

# Aprendizaje no supervisado

72.27 - Sistemas de Inteligencia Artificial

# Grupo 7

Luque Meijide, Manuel - 57386

Karpovich, Lucía - 58131

Tarradellas del Campo, Manuel - 58091

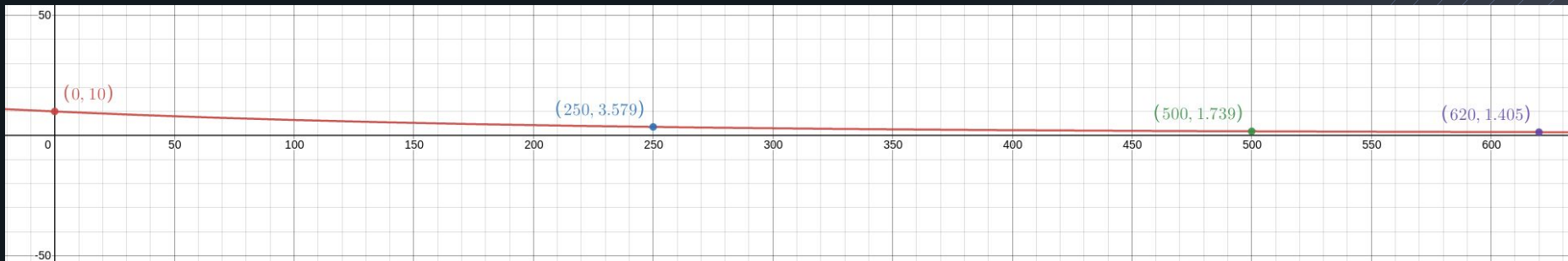
1.

# Modelo de Kohonen

# Implementación

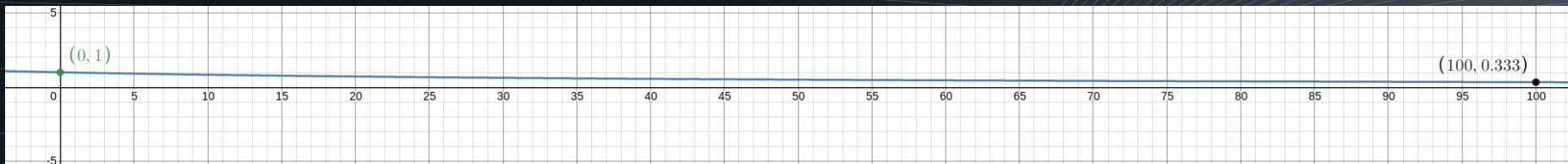
Funcion de actualizacion del radio de vecinos:

$$1 + (10 - 1) \cdot e^{-0.005x}$$



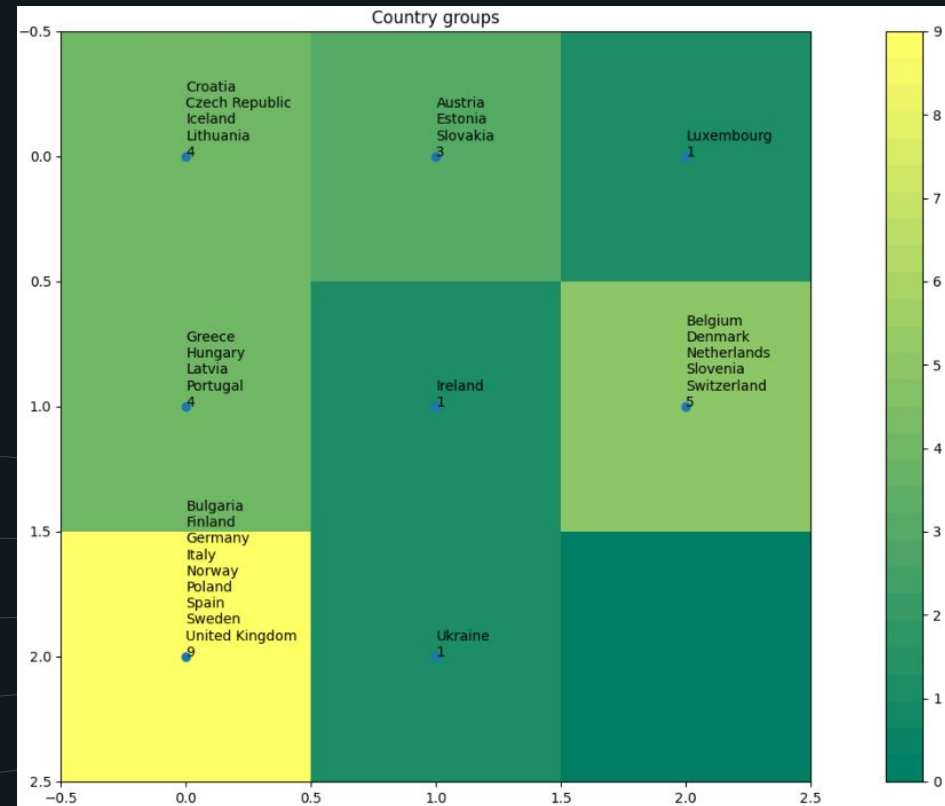
Funcion de actualizacion del learning rate:

$$\frac{1}{(0.02x) + 1}$$

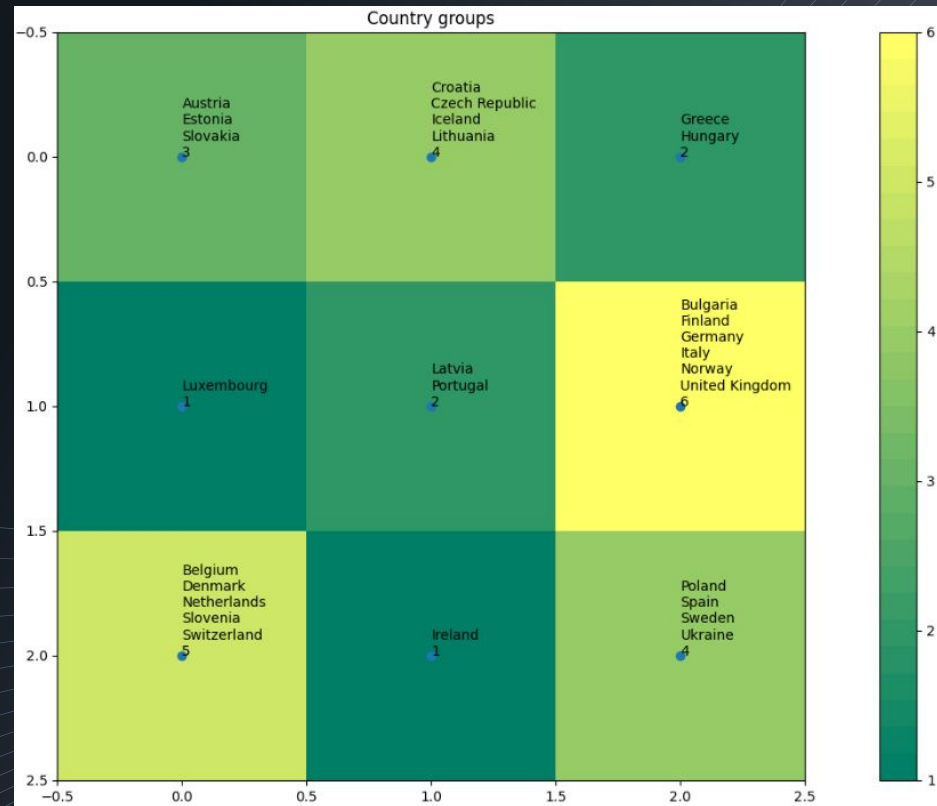


Rectangular ~ 1000\*input iteraciones ~ Wo random

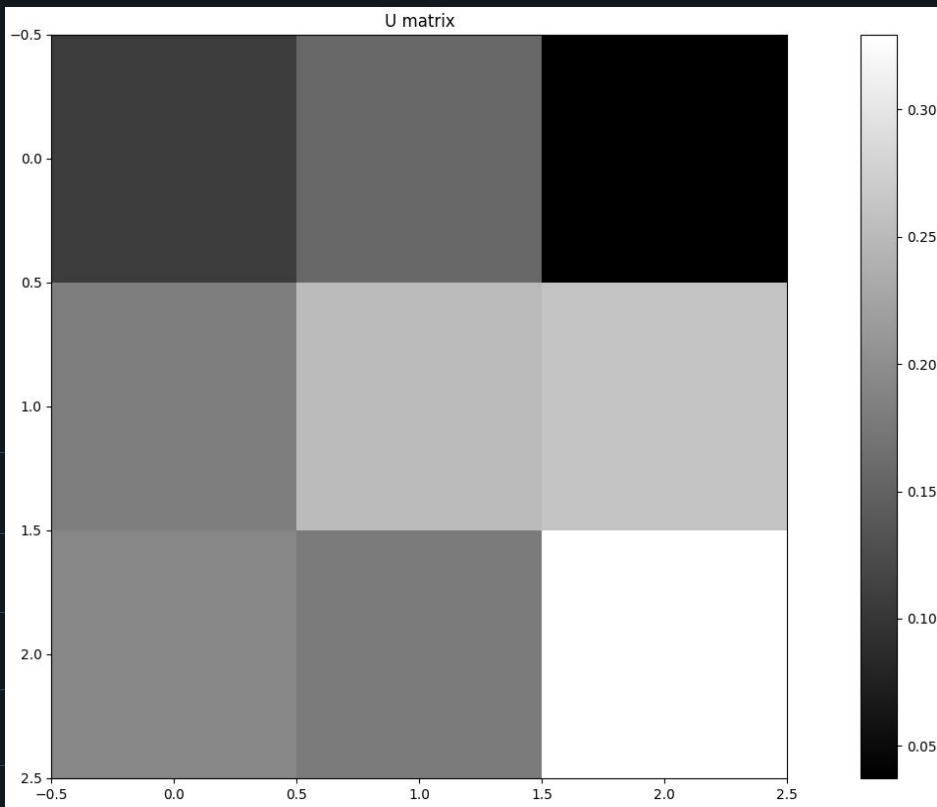
$R_0 = 10$



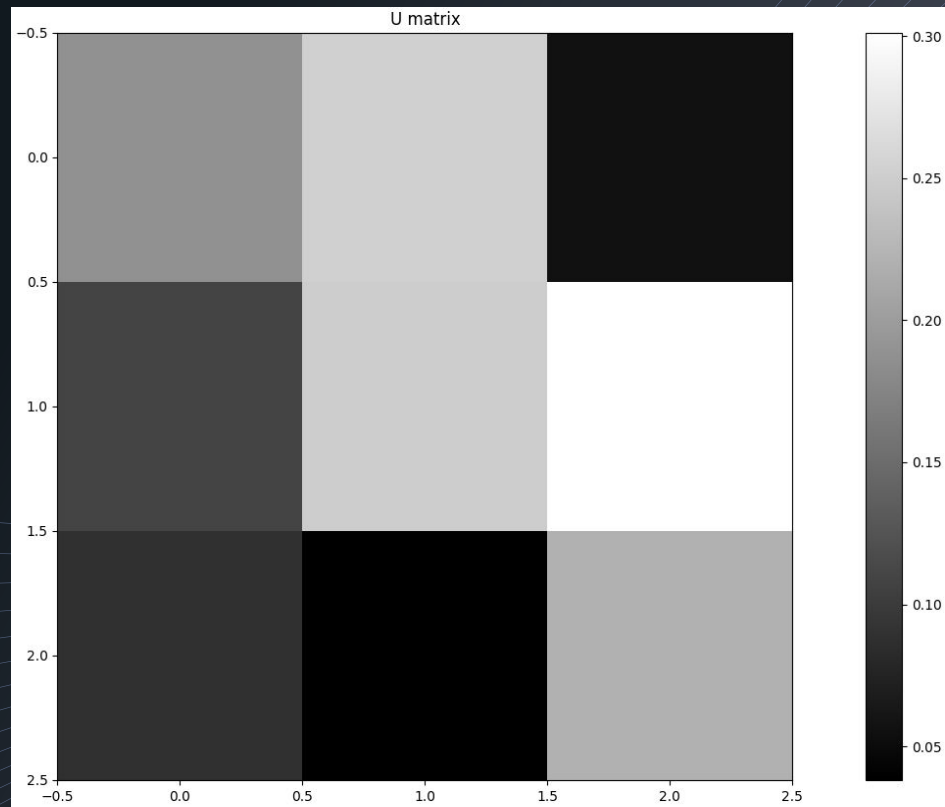
$R_0 = 1$



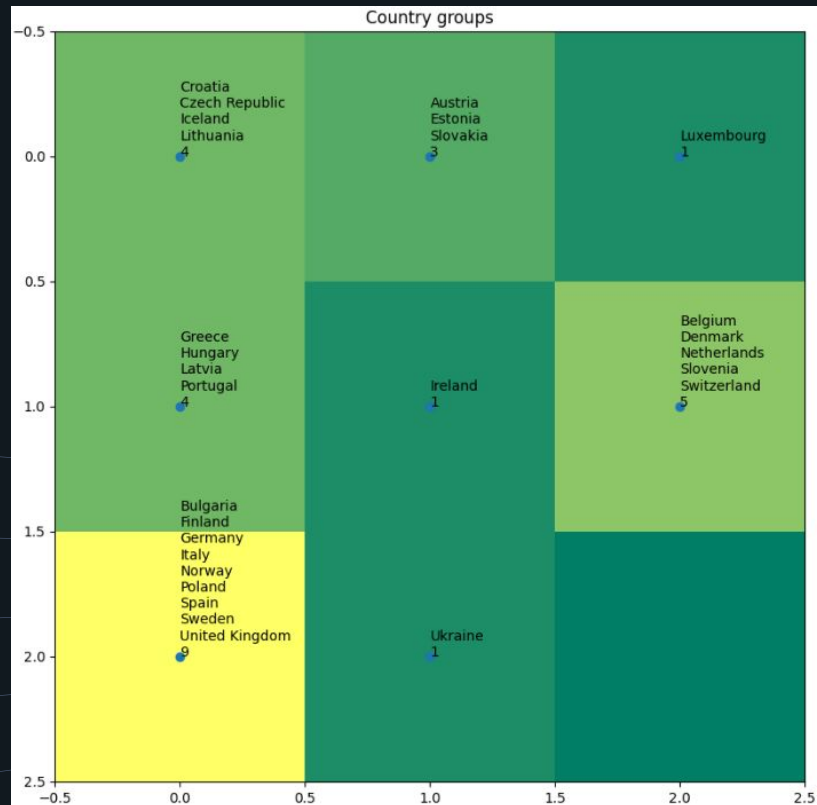
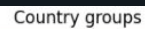
$Ro = 10$



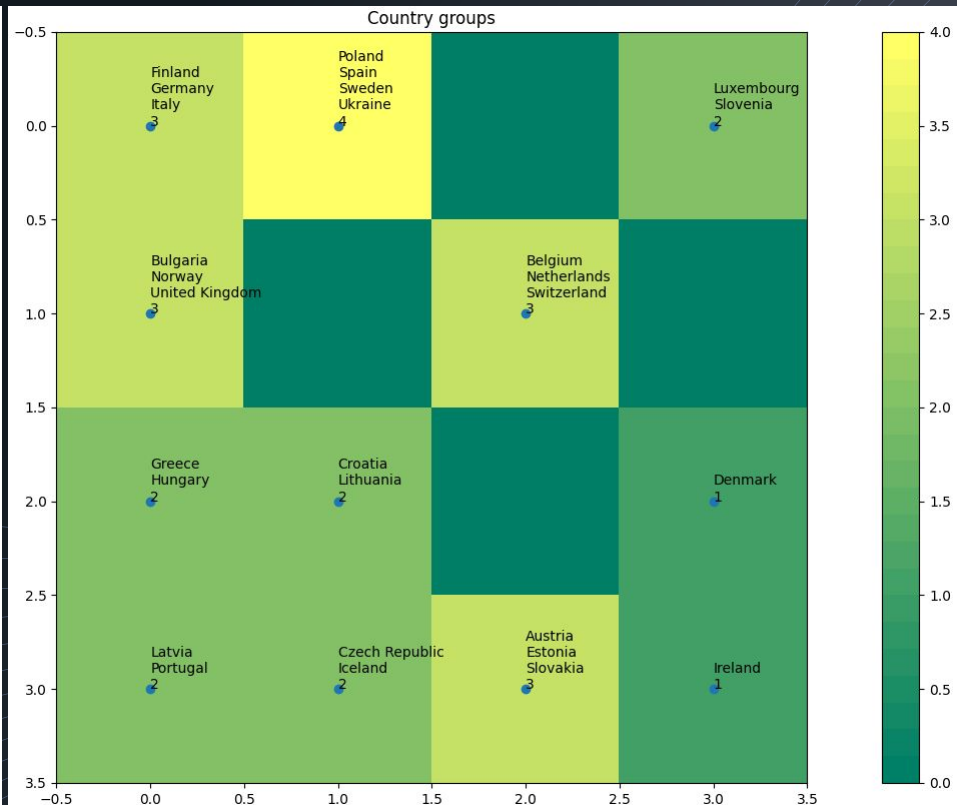
$Ro = 1$



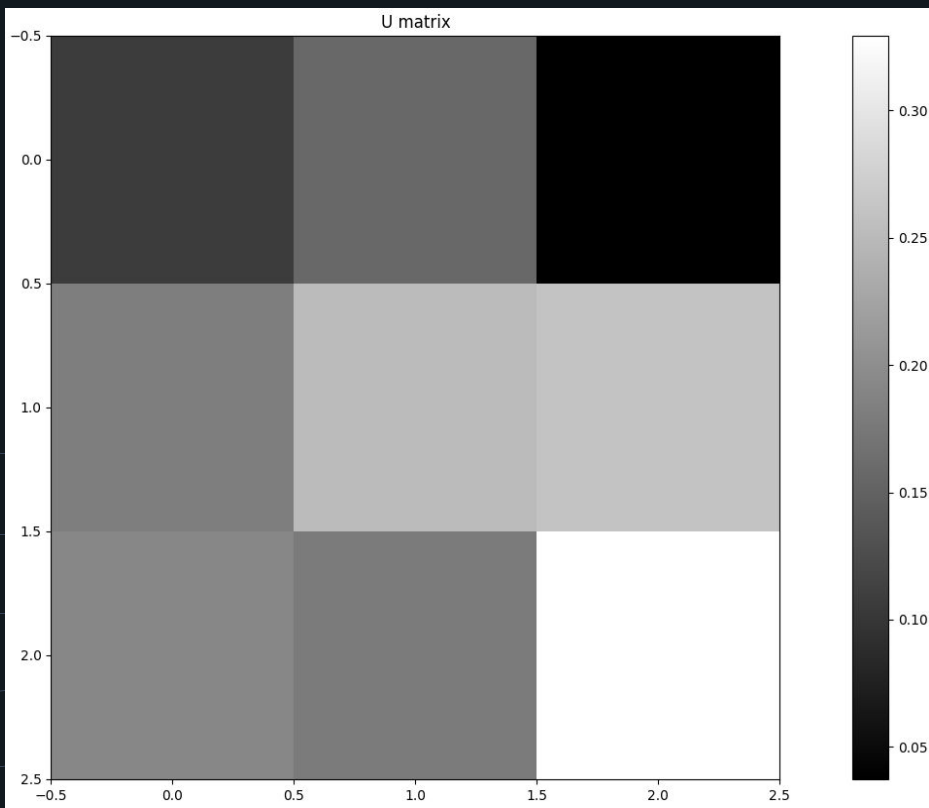
# K = 3



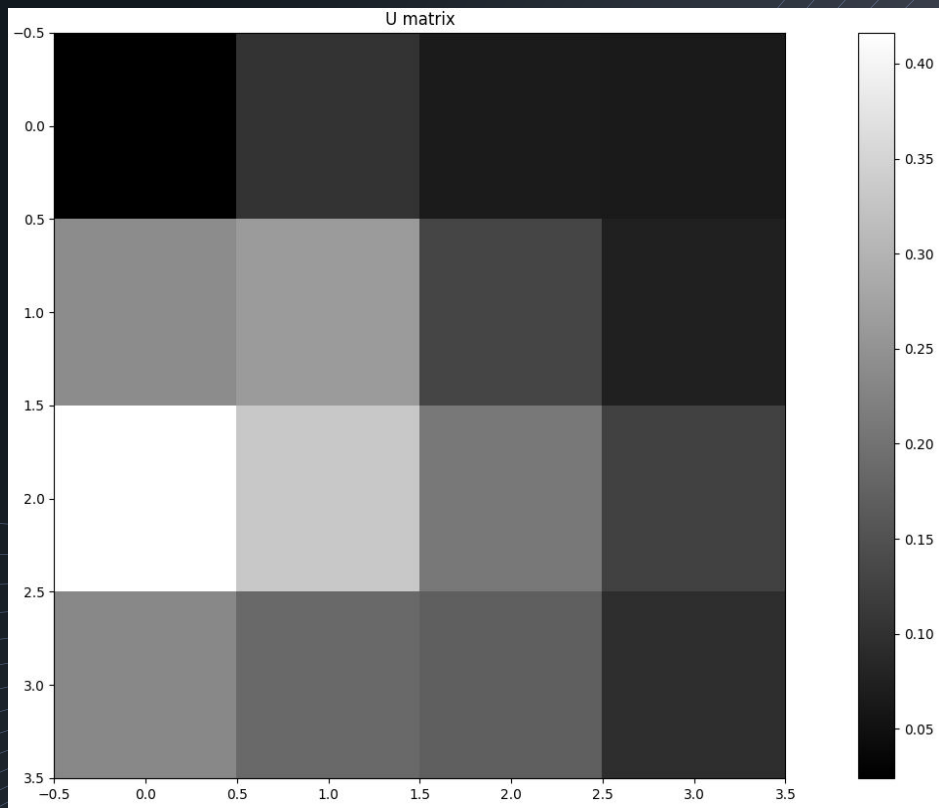
**K = 4**



**K = 3**



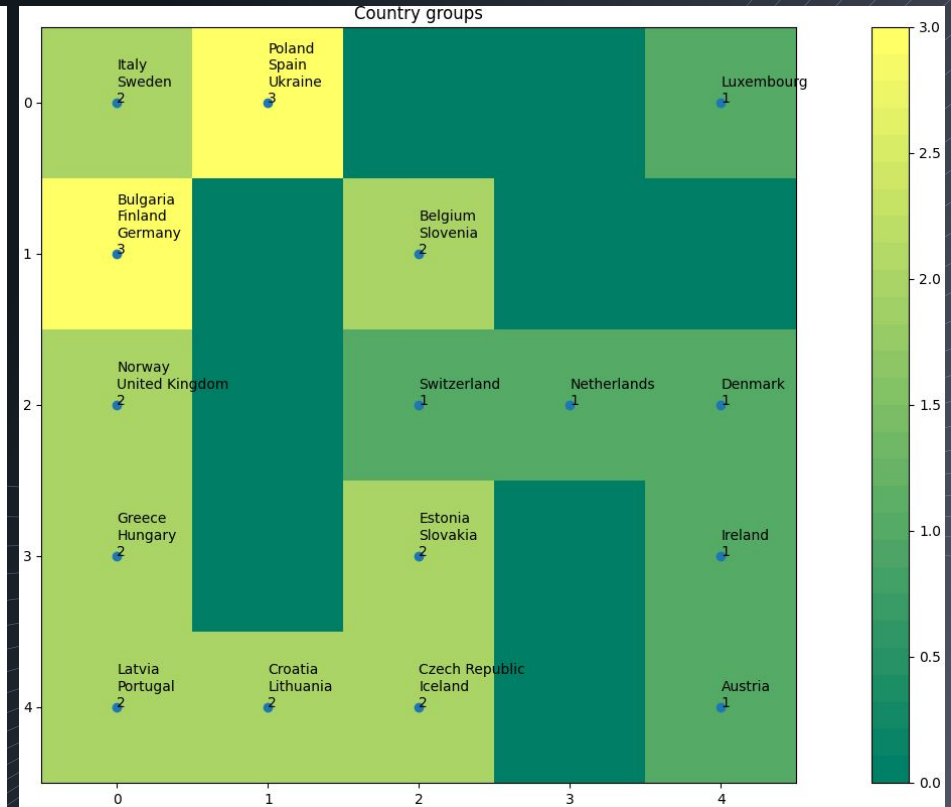
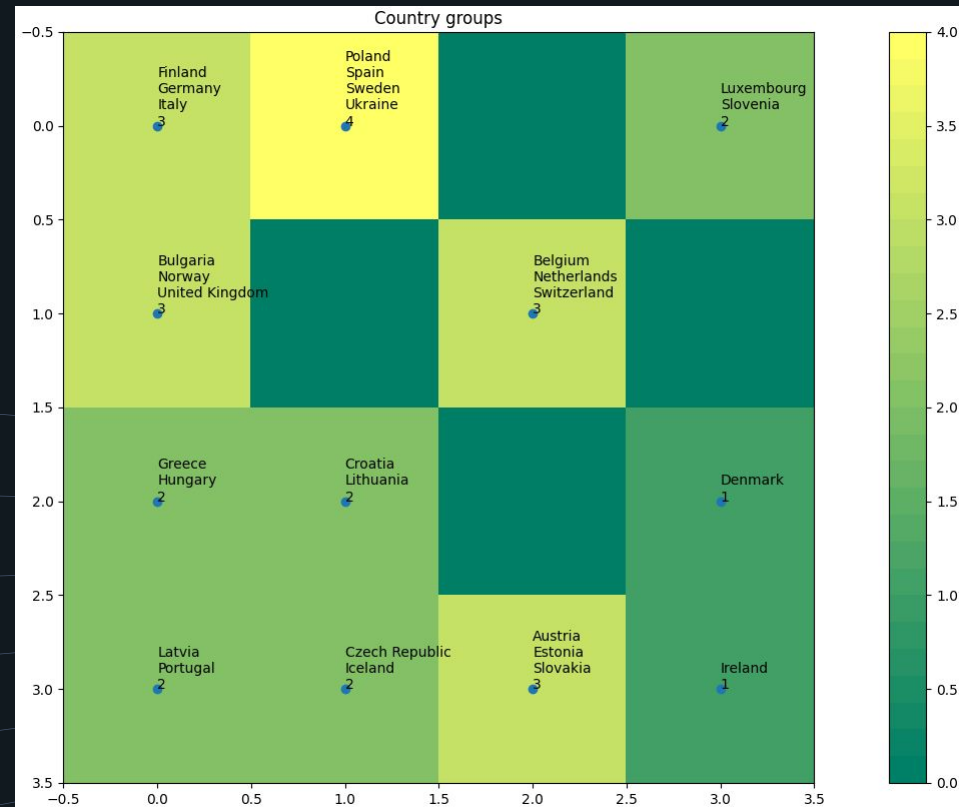
**K = 4**



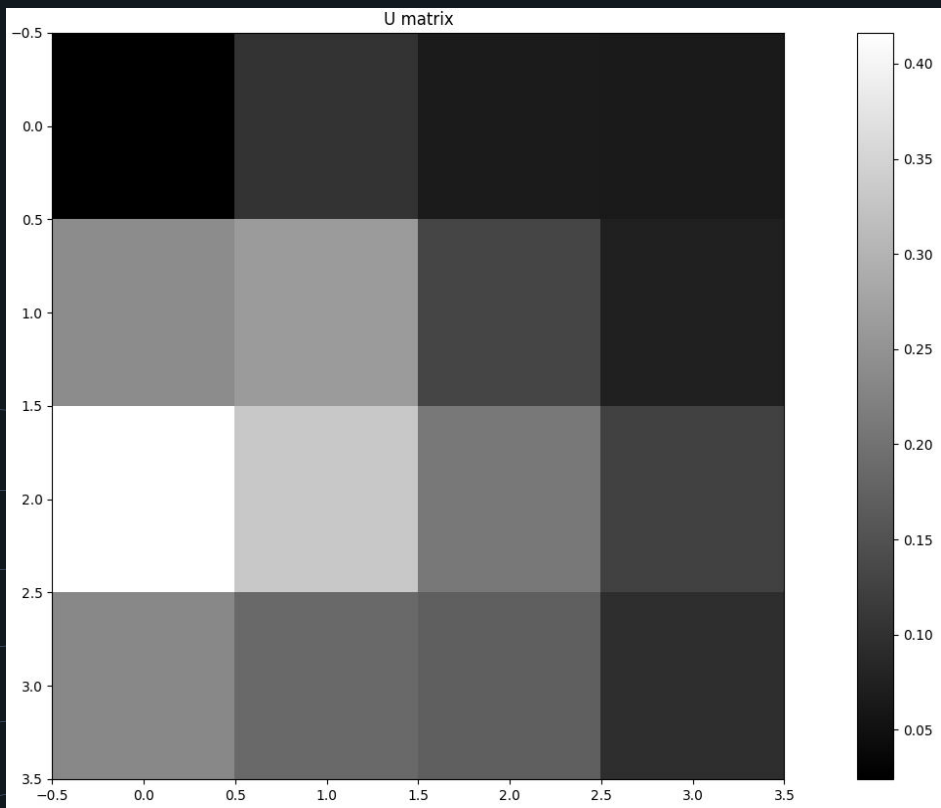


K = 4

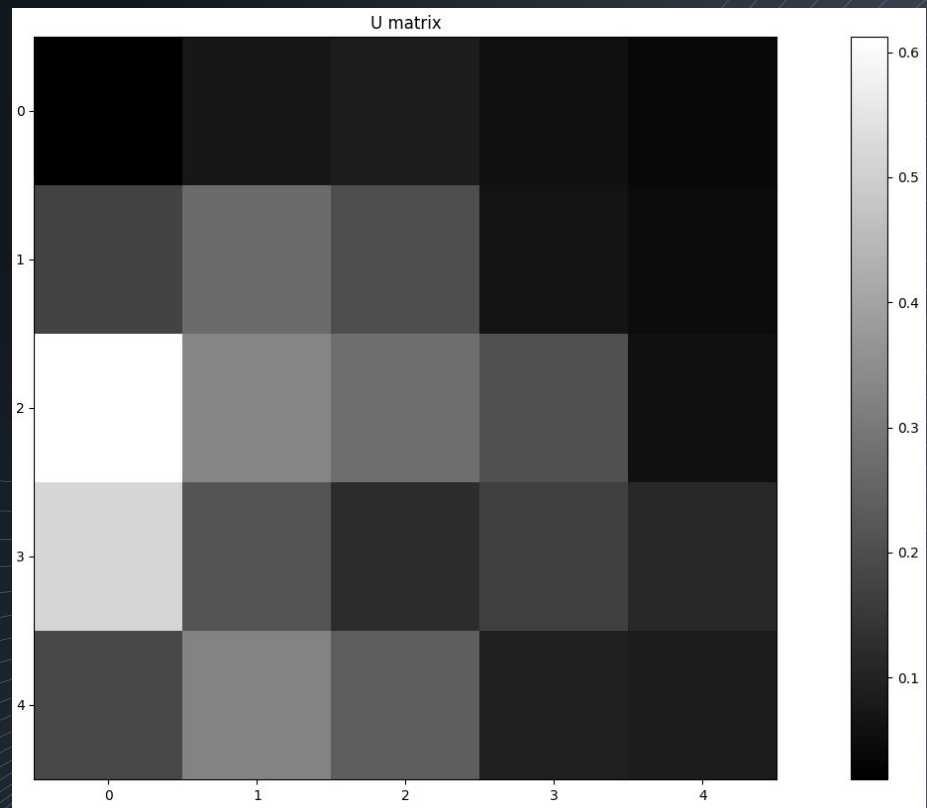
K = 5



$K = 4$



$K = 5$



2.

## Regla de Oja

# Regla de Oja

Hebb

$$\Delta w_j = \eta y^n x_j^i$$

La norma de W aumenta a cada paso

A partir de la propuesta de Oja podemos obtener la primera componente de PCA

W converge al autovector correspondiente al mayor autovalor de la matriz de correlaciones de los datos de entrada

Oja

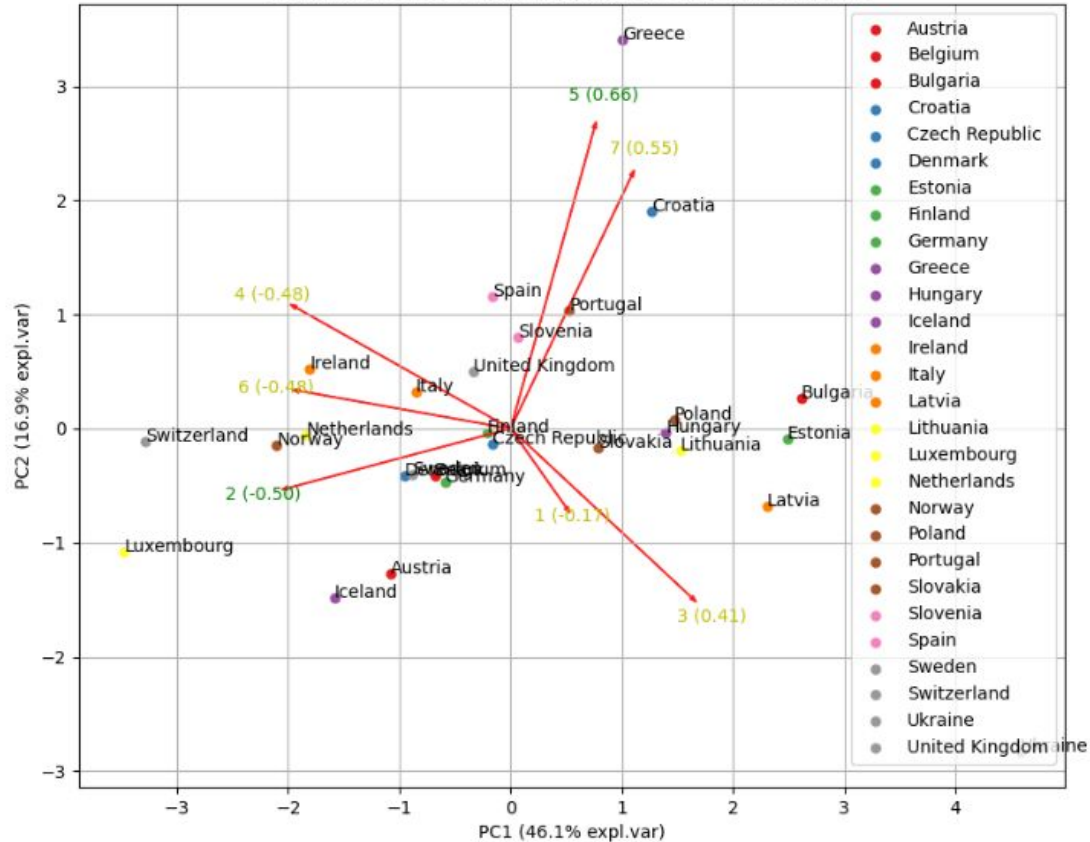
$$\Delta w_j = \eta (y * x_j^n - y^2 * w_j^n)$$

1	OJA	PCA	ERROR
2	1.08174762	-1.08174766	0.00000004
3	0.68109405	-0.68109407	0.00000002
4	-2.60987882	2.60987882	0
5	-1.27014881	1.27014885	0.00000004
6	0.16720948	-0.16720949	0.00000001
7	0.95519078	-0.95519080	0.00000002
8	-2.48773523	2.48773522	0.00000001
9	0.21056317	-0.21056316	0.00000001
10	0.59239365	-0.59239365	0
11	-1.00047185	1.00047196	0.00000011
12	-1.39689832	1.39689831	0.00000001
13	1.58371966	-1.58371970	0.00000004
14	1.80891762	-1.80891761	0.00000001
15	0.85322399	-0.85322396	0.00000003
16	-2.30605944	2.30605941	0.00000003
17	-1.53009993	1.53009991	0.00000002
18	3.47843492	-3.47843496	0.00000004
19	1.8400534	-1.84005341	0.00000001
20	2.10651084	-2.10651083	0.00000001
21	-1.47177382	1.47177383	0.00000001
22	-0.52649331	0.52649333	0.00000002
23	-0.78296599	0.78296597	0.00000002
24	-0.06754337	0.06754338	0.00000001
25	0.16376702	-0.16376696	0.00000006

26	0.88510532	-0.88510531	0.00000001
27	3.28158612	-3.28158613	0.00000001
28	-4.58026812	4.58026807	0.00000005
29	0.34081938	-0.34081935	0.00000003

Learning rate: 0.005  
 Librería de comparación: PCA

2 Principal Components explain [63.06%] of the variance



**Biplot de los dos  
componentes  
principales obtenidos  
de la librería PCA**

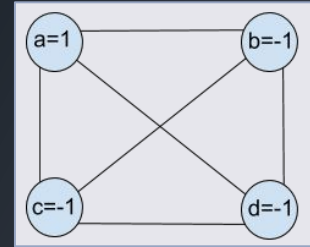
3.

## Redes de Hopfield

# Red de Hopfield

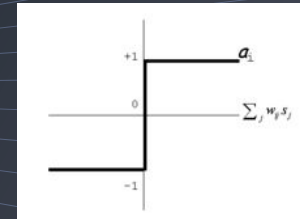
- Basado en memoria asociativa
- R.N.A recurrente (todos los nodos están conectados) - *fig.1*
- Función de activación escalonada - *fig.2*
- Ningún está conectado consigo mismo

Fig.1 - Red de Hopfield



$$V = \begin{bmatrix} +1 \\ -1 \\ -1 \\ +1 \end{bmatrix}, W = \begin{bmatrix} w_{aa} & w_{ab} & w_{ac} & w_{ad} \\ w_{ba} & w_{bb} & w_{bc} & w_{bd} \\ w_{ca} & w_{cb} & w_{cc} & w_{cd} \\ w_{da} & w_{db} & w_{dc} & w_{dd} \end{bmatrix}$$

Fig.2 - Función de activación escalonada



Un vector de estado y una matriz de peso describen a la red en un punto del tiempo.



# Pero, ¿por qué funciona?

- La red tiene asociada una función de energía (demostrado por Hopfield)
  - Esta función depende de  $W$  y los estados de la red:

$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} V_i V_j$$
$$E = -\frac{1}{2} V^T W V$$

- La función de energía es siempre decreciente (o constante) hasta que llegue a un patrón estable.
- Además, ese patrón estable es el que se corresponde con el patrón almacenado, un mínimo local de la función de energía.

**Limitaciones:** El número máximo de patrones que puede almacenar es igual al 15 % del número de neuronas de la red.

# Implementación

## Patrones almacenados de 5x5

-1	1	1	1	-1
1	-1	-1	-1	1
1	1	1	1	1
1	-1	-1	-1	1
1	-1	-1	-1	1



A

## Limitaciones

N = cantidad de neuronas

P = cantidad de patrones a almacenar

Dado que  $N = 25$ ,

$P \leq 0,15 * N \rightarrow P \leq 3,75$

Luego, nuestra red puede almacenar 3 patrones.

# 1º experimento: variar la cantidad de patrones almacenados P

P = 3

Patrones almacenados: [A, C, H]

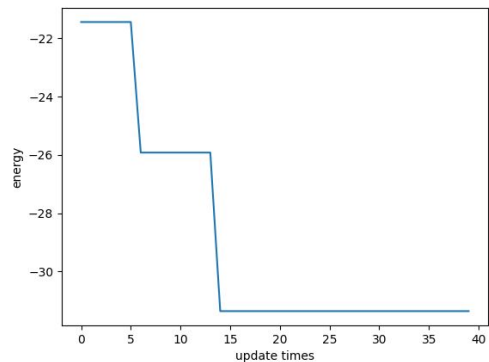
<p>Pattern to test:</p> <pre>111 1 1 11 1 1 1 1 1</pre> <p>RESULT:</p> <pre>111 1 1 11111 1 1 1 1</pre>	<p>Pattern to test:</p> <pre>1 1 1 1 1 1111</pre> <p>RESULT:</p> <pre>1111 1 1 1 1111</pre>	<p>Pattern to test:</p> <pre>1 1 1 1 1 11 1 1 1 1</pre> <p>RESULT:</p> <pre>1 1 1 1 11111 1 1 1 1</pre>
---	---	---

Nota: para estos experimentos, la cantidad de valores alterados es aleatoria e igual a 2

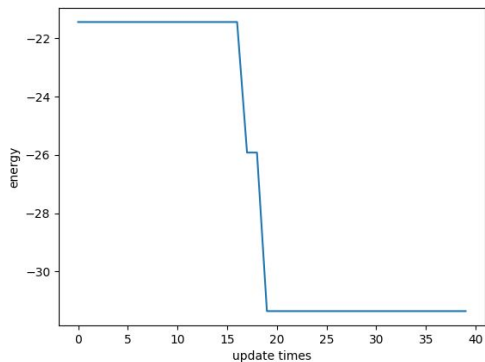
# Energía

$$P = 3$$

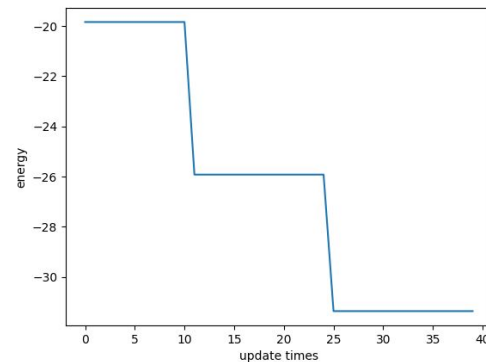
A



C



H



# 1º experimento: variar la cantidad de patrones almacenados P

P = 4

Patrones almacenados: [A, C, E, H]

Pattern to test:

```
111
1 1
11 1
1 1
1 1
```

RESULT:

```
11111
1 1
11111
1 1
1 1
```

Pattern to test:

```
1 1
1
1
1
1111
```

RESULT:

```
1111
1
1
1
1111
```

Pattern to test:

```
11111
1
11111
1
1111
```

RESULT:

```
11111
1
1111
1
11111
```

Pattern to test:

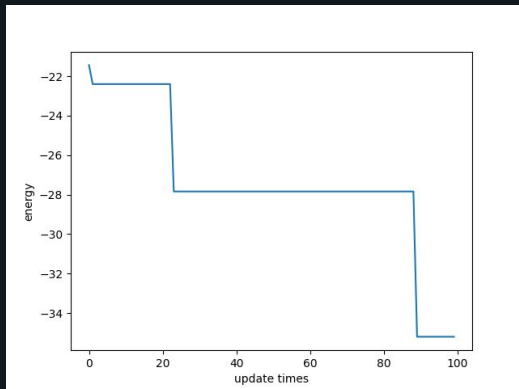
```
1 1
1 1
1 11
1 1
1 1
```

RESULT:

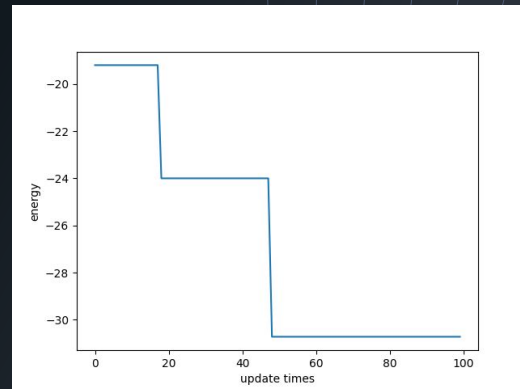
```
1 1
1 1
11111
1 1
1 1
```

# Energia

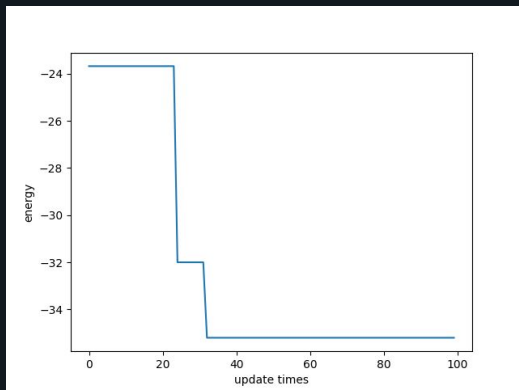
A



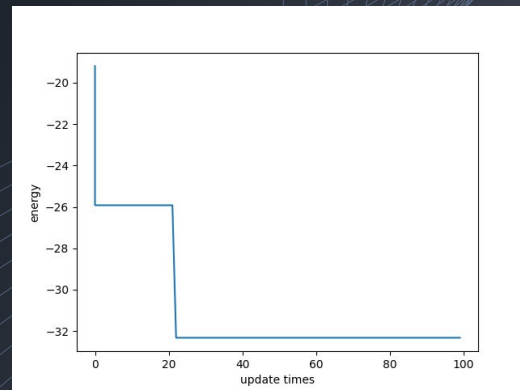
C



E



H



# 1º experimento: variar la cantidad de patrones almacenados P

P = 5

Patrones almacenados: [A, C, E, H, J]

Pattern to test:

```
111
1  1
11 1
1  1
1  1
```

RESULT:

```
11111
1  1
11111
1  1
1  1
```

Pattern to test:

```
 1 1
1
1
1
 1111
```

RESULT:

```
11111
1
1111
1
11111
```

Pattern to test:

```
11111
1
11111
1
1111
```

RESULT:

```
11111
1
1111
1
11111
```

Pattern to test:

```
1  1
1  1
 1 11
1  1
1  1
```

RESULT:

```
1  1
1  1
11111
1  1
1  1
```

Pattern to test:

```
11 11
  1
  1
1
1111
```

RESULT:

```
11111
  1
  1
1  1
1111
```

## 2º experimento: variar ruido

Patrones almacenados: [A, H, J]

Pattern to test:

```
1 1
1 1
11 1
1
1
```

RESULT:

```
1 1
11111
1 1
1 1
```

Pattern to test:

```
1
1 1
1 11
1 1
1
```

RESULT:

```
1 1
11111
1 1
1 1
```

Pattern to test:

```
11 11
1
1
1
11
```

RESULT:

```
11111
1
1
1 1
111
```



## 2º experimento: variar umbral

Patrones almacenados: [A, H, J]

Mismo experimento, pero variando el umbral

$\alpha = 0.2$ ;  $\alpha = 0.5$ ;  $\alpha = 0.7$ ;

Pattern to test:

```
1 1
1 1
11 1
1
1
```

RESULT:

```
1 1
11111
1 1
1 1
```

Pattern to test:

```
1
1 1
1 11
1 1
1
```

RESULT:

```
1 1
11111
1 1
1 1
```

### Estado espúreo

- Patrones de actividad que no se han incorporado explícitamente en la matriz sináptica, pero que, sin embargo, son estables.
- Son atractores "no deseados" que se producen como un mínimo local en la función de energía.