

Trabajo Práctico 5

Deep Learning

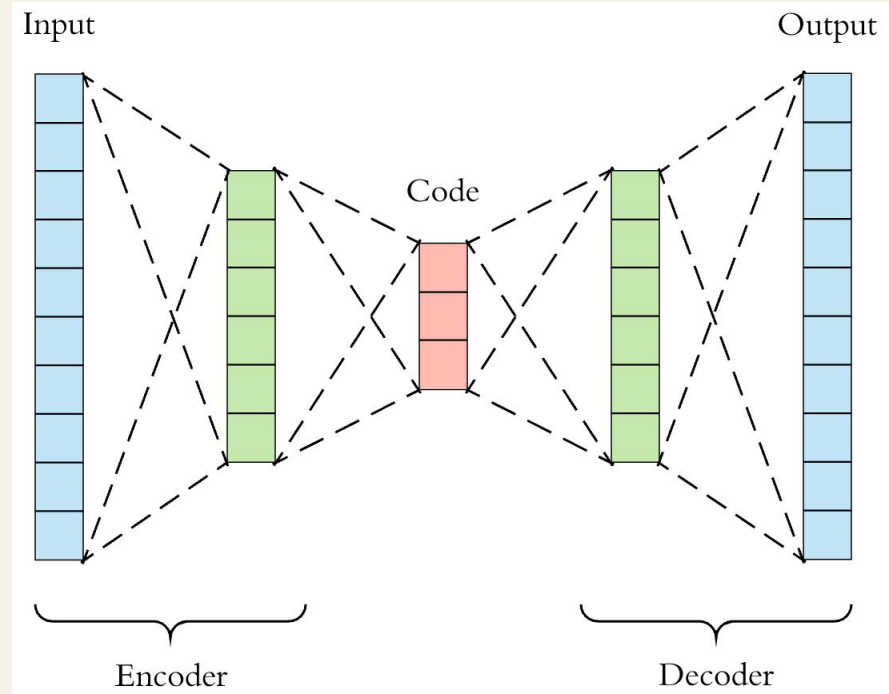
Eugenio Damm
Santiago Terenziani

Ejercicio 1

Arquitectura

La red se crea automáticamente dado un conjunto de entrenamiento y un “factor de división”. Dado el tamaño del input y el valor del factor, se determinará la cantidad de capas de la red y la cantidad de nodos en cada una.

En la capa central siempre se hallarán dos nodos que conforman el “espacio latente” y un nodo adicional que actúa como el bias.



Técnicas de Optimización

01

Momentum

El ajuste de pesos utiliza momentum

02

Aprendizaje progresivo

La red se entrena con un conjunto limitado creciente

03

Learning Rate Variable

Si el error baja/sube consistentemente, se modifica el LR

04

Tiempo Extra

Si queda poco tiempo y el error viene bajando, se le da tiempo adicional

Resultados

Dataset 2 / 5 letras por tanda / Máx 5000 épocas por tanda
3 minutos base por tanda / Shuffling desactivado / Learning rate 0.05

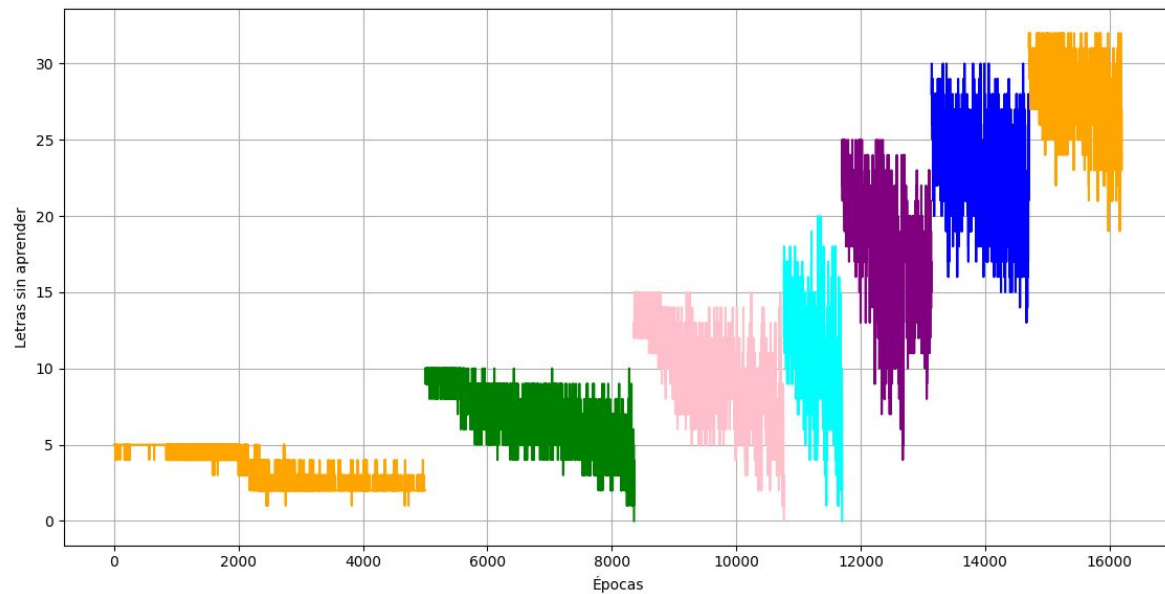
	Sin Momentum	Momentum 0.75	Momentum 0.5
35→17→8→4→2→4→8→17→35	21.8 letras	22.4 letras	21.6 letras
35→11→3→2→3→11→35	14.3 letras	15 letras	15.6 letras
35→8→2→8→35	18.6 letras	17.6 letras	17.6 letras

Resultados

Dataset 1 / 5 letras por tanda / Máx 5000 épocas por tanda
3 minutos base por tanda / Shuffling desactivado / Learning rate 0.05

	Sin Momentum	Momentum 0.75	Momentum 0.5
35→17→8→4→2→4→8→17→35	20 letras	22 letras	21.5 letras
35→11→3→2→3→11→35	15 letras	15 letras	15.5 letras
35→8→2→8→35	20 letras	21 letras	20 letras

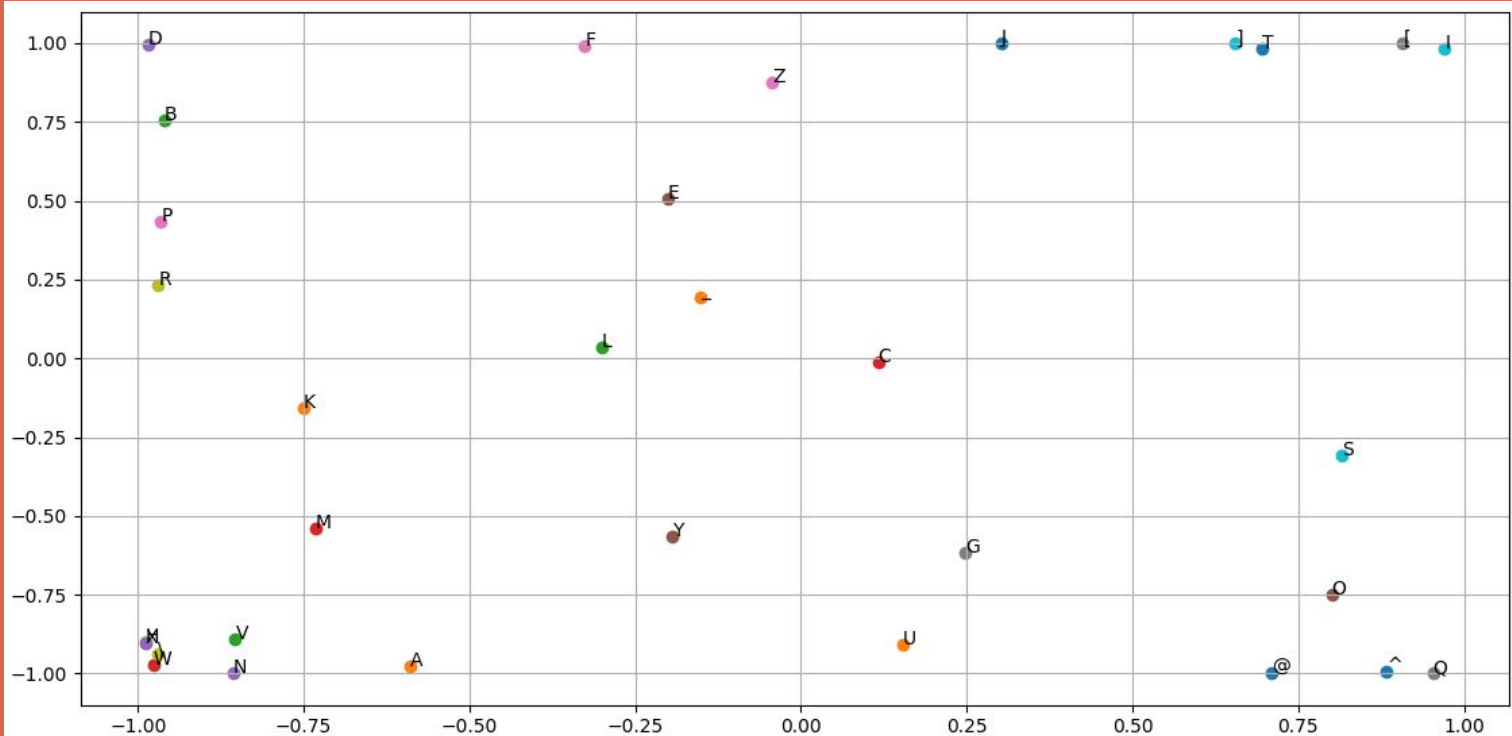
Evolución



Espacio Latente

Recta a la izquierda

Recta en el centro



Rectas a ambos costados

Círculos

¡Nuevas letras!

A partir de dos coordenadas del espacio latente, podemos decodificarlas para obtener una pseudo-letra

```
En ( -0.5 , 0.0 ):  
Decoding complete:  
  XXX  
X  
X  
X  
X  
X  
XXXXX
```

```
En ( 0.0 , 1.0 ):  
Decoding complete:  
XXXXX  
X  
X  
XXXXX  
X X  
X X  
X X
```

```
En ( 0.0 , 0.0 ):  
Decoding complete:  
  XXX  
X  X  
X  
X XX  
X  X  
X  X  
  XXX
```

Denoising

Si se habilita “Noise Training”, la red se entrenará cargando letras con ruido a la capa de input.

Al habilitar el flag “noise” en la función de testeo, se evaluará el output para un input ruidoso. Según su cercanía dentro del espacio latente, es posible predecir de qué letra se trata.

```
XXXXX XXXXX
X      X
X      X
XXX    XXX
X X    X
X      X
XXXXX XXXXX
```

```
X  X  X  X
X  X  X  X
X  X  X  X
  XXXX XXXXX
X  X  X  X
X  X  X  X
X  X  X  X
```

```
X  X  X  X
XX XX XX XX
X X X  X X X
X  X  X  X
X  X  X  X
X  X  X  X
X      X  X
```

```
XXX    XXX
X  X  X  X
X      X
X      X
XX XX  X  XX
X  X  X  X
  XXXX XXXX
```

Resultados

Dataset 2 / 5 letras por tanda / Máx 25000 épocas por tanda / Learning rate 0.05
2 minutos base por tanda / Shuffling y momentum desactivados / Probabilidad de ruido 0.05

	Entrenamiento sin Ruido	Entrenamiento con Ruido
35→17→8→4→2→4→8→17→35	16 aciertos	14 aciertos
35→8→2→8→35	13 aciertos	11 aciertos



Ejercicio 2

Set de Kanji

田

CAMPO

山

MONTAÑA

出

AFUERA

上

ARRIBA

下

ABAJO

中

MEDIO

百

CIEN

子

NIÑO

日

SOL

本

LIBRO

目

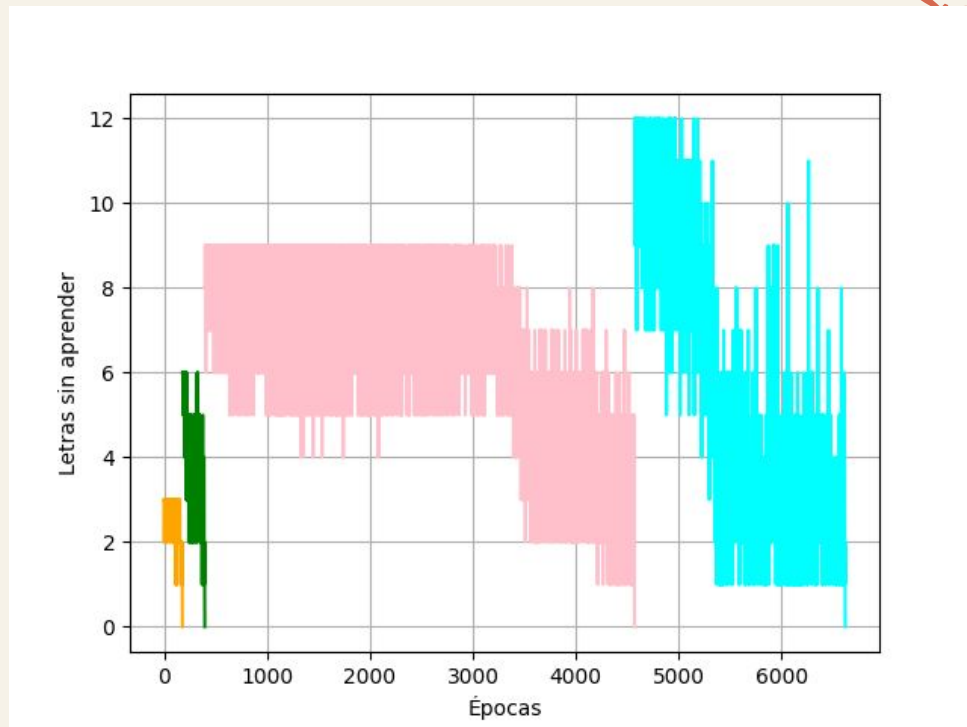
OJO

力

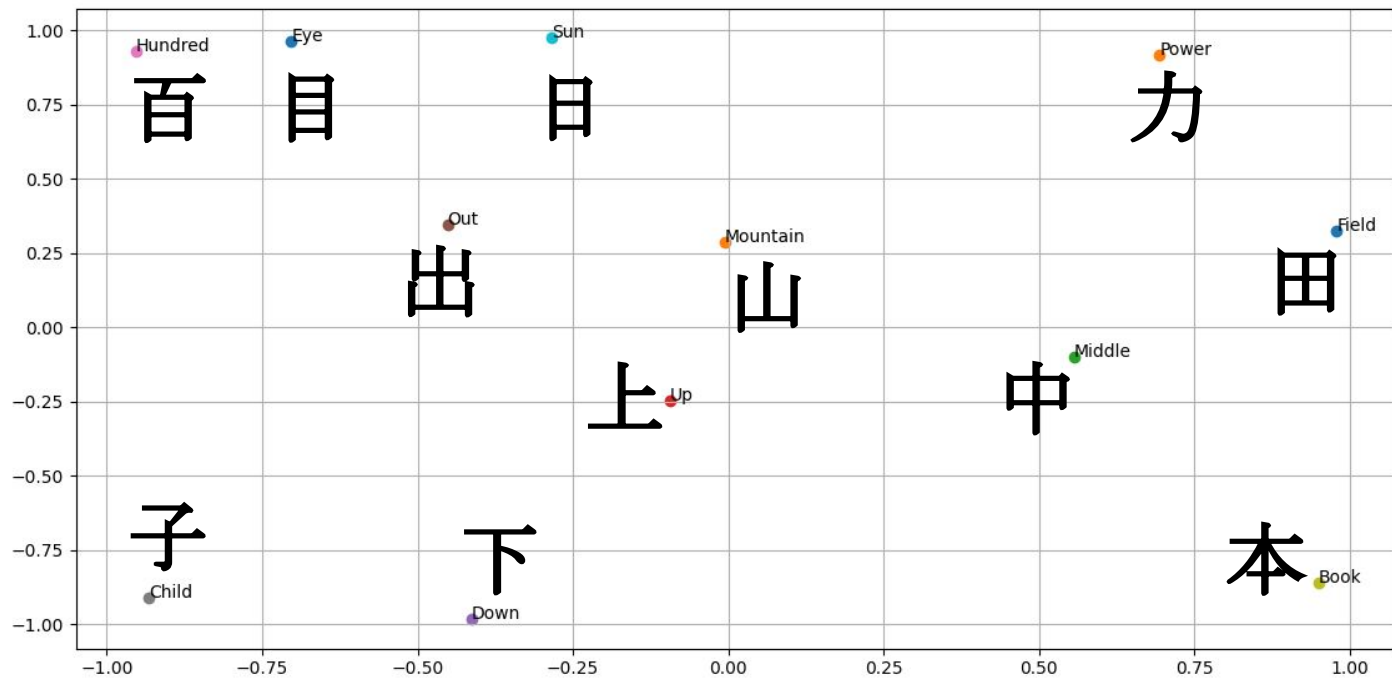
PODER

Entrenamiento

- **Shuffling y Momentum**
Desactivados
- **Tasa de Aprendizaje**
0.05
- **Arquitectura**
35 → 14 → 5 → 2 → 5 → 14 → 35 (factor 2.5)
- **Condición de Corte**
3 minutos por tanda / 5000 épocas
- **Tamaño de salto**
3 letras nuevas por tanda



Espacio Latente



Nuevos Kanji

上

X
X
XXX
X
X X X
X X
XXXXX

上

X
X
XXX
X X
X
X
XXXXX

忠

X
XXXXX
X X X
XXXXX
X
X X X
X XXX

山

X
X X
X X X
X X X
X X X
X X X
XXXXX

力

X
XXXXX
X X
X X
X
XX X
X XX

十

X
XXXXX
X
XXX
X
X
X



Demo



¡Gracias!

