



Aprendizaje No Supervisado - TP4

Grupo 11

Francisco Pérez Sammartino
& Gabriel Silvatici



1

Análisis de Datos Europa

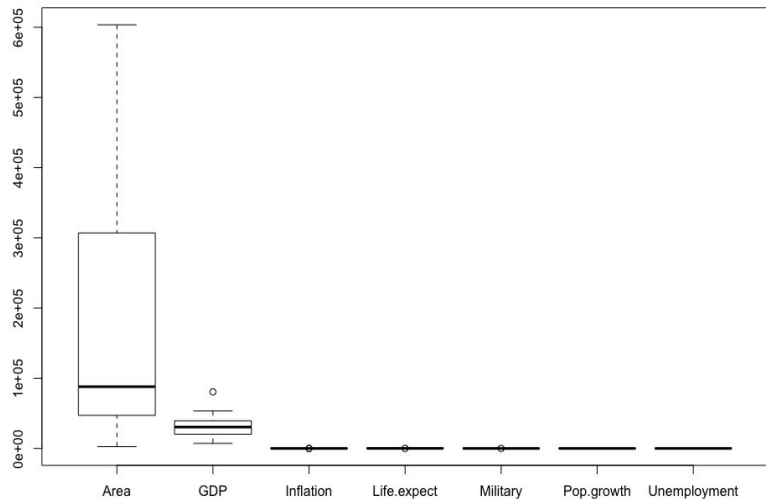
¿Cómo son los datos?

- 8 variables -> 7 numéricas
- 28 registros

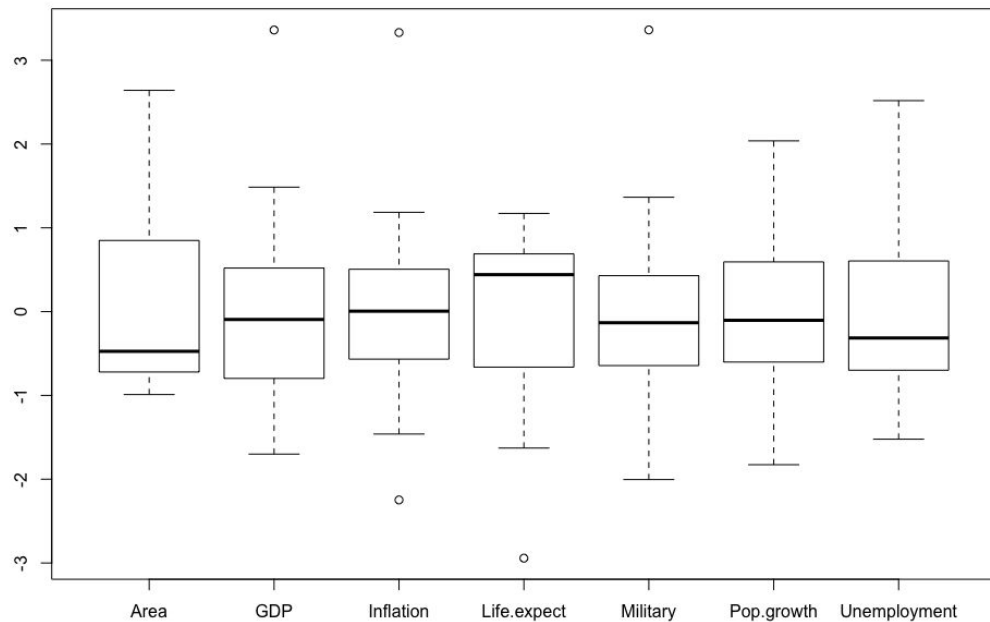
Country	Area	GDP	Inflation	Life.expect	Military	Pop.growth	Unemployment
Austria	83871	41600	3.5	79.91	0.8	0.03	4.2
Belgium	30528	37800	3.5	79.65	1.3	0.06	7.2
Bulgaria	110879	13800	4.2	73.84	2.6	-0.8	9.6
Croatia	56594	18000	2.3	75.99	2.39	-0.09	17.7
Czech Republic	78867	27100	1.9	77.38	1.15	-0.13	8.5
Denmark	43094	37000	2.8	78.78	1.3	0.24	6.1
Estonia	45228	20400	5	73.58	2	-0.65	12.5

Boxplot

Raw Data



Standardized Data





a

Red de Kohonen

Implementación

- Estandarización de datos
- Función vecinos
Decreciente con un porcentaje.

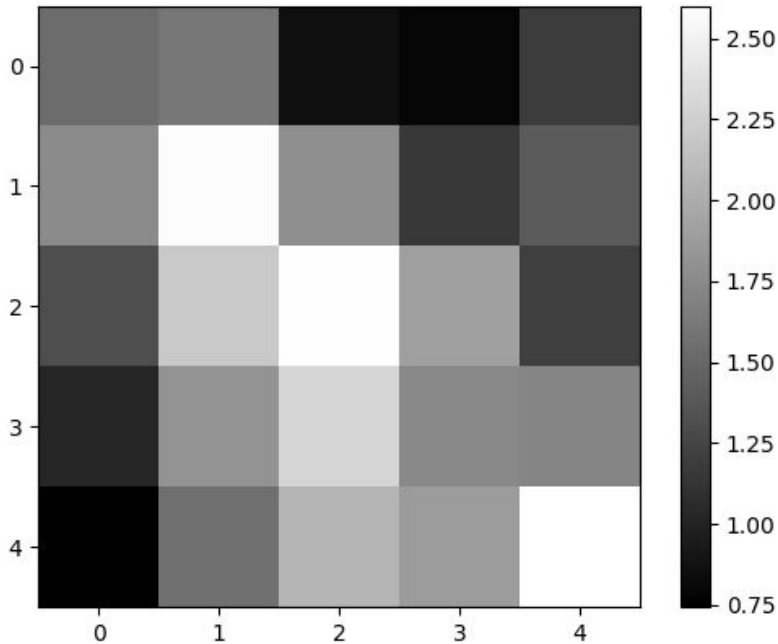
Comenzando en $(\frac{2}{3})^*$
Alto

- Coeficiente de aprendizaje decreciente
con un porcentaje.
Comienza en 0.9

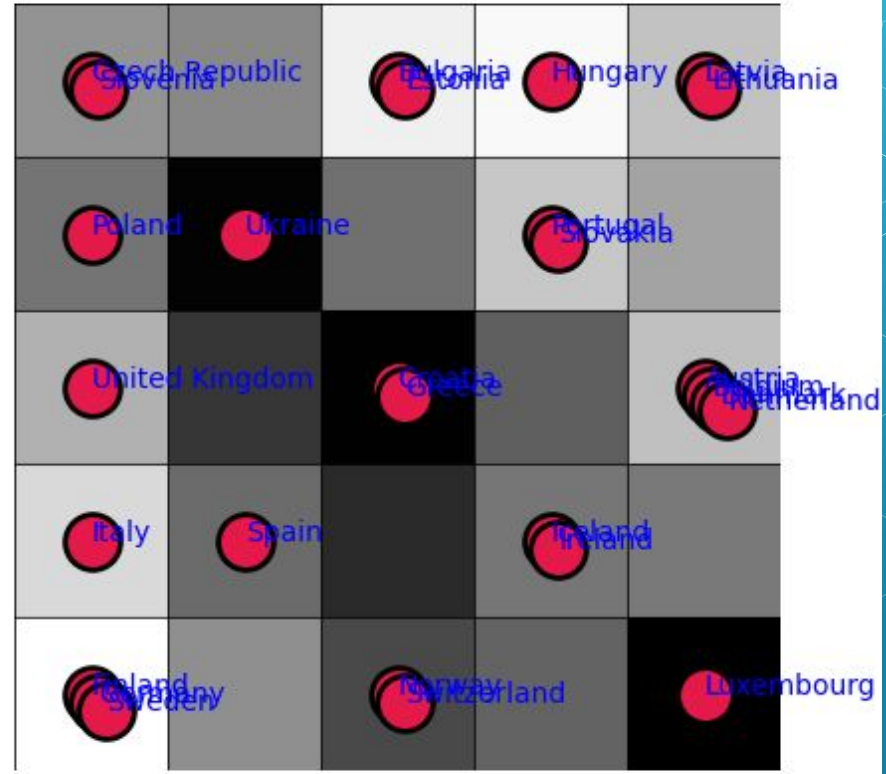
- Mapa rectangular
- Variación del tamaño del mapa

500 épocas

Resultados Grilla Rectangular 5x5

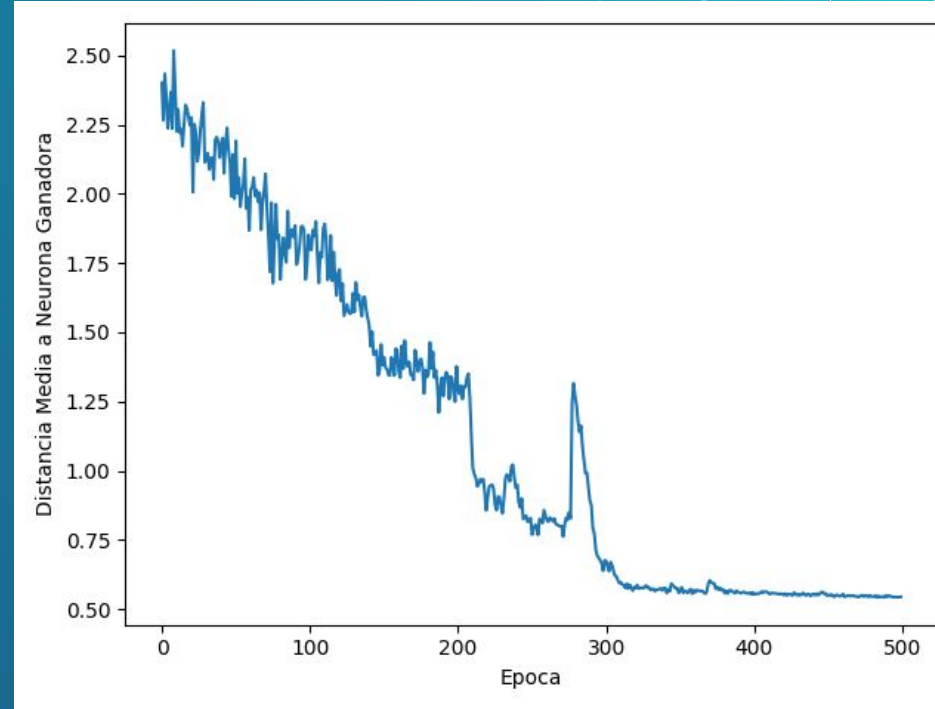
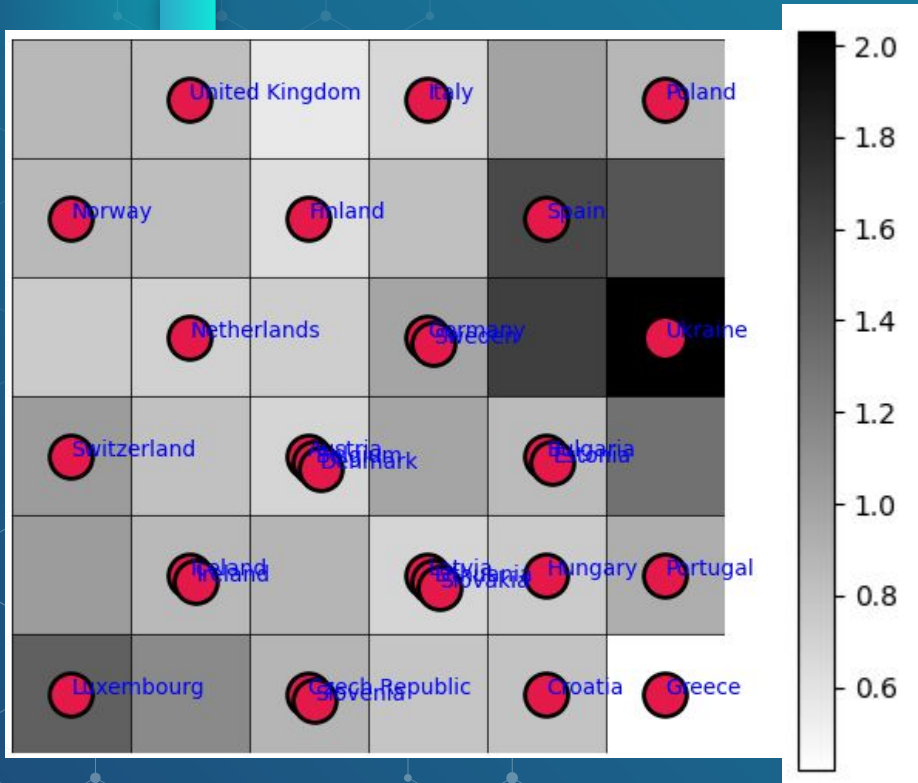


Matriz U



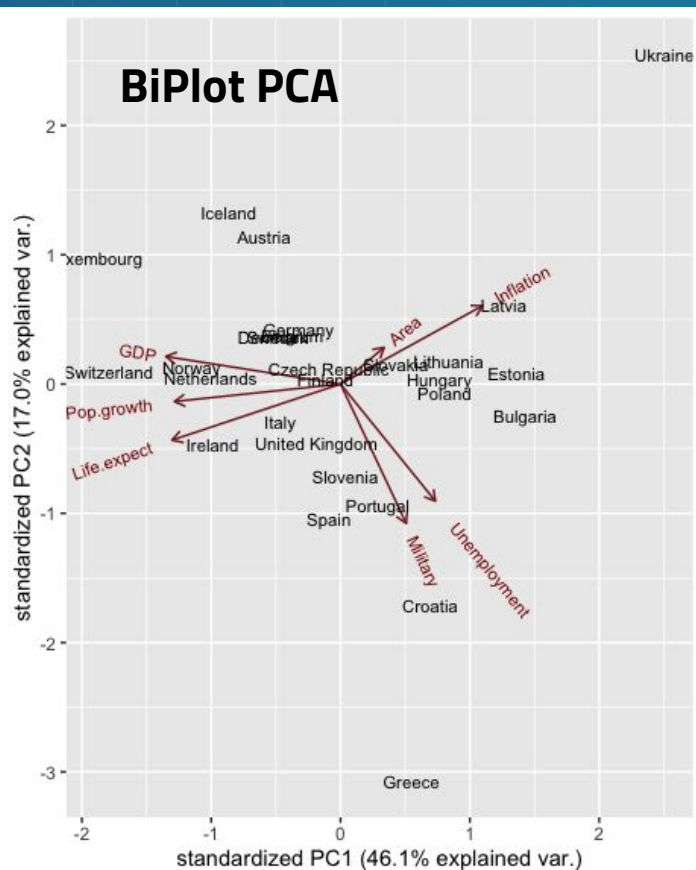
Matriz U invertida con mapeo

Resultados Grilla Rectangular 6x6



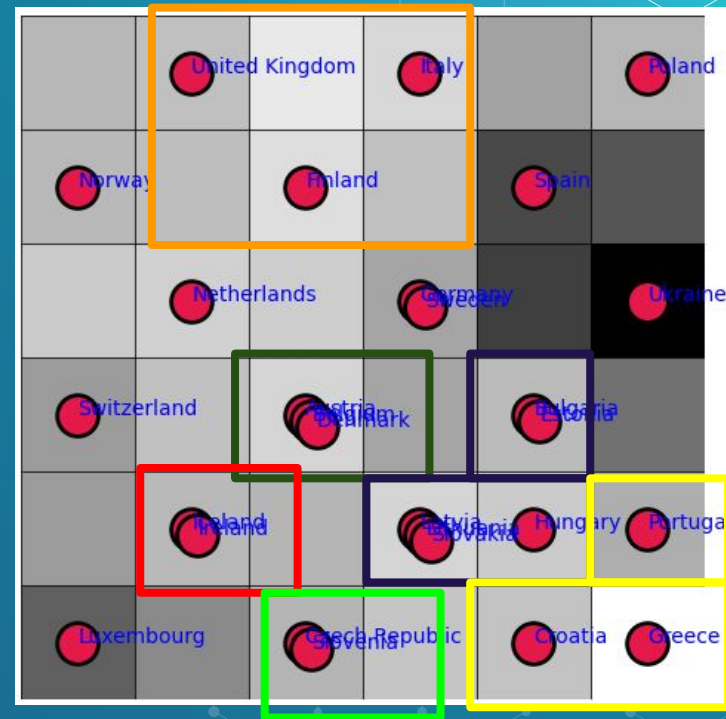
Resultados

BiPlot PCA

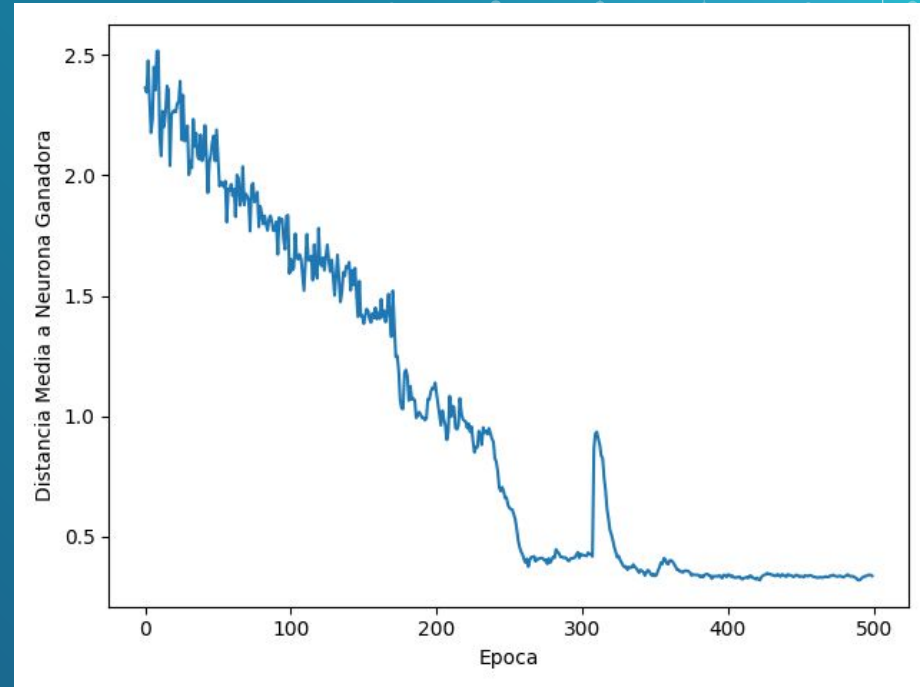
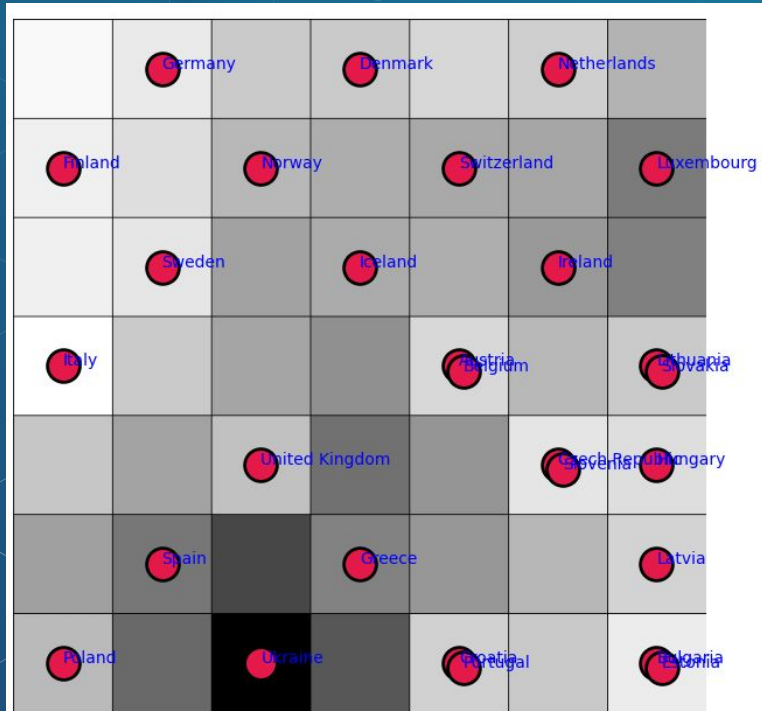


	Comp. 1
Luxembourg	-3.41575532
Switzerland	-3.22245361
Norway	-2.06855257
Netherlands	-1.80689657
Ireland	-1.77632183
Iceland	-1.55518187
Austria	-1.06225512
Denmark	-0.93797874
Sweden	-0.86915616
Italy	-0.83784930
Belgium	-0.66882110
Germany	-0.58171902
United Kingdom	-0.33467796
Finland	-0.20676891
Czech Republic	-0.16419646
Spain	-0.16081596
Slovenia	0.06632628
Portugal	0.51700619
Slovakia	0.76885732
Greece	0.98244396
Croatia	1.24726141
Hungary	1.37172691
Poland	1.44525321
Lithuania	1.50252828
Latvia	2.26450539
Estonia	2.44290749
Bulgaria	2.56285012
Ukraine	4.49773395

Kohonen Grupos

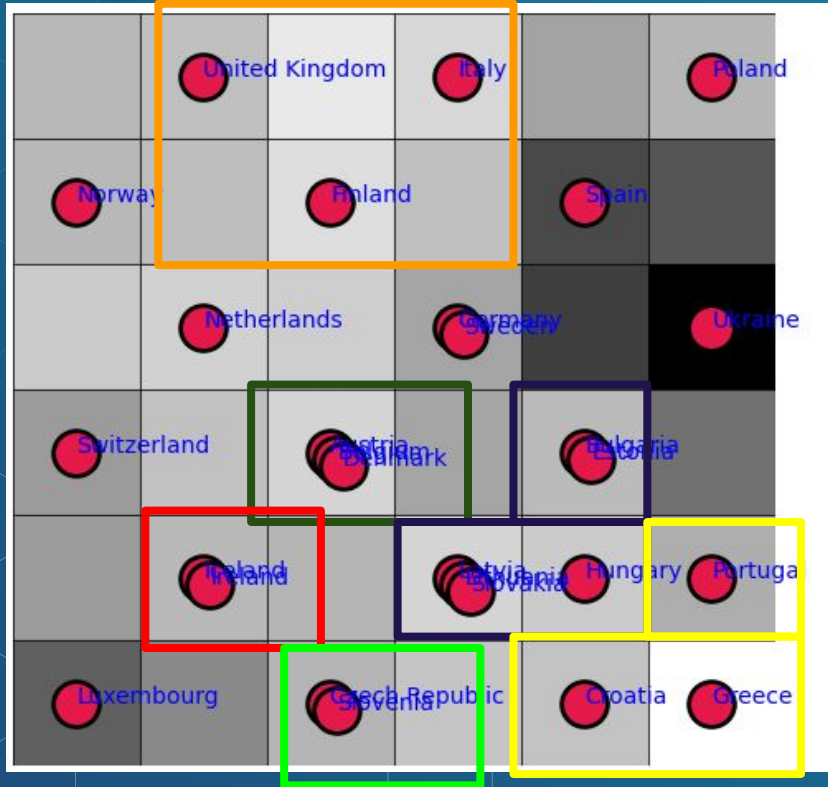


Resultados Grilla Rectangular 7x7

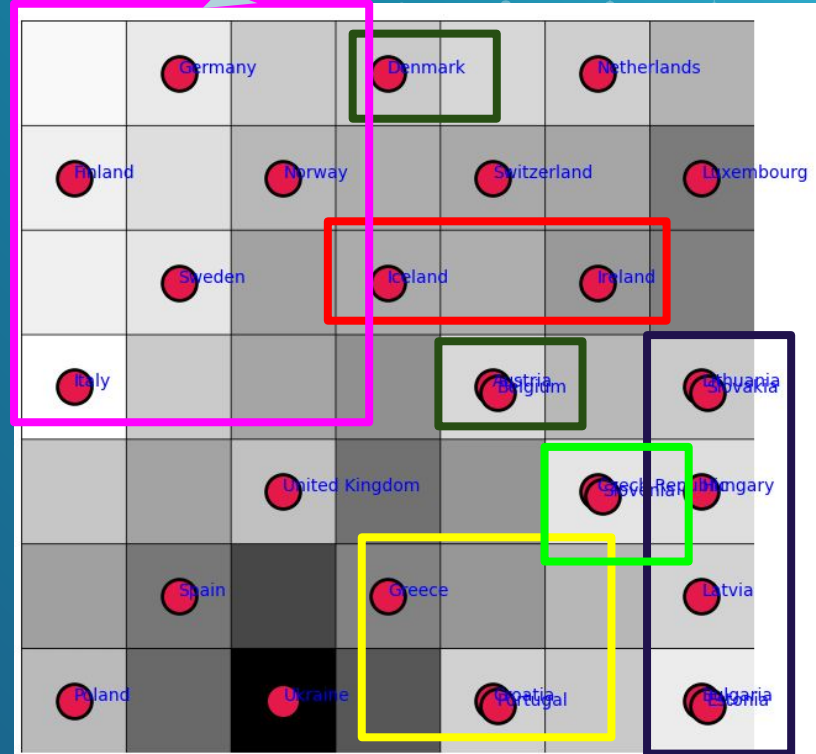


Grilla Rectangular 7x7 vs 6x6

SOM 6x6



SOM 7x7





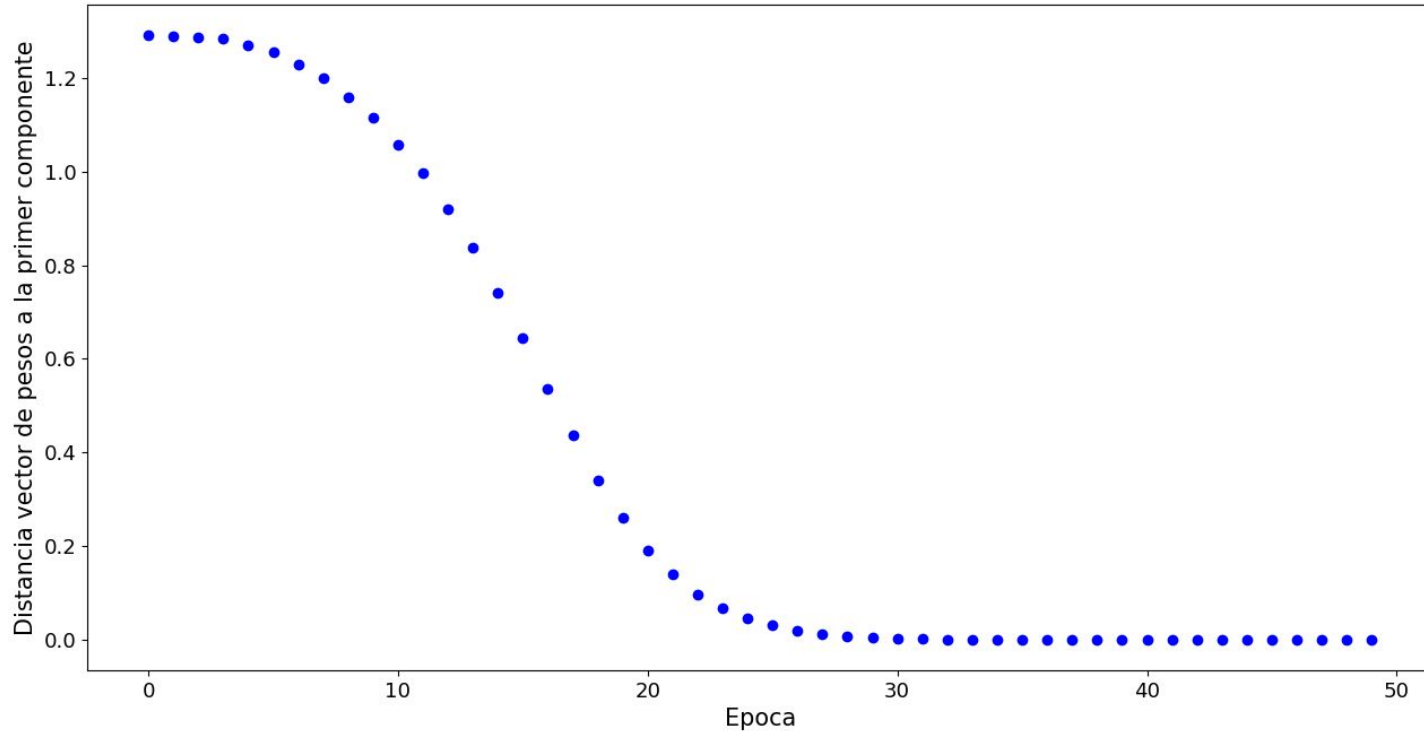
a

Red Neuronal con Regla de Oja

Implementación

- Perceptron simple lineal.
- Estandarización de los datos de entrada.
- Inicialización de pesos aleatorios.
- Tasa de aprendizaje adaptativa.
- Condicion de corte: vector de pesos estacionario y norma de W oscila en 1

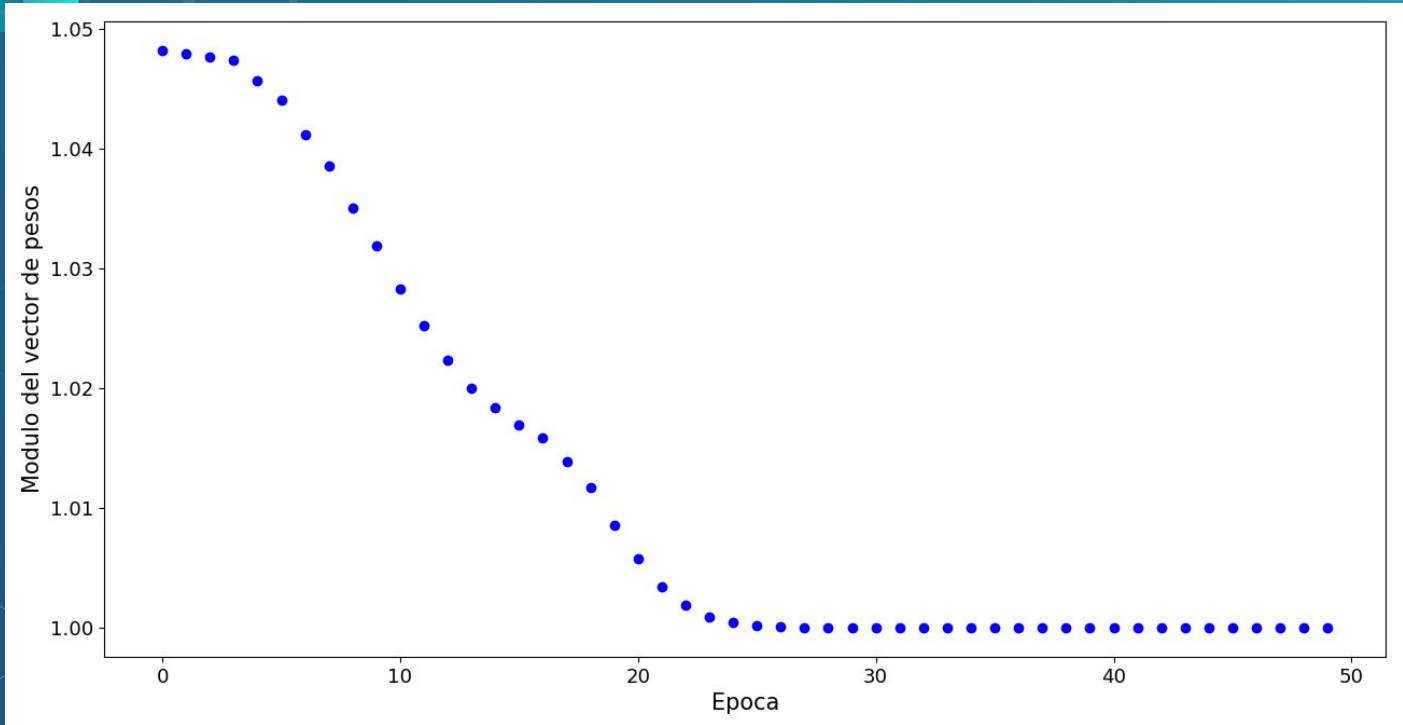
Convergencia a la primer componente



Entrenando los
datos con la red
de Oja
comparando el
vector de pesos
con la primer
componente pre
calculada con
PCA

$$\eta_0 = 0.0001$$

Convergencia a la primer componente



**Evolución del
módulo del
vector de pesos.**

$$\eta_0 = 0.0001$$

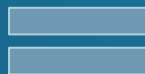
Cálculo de la primer componente principal

Calculando el primer autovector de la matriz de correlaciones

Entrenando los datos con la red de Oja

$\lambda:$ 3.2271657

Area	0.1248739
GDP	-0.5005059
Inflation	0.4065182
Life.expec	-0.4828733
Military	0.1881116
Pop.growth	-0.4757036
Unemployme	0.2716558



Area	-0.1248739
GDP	0.5005059
Inflation	-0.4065182
Life.expec	0.4828733
Military	-0.1881116
Pop.growth	0.4757036
Unemployme	-0.2716558

Interpretando los datos

Entrenando los datos con la red de Oja obtenemos el autovector de autovalor más grande, es decir, la primer componente, este autovector estará en alguno de los 2 posibles sentidos, que estará arbitrado en función de cómo queremos ordenar los datos obtenidos

	Comp.1
Luxembourg	-3.41575532
Switzerland	-3.22245361
Norway	-2.06855257
Netherlands	-1.80689657
Ireland	-1.77632183
Iceland	-1.55518187
Austria	-1.06225512
Denmark	-0.93797874
Sweden	-0.86915616
Italy	-0.83784930
Belgium	-0.66882110
Germany	-0.58171902
United Kingdom	-0.33467796
Finland	-0.20676891
Czech Republic	-0.16419646
Spain	-0.16081596
Slovenia	0.06632628
Portugal	0.51700619
Slovakia	0.76885732
Greece	0.98244396
Croatia	1.24726141
Hungary	1.37172691
Poland	1.44525321
Lithuania	1.50252828
Latvia	2.26450539
Estonia	2.44290749
Bulgaria	2.56285012
Ukraine	4.49773395

The background of the slide features a repeating pattern of hexagons and dots in a light blue color against a darker blue gradient. The hexagons are arranged in a honeycomb-like structure, with dots at their vertices. Some hexagons are more prominent than others, creating a sense of depth.

2

Red de Hopfield



Implementación

- Orden Aleatorio
- Selección de mejores y peores grupos.
- Ruido de 0,10,30,50 y 70% para testeo.

Elección de Patrones

Suma de valor absoluto de producto escalar.

$$\sum_{i \neq j} |letra_i letra_j|$$

Z-M-O-R:8.0

M-O-T-R:8.0

T-B-W-V:8.0

I-L-Y-F:10.0

I-L-Y-P:10.0

H-V-S-Q:10.0

L-X-T-R:10.0

L-Y-W-V:10.0

L-Y-W-F:10.0

L-Y-W-P:10.0

L-Y-V-F:10.0

L-T-W-V:10.0

L-W-V-A:10.0

B-E-F-P:90.0

O-C-D-Q:92.0

O-X-Y-D:94.0

O-X-B-D:94.0

O-C-Q-G:94.0

R-E-F-P:94.0

O-X-Y-G:96.0

O-B-D-G:96.0

O-D-Q-G:96.0

B-R-F-P:96.0

O-C-D-G:98.0

O-U-D-G:98.0

O-X-D-G:100.0

Z:

```

1 1 1 1 1
      1
      1
    1 1 1 1 1
    
```

M:

```

1      1
1 1    1 1
1      1 1
1      1 1
1      1 1
    
```

O:

```

      1 1 1
1      1
1      1
1      1
      1 1 1
    
```

R:

```

1 1 1 1
1      1
1 1 1 1
1      1
1      1
    
```

O:

```

      1 1 1
1      1
1      1
1      1
      1 1 1
    
```

X:

```

1      1
      1 1
      1
      1 1
1      1
    
```

D:

```

1 1 1 1
1      1
1      1
1      1
1 1 1 1
    
```

G:

```

      1 1 1 1
1      1 1
1      1
1      1
      1 1 1
    
```

30% alterado

The image displays a 3x3 grid of 9 sparse matrices, each with 5 rows and 5 columns. The matrices are connected by lines, suggesting a sequence or relationship. Each matrix contains a different pattern of 1s and 0s.

Matrix 1 (Top Left)	Matrix 2 (Top Middle)	Matrix 3 (Top Right)
0 0 0 0 1	1 0 0 0 1	1 1 1
1 1 1 0 0	0 1 1 1 0	0 1
0 0 0 1 0	0 0 1 0 0	1 1
1 1 0 0 0	0 1 1 0 0	1 1
1 1 1 0 0	1 0 1 1 0	1 1

Matrix 4 (Middle Left)	Matrix 5 (Middle Middle)	Matrix 6 (Middle Right)
1 1 1 0 1	1 1 1 0 1	1 0 0 1
1 1 0 1 1	1 1 1 1 1	0 1 1
1 0 1 1 1	1 0 1 1 1	0 1 1
0 0 0 0 0	0 1 1 1 1	1 1 1
0 1 1 1 1	1 0 0 1 1	1 0 1

Matrix 7 (Bottom Left)	Matrix 8 (Bottom Middle)	Matrix 9 (Bottom Right)
1 1 1 1 1	0 0 0 0 1	1 1 1
0 0 0 1 0	1 1 1 1 1	1 0 1
0 1 1 1 0	1 0 0 1 1	1 1 1
0 0 1 1 0	1 0 1 1 1	1 1
1 1 1 1 1	0 1 1 1 0	1 1

Z - M - O - R

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	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	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
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	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

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 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      1
    1

1 1 1 1 1

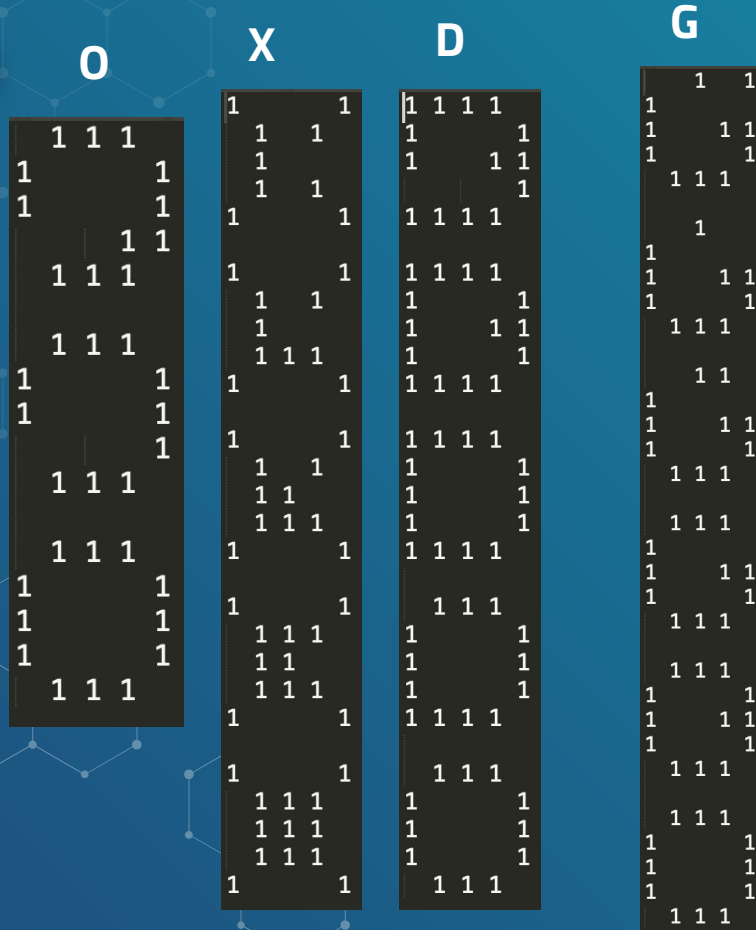
1 1 1 1 1
  1      1
    1
      1
1 1 1 1 1

1 1 1 1 1
  1      1
    1
      1
1 1 1 1 1

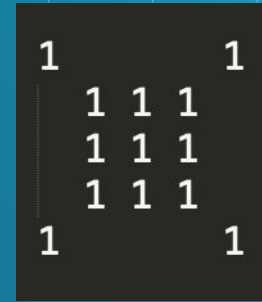
```

Entrenamiento y Reconocimiento 0 - X - D - G

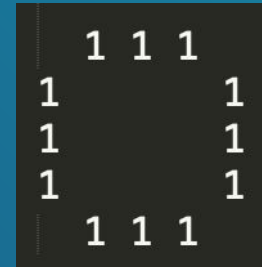
10%
alterado



Patrones Espúreos



Patrones Aprendidos (Sin los espúreos)





Conclusiones

- La red funciona con patrones más o menos ortogonales.
- En el caso de O-X-D-G la red solo aprende el patrón 0 y su patrón invertido.