



TP5: Deep Learning

Grupo 2

Tomás Álvarez Escalante (60127)

Alejo Francisco Caeiro (60692)

Lucas Agustín Ferreiro (61595)

Román Gómez Kiss (61003)

TABLA DE CONTENIDOS

01

Linear
Autoencoder

Ejercicio 1.a

02

Denoising
Autoencoder

Ejercicio 1.b

03

Variational
Autoencoder

Ejercicio 2

...

The background features a light blue-grey color with abstract geometric shapes in a slightly darker shade. A network of thin, dark blue lines crisscrosses the frame, some ending in small circles. A green plus sign is located in the upper left, and a red vertical bar with diagonal stripes is on the right. In the bottom left, there are four red 'x' marks arranged in a small cluster.

01

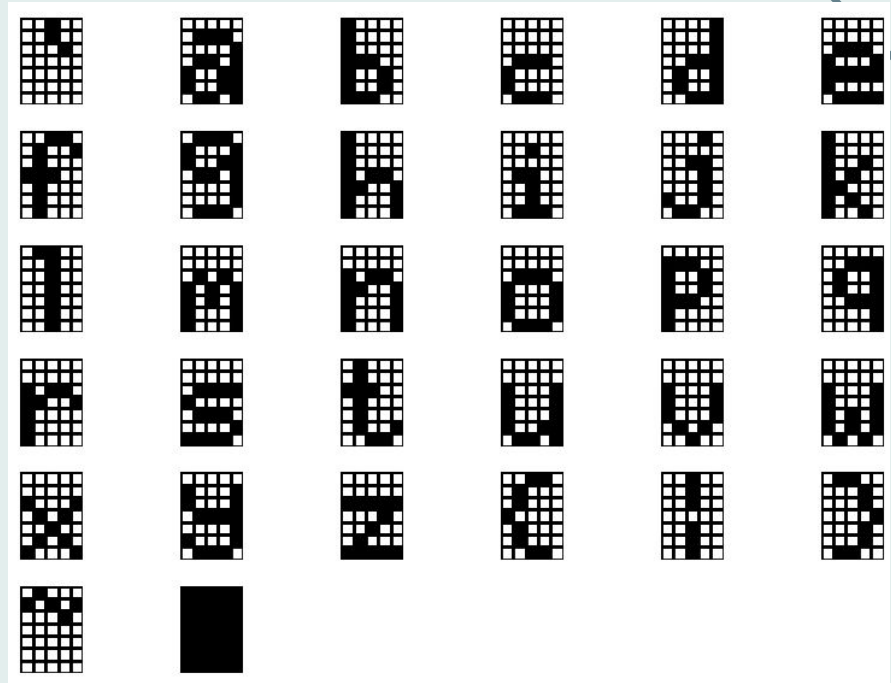
Linear Autoencoder

Problema

Se tiene un conjunto de 32 imágenes binarias de tamaño 7x5 para representar ciertos caracteres.

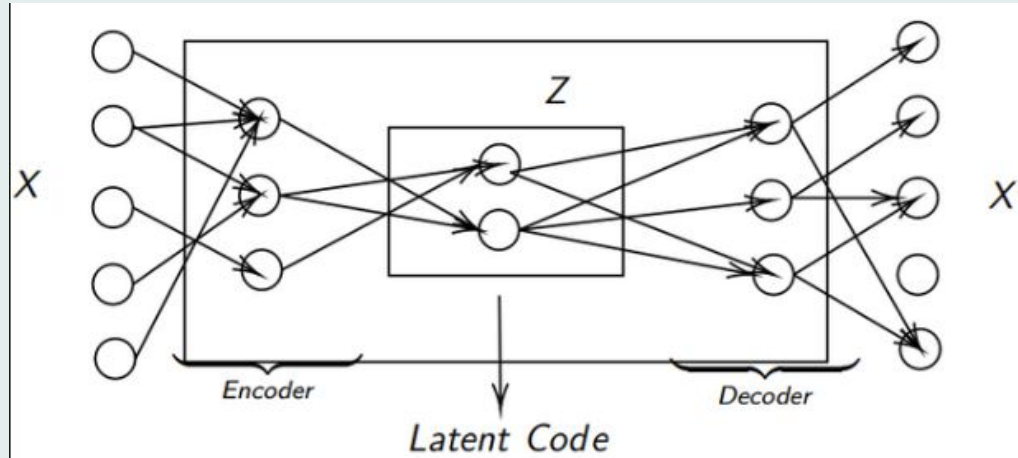
Se busca representar estos datos mediante un autoencoder con un espacio latente de dos dimensiones y con un error máximo de 1 pixel.

También se busca mostrar la capacidad de la red de generar un nuevo carácter que no pertenece al conjunto de entrenamiento.



Autoencoder

Son redes neuronales de aprendizaje no supervisado cuyo objetivo principal es reducir la dimensionalidad. Están conformados por dos perceptrones multicapa, donde la salida de la primera red se conecta con la entrada de la segunda red, la cual tiene la distribución invertida de neuronas en las capas y como salida tiene la misma dimensión que la entrada de la primera red. La idea es minimizar $\|X-X'\|^2$

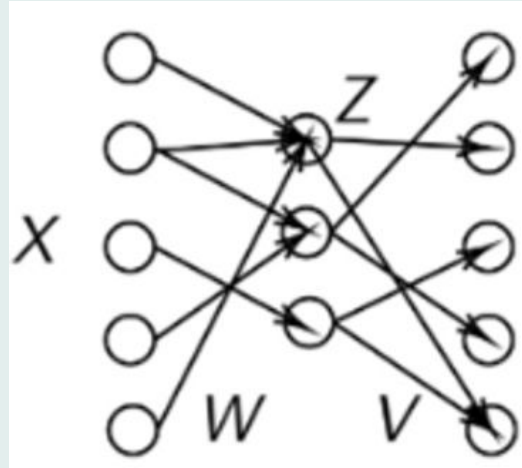


Z nos permite tener una nueva codificación de X en una dimensionalidad más pequeña.

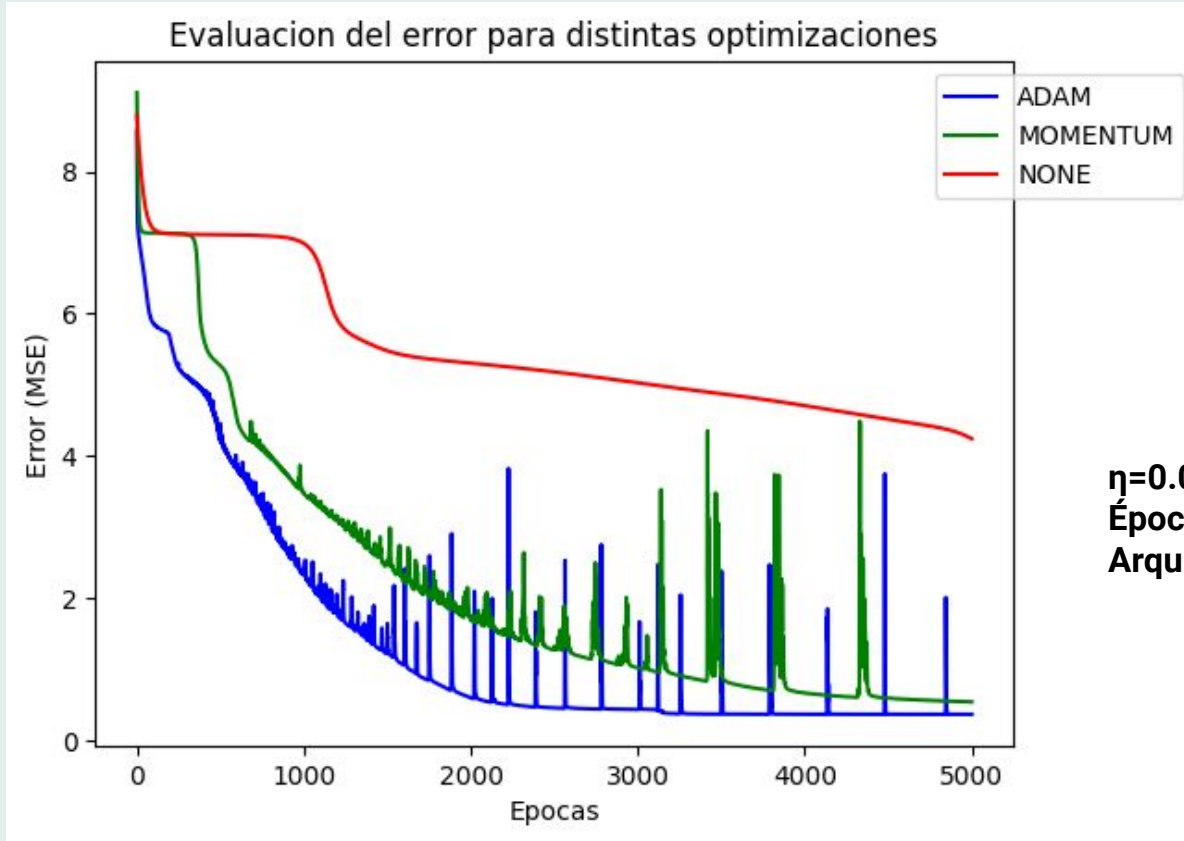
Autoencoder

Los perceptrones involucrados son no lineales con funciones de activación sigmoideas; una vez que el autoencoder aprende minimiza $\|X - ZV^T\|$

La salida del espacio latente Z del autoencoder lineal son las salidas de las proyecciones de los datos en los componentes principales de PCA.



Optimizaciones



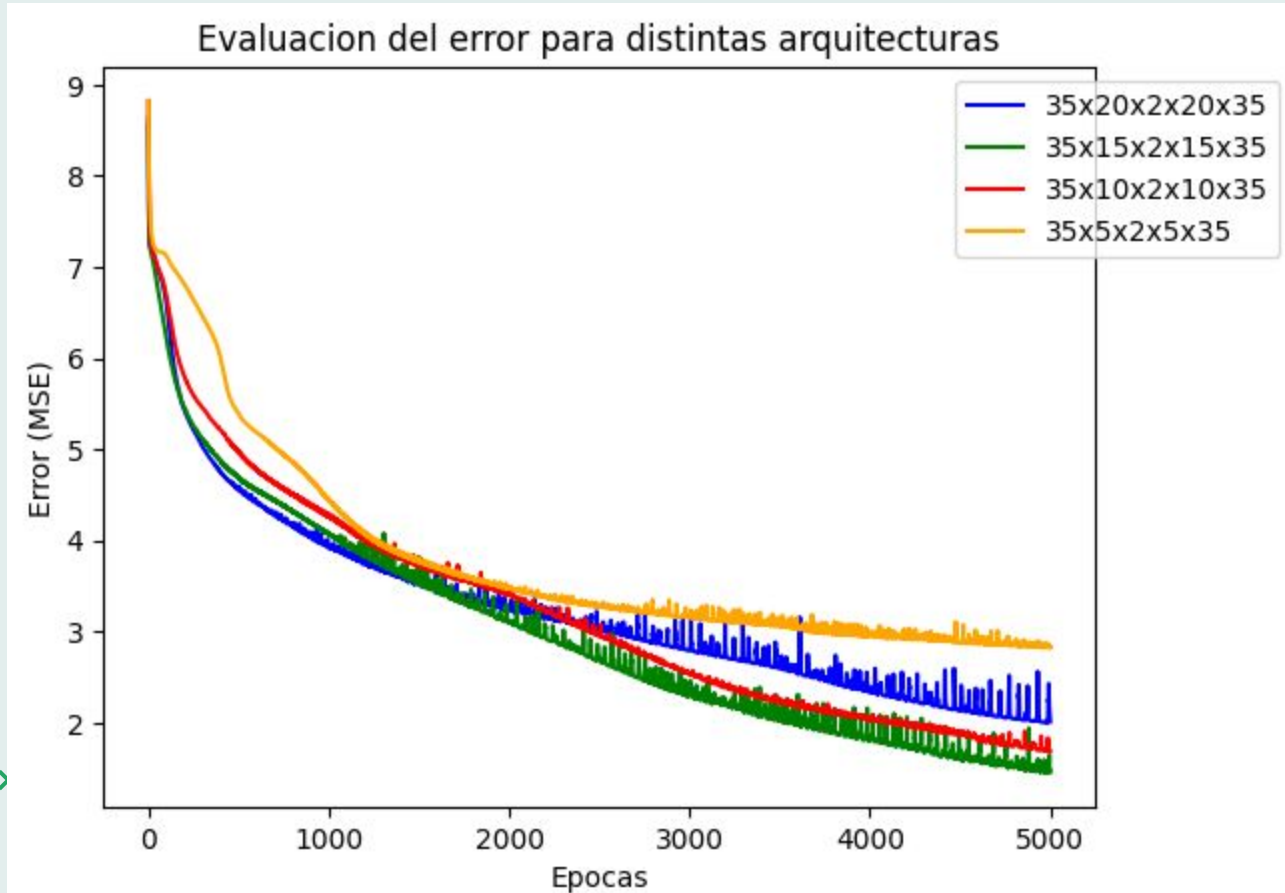
$\eta=0.001$

Épocas: 5000

Arquitectura: 35-10-2-10-35

```
"alpha": 0.8,  
"beta1": 0.9,  
"beta2": 0.999,  
"epsilon": 1e-8,
```

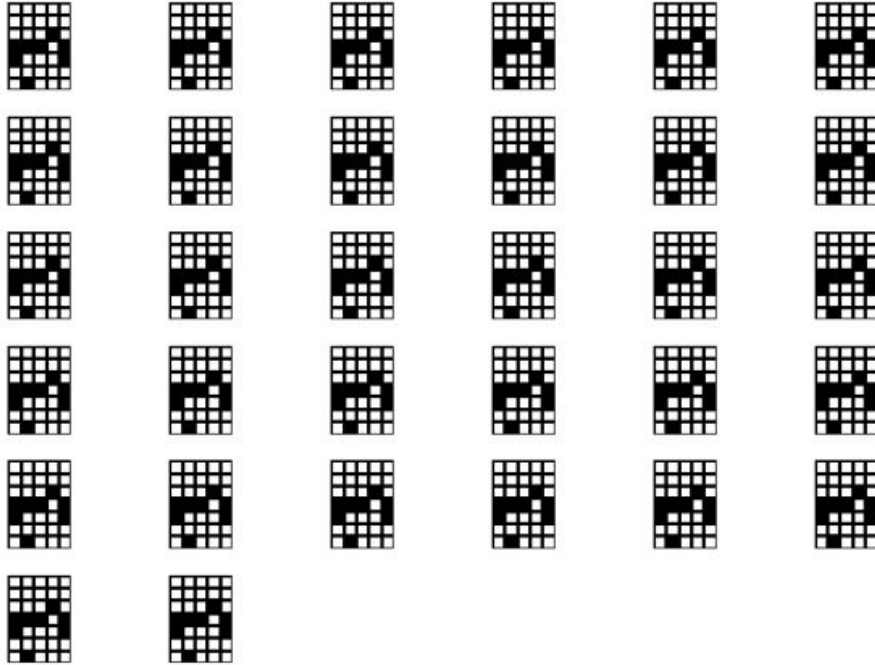
Variando la cantidad de nodos de la capa intermedia



Variando el learning rate



Predicted



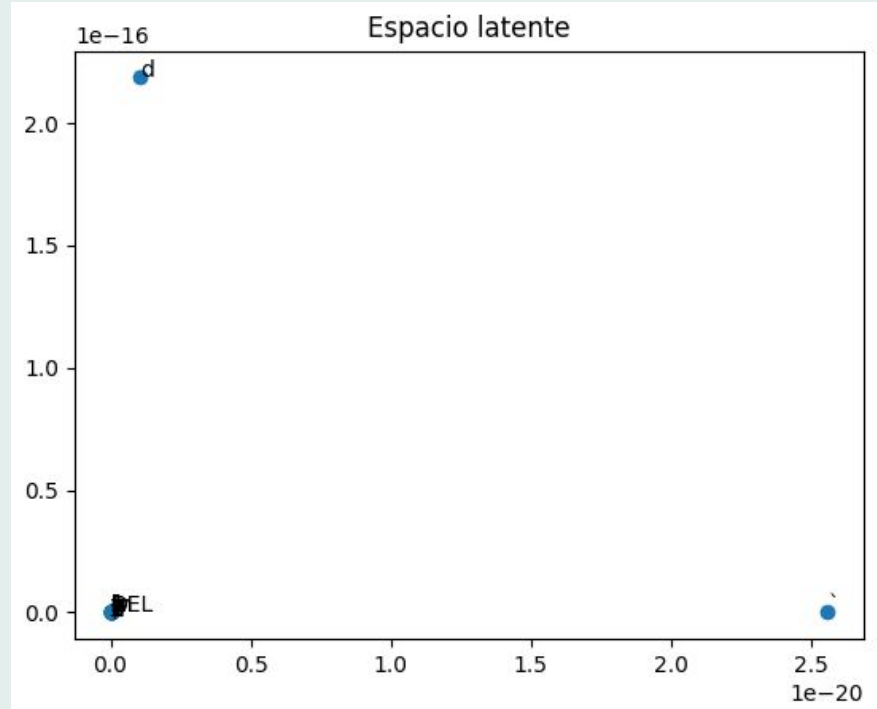
x x
x x

$\eta=0.1$

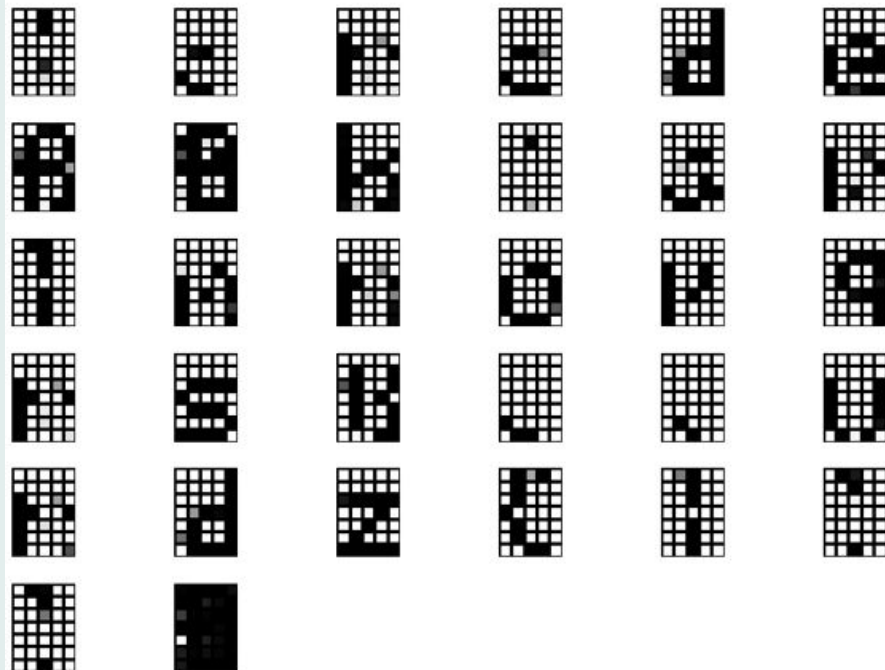
Épocas: 20000

Arquitectura: 35-10-2-10-35

MSE: 12.34375



Predicted



x x
x x

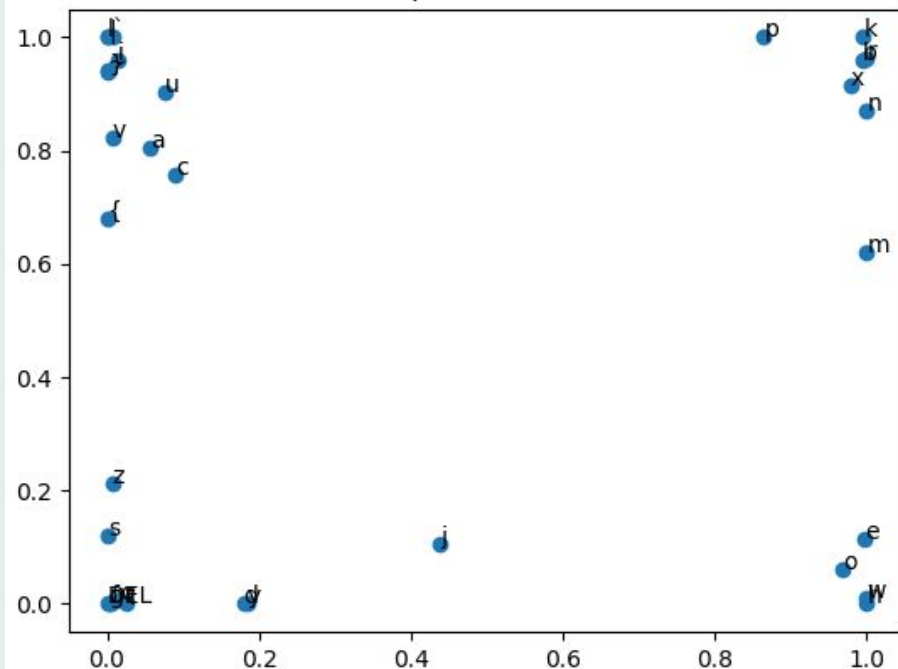
$\eta=0.01$

Épocas: 20000

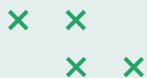
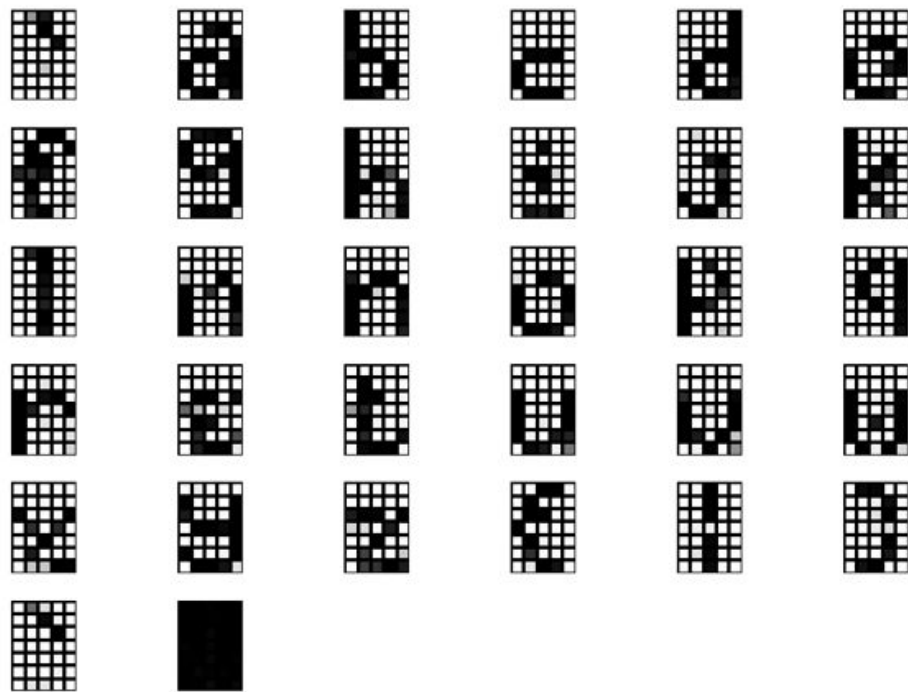
Arquitectura: 35-10-2-10-35

MSE: 4.30261

Espacio latente



Predicted



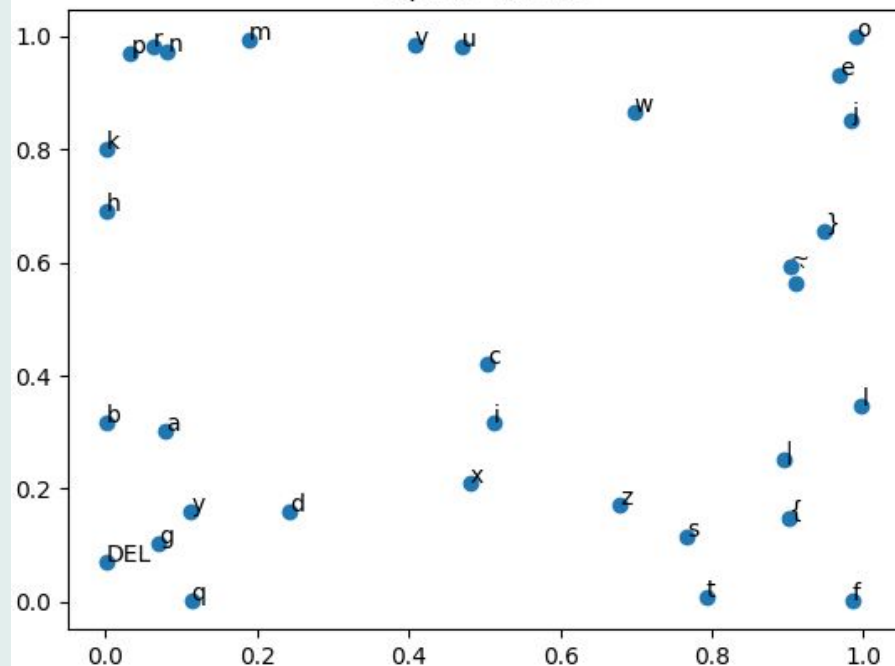
$\eta=0.001$

Épocas: 20000

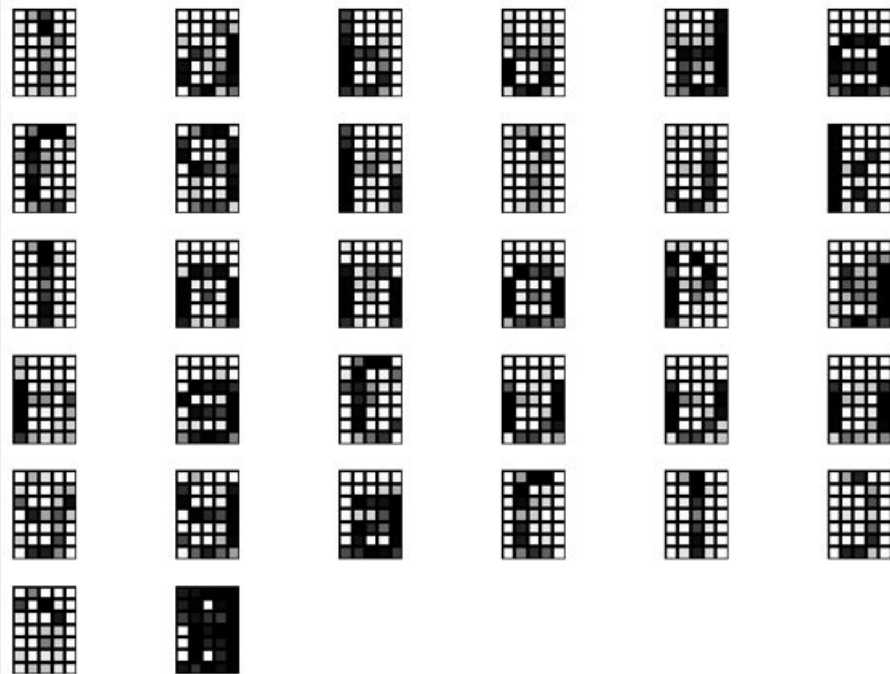
Arquitectura: 35-10-2-10-35

MSE: 1.15914

Espacio latente



Predicted



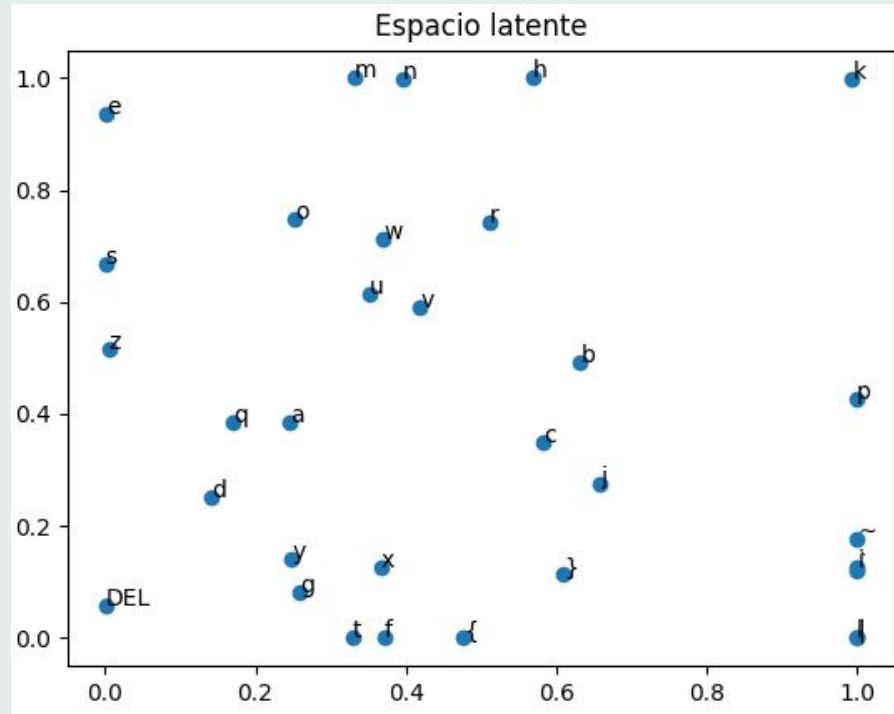
x x
x x

$\eta=0.0001$

Épocas: 20000

Arquitectura: 35-10-2-10-35

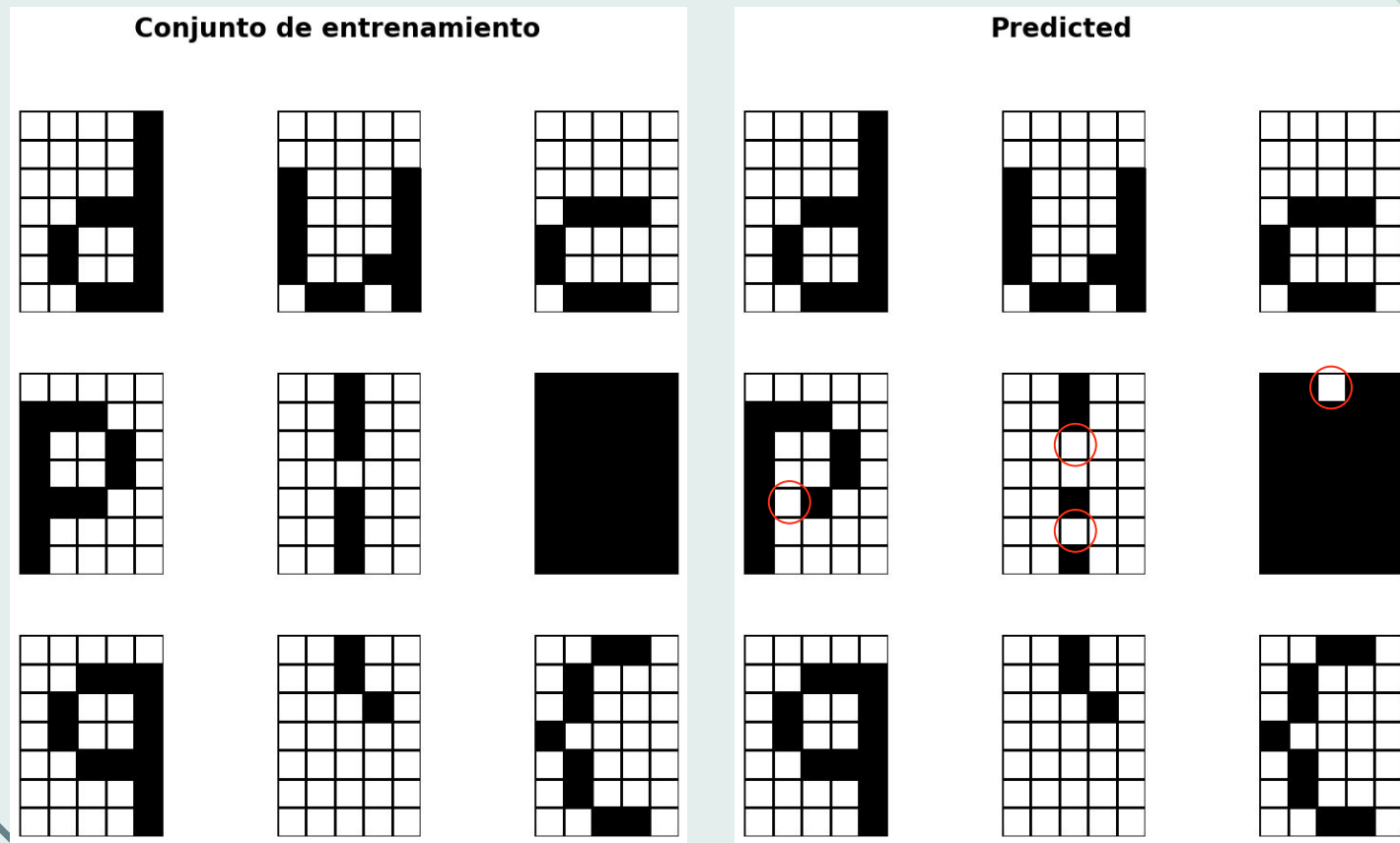
MSE: 2.63081



Testeando con subconjuntos



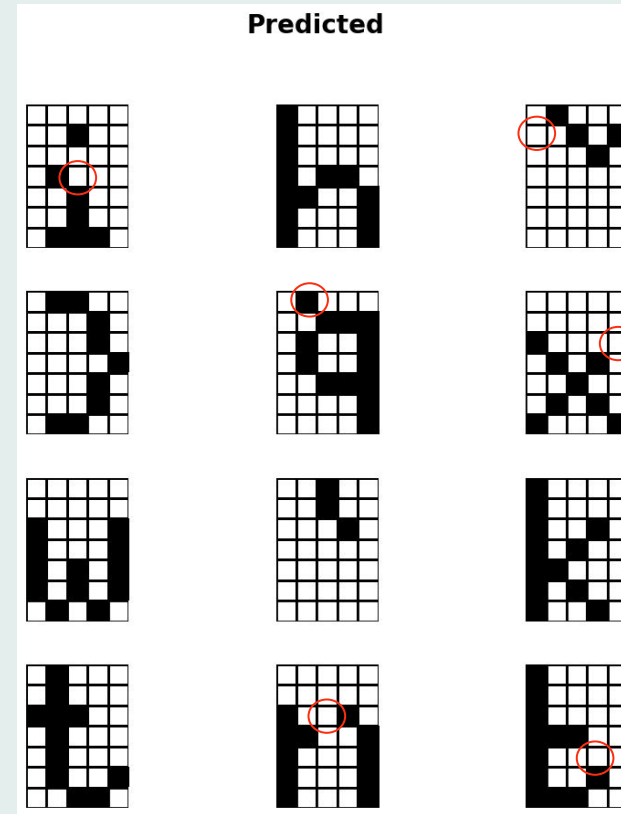
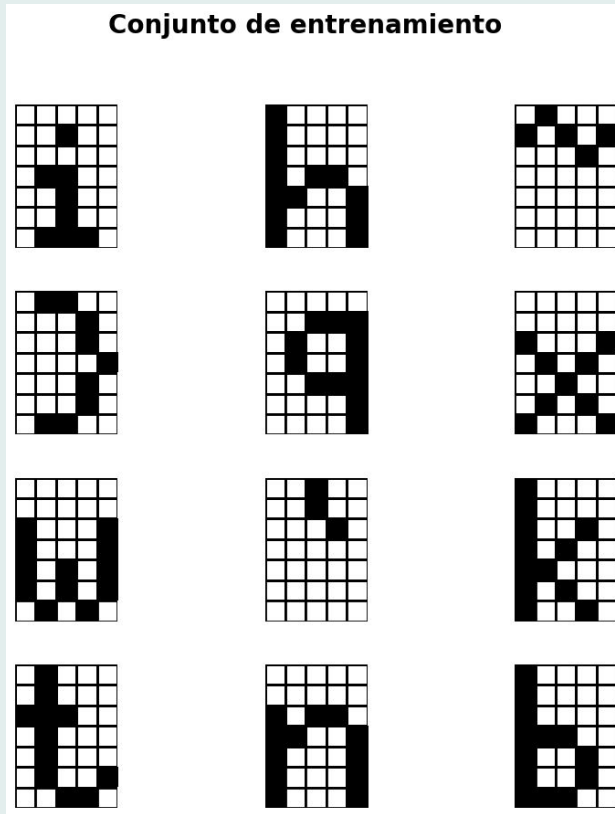
Subconjunto del 30% de los datos



$\eta=0.001$ Épocas: 10000 Arquitectura: 35-15-2-15-35 MSE: 0.43291



Subconjunto del 40% de los datos



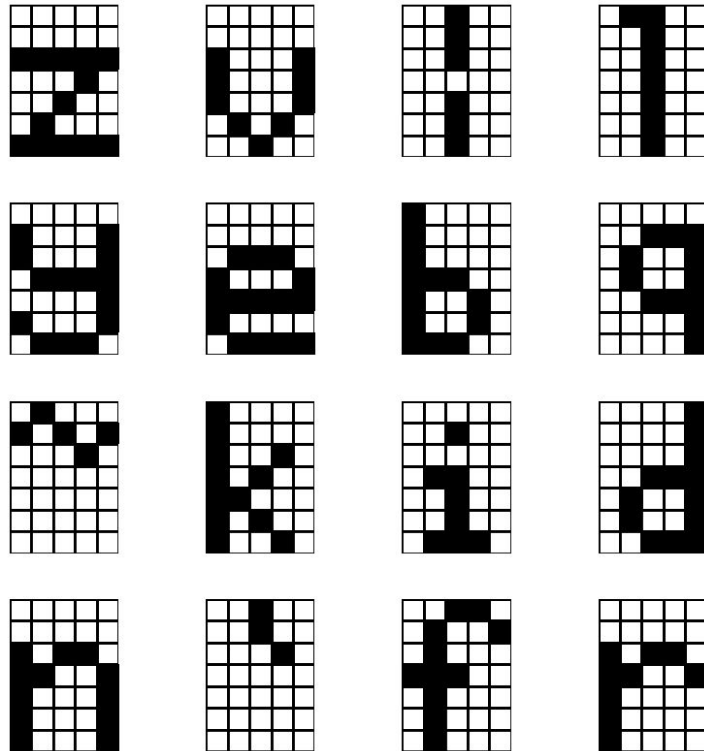
× ×

$\eta=0.001$ Épocas: 15000 Arquitectura: 35-15-2-15-35 MSE: 0.50001

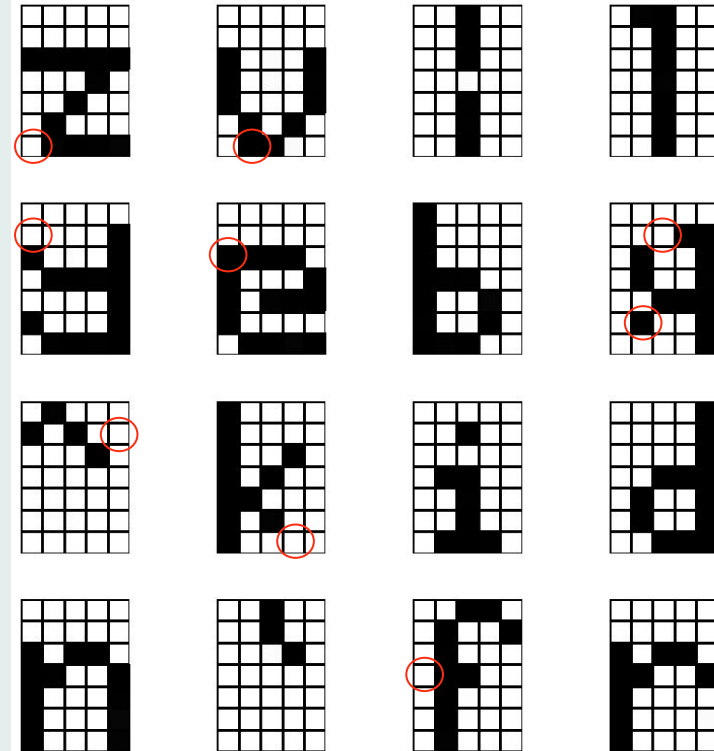


Subconjunto del 50% de los datos

Conjunto de entrenamiento



Predicted

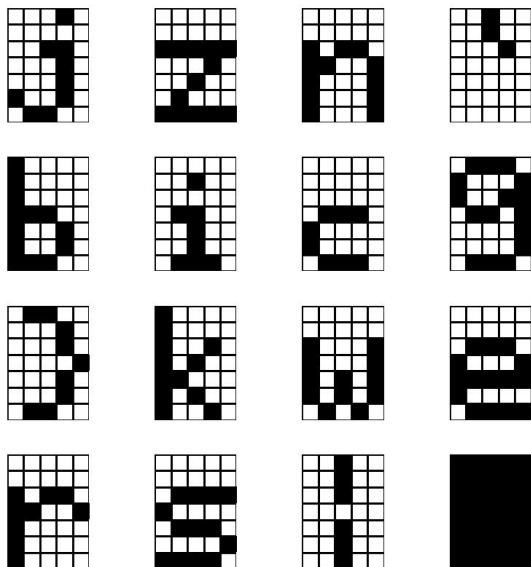


$\eta=0.001$ Épocas: 15000 Arquitectura: 35-15-2-15-35 MSE: 0.68778

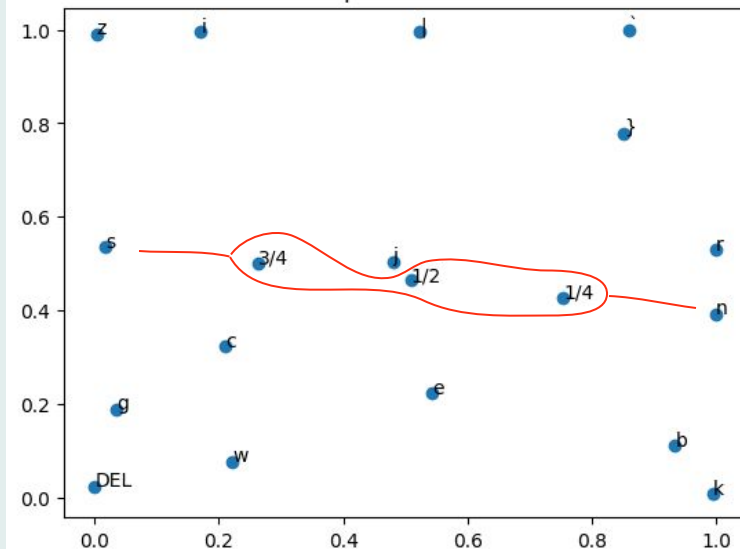
**Capacidad de
generar nuevos
caracteres**



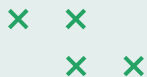
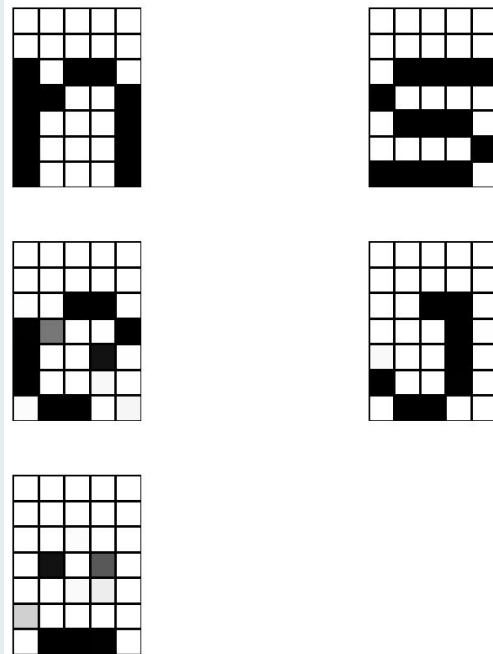
Conjunto de entrenamiento



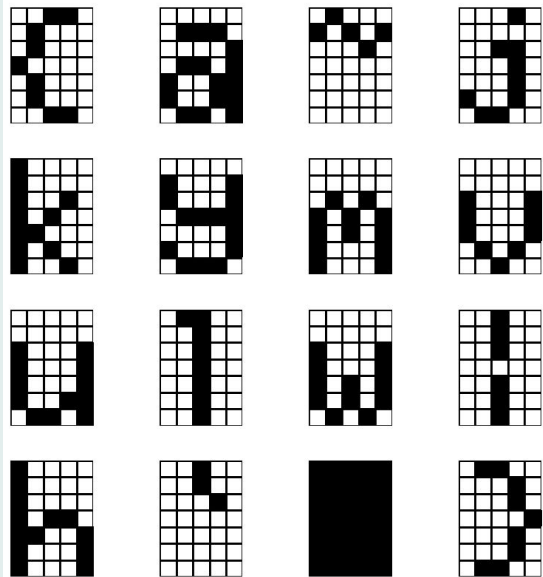
Espacio latente



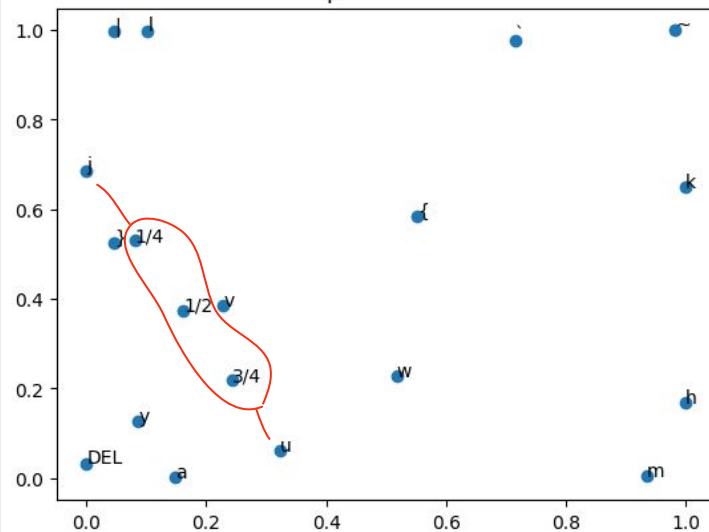
Espacio latente medio



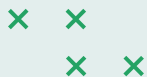
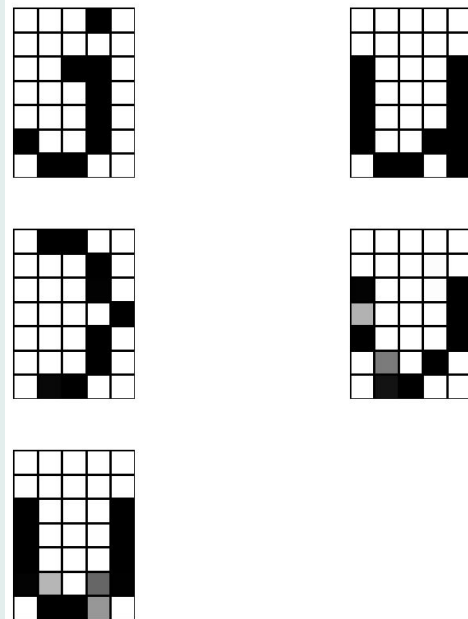
Conjunto de entrenamiento



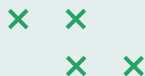
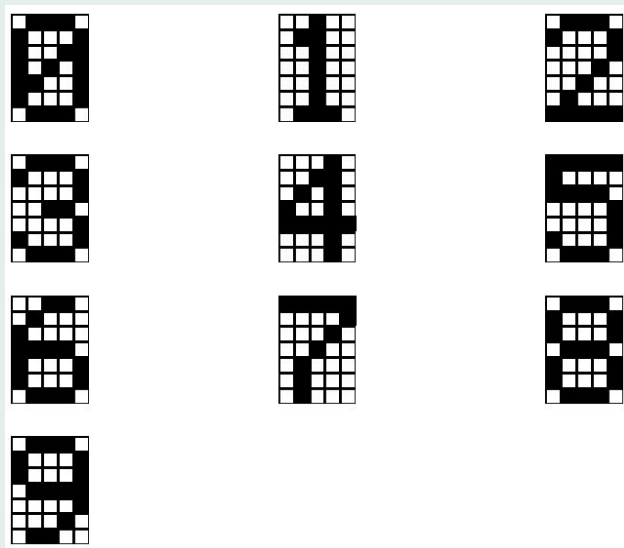
Espacio latente

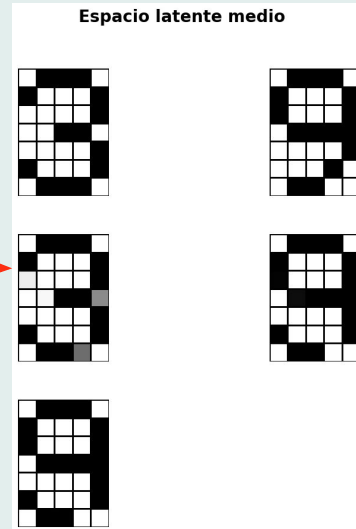
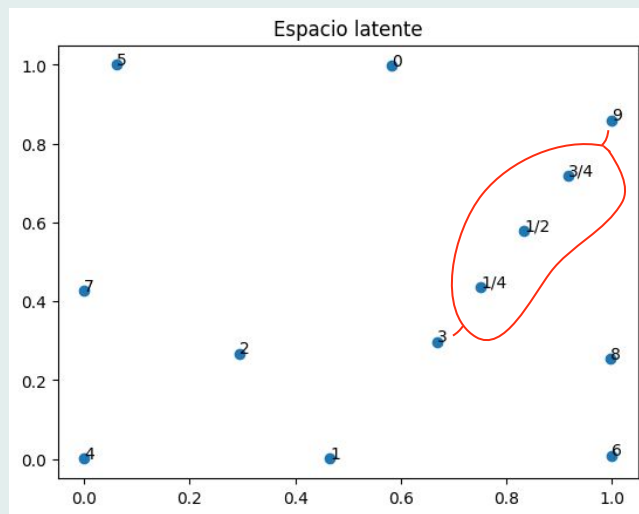
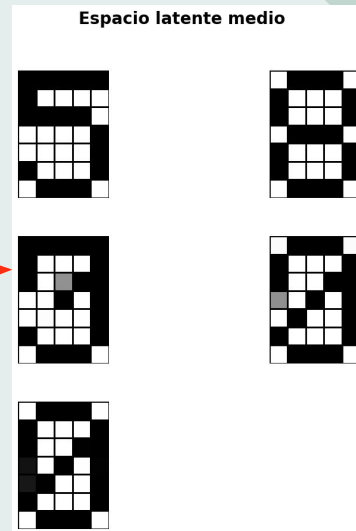
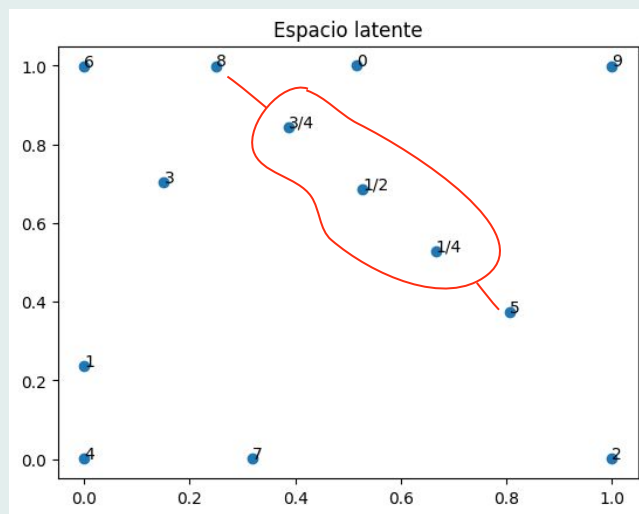


Espacio latente medio



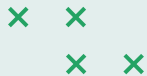
Capacidad de generar nuevos caracteres con otro conjunto de datos





Conclusiones

- Cuanto más grande es el conjunto de entrenamiento, mayor es el error obtenido, es decir, será más difícil de aprender dicho conjunto por el autoencoder.
- Al analizar la capa latente, se puede observar que letras similares tienen codificaciones similares (lemma PCA).
- El método de optimización más eficiente es ADAM.
- No existe un número de capas ideal.
- Tampoco existe un número de neuronas por capa ideal.
- No se pueden generar caracteres nuevos, sino que se encuentran deformaciones de los aprendidos, o caracteres cercanos a ese espacio latente





02

Denoising Autoencoder

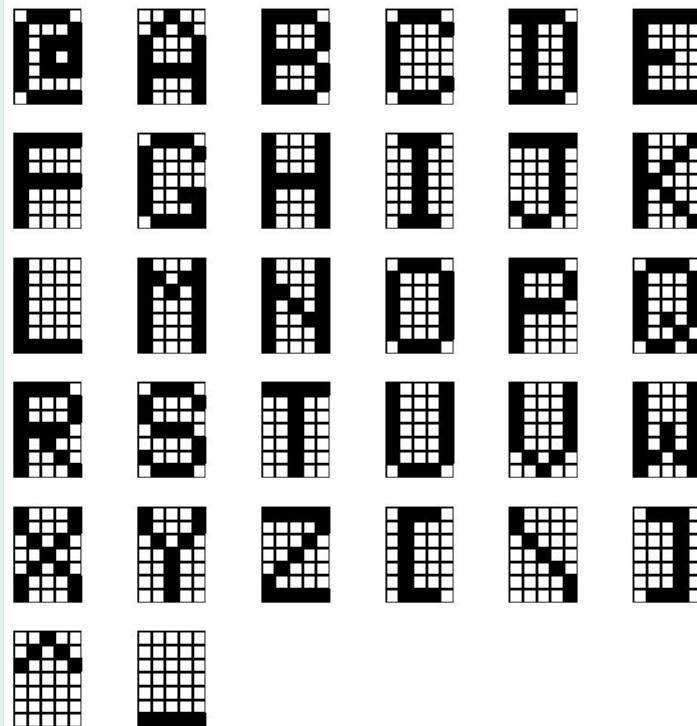
Problema

Se busca estudiar la capacidad del autoencoder de eliminar ruido sobre los caracteres del conjunto de entrenamiento. Para este problema se decidió utilizar el siguiente conjunto de caracteres:

Para realizar esto se utiliza el concepto de un Denoising Autoencoder.

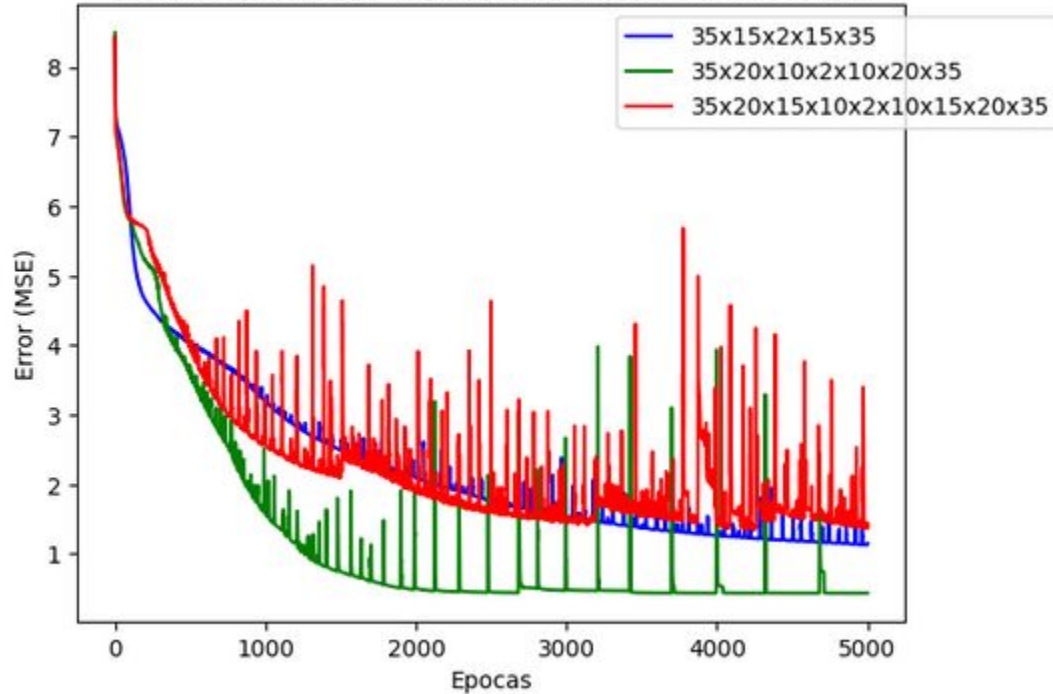
La mutación o ruido aplicado consiste en elegir aleatoriamente N caracteres a ser mutados, donde se varía el valor de cada píxel por un número delta con probabilidad uniforme entre (0,0.5).

Conjunto de entrenamiento



Arquitecturas

Evaluación del error de DAE con distintas arquitecturas

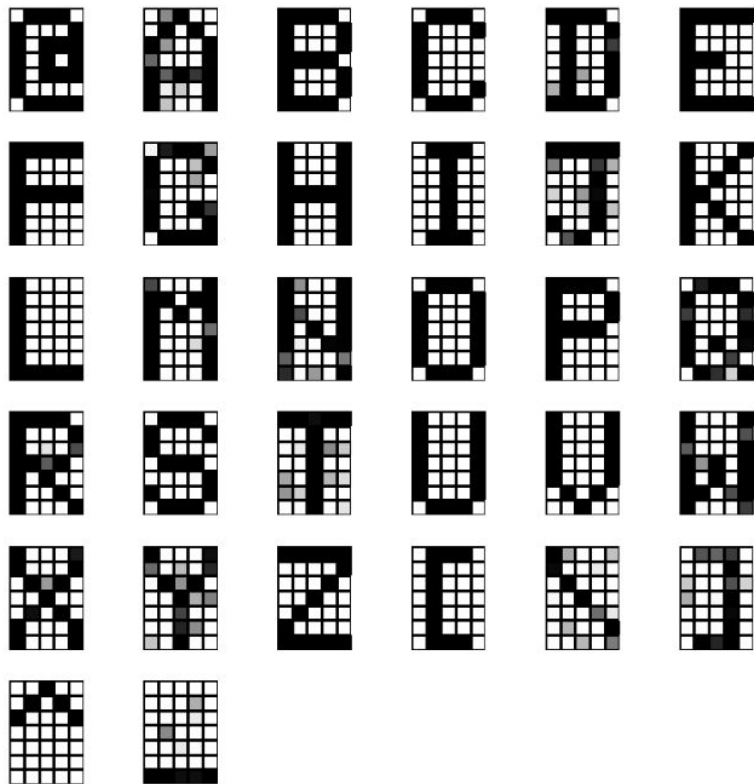


Variando la probabilidad de mutación

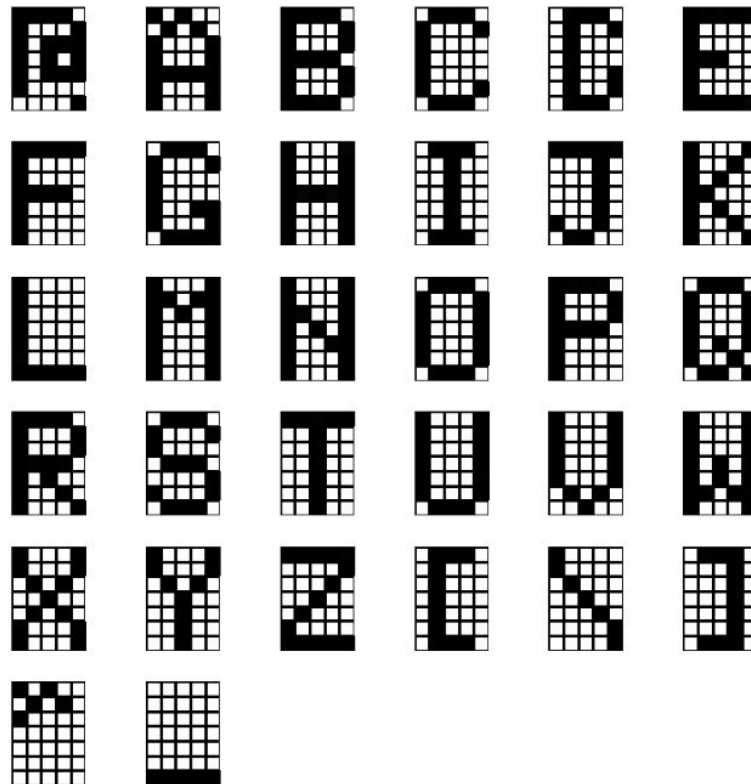


Probabilidad 0.2 de mutar cada pixel

Conjunto de entrenamiento con 15 caracteres mutados

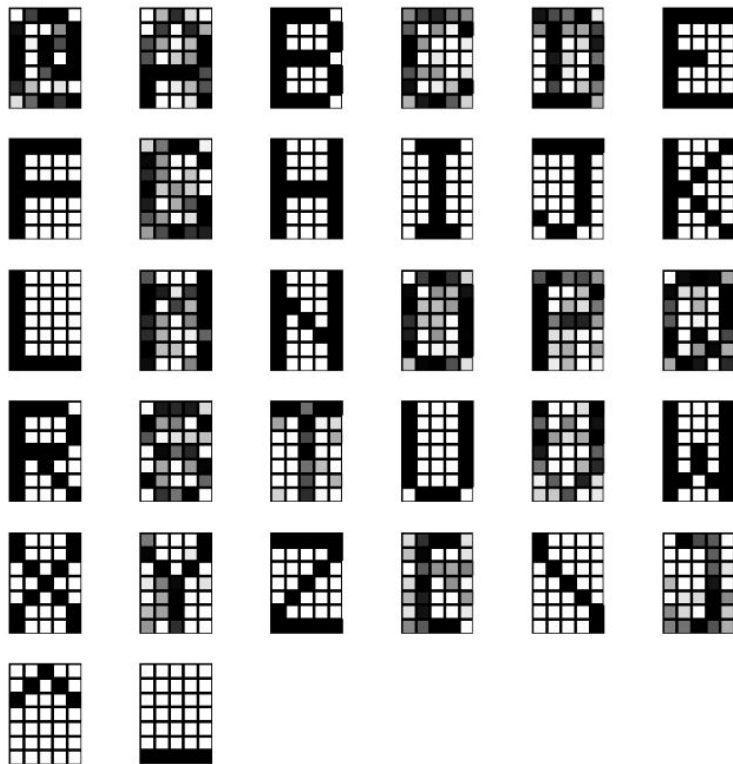


Eliminacion del ruido

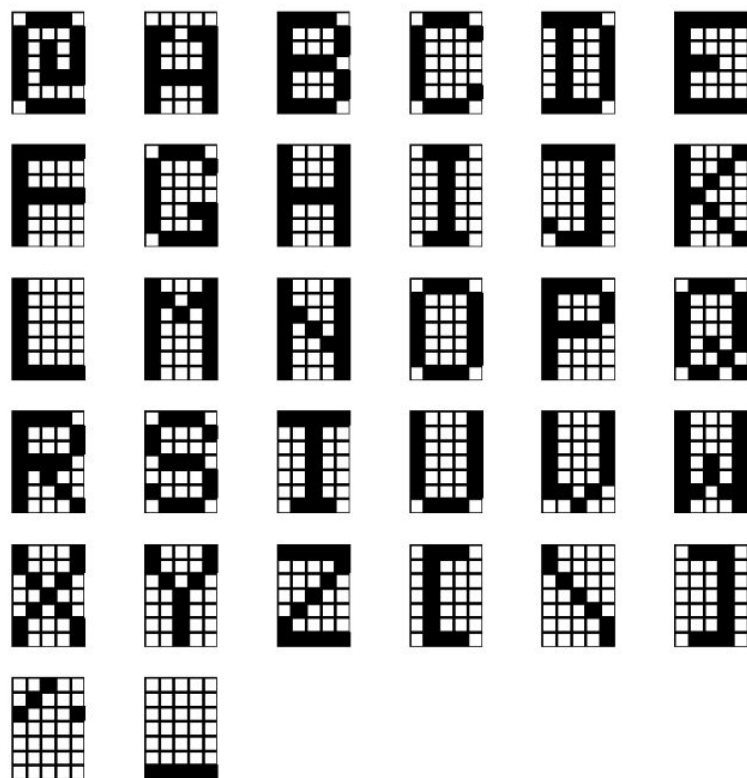


Probabilidad 0.5 de mutar cada pixel

Conjunto de entrenamiento con 15 caracteres mutados

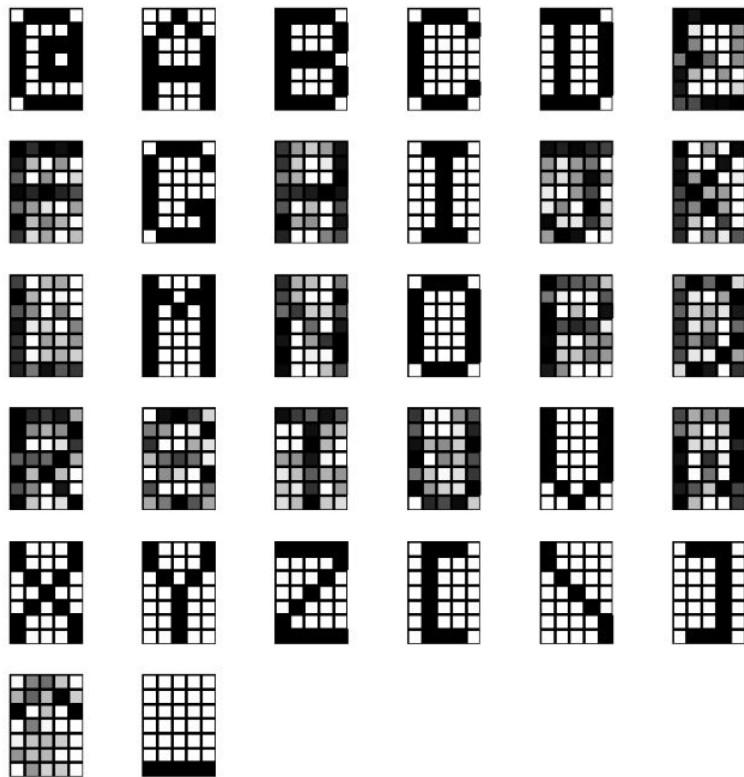


Eliminacion del ruido

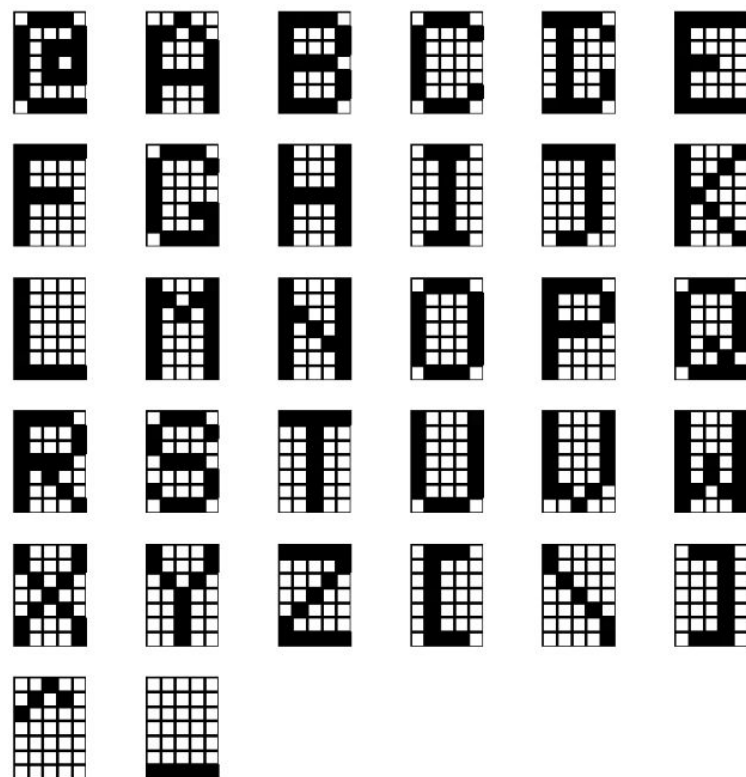


Probabilidad 0.7 de mutar cada pixel

Conjunto de entrenamiento con 15 caracteres mutados

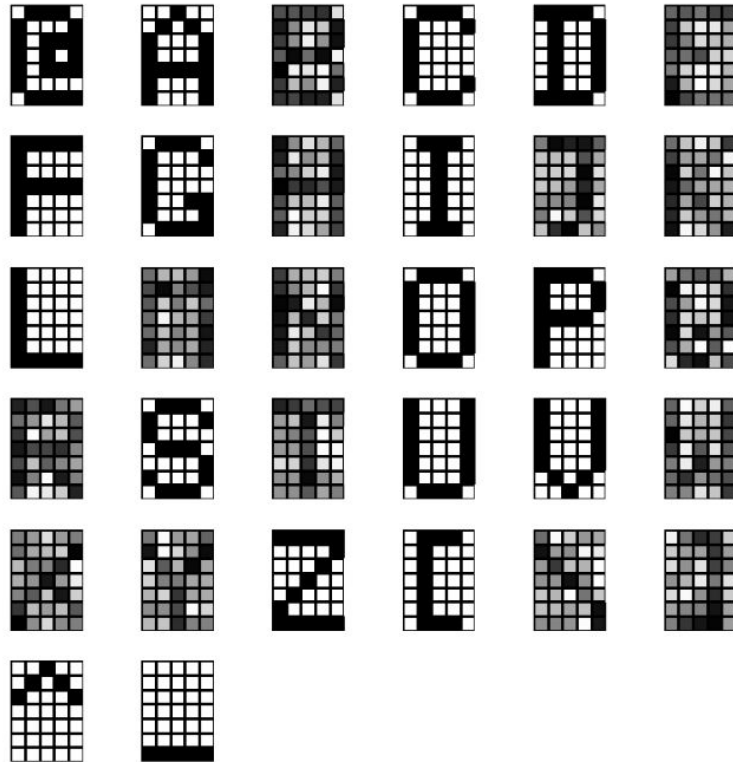


Eliminacion del ruido

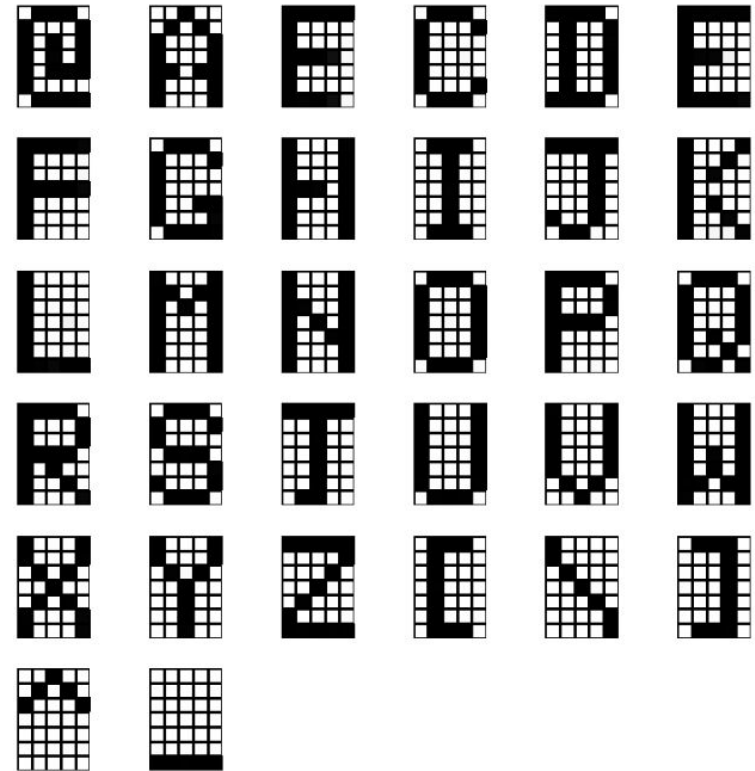


Probabilidad 1 de mutar cada pixel

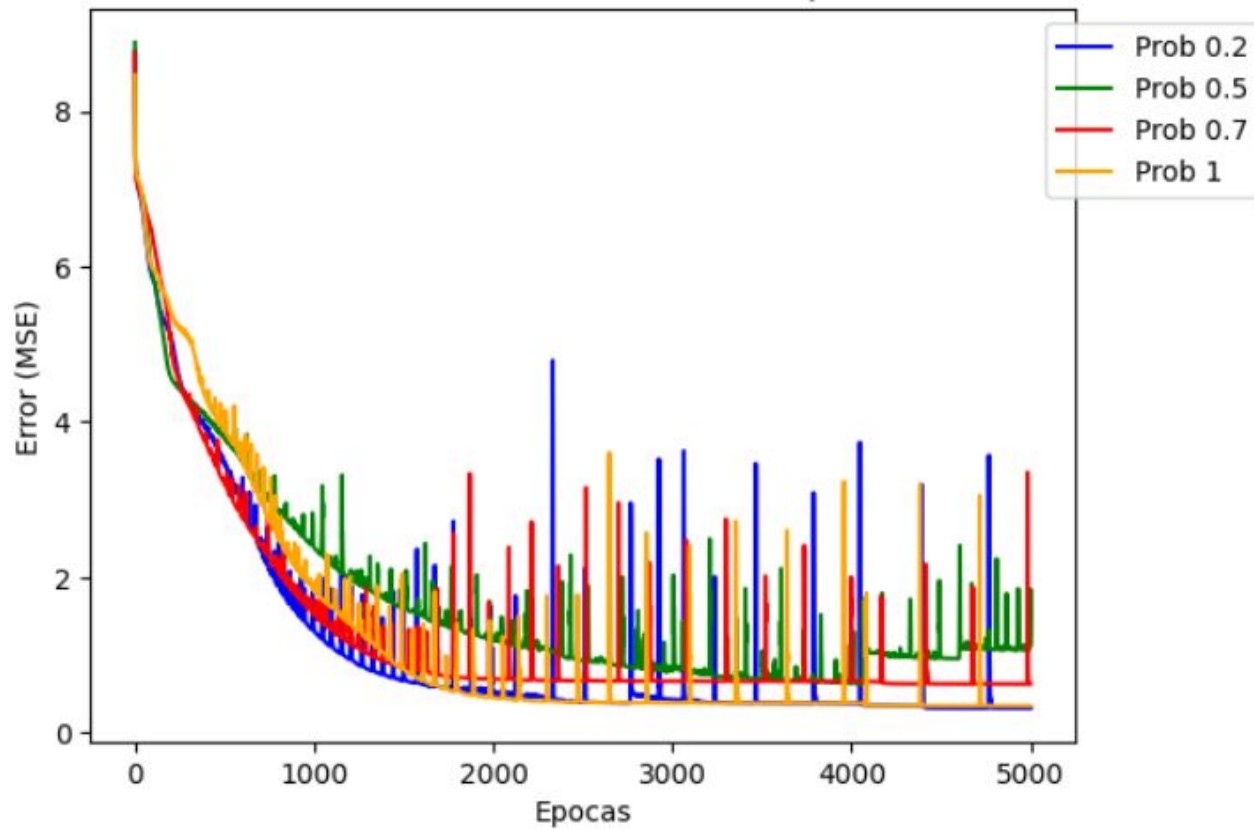
Conjunto de entrenamiento con 15 caracteres mutados



Eliminacion del ruido

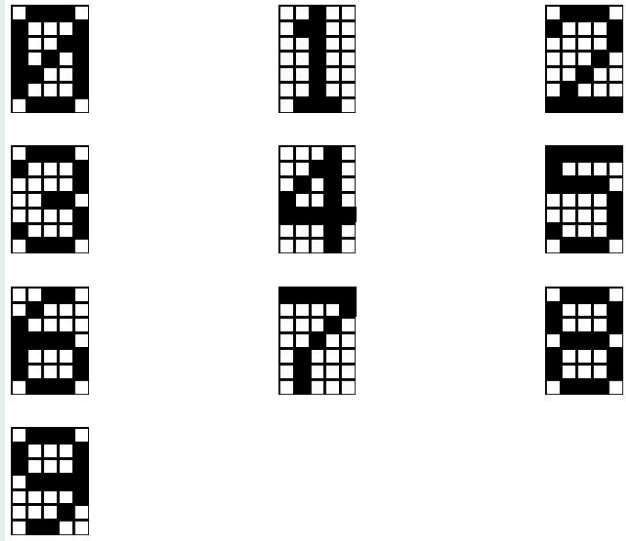


Evaluacion del error de DAE con distintas probabilidades



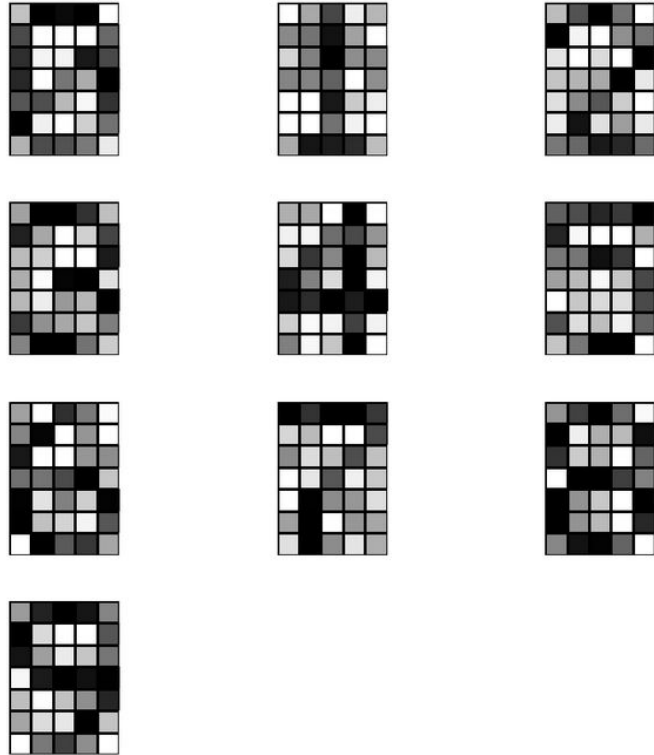
x x

Variando la probabilidad de mutación con un nuevo conjunto de datos

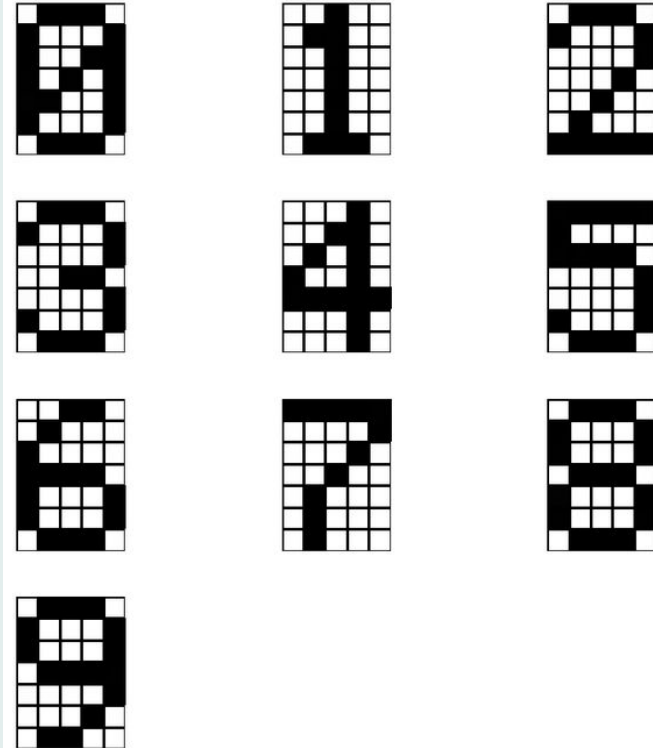


Probabilidad 0.7 de mutar cada pixel

Conjunto de entrenamiento

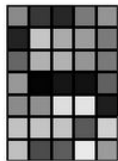
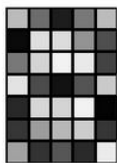
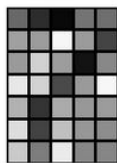
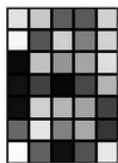
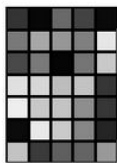
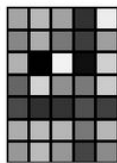
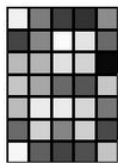
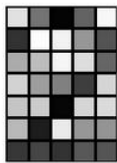
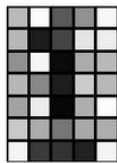
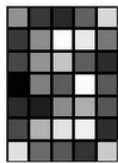


Eliminacion del ruido

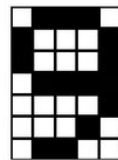
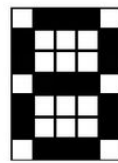
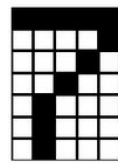
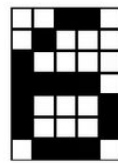
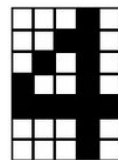
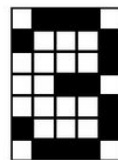
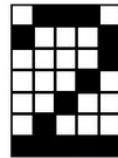
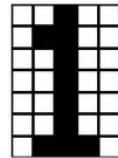


Probabilidad 1 de mutar cada pixel

Conjunto de entrenamiento



Eliminacion del ruido



Conclusiones

- El DAE es capaz de asociar el valor ruidoso al valor original en la gran mayoría de los casos.
- A mayor cantidad de datos del conjunto de entrenamiento, el DAE tiene mayores dificultades para eliminar el ruido.
- A mayor cantidad de capas, se asocian mejor las letras ruidosas. Pero con estos hay que tener en cuenta que el tiempo de aprendizaje del DAE aumenta considerablemente.
- A mayor ruido, más le cuesta asociar las entradas a la red.

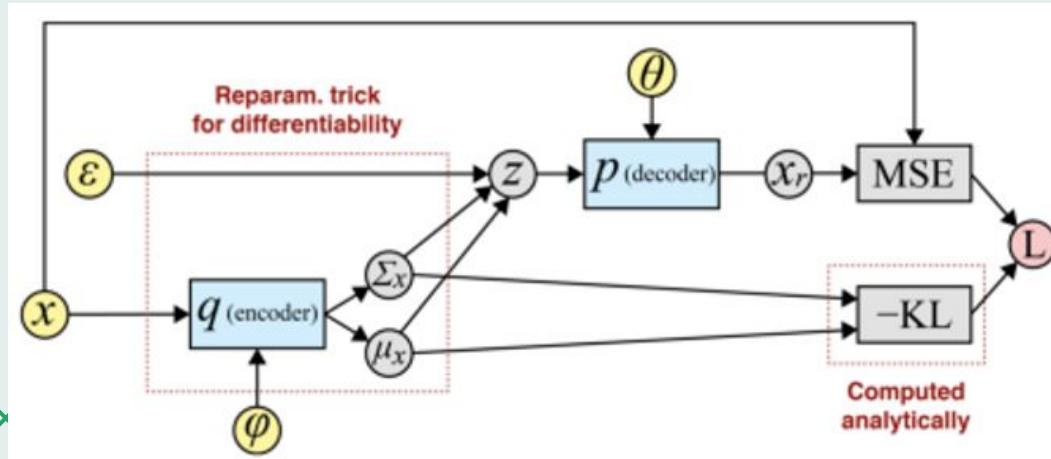
The background features a light blue-grey color with abstract geometric patterns. On the left, there are green and blue shapes with a green plus sign and a blue line with a circle. On the right, there are blue and green shapes with a red and white striped vertical bar. At the bottom left, there are four red 'x' marks.

03

Variational Autoencoder

Variational Autoencoder

En vez de construir un autoencoder que emita una solo valor para describir cada atributo del estado latente, el autoencoder variacional provee una forma probabilística de describir una observación en el espacio latente. De esta forma, nos podemos mover dentro de un vector de representación para generar nuevas muestras a la salida del decodificador.



Fashion MNIST

Conjunto de datos de Keras

Set de 60000 muestras

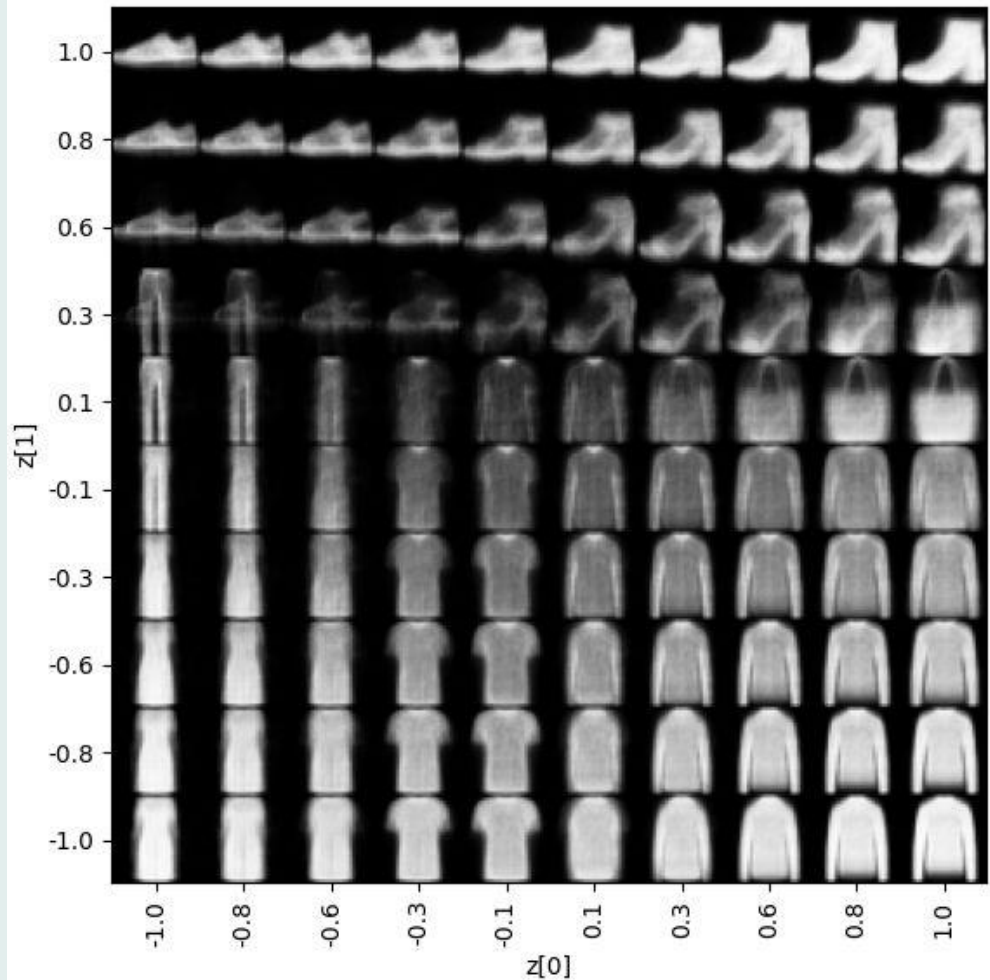
Entrenamiento con los primeros 10000

30 Épocas

Arquitectura:

$N \times 300 \times 200 \times 100 \times 2 \times 100 \times 200 \times 300 \times N$

El VAE nos permite explorar salidas del autoencoder que no necesariamente tienen una entrada asociada, lo que con un buen conjunto de datos permite generar nueva data.



MNIST

Conjunto de datos de Keras

Set de 60000 muestras

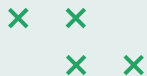
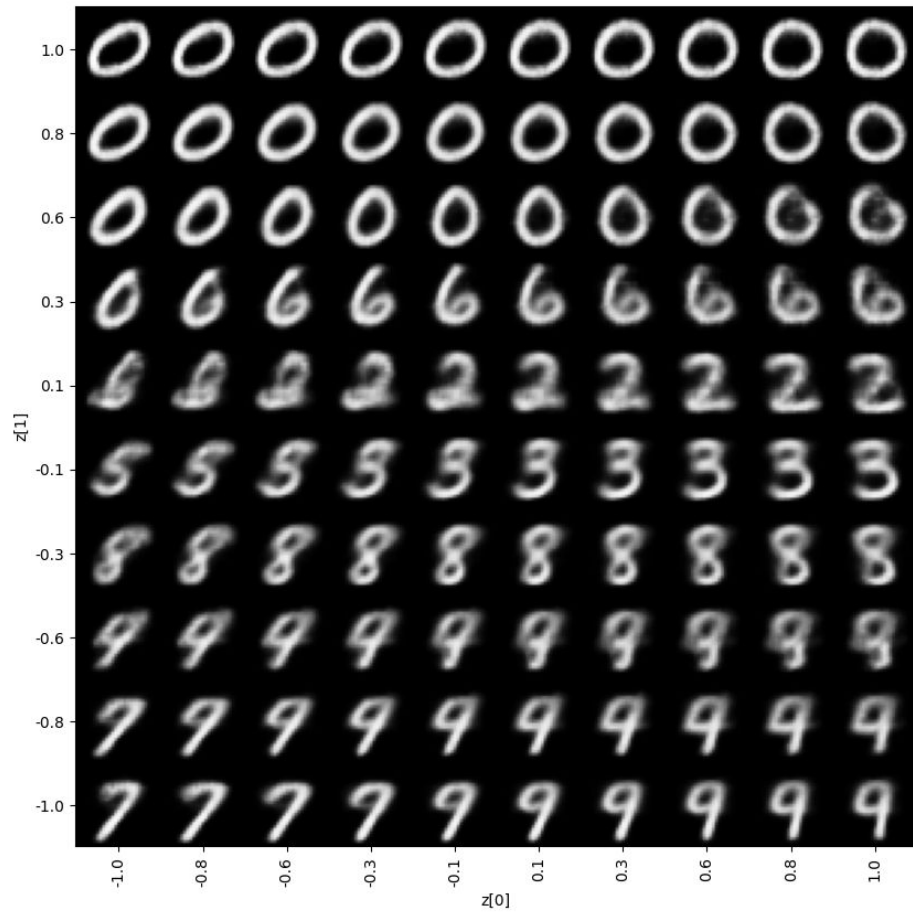
Entrenamiento con los primeros 10000

50 Épocas

Arquitectura:

$N \times 300 \times 200 \times 100 \times 2 \times 100 \times 200 \times 300 \times N$

El VAE nos permite explorar salidas del autoencoder que no necesariamente tienen una entrada asociada, lo que con un buen conjunto de datos permite generar nueva data.



Conclusiones

- La representación del espacio latente en un autoencoder no variacional no tienen una estructura probabilística, lo que limita la capacidad de generar nuevas muestras.
- Los autoencoders utilizados en el ejercicio 1 son más rápidos de entrenar en comparación con los VAE, ya que el backpropagation es menos complejo.
- Los VAEs permiten generar nuevas muestras y realizar interpolaciones suaves sobre todo el espacio latente.



¡Gracias!

