

AUTOENCODERS

TABLE OF CONTENTS

01

Autoencoder

02

Denoiser

03

VAE

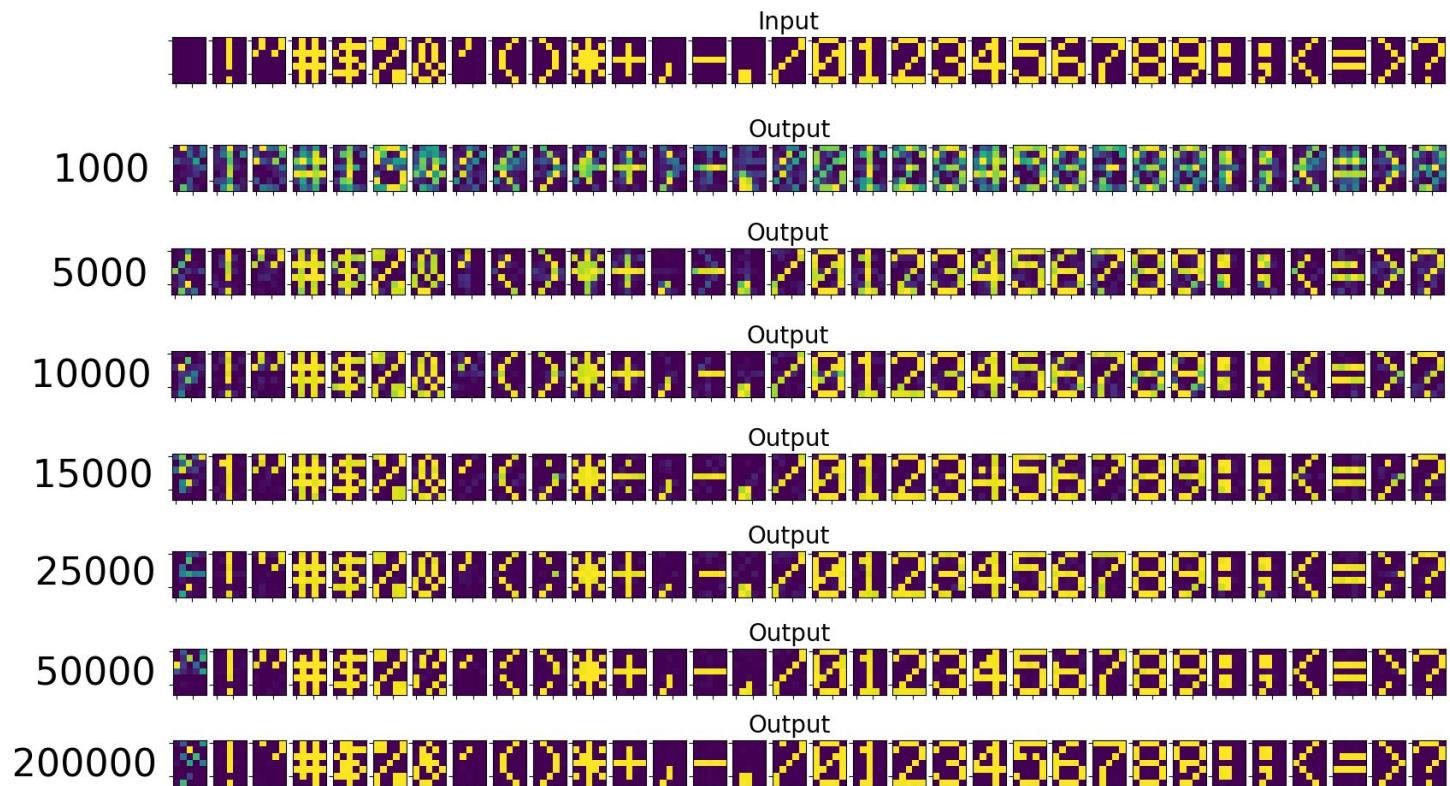
04

CONCLUSION

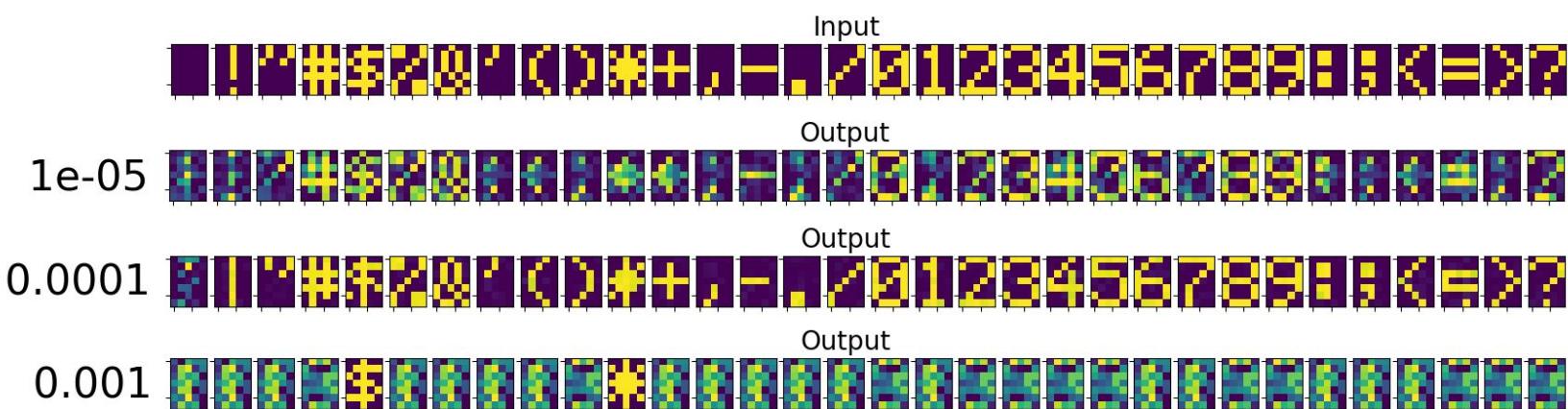


Autoencoder

Variando la Cantidad de Epochs



Variando el Learning Level



Variando la Estructura

Epochs: 25000 - Learning Rate: 0.0001

Input



[30:25:20:15:10:5:2:5:10:15:20:25:30]



[20:15:2:15:20]



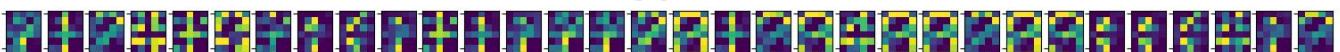
[20:2:20]



[10:2:10]



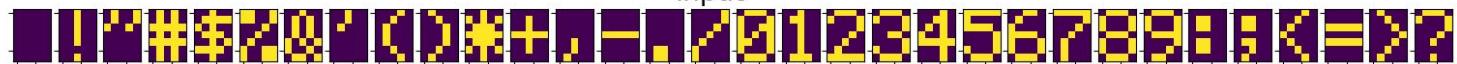
[2]



Con Estructuras Asimétricas

Epochs: 25000 - Learning Rate: 0.0001

Input



[30:25:20:15:10:5:2:10:20:30]

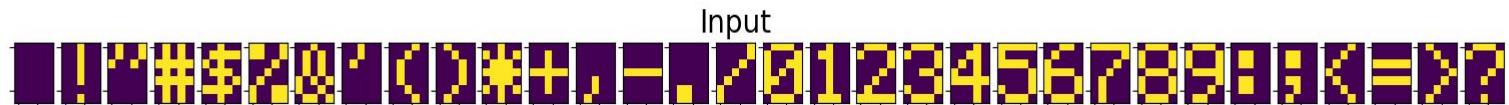


[30:15:2:7:21]

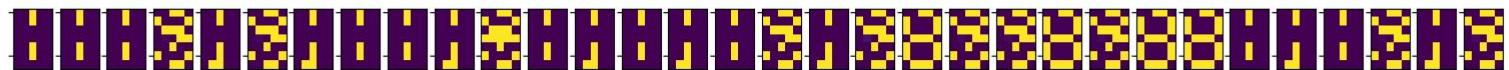


Variando las Funciones de Activación

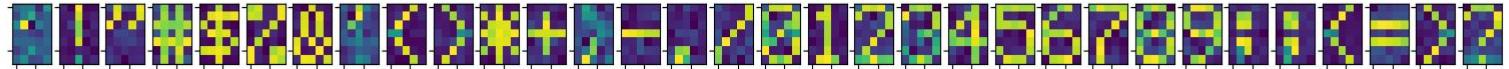
Epocts: 25000 - Learning Rate: 0.0001



20(linear) -> 15(relu) -> 10(relu) -> 2(tanh) -> 10(relu) -> 15(relu) -> 20(linear) -> 35(tanh)

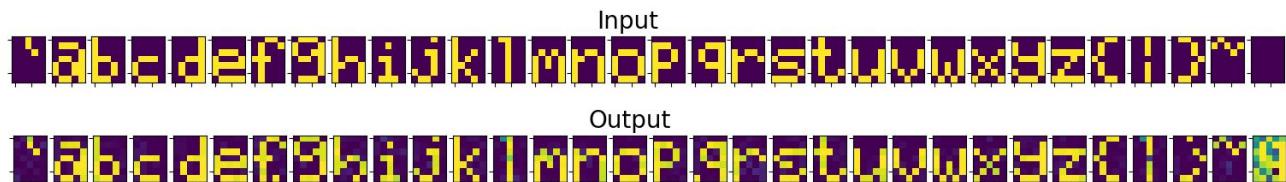
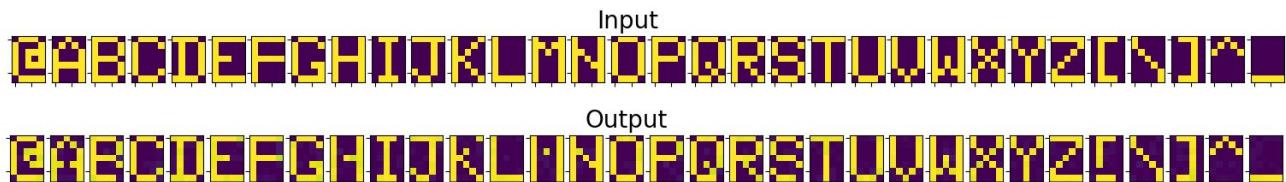
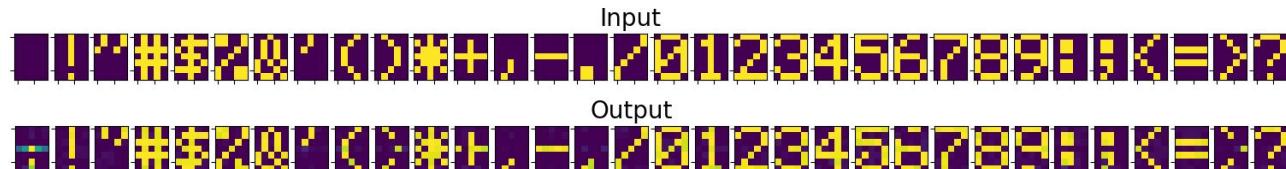


20(linear) -> 15(tanh) -> 10(tanh) -> 2(tanh) -> 10(tanh) -> 15(tanh) -> 20(tanh) -> 35(linear)



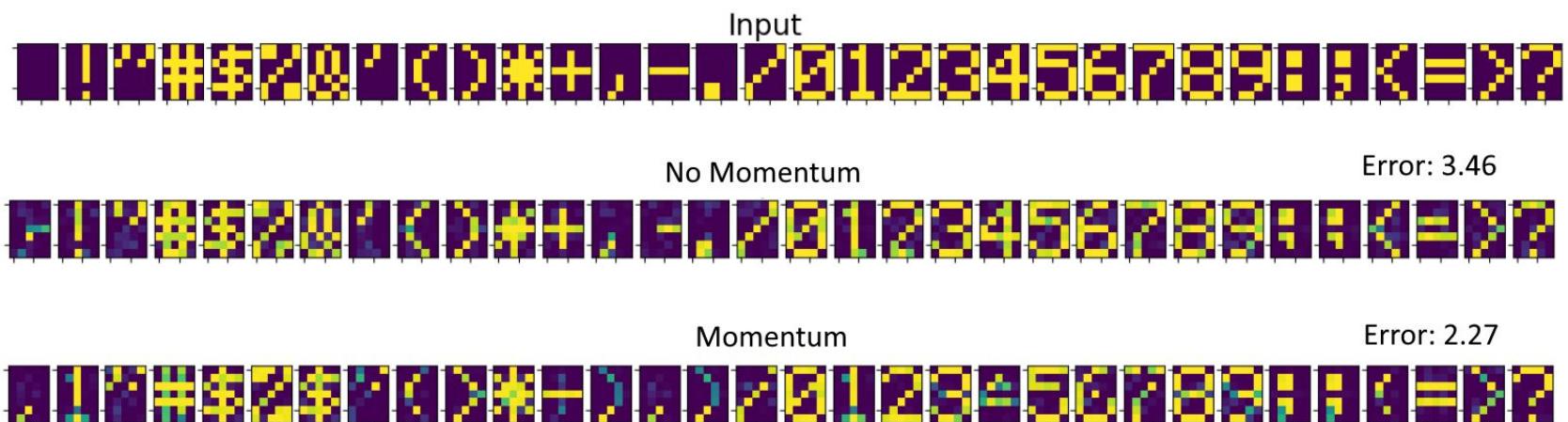
Con las Distintas Fonts

Epcos: 25000 - Learning Rate: 0.0001



Momentum

Epochs: 10000 - Learning Rate: 0.0001 - Structure: [20:10:2:10:20]



Learning Rate Adaptativo

$$LRn = LR0 \cdot e^{-\frac{n}{d}}$$

$$-\frac{\left(\frac{epochs}{limit}\right)}{d}$$

$$finalLR = LR0 \cdot e$$

LR0: learning rate inicial

Final LR: learning rate final

Epochs : cantidad de epochs totales

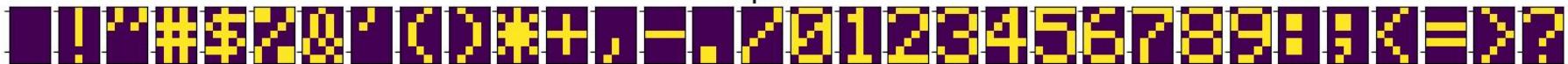
Epochs/limit : en qué epoch alcanzar el LR final

d: incognita

Learning Rate Adaptativo

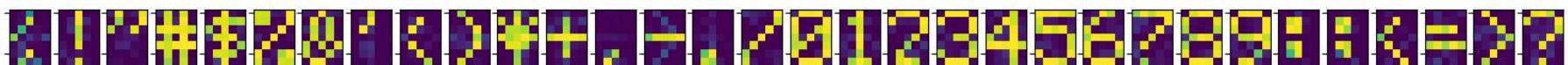
Epcos: 5000

Input



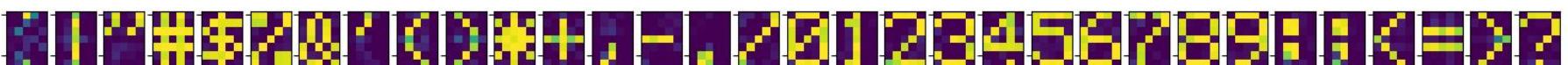
Fixed Learning Rate (0.0001)

Error: 4.83



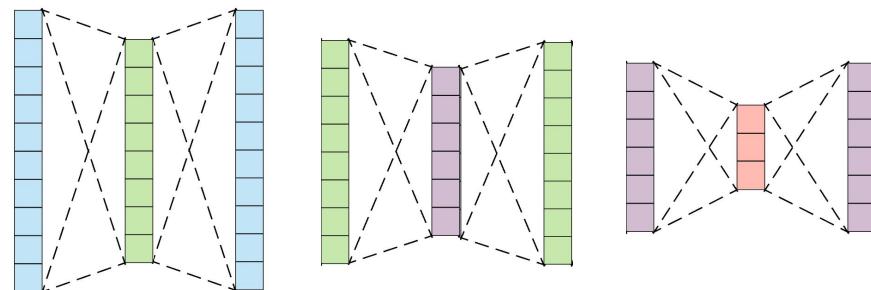
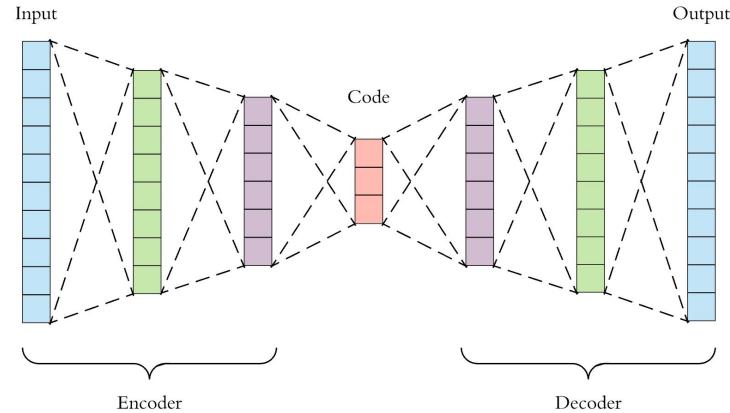
Adaptative Learning Rate (0.001 -> 0.0001)

Error: 2.42



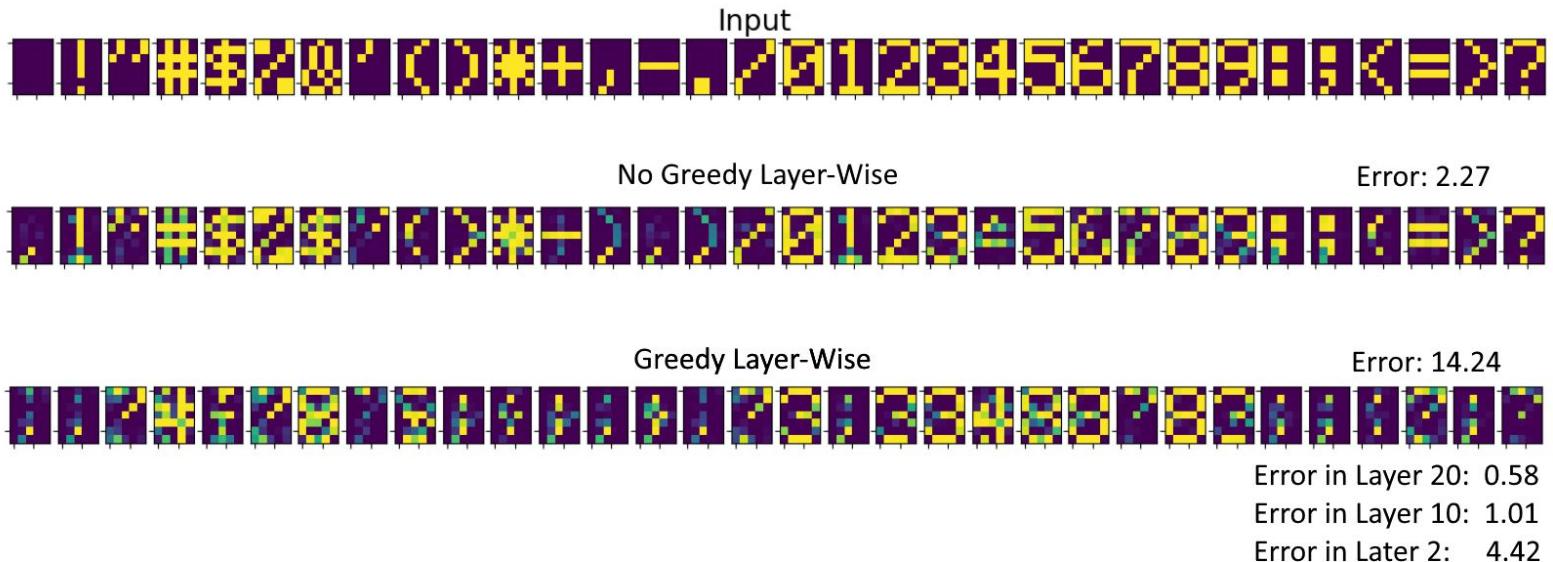
Greedy Layer-Wise

- Se entrena las capas de color juntas
- Se busca minimizar la pérdida de información de dimensión a dimensión
- Sin embargo, no significa que los resultados para transformar a 2 dimensiones sean mejores

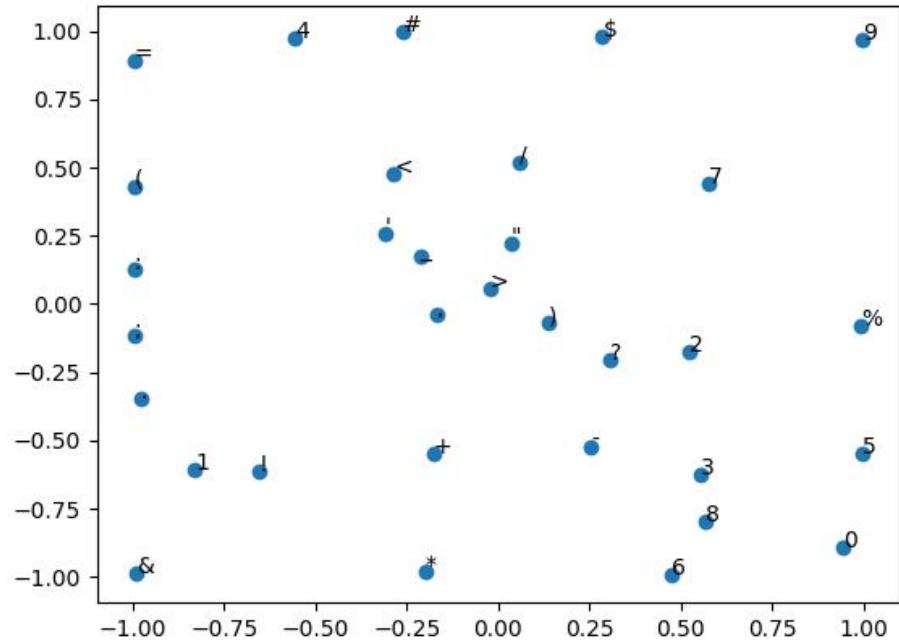


Greedy Layer-Wise

Epochs: 10000 - Learning Rate: 0.0001 - Structure: [20:10:2:10:20]

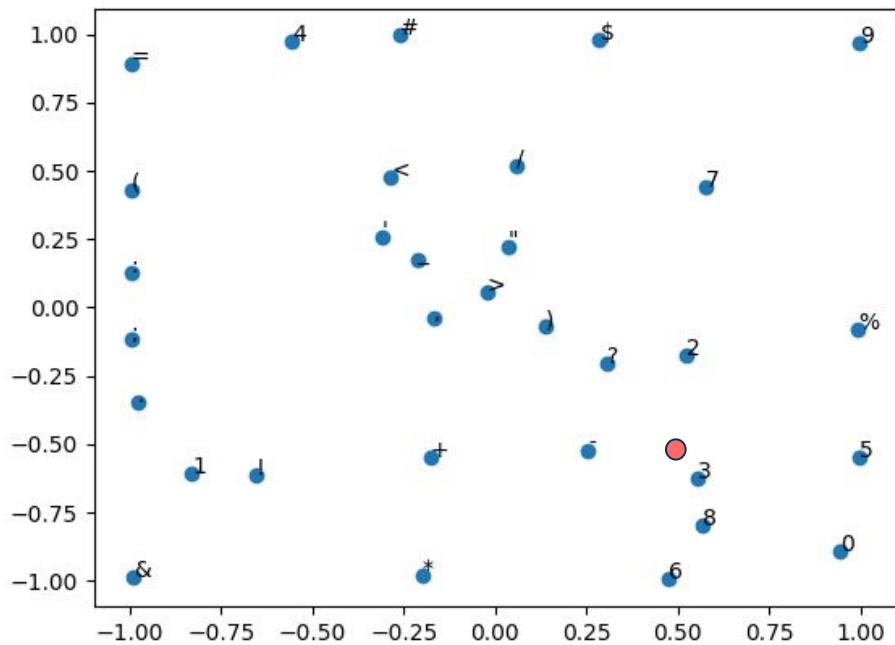
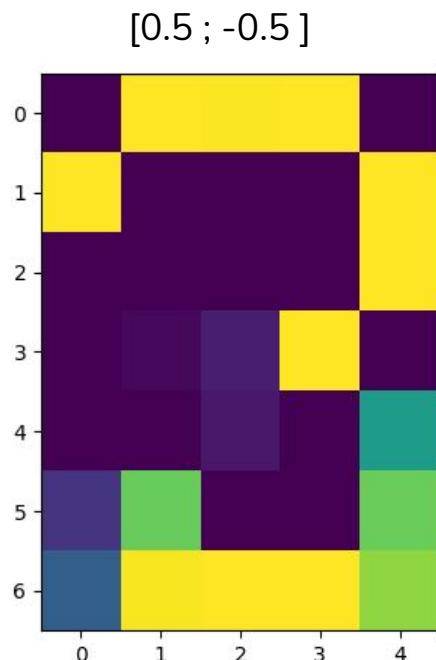


El Espacio Latente

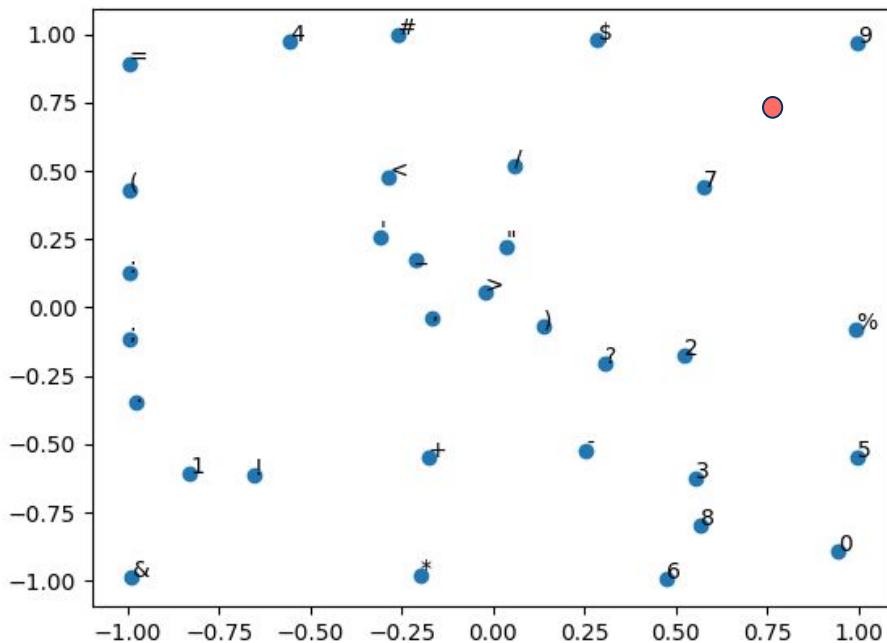
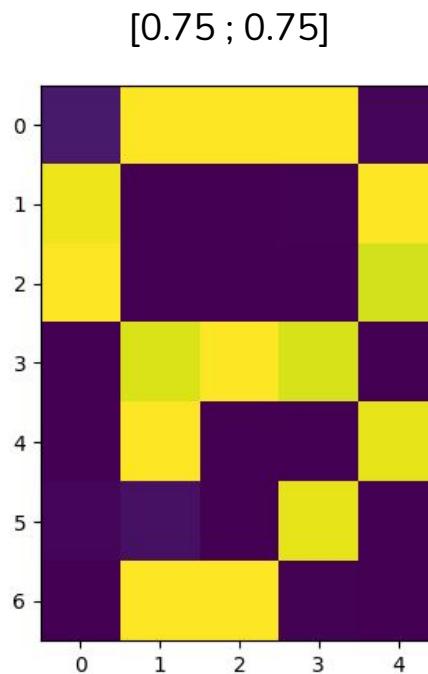


*Gráfico realizado con 200000 epochs y learning level 0.0001

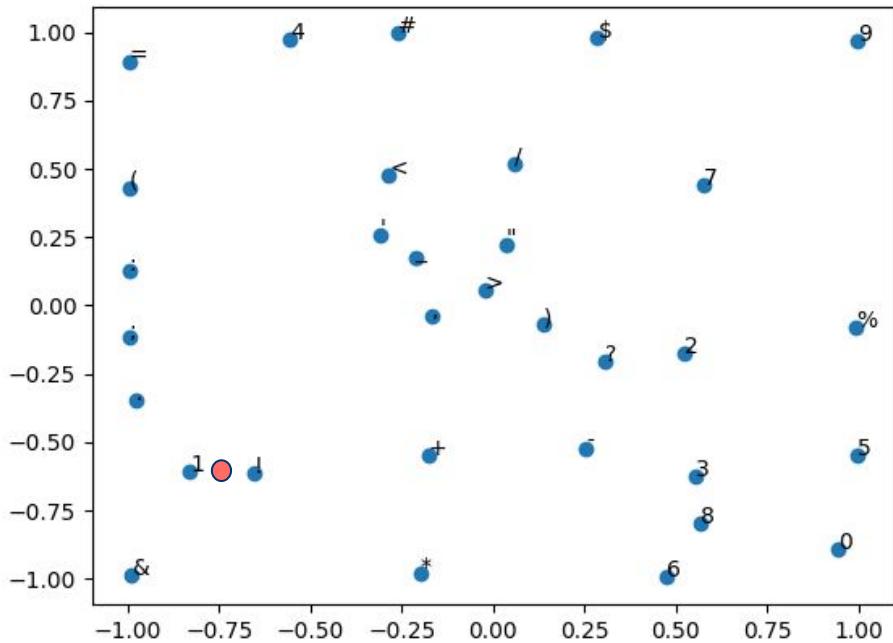
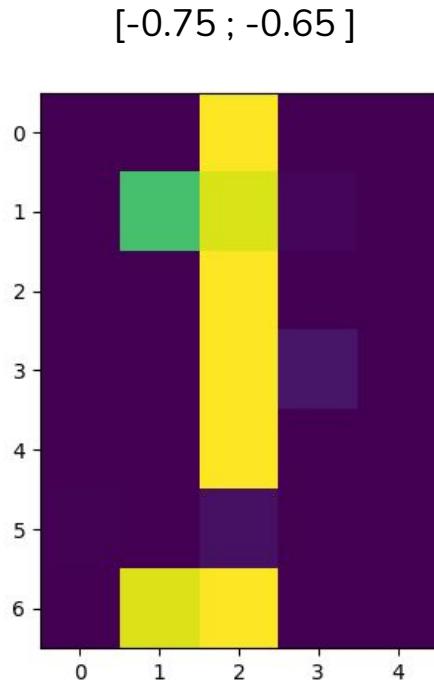
Un Nuevo Símbolo



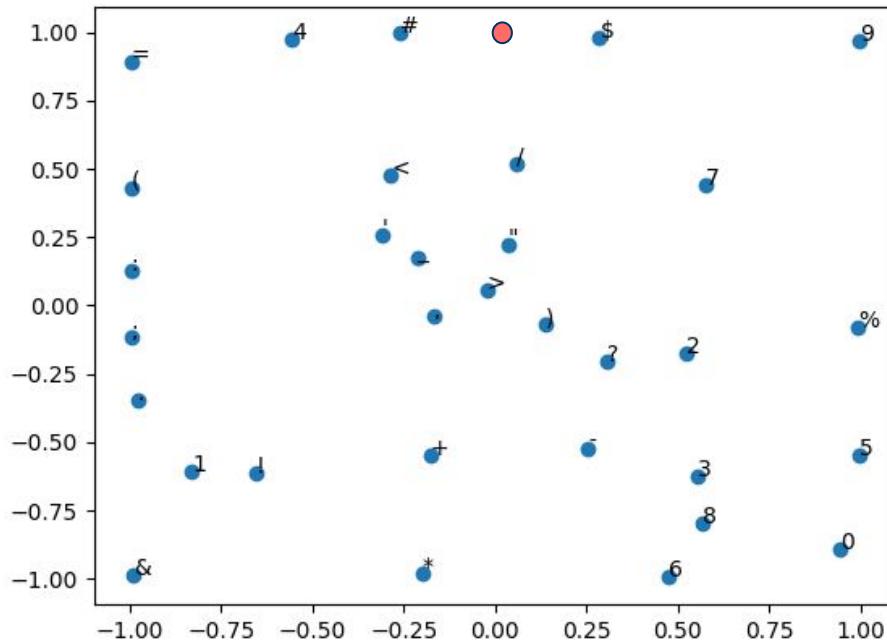
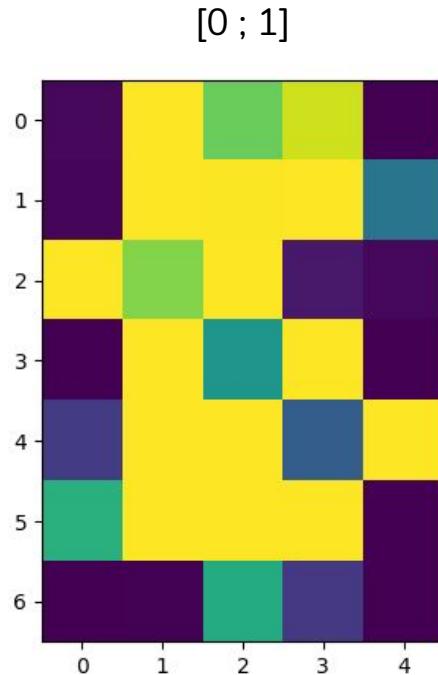
Un Nuevo Símbolo



Un Nuevo Símbolo



Un Nuevo Símbolo



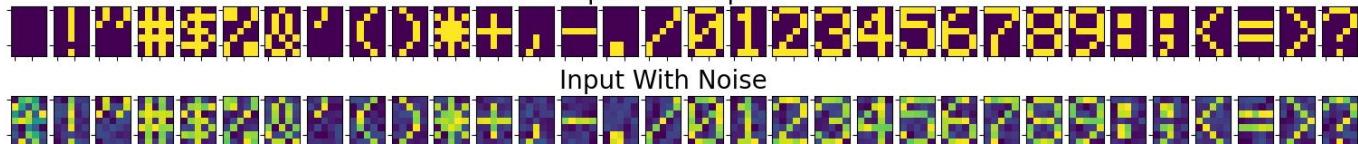


Denoiser

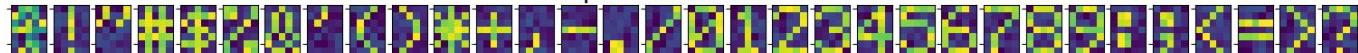
Variando la Estructura

Epochs: 25000 - Learning Rate: 0.0001

Expected Output



Input With Noise

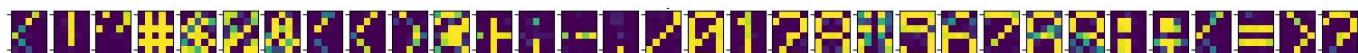


Outputs:

20 -> 15 -> 10 -> 2 -> 10 -> 15 -> 20



20 -> 15 -> 10 -> 15 -> 20



20 -> 15 -> 20



20



Variando el Ruido

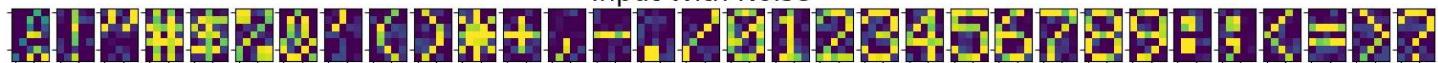
Epochs: 25000 - Learning Rate: 0.0001

Expected Output



Noise Coverage:0.5 Noise Pct: 0.5

Input With Noise

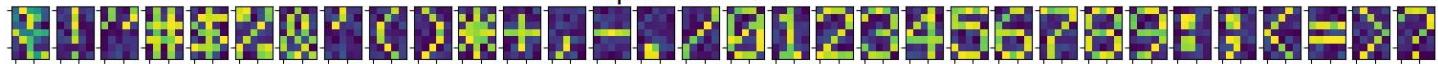


Output



Noise Coverage: 1 Noise Pct: 0.5

Input With Noise

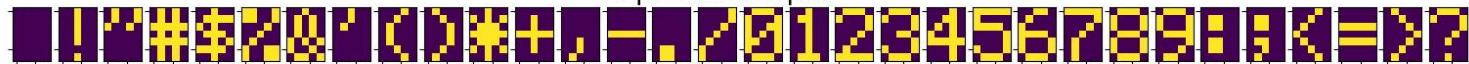


Output

Variando el Ruido

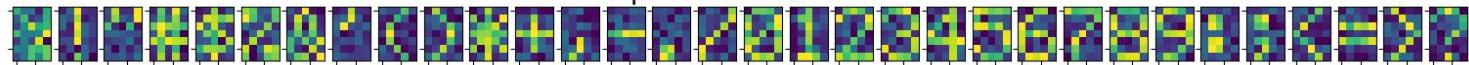
Epcos: 25000 - Learning Rate: 0.0001

Expected Output

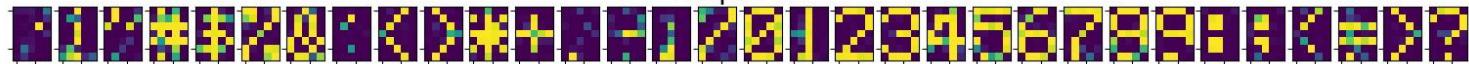


Noise Coverage: 1 Noise Pct: 0.75

Input With Noise

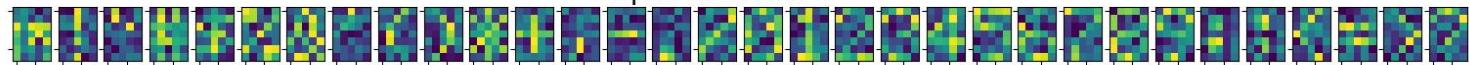


Output

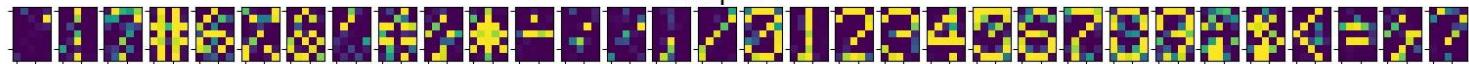


Noise Coverage: 1 Noise Pct: 1

Input With Noise



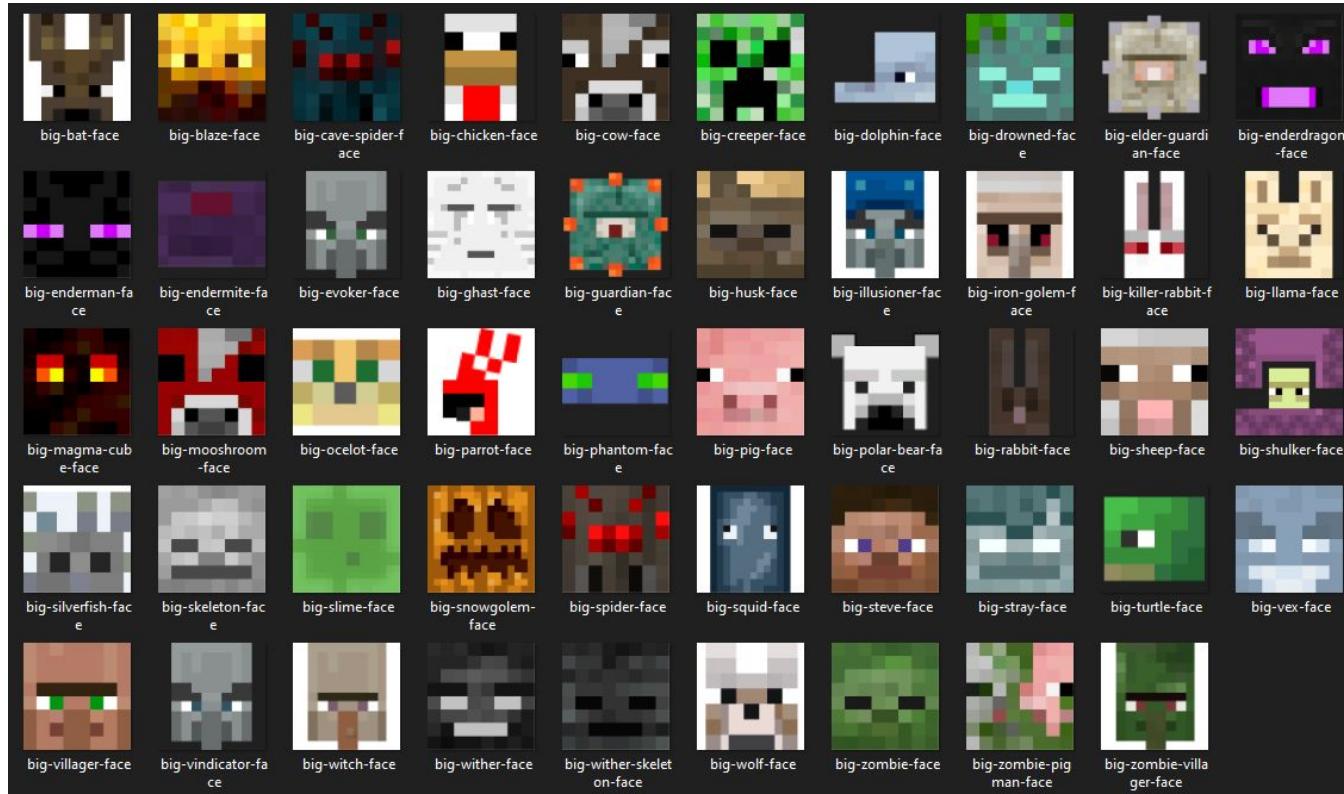
Output



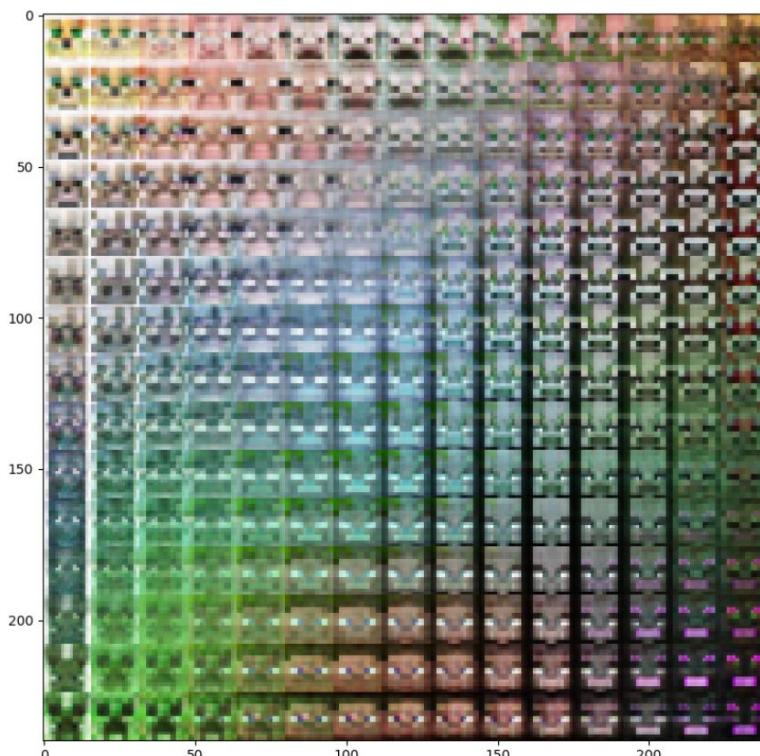
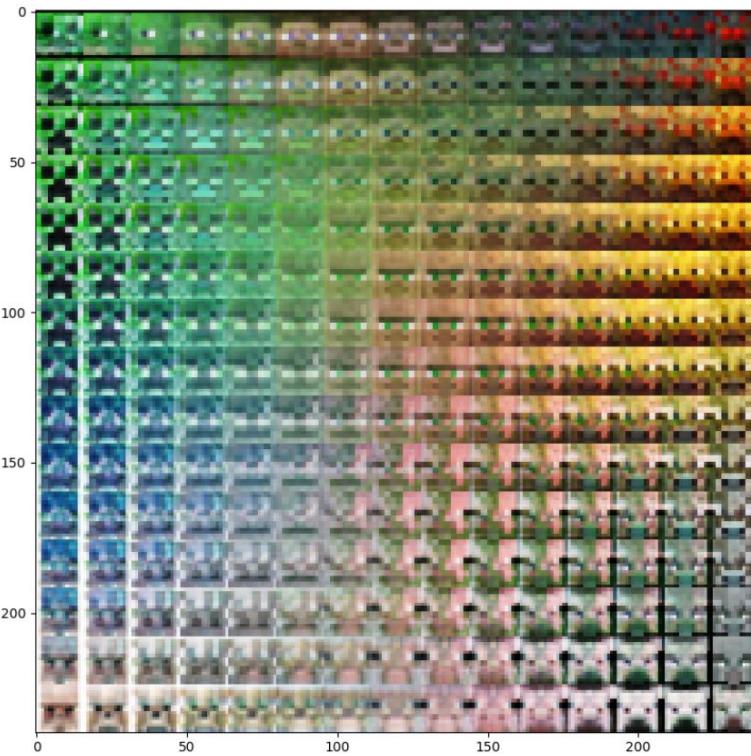


VAE

VAE con Caras de Minecraft



VAE con Caras de Minecraft



VAE con Caras de Minecraft



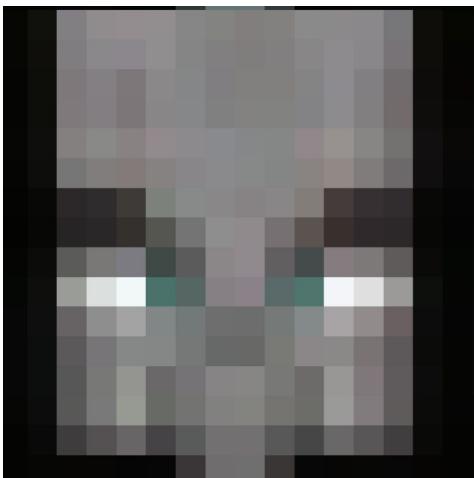
+



=



VAE con Caras de Minecraft



+



=



VAE con Caras de Celebridades



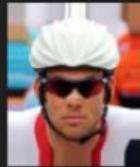
000332



000333



000334



000335



000336



000337



000338



000339



000340



000341



000342



000343



000345



000346



000348



000349



000350



000351



000352



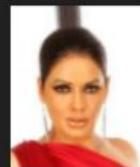
000353



000354



000355



000356



000357



000358

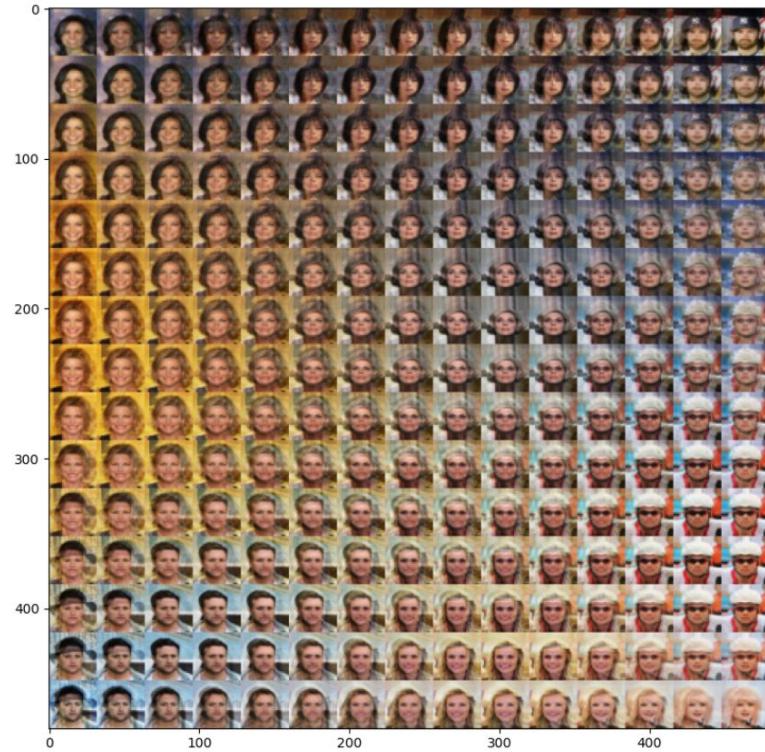
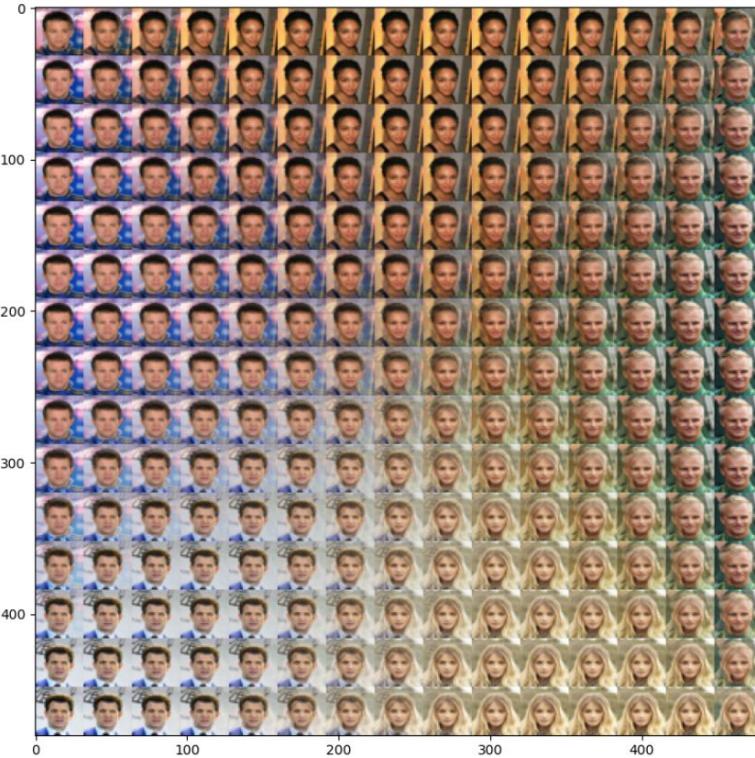


000359



000360

VAE con Caras de Celebridades



VAE con Caras de Desarrolladores



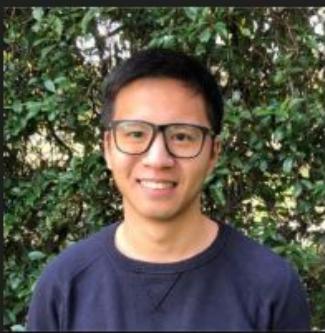
bailevich



bmaiges



cnomerci

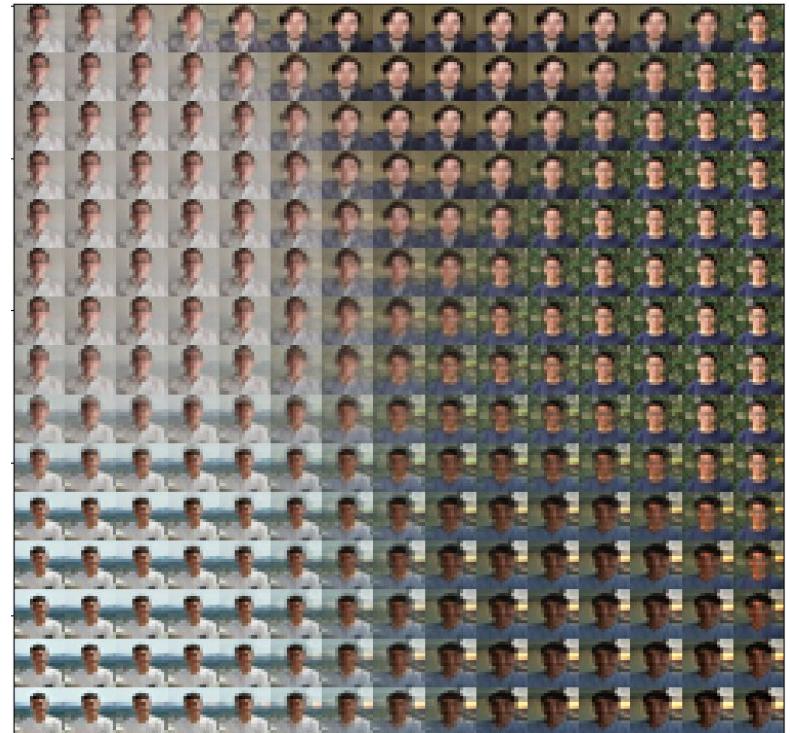
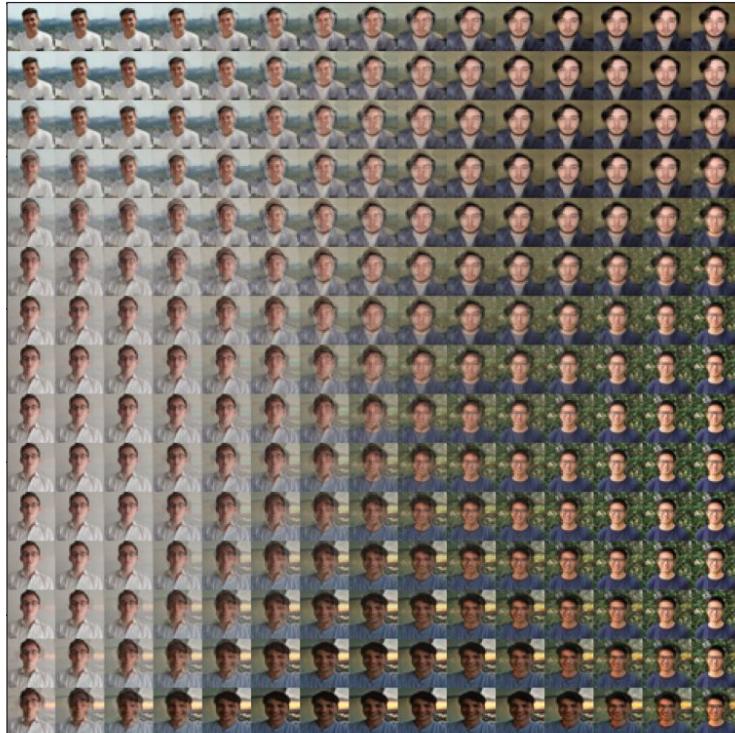


lsinn



mgargossian

VAE con Caras de Desarrolladores



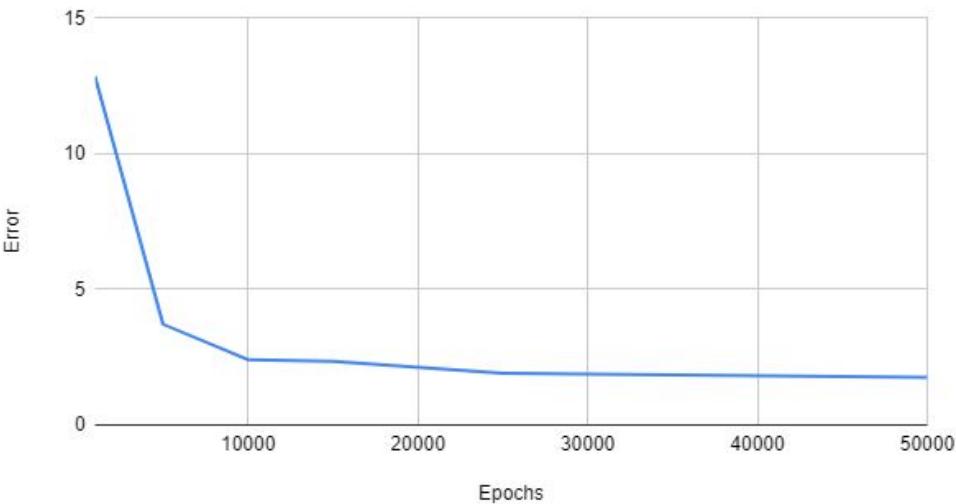
04

Conclusiones

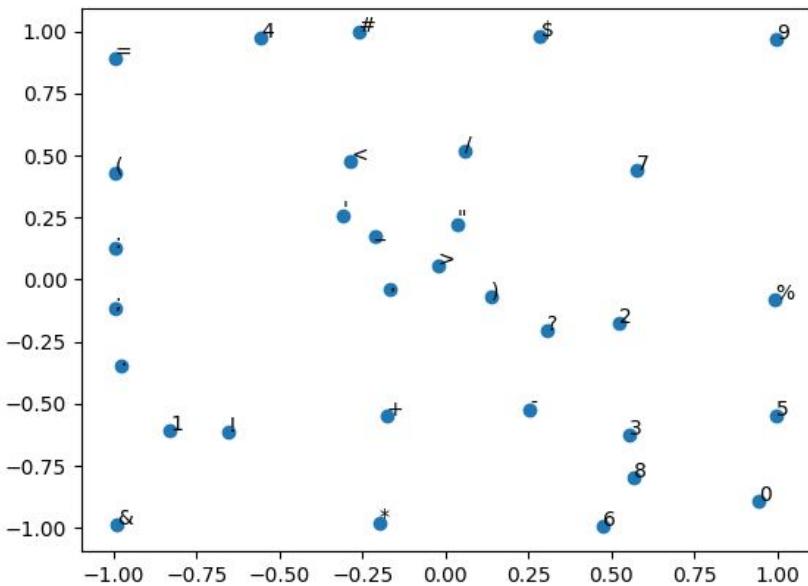
Epochs

Pudimos ver un gran incremento en la precisión del autoencoder variando la cantidad de epochs, sobre todo en el rango de las 1000-10000. Al aumentar mas el numero se presentan rendimientos decrecientes

Error frente a Epochs



El Espacio Latente

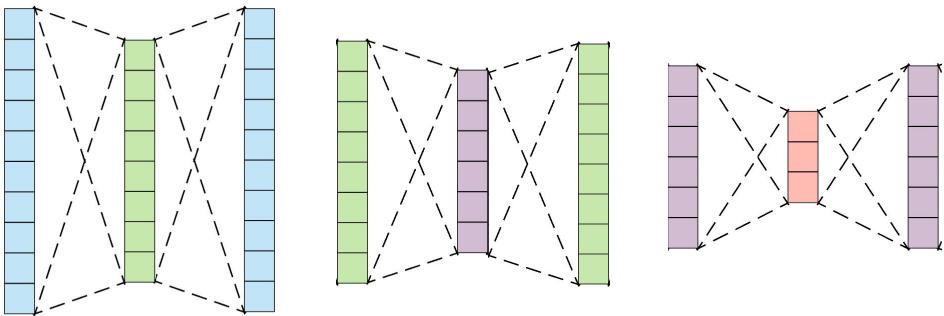
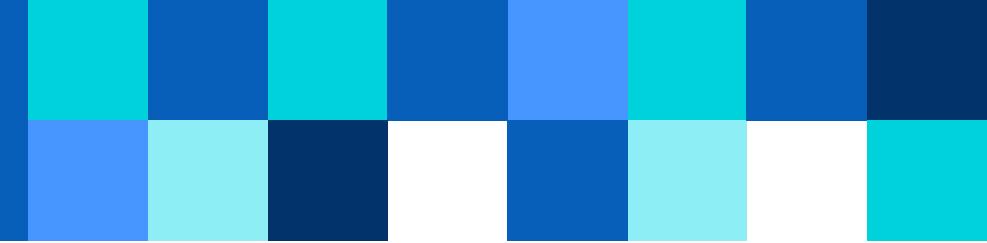


Nos pareció muy interesante como tirando números en el espacio latente y alimentando directamente al decoder se observó como este intentaba aproximar los caracteres y a veces mezclaba los mismos.

Los caracteres parecidos tambien, se encontraban más cercanos entre sí

Greedy Layer-Wise

Observamos que minimizar el error en cada transformación de dimensión no lleva necesariamente a una mejora en la codificación de la información a 2 dimensiones. Sin embargo, es interesante conocer este algoritmo para casos donde se necesite perder la mínima cantidad de información en cada dimensión.



VAE

Observamos que si las dimensiones de las imágenes eran muy grandes, rápidamente ocurría un overflow, con lo cual era importante reducir la dimensionalidad del problema para un mejor procesamiento.

Por otra parte, observamos que en ciertos casos, lo que aprendía la red era una distribución que permitía un morphing entre imágenes similar a desvanecer una imagen sobre la otra.

¡Gracias Totales!

- Baiges, Matías Sebastián 59076
- Bilevich, Andres Leonardo 59108
- Margossian, Gabriel Viken 59130

