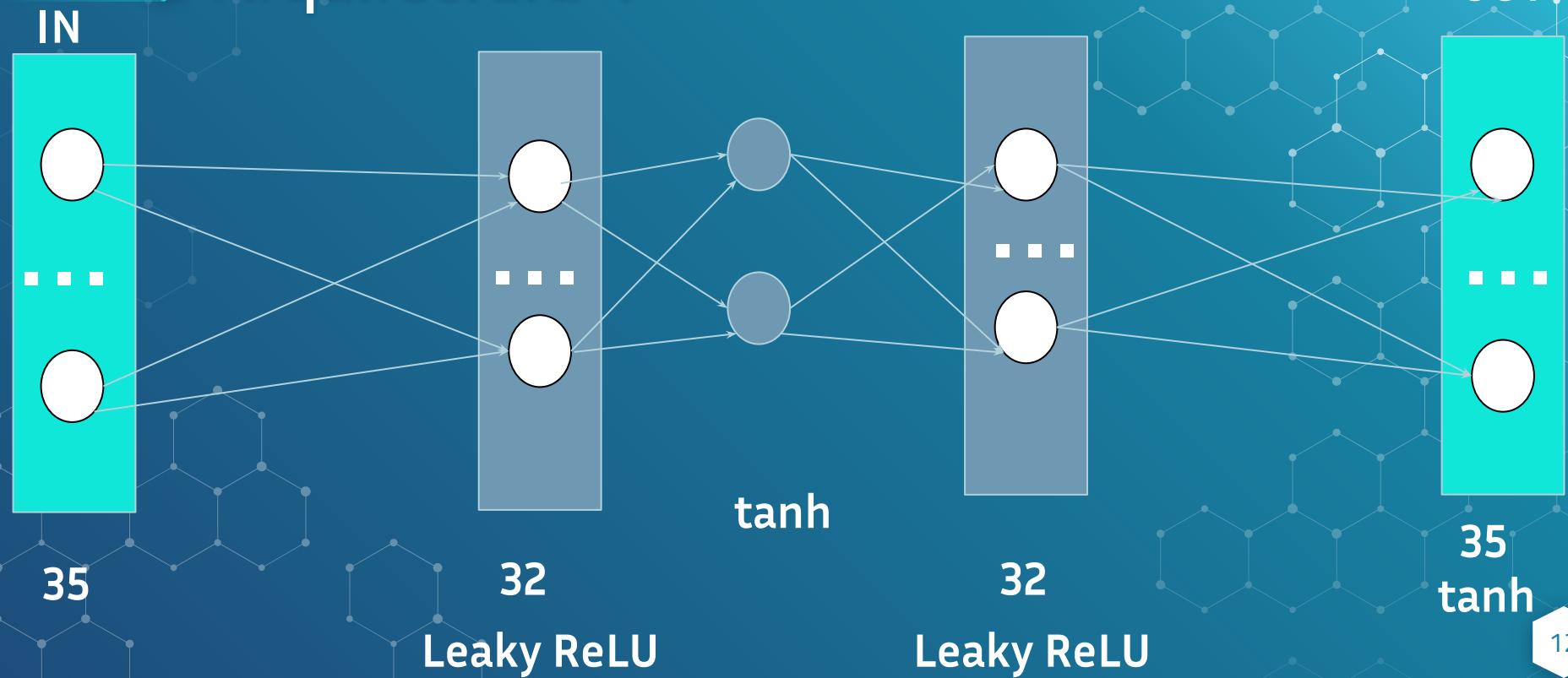


Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch = 5

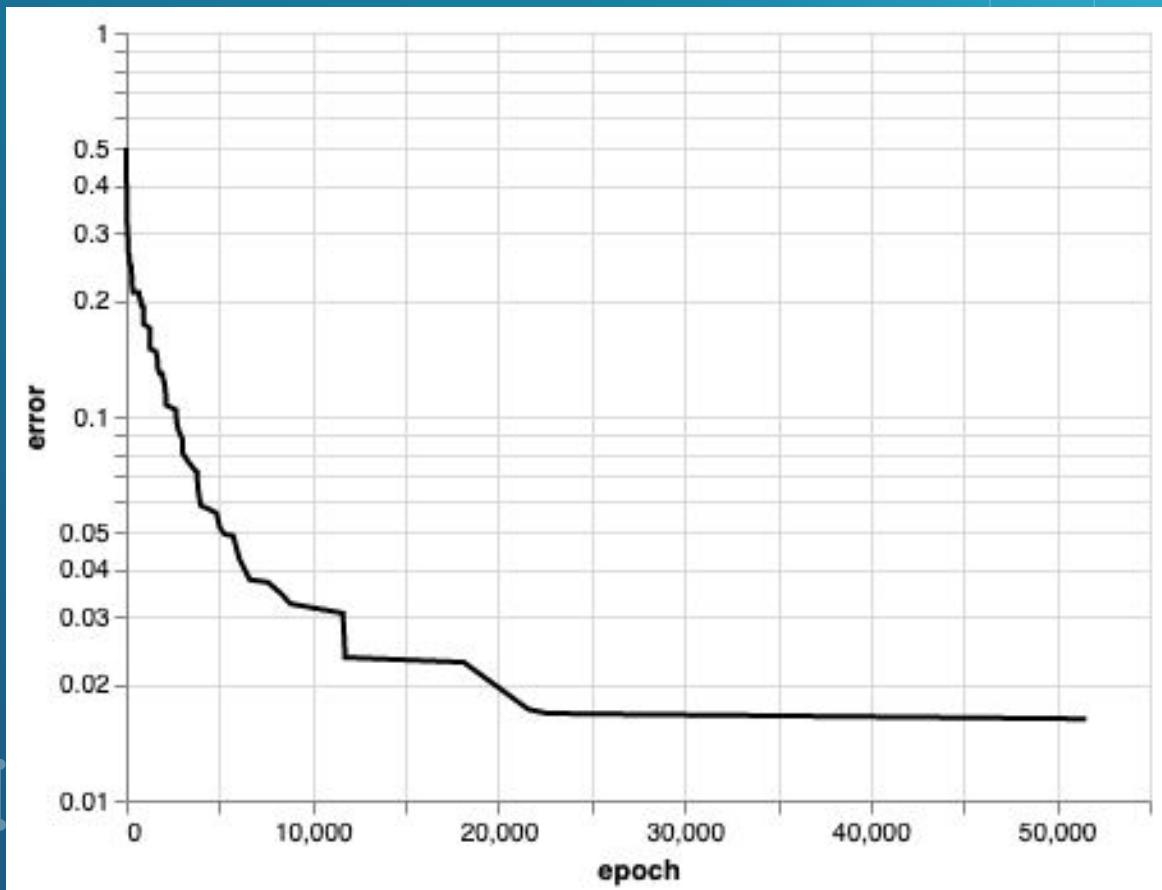


Arquitectura 4

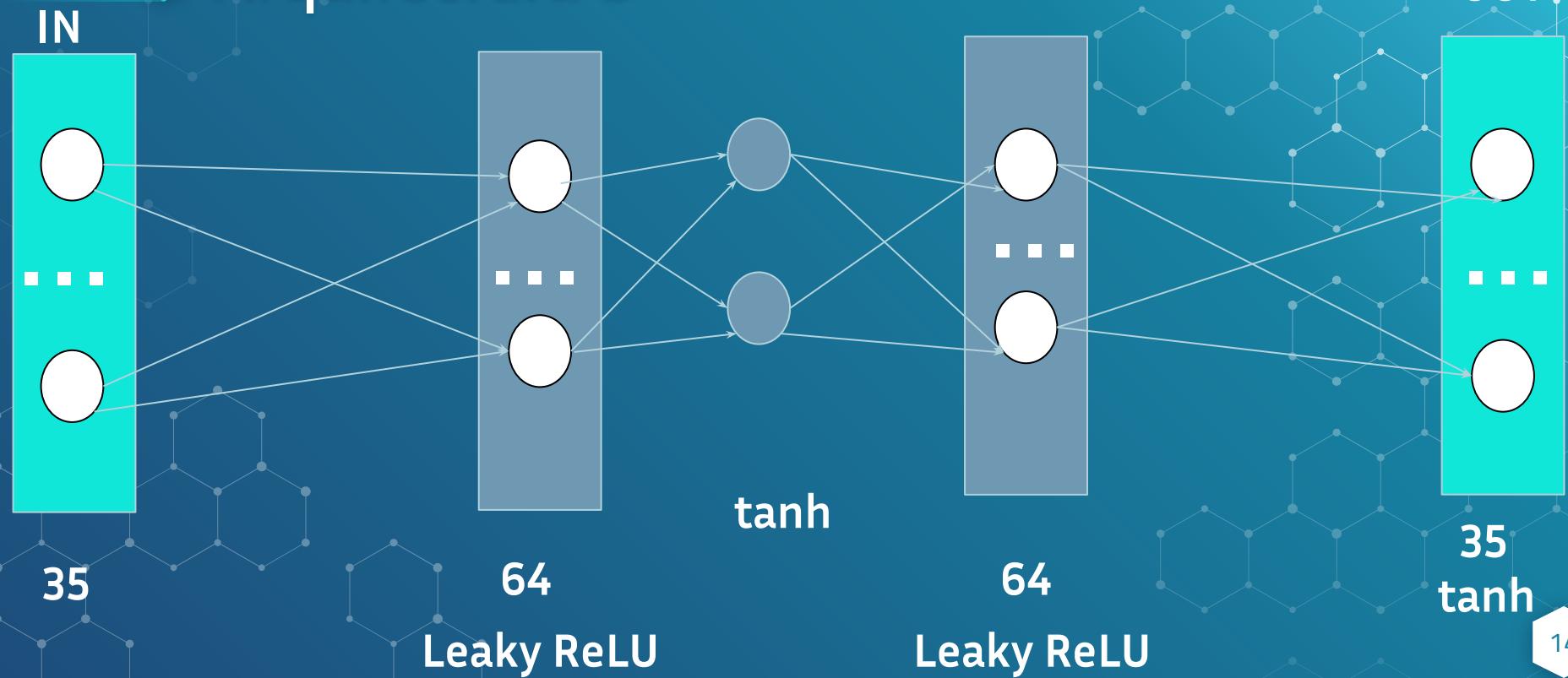


Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch = 5

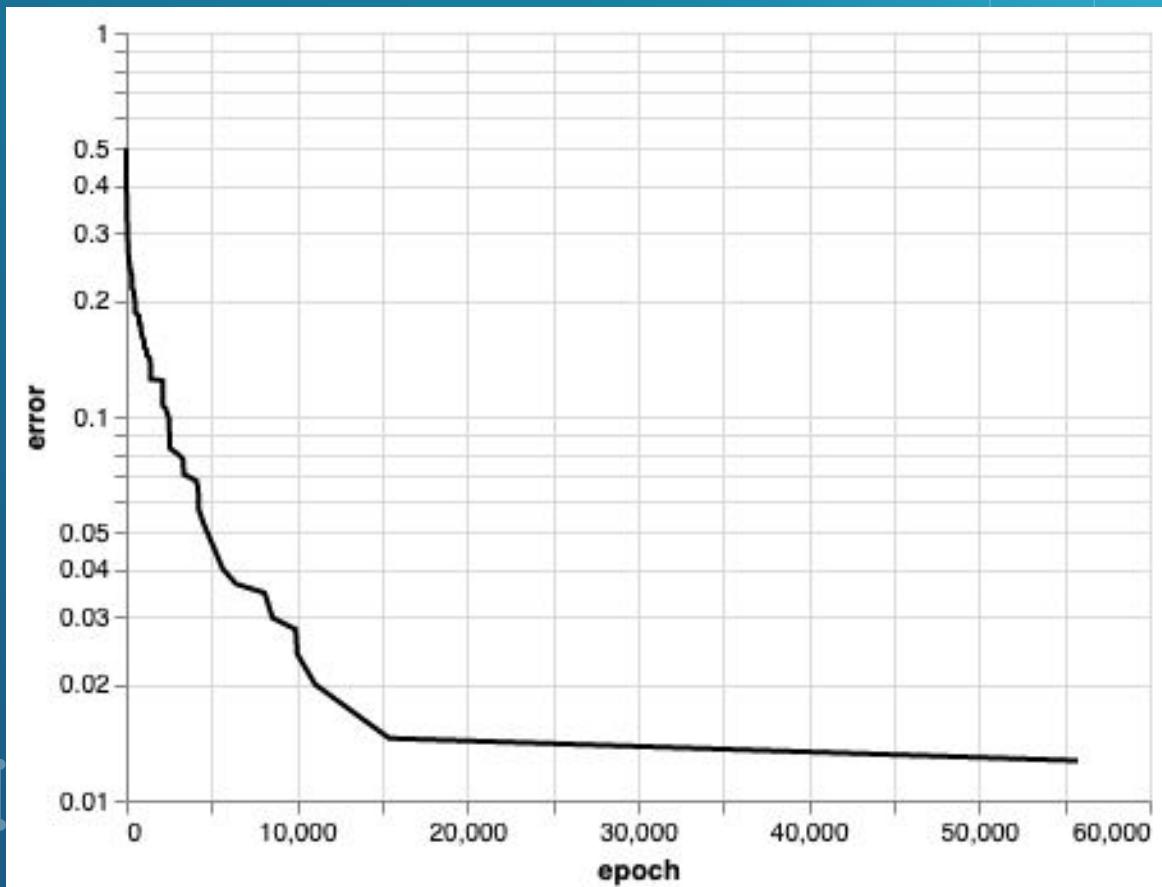


Arquitectura 5

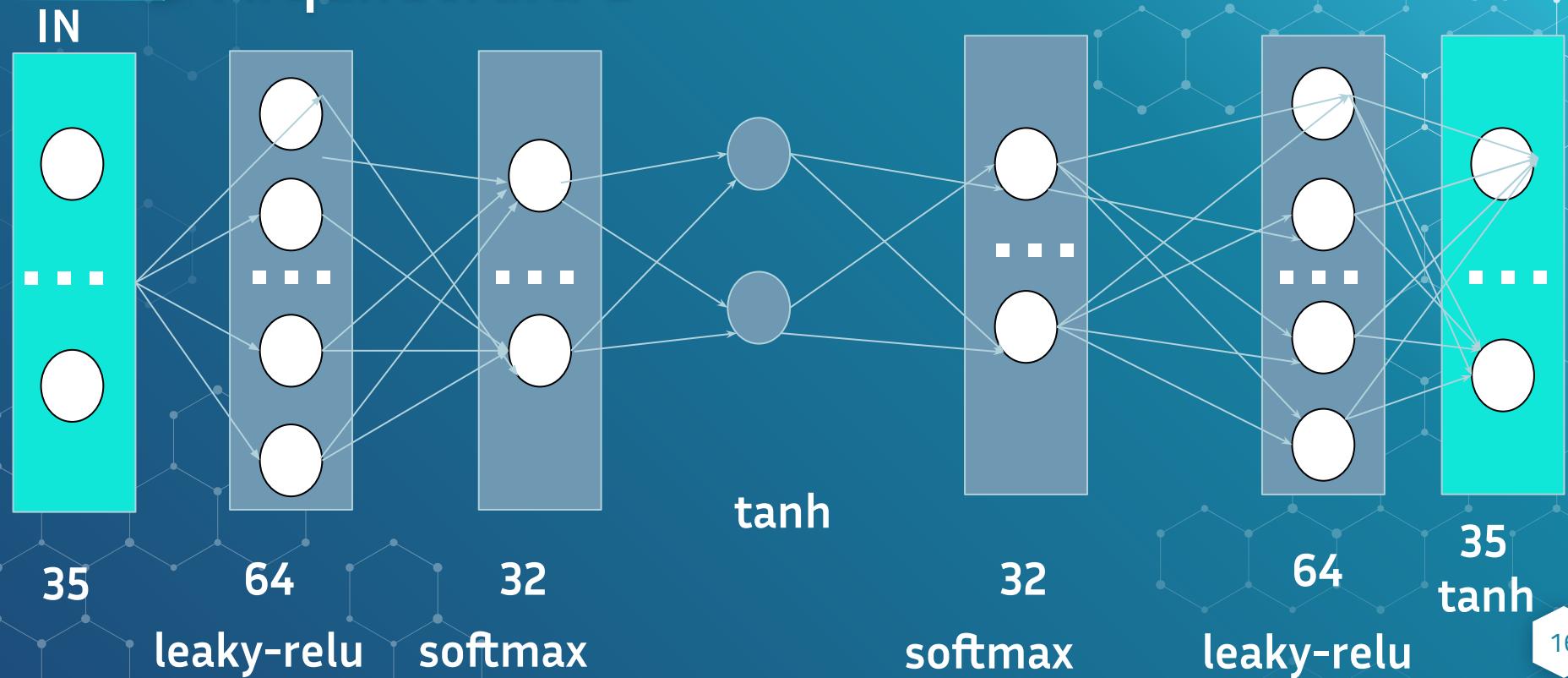


Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch =5

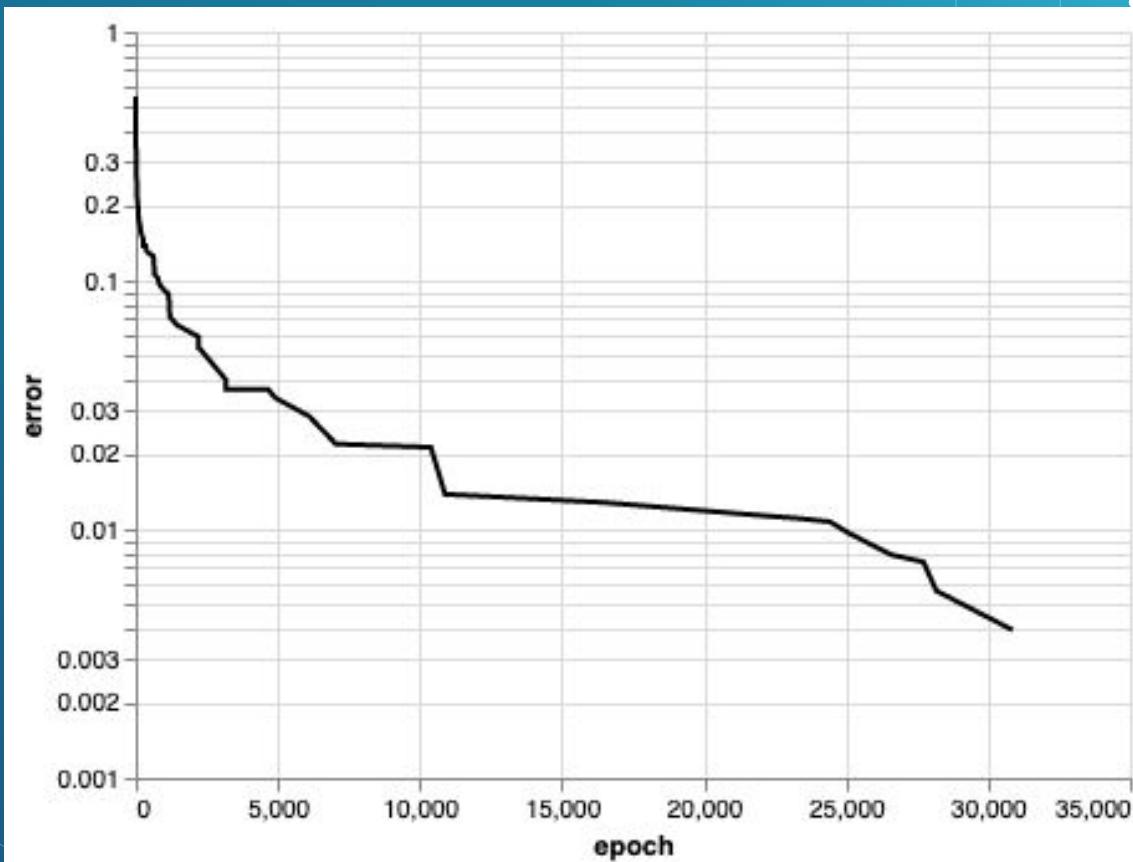


Arquitectura 6

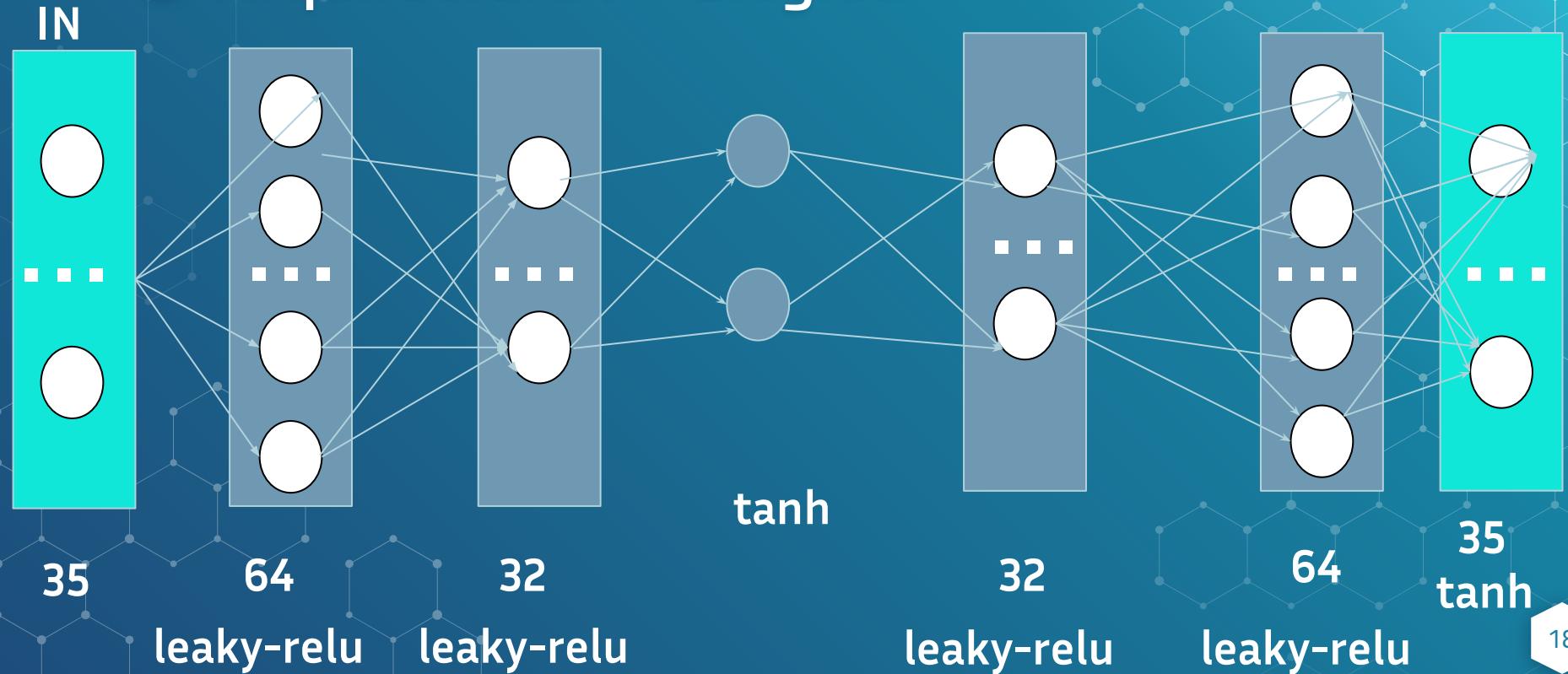


Evolución del Error

- Puntos donde error es más bajo.
- Hasta 50 mil epochs.
- Batch =5

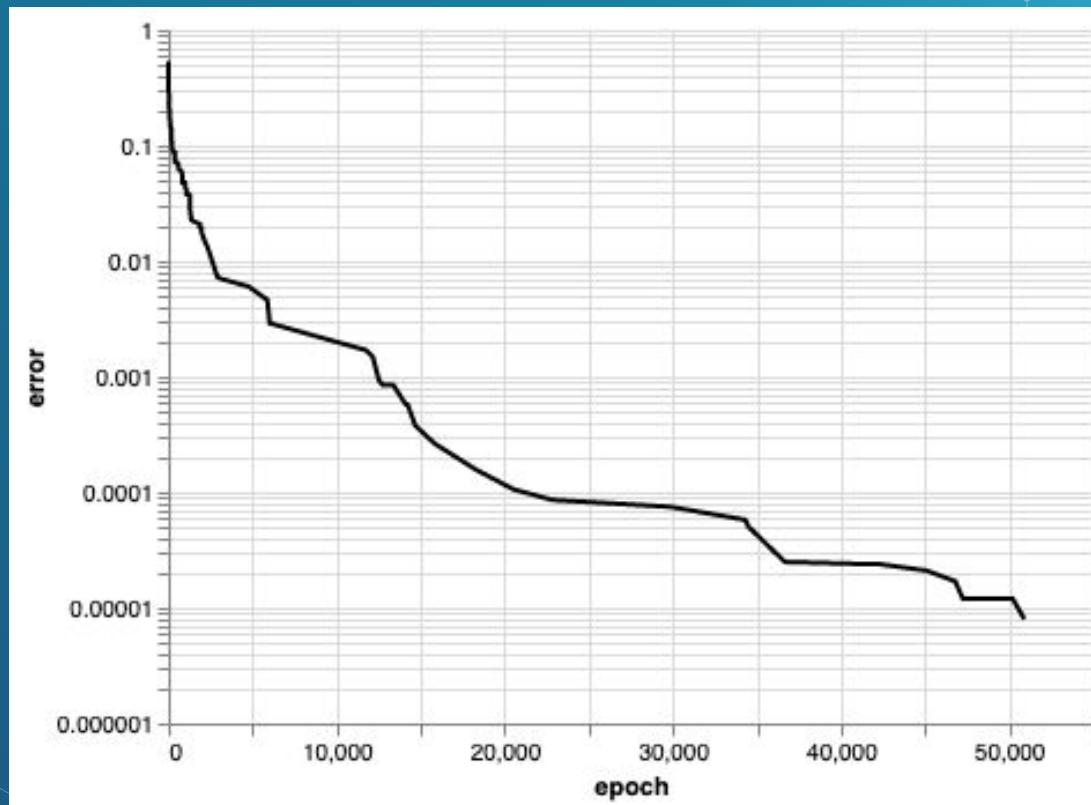


Arquitectura 7 - Elegida

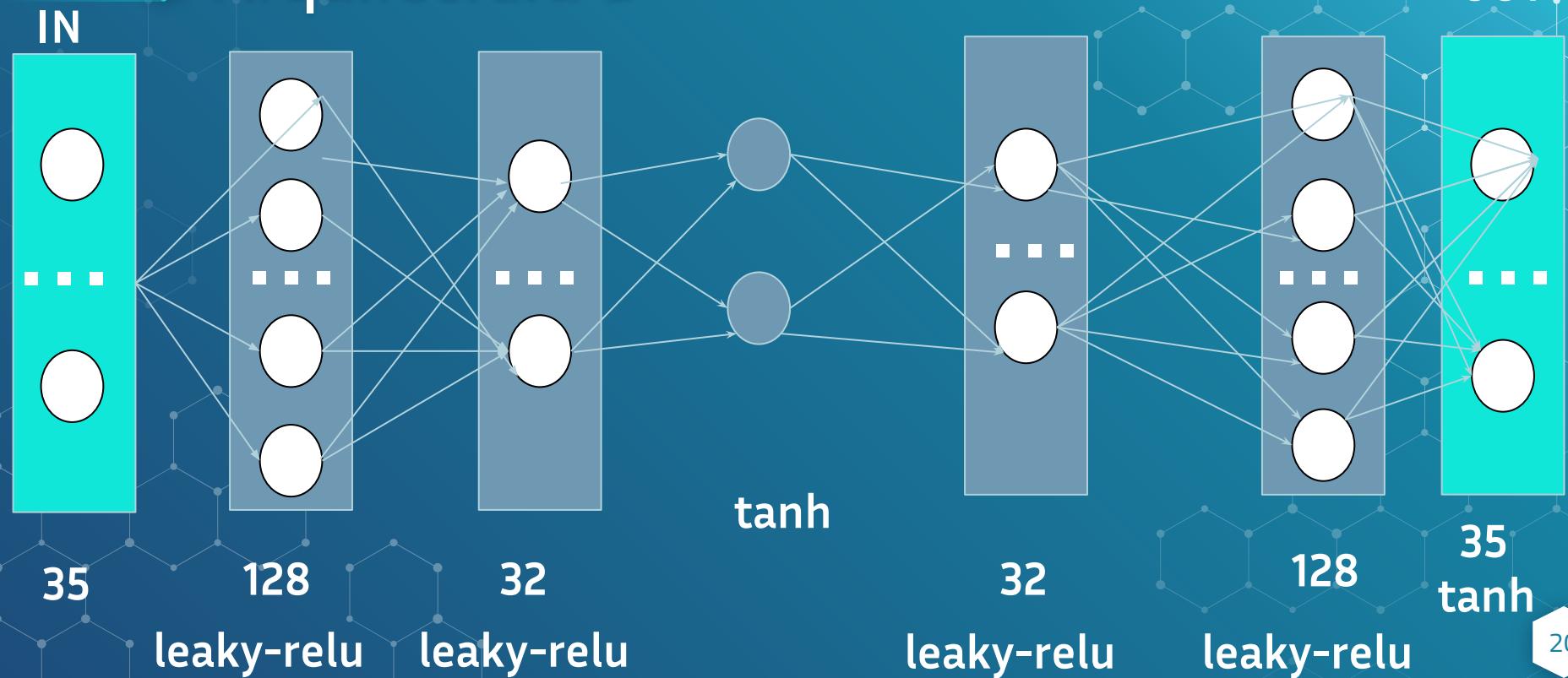


Evolución del Error

- Puntos donde error es más bajo.
- Hasta 150mil epochs.
- Desde 50000 queda estable.
- Batch = 5

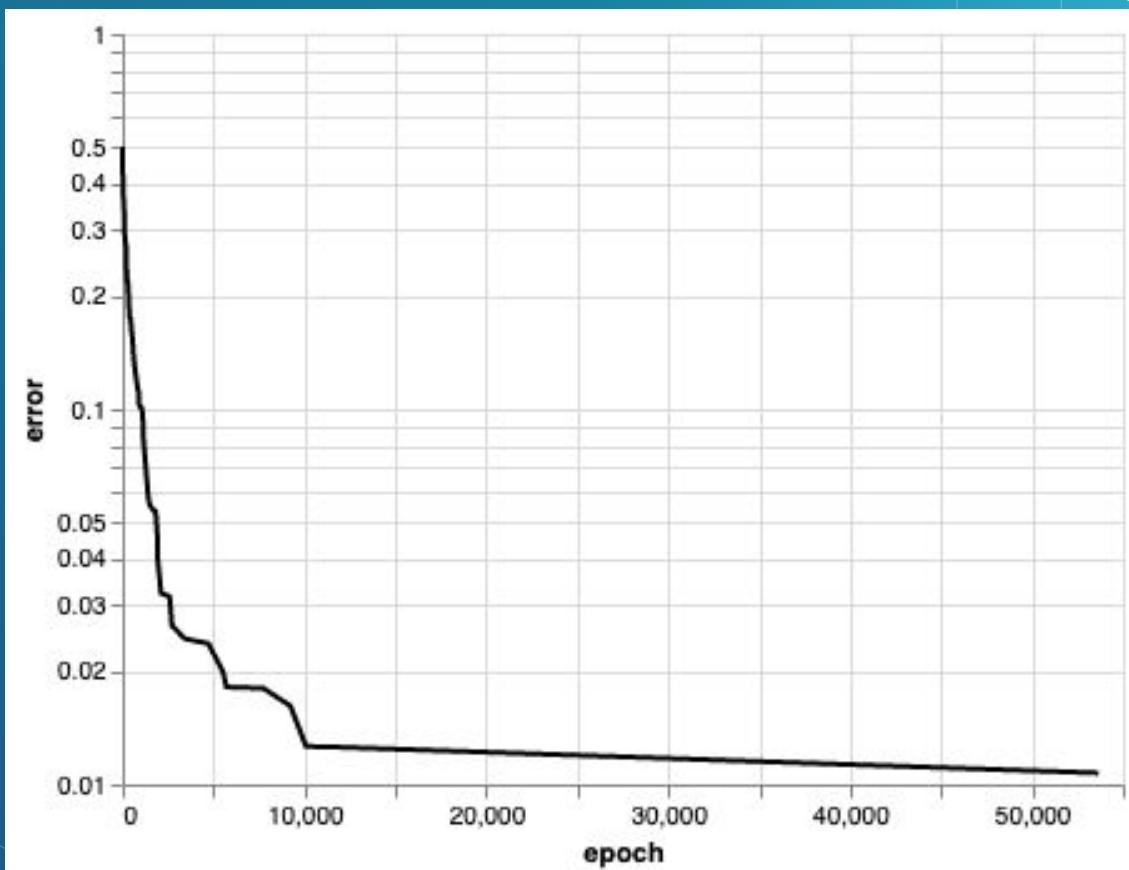


Arquitectura 8



Evolución del Error

- Puntos donde error es más bajo.
- Hasta 60 mil epochs.
- Batch = 5

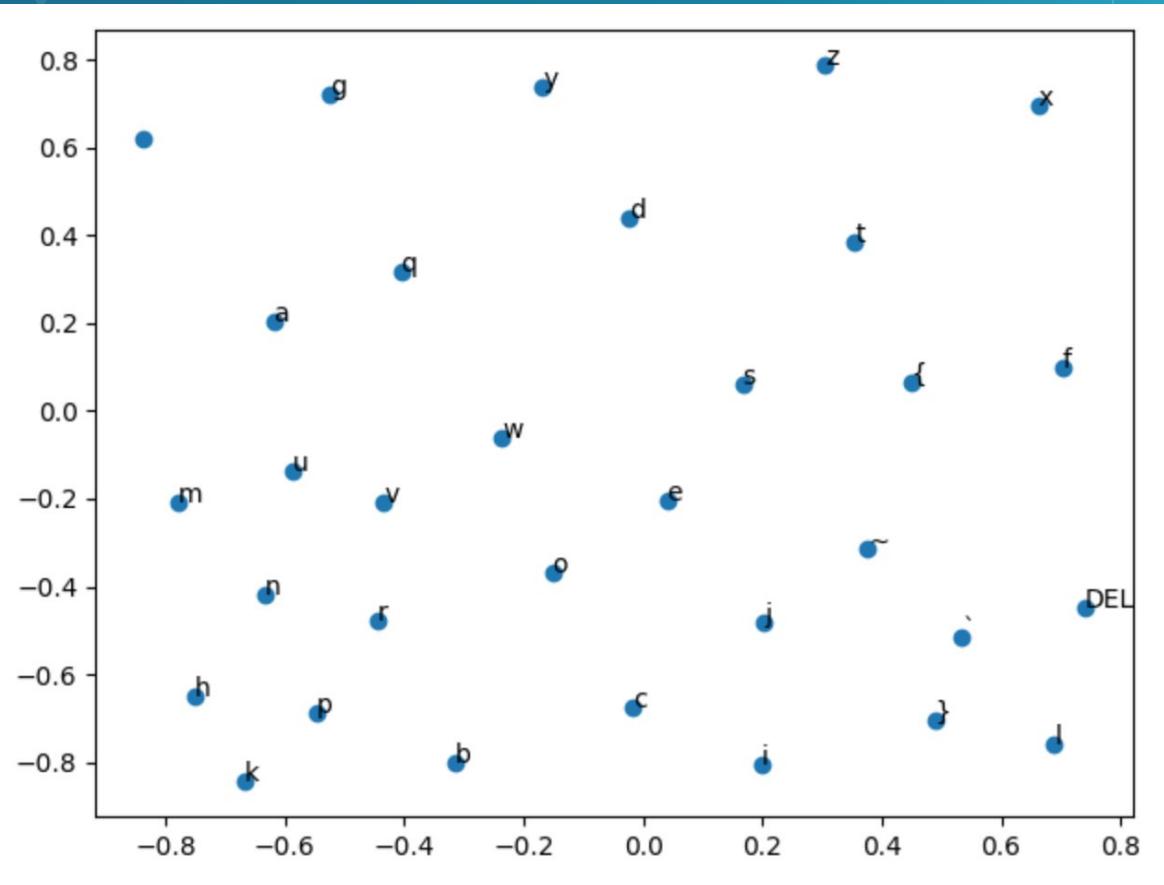


Codificación - Decodificación

Algoritmo de codificación

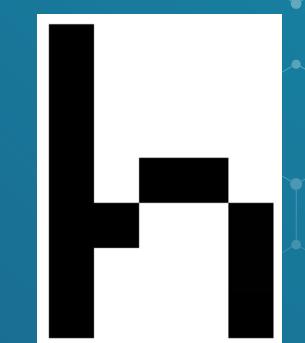
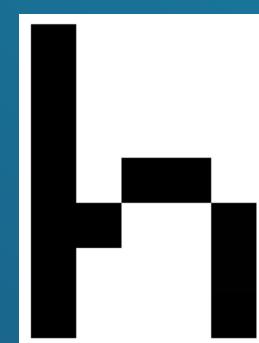
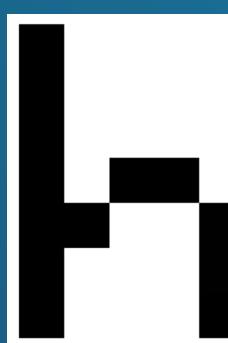
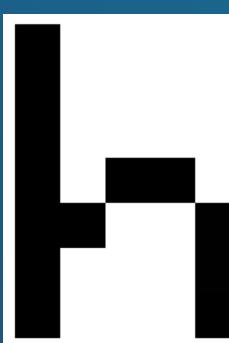
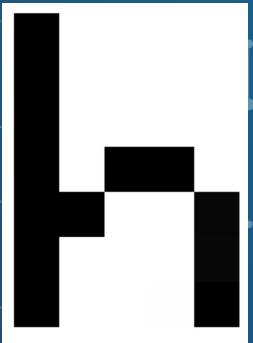
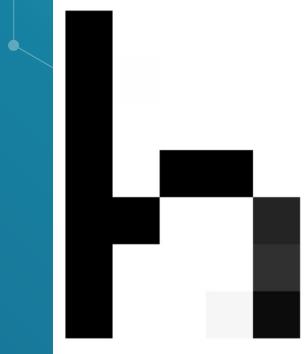
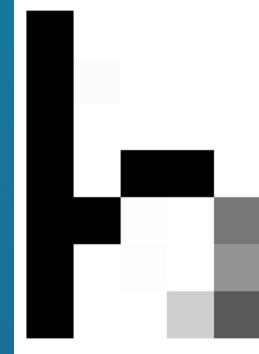
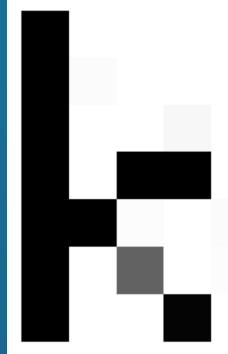
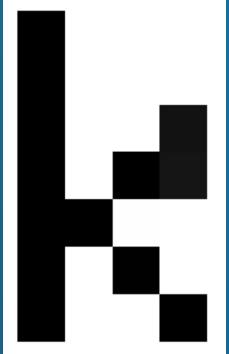
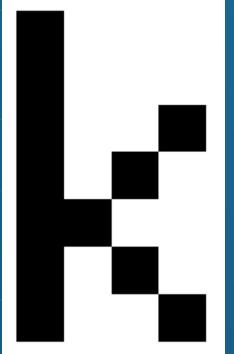
Algoritmo de decodificación

Espacio Latente



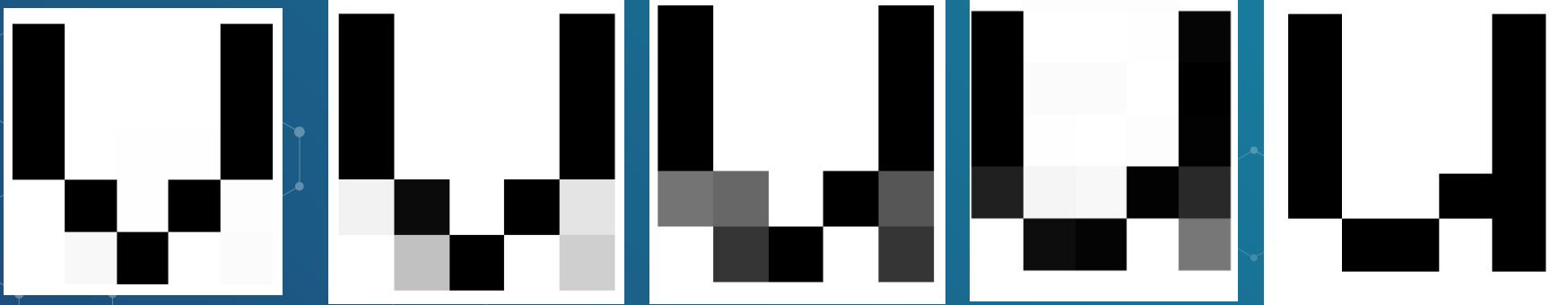
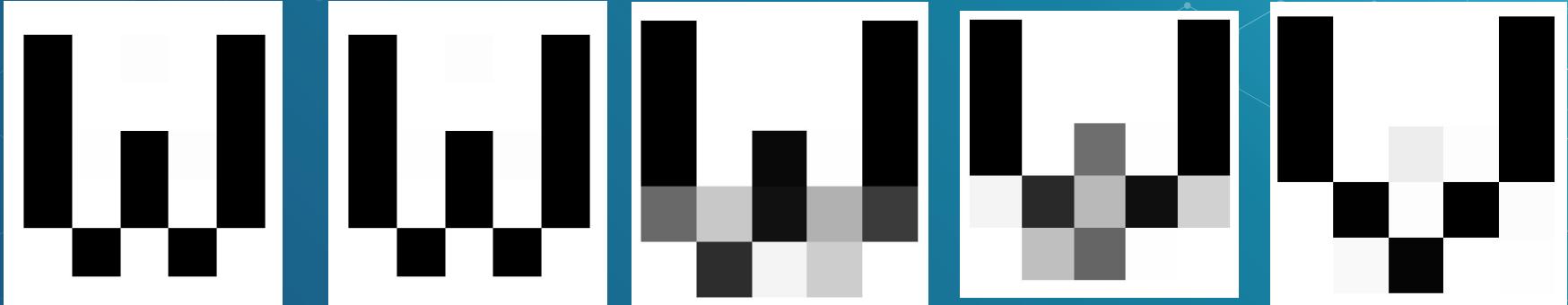
Transiciones

K - H



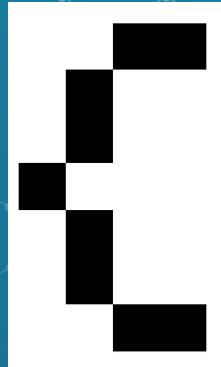
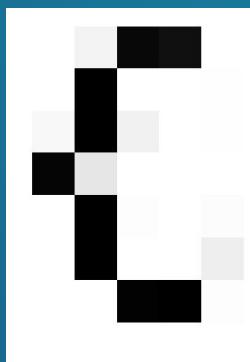
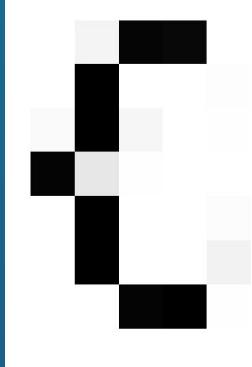
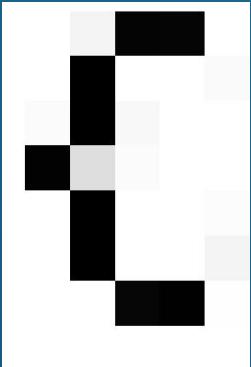
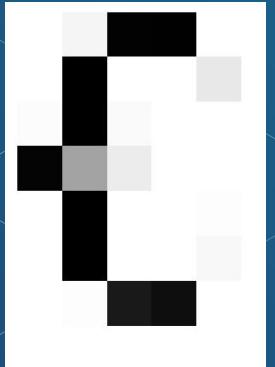
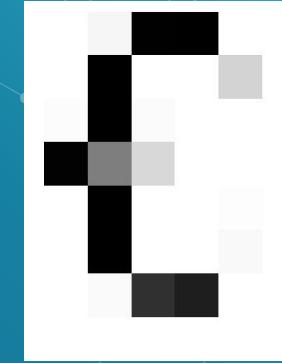
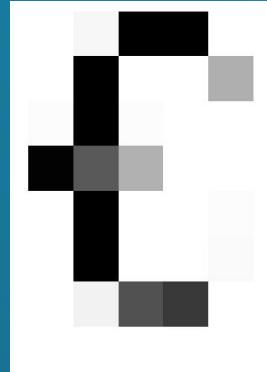
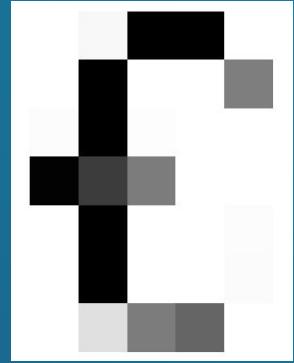
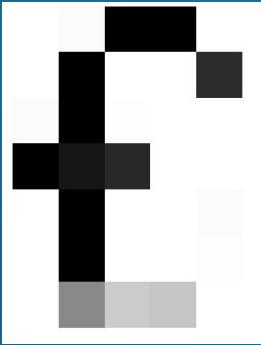
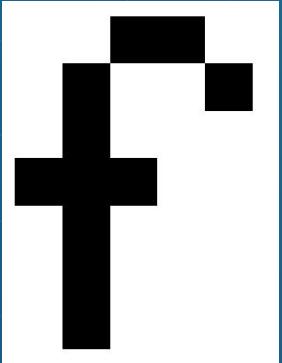
Transiciones

W - U



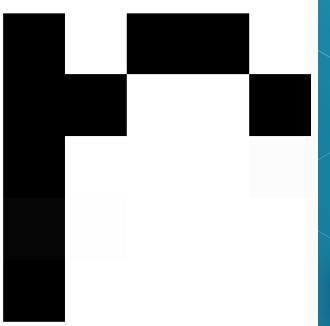
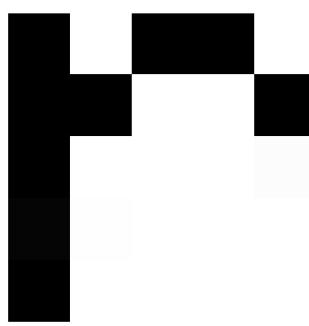
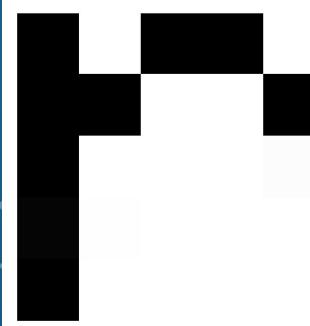
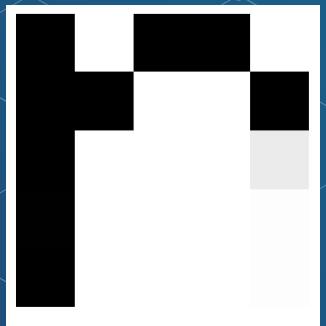
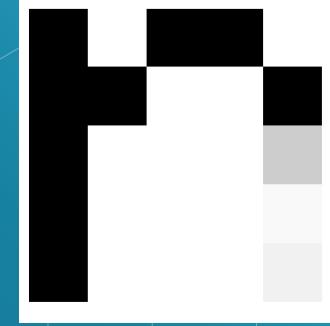
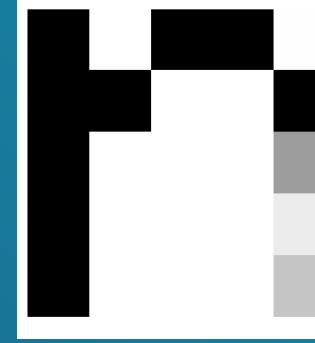
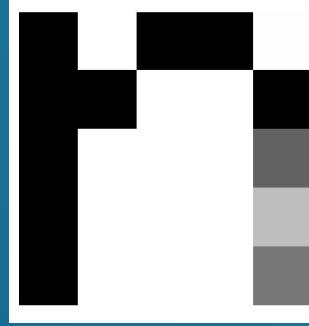
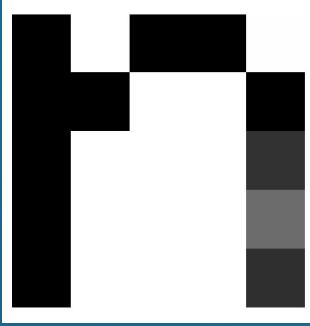
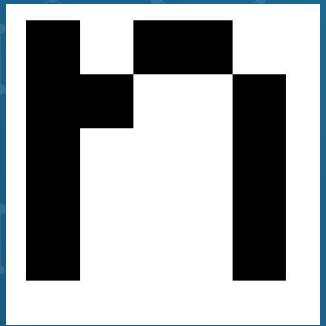
Transiciones

F - {



Transiciones

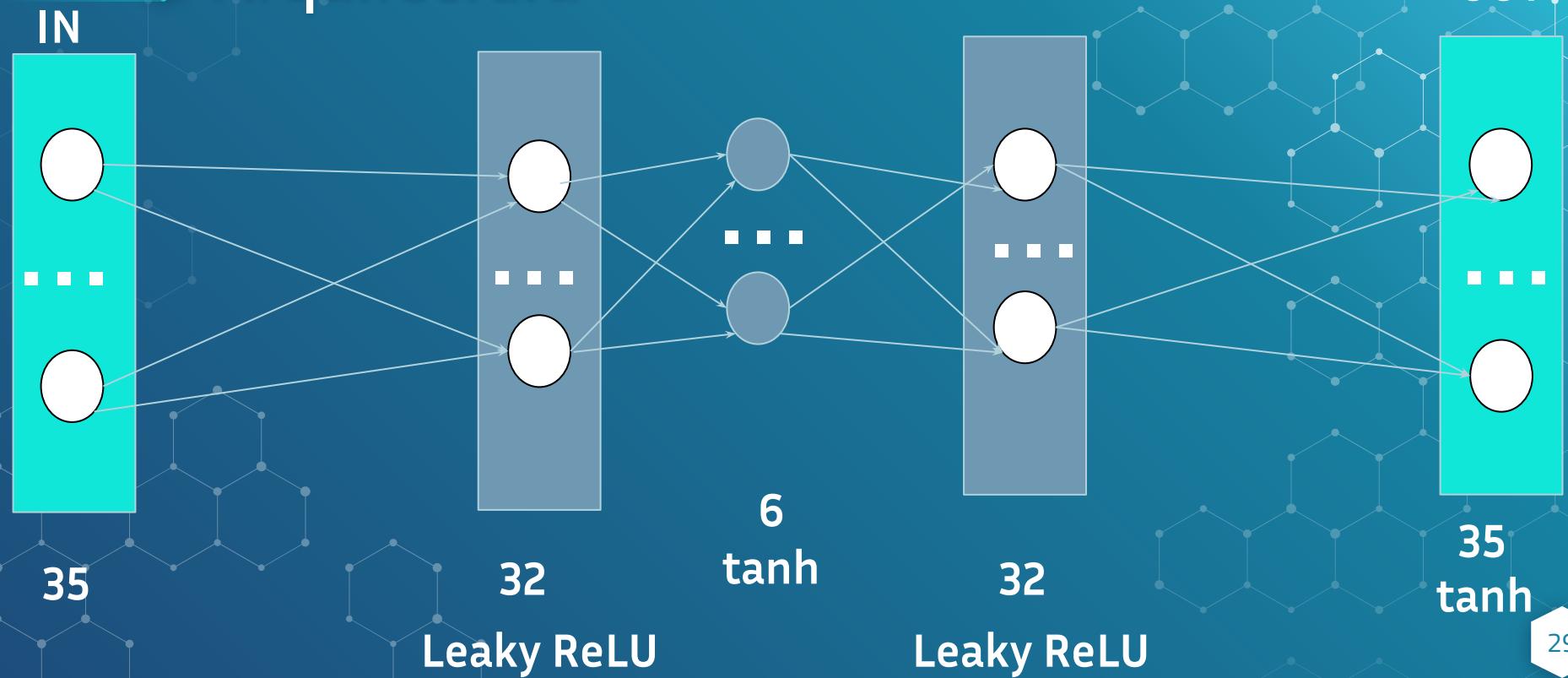
N-R





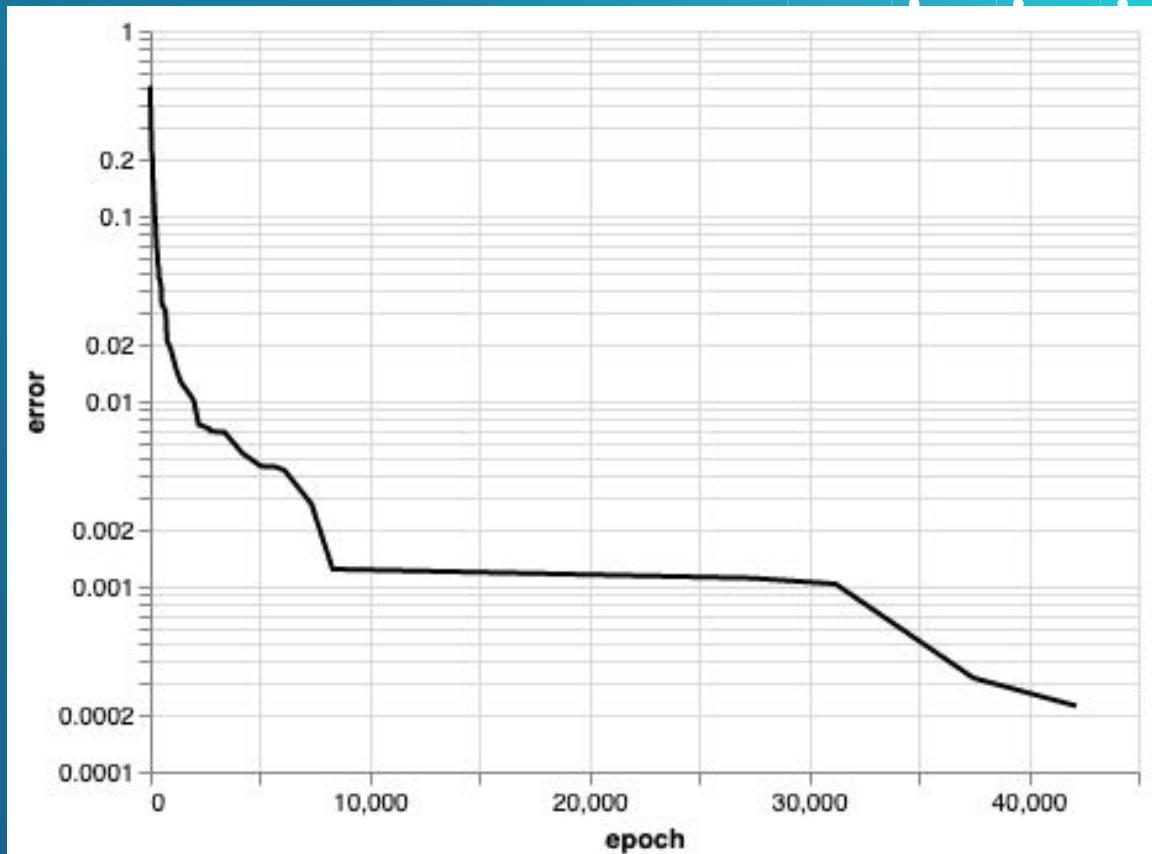
Imágenes Binarias - Denoising

Arquitectura



Evolución del Error

- Puntos donde error es más bajo.
- Batch = 5
- Hasta 50mil epochs.
- Entrenamiento con ruido de :
 - 0,0.05,0.1,0.15,0.2.
 - 64 conjuntos con cada ruido.
- Total $320 * 32$ elementos



Denoising - Sin entrenamiento

10 %

$$E = 1 \times 10^{-3}$$

20 %

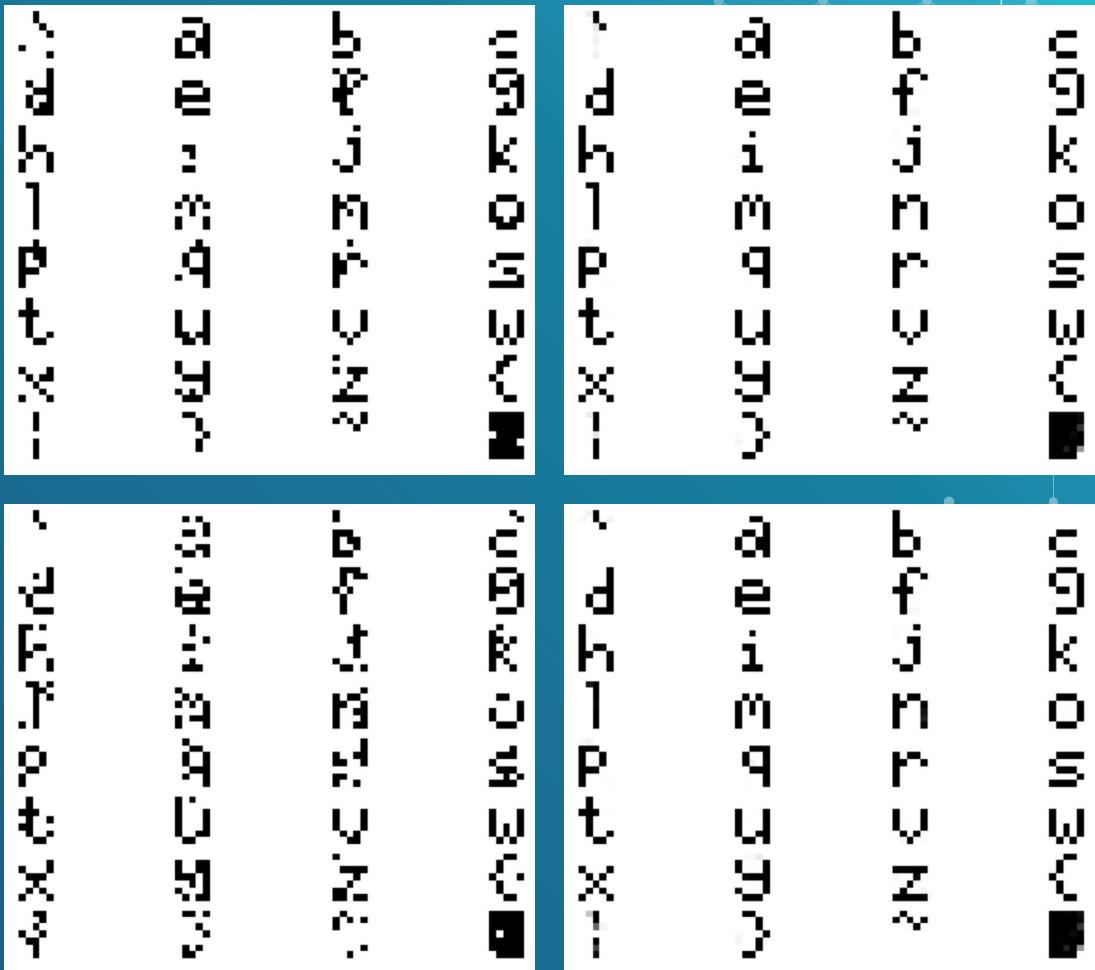
$$E = 2 \times 10^{-3}$$



Denoising

5 %

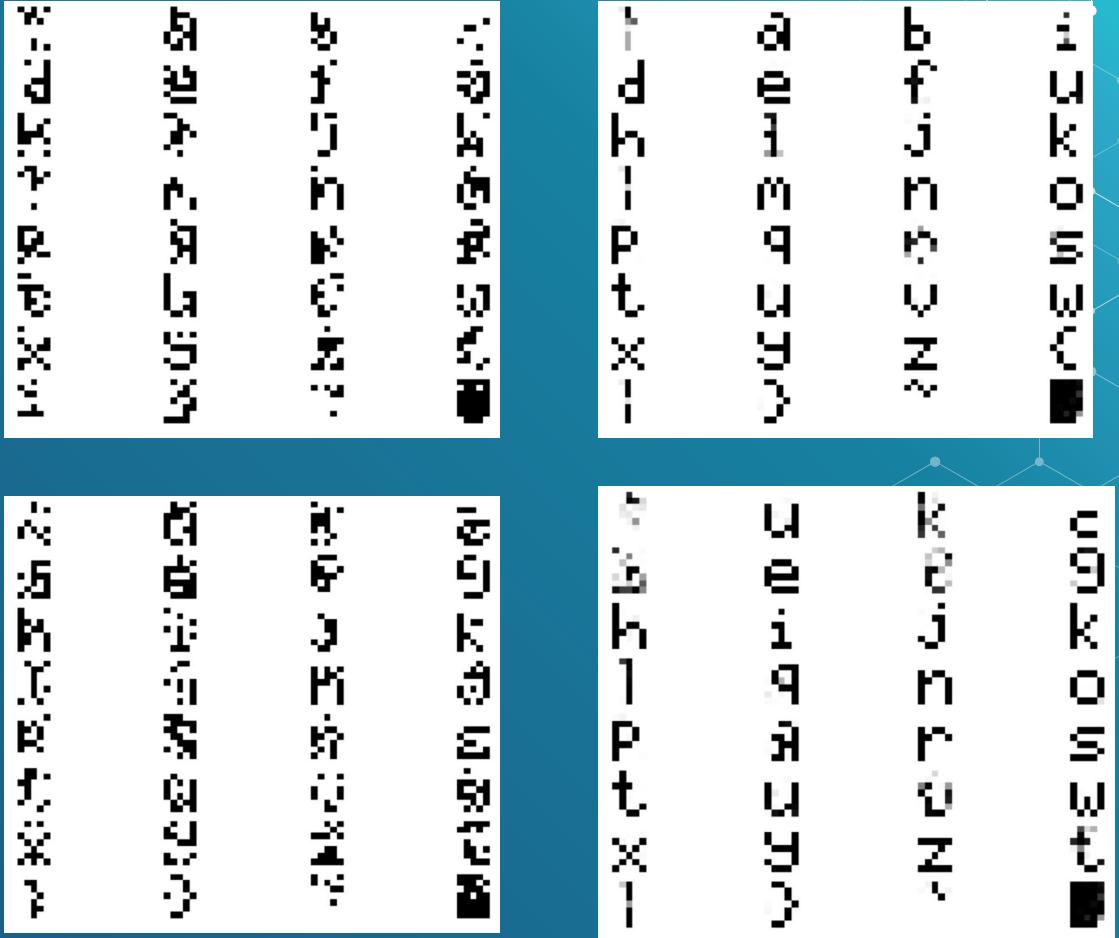
10 %



Denoising

15 %

20 %





3

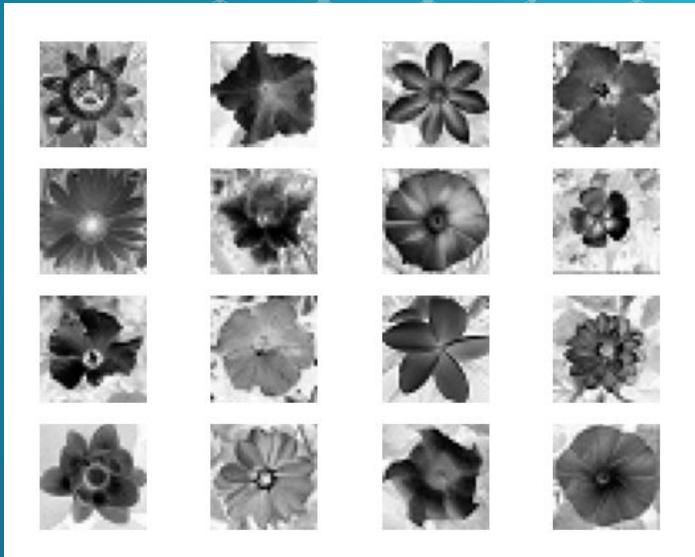
Data Set - Flores

Data Set

Colección de imágenes de flores de la biblioteca de oxford.

Se preseleccionan 24 que tengan una puesta de cámara similar.

Luego se preprocesa cada una para que este 32x32 píxeles en escala de grises normalizada en [-1, 1]

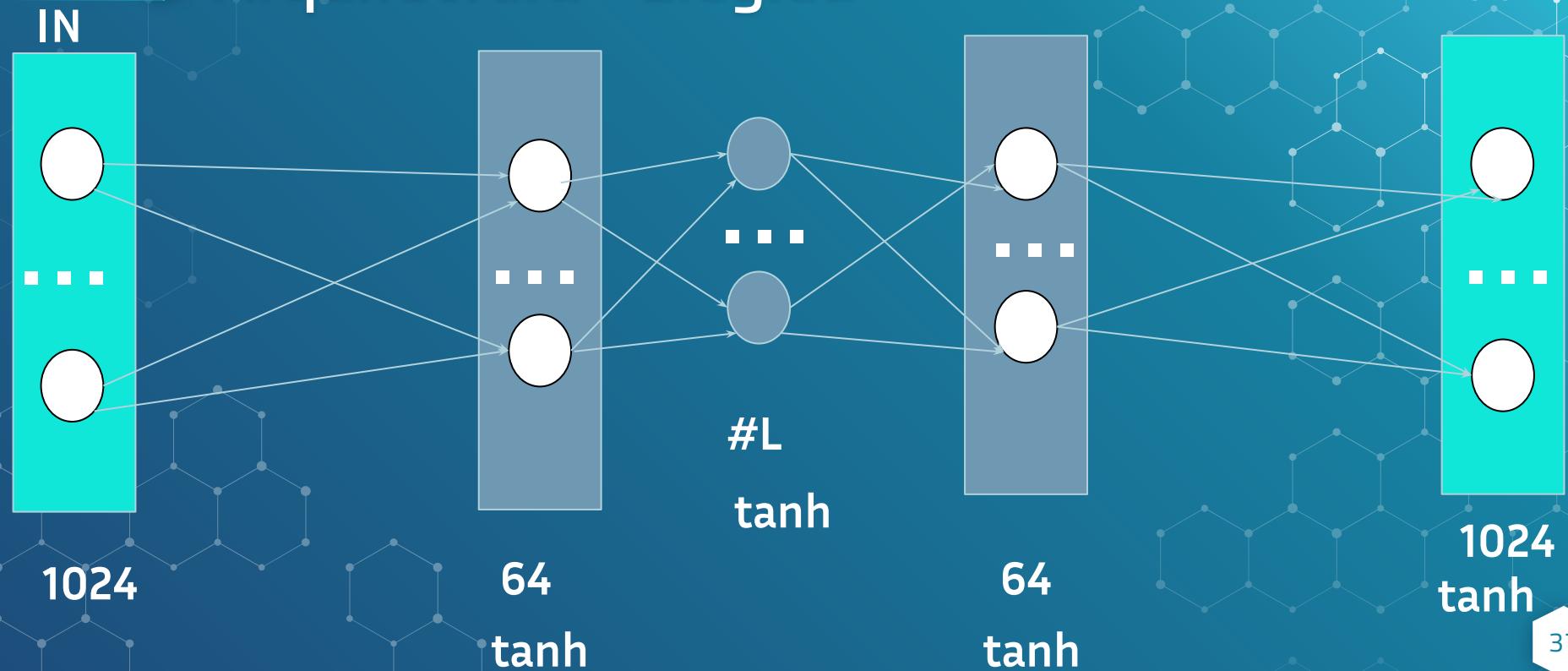


Dataset: <http://www.robots.ox.ac.uk/~vgg/data/flowers/102/index.html>

Entrenamiento:

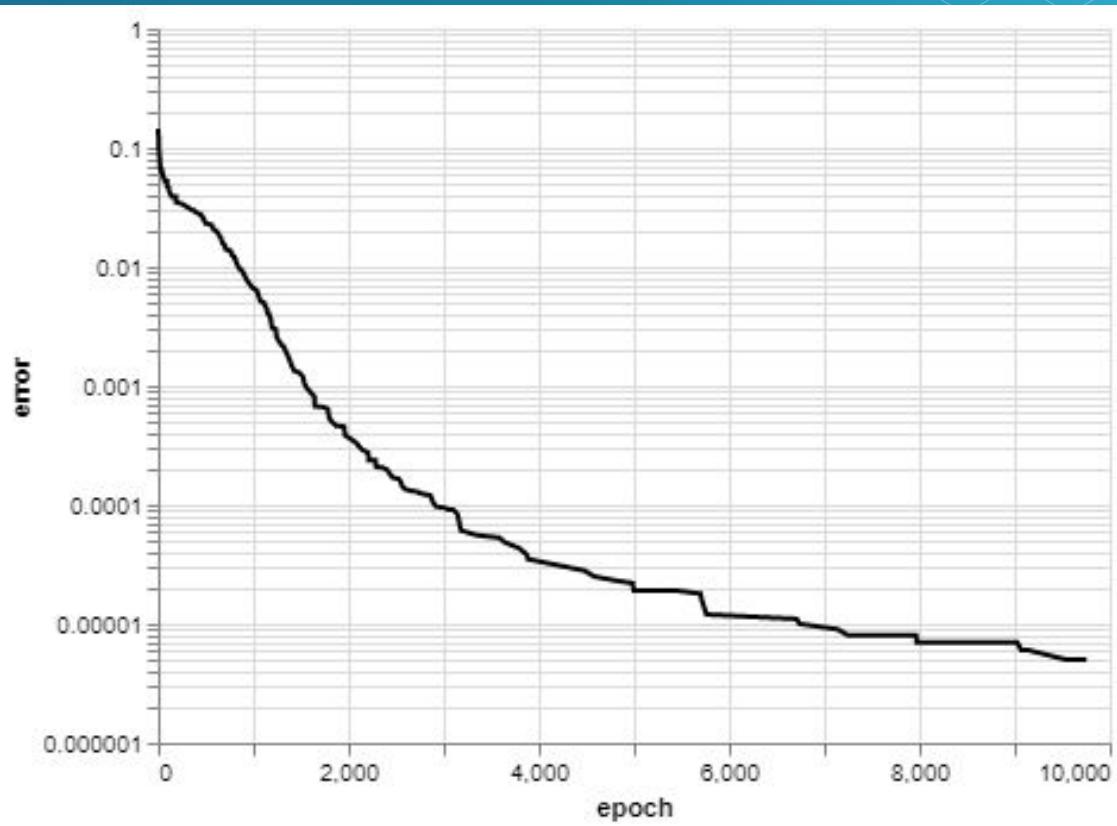
Buscamos la arquitectura óptima para conseguir el valor de dimensionalidad para el espacio latente más bajo en el que pueda aprender

Arquitectura - Elegida



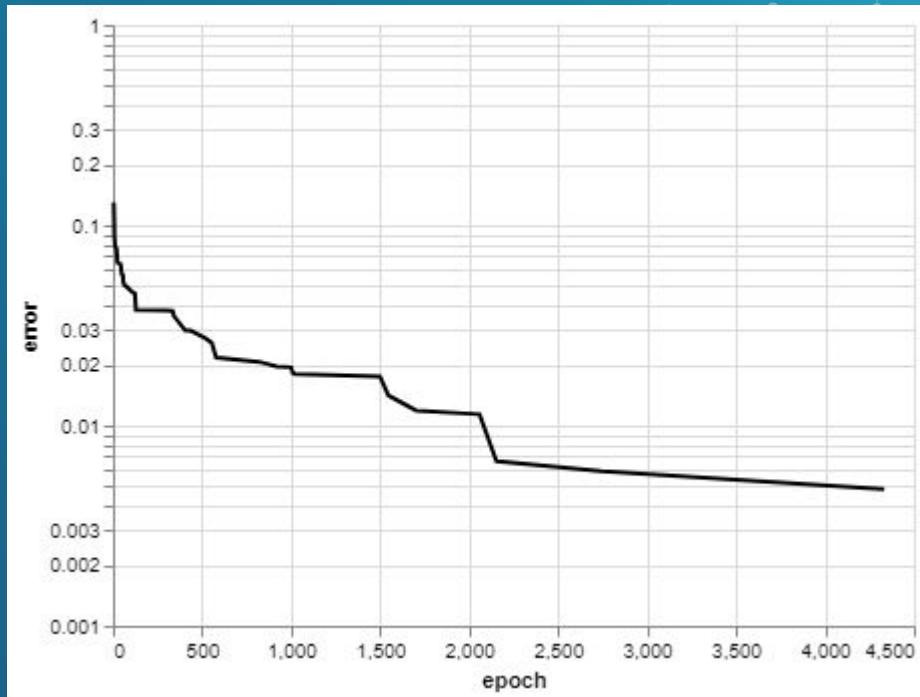
Evolución del Error

- $\#L = 6$
- Estable a partir de 10 mil epochs.
- Batch = 20

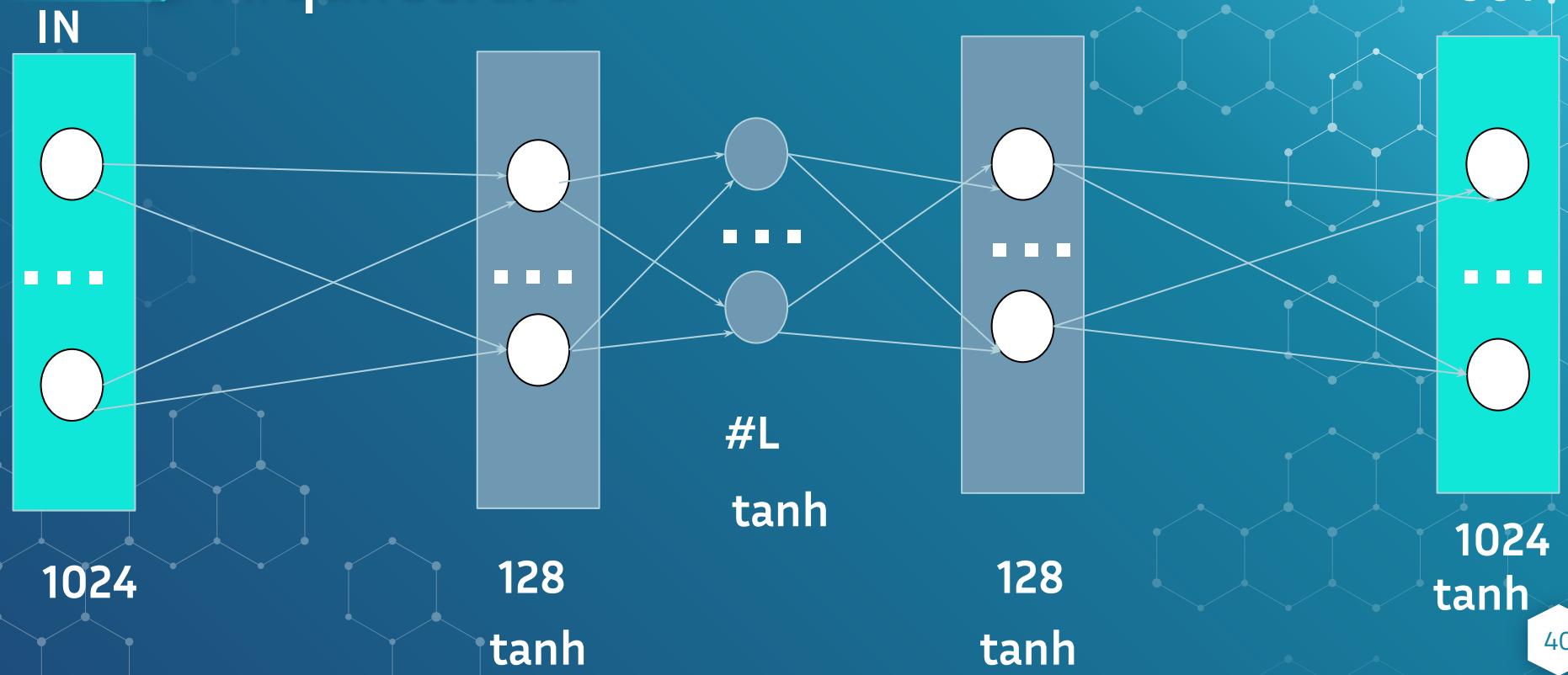


Evolución del Error

- $\#L = 5$
- Al disminuir el espacio latente el error crece.
- Batch = 20

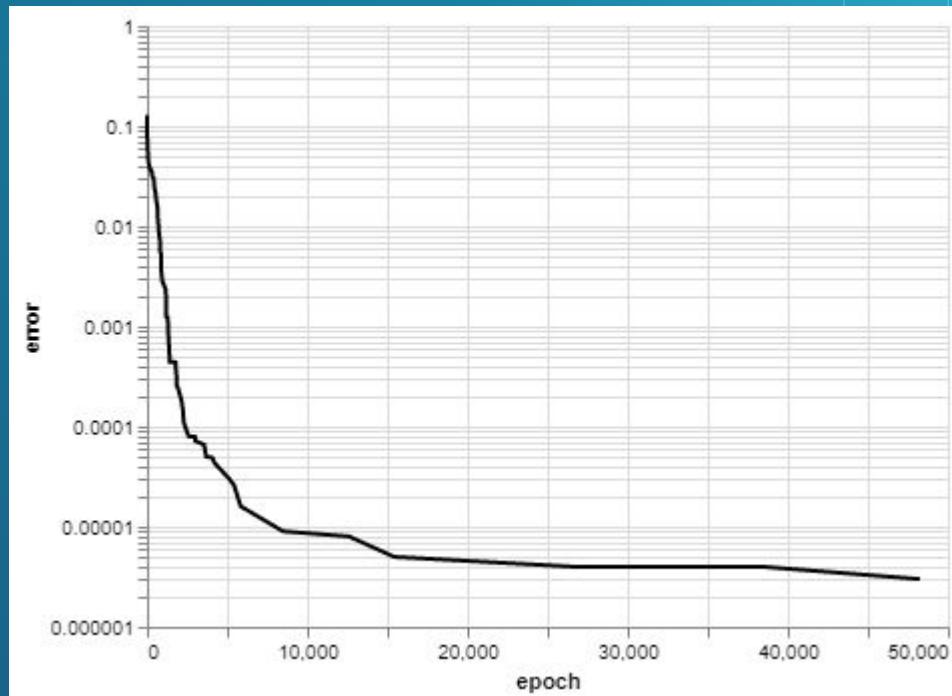


Arquitectura

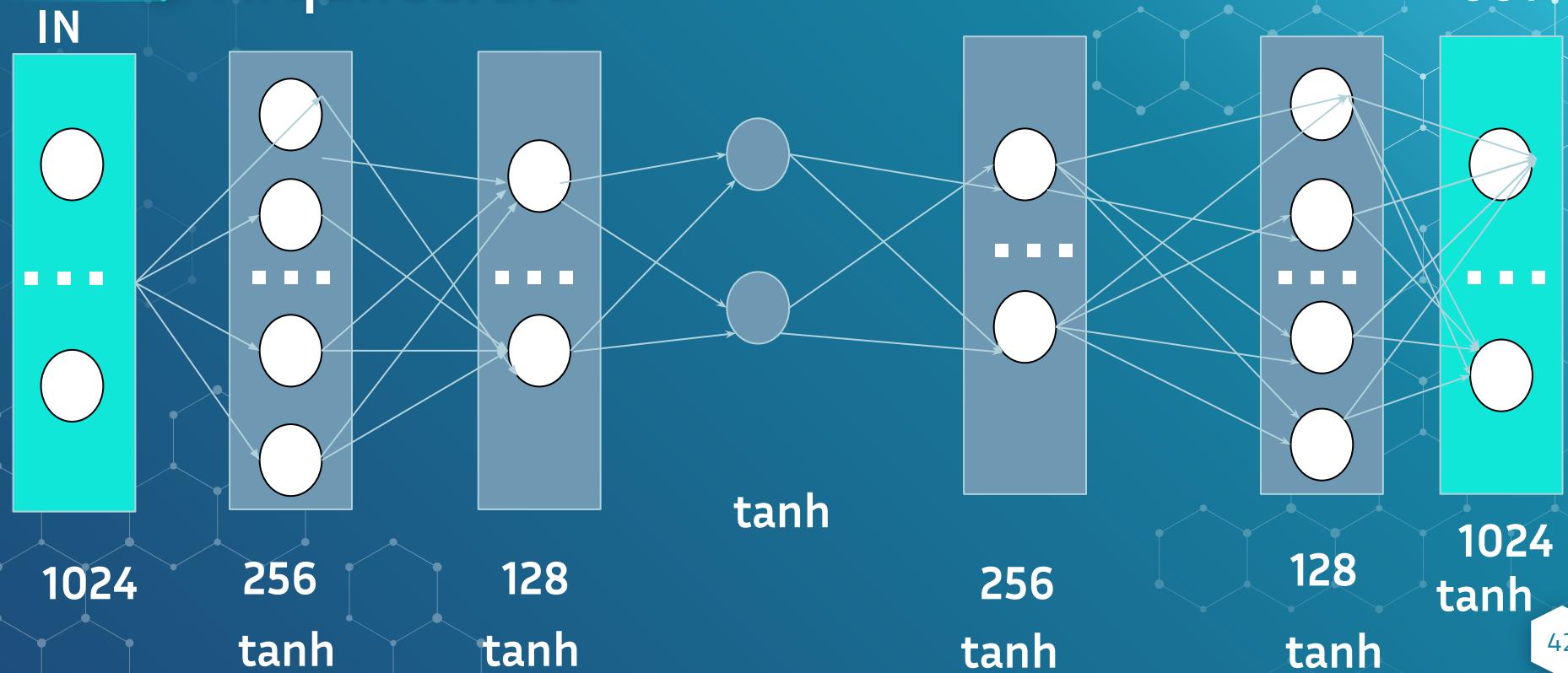


Evolución del Error

- $\#L = 6$
- Estable a luego de 50 mil epochs.
- Batch = 20

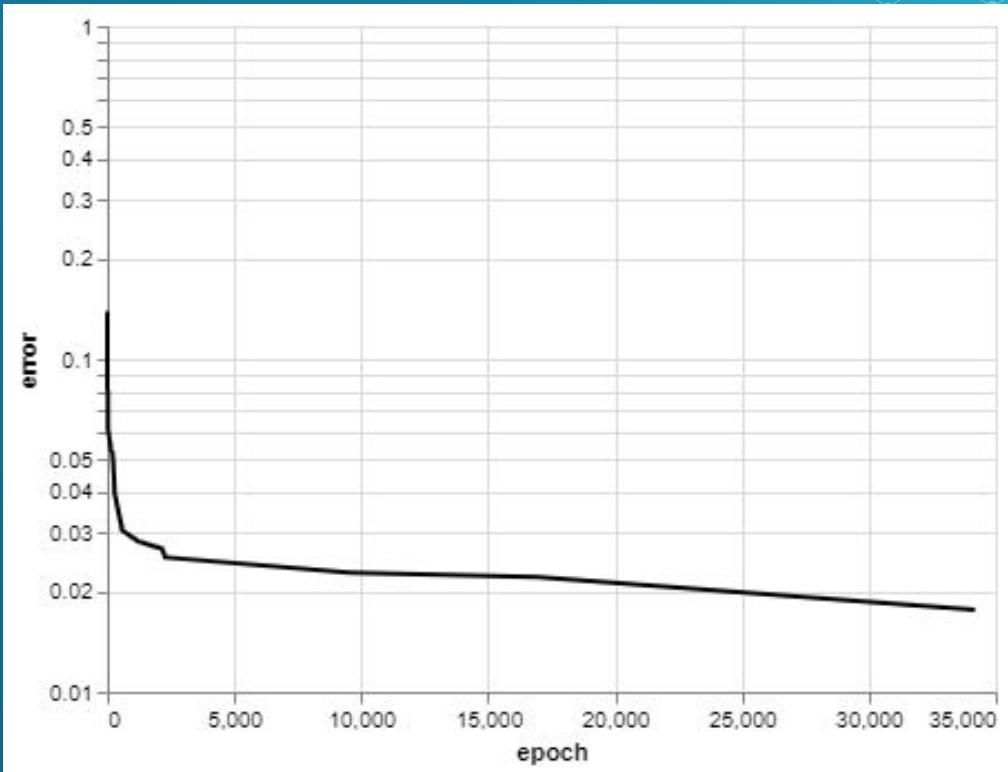


Arquitectura

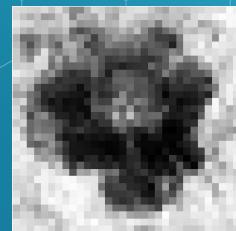
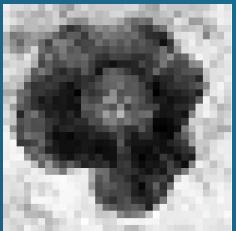
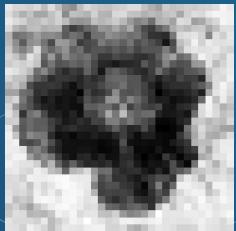
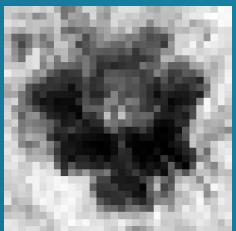
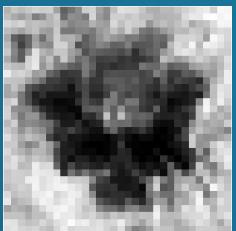
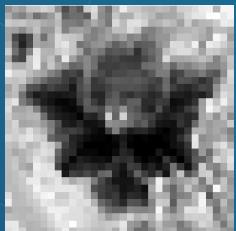


Evolución del Error

- $\#L = 6$
- Valor de pérdida más elevado respecto a los escenarios anteriores.
- Batch = 20



Buscando flores en el espacio latente



Navegación del espacio latente

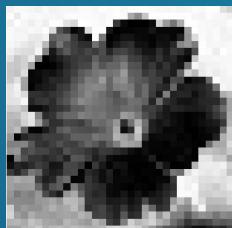
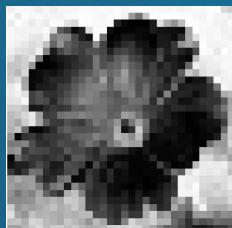
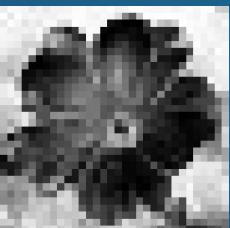


Transiciones

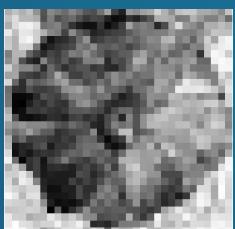
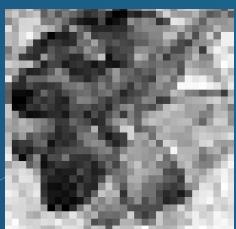
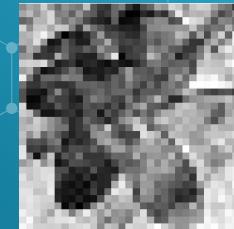
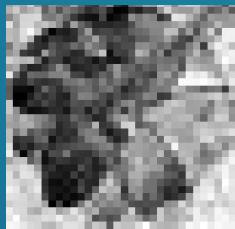
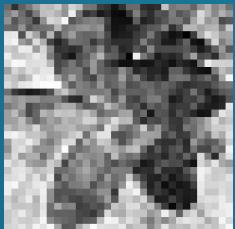
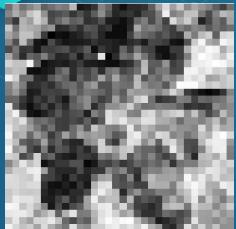
Solo fueron variadas las primeras 2 componentes en el espacio latente.

Para realizar las transiciones en el espacio latente, se realizaron variaciones en las primeras 2 coordenadas solamente.

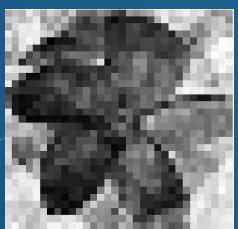
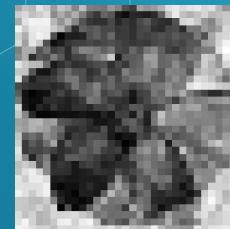
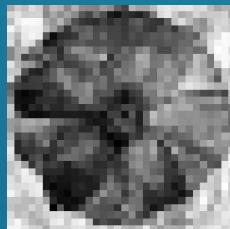
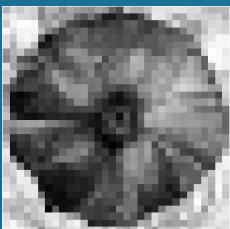
Buscando flores en el espacio latente



Buscando flores en el espacio latente



Buscando flores en el espacio latente



Conclusiones

Ejercicio 1:

- Existen letras que admiten una representación de mayor área en espacio latente. (~atractoras)
- Las letras que están más cerca en espacio latente son aquellas más confundidas en presencia de ruidos o alteraciones.

Ejercicio 2:

- Gracias a la homogeneidad de las imágenes la red puede aprender relativamente bien las imágenes
- Debido a distinta naturaleza de los datos respecto al ejercicio 1 se debe diseñar una nueva arquitectura

Reducir la dimensión del espacio latente manteniendo el error implica una complejización de la red.

4

Gracias!