

# Trabajo práctico 4

Métodos de aprendizaje no supervisados

Julian Neuss, Davor Vindis y Reurison Silva



# Contenido

1. Ejercicio 1 - Kohonen
2. Ejercicio 2 - Hopfield
3. Conclusiones

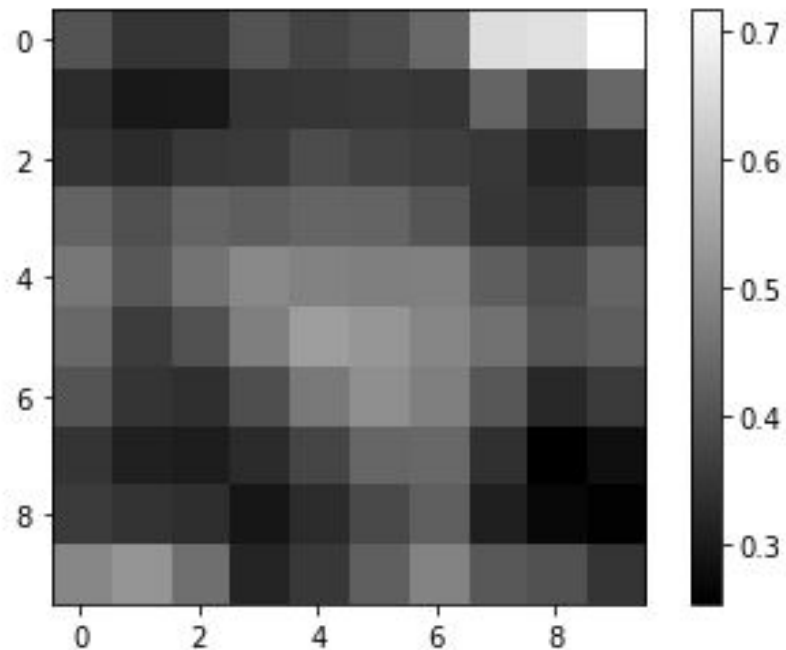
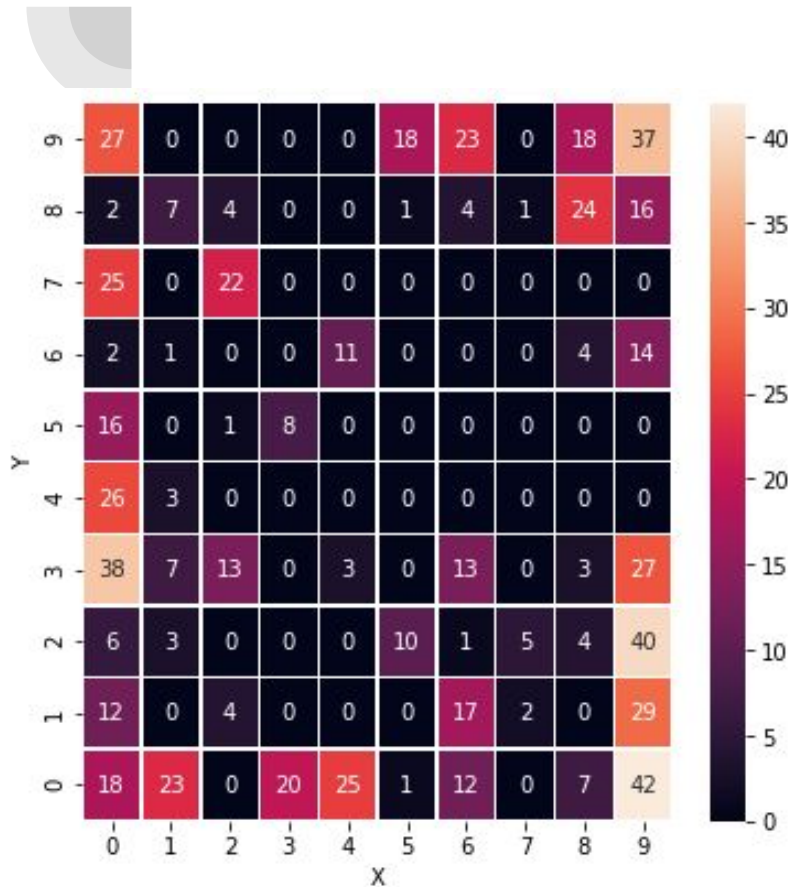


# Ejercicio 1 - Kohonen

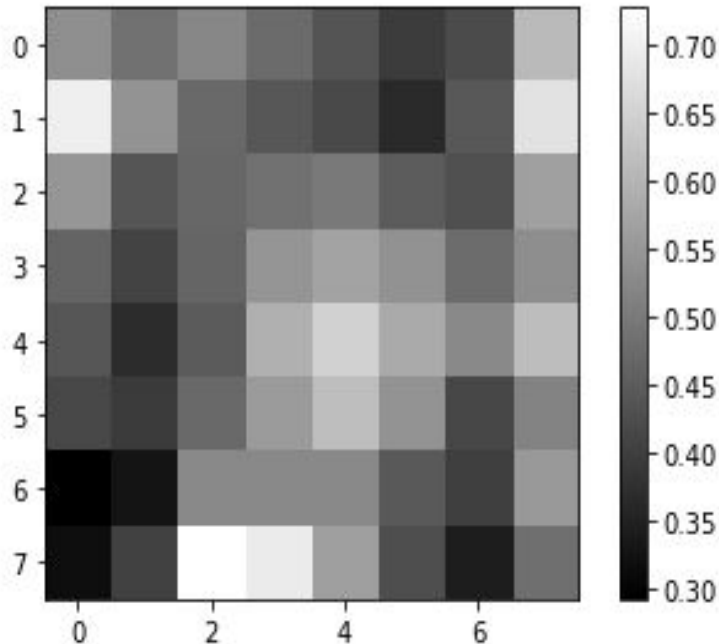
# 1a - Red de Kohonen



**$K=10$  / Iter = 700 /  $\eta(0) = 0.1$  /  $R(0) = K$**



Greece					UK	Finland			Germany Sweden
								Italy	Spain
Ukraine		Poland							
				Portugal					Norway
Bulgaria									
Estonia									
Latvia		Hungary				Denmark			Holanda
					Belgium				Swiss
Lithuania						Austria			Luxem
Croatia	Slovakia		Slovenia	Czech Repu					Iceland Ireland



Ireland	Luxem Swiss			Norway		Spain	Germany Italy Sweden
Holanda			Iceland	Hungary			UK
							Finland
Austria Belgium Denmark							
							Poland
Czech Rep							Ukraine
Slovenia		Iceland		Hungary			
	Croatia Greece Portugal			Lithuania	Bulgaria Estonia Latvia		



# 1b - Regla de Oja



## Calcular la primer componente principal

$\eta = 0.0001$  / épocas = 1000

```
W_Oja      = Oja(0.0001, 1000, X_std)
W_Oja_norm = W_Oja/np.linalg.norm(W_Oja) #|W_oja_norm| = 1
W_Pca      = pca.components_[0,:] # |W_Pca| = 1
```

```
print('W_Oja : ', W_Oja.T)
print('W_Pca : ', W_Pca)
```

```
W_Oja : [[ 0.13287485 -0.49920776  0.41036008 -0.48212269  0.18698564 -0.4750181
          0.26825696]]
```

```
W_Pca : [ 0.1248739  -0.50050586  0.40651815 -0.48287333  0.18811162 -0.47570355
          0.27165582]
```



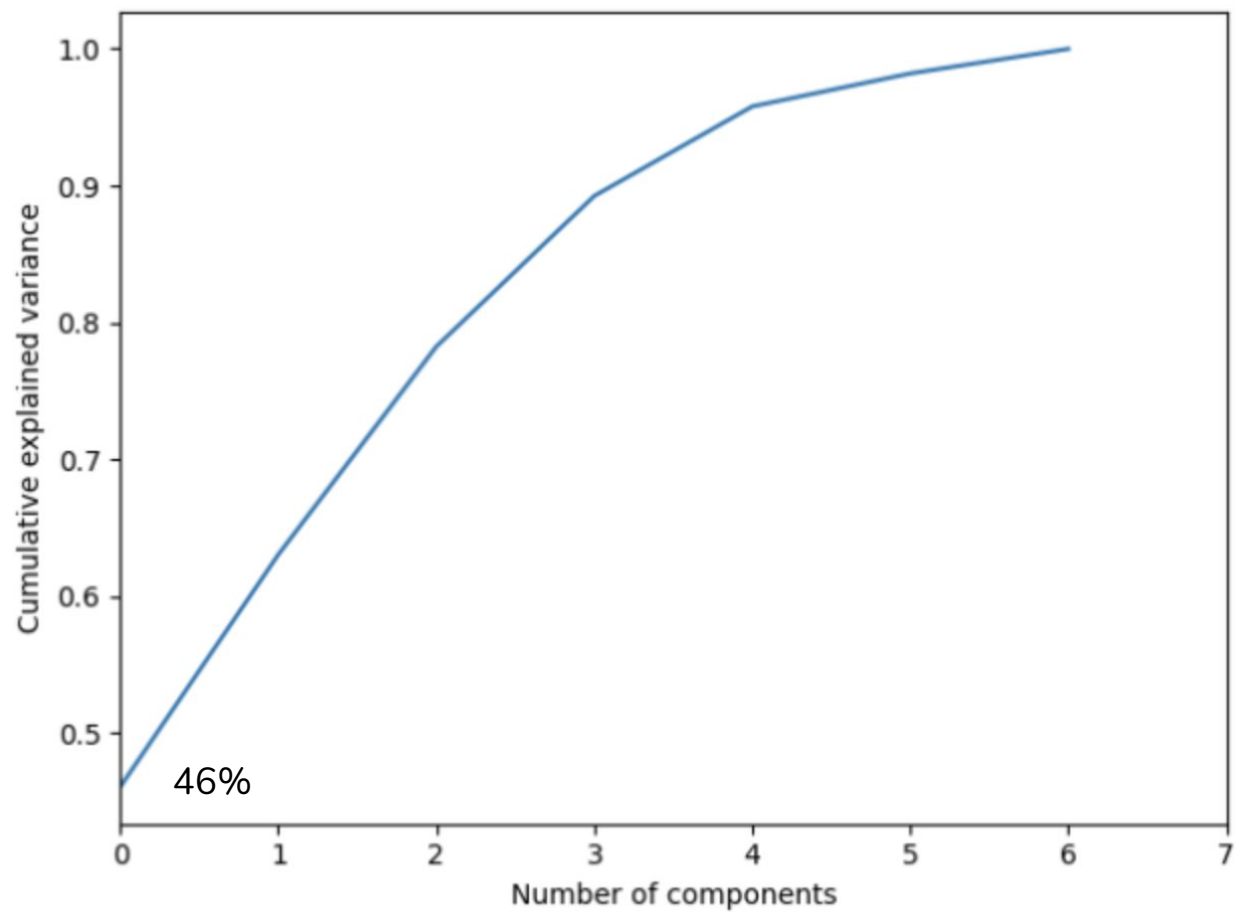
## Comparación entre librería y regla de Oja.

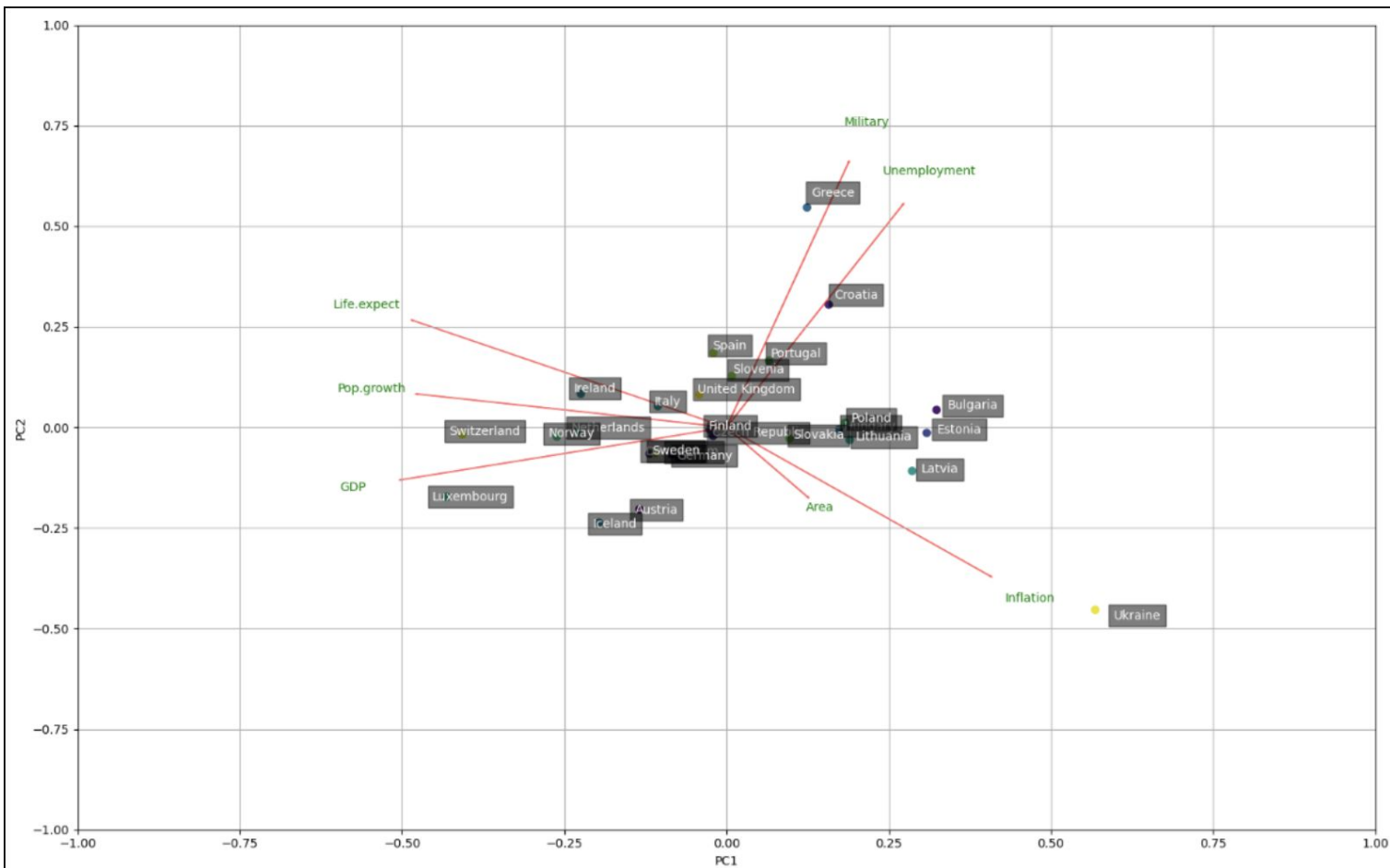
	Librería : Sklearn	Regla de Oja
<b>Area</b>	<b>0.1248739</b>	<b>0.13287485</b>
<b>GDP</b>	<b>-0.50050586</b>	<b>-0.49920776</b>
<b>Inflation</b>	<b>0.40651815</b>	<b>0.41036008</b>
<b>Life Expect.</b>	<b>-0.48287333</b>	<b>-0.48212269</b>
<b>Military</b>	<b>0.18811162</b>	<b>0.18698564</b>
<b>Pop. Growth</b>	<b>-0.47570355</b>	<b>-0.4750181</b>
<b>Unemployment</b>	<b>0.27165582</b>	<b>0.26825696</b>



# Interpretación

El método de Oja, converge al primer autovector corresponde al mayor autovalor.





## Ejercicio 2 - Hopfield



# Conjunto de letras

-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

Por qué esas?

- ❖ Producto vectorial interno entre todas más cercano a 0.
- ❖ El método alcanza su pico de efectividad con vectores ortogonales

$$J.P = 5$$

$$J.T = 7$$

$$J.W = 7$$

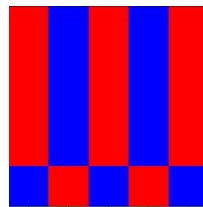
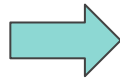
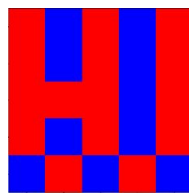
$$P.T = 7$$

$$P.W = 7$$

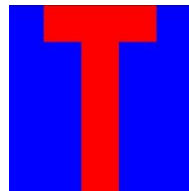
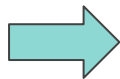
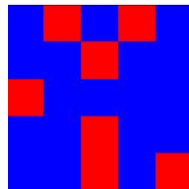
$$T.W = 1$$



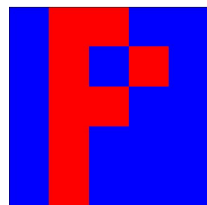
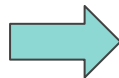
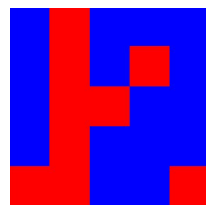
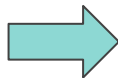
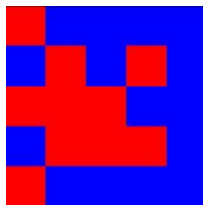
**Steps W poco ruido**



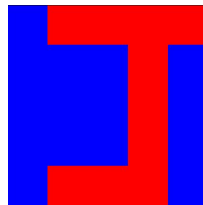
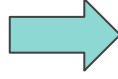
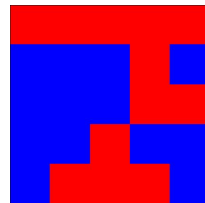
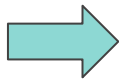
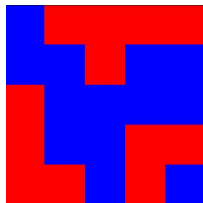
**Steps T poco ruido**



**Steps P mucho ruido**



**Steps J mucho ruido**



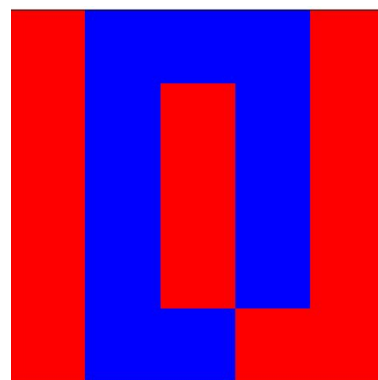
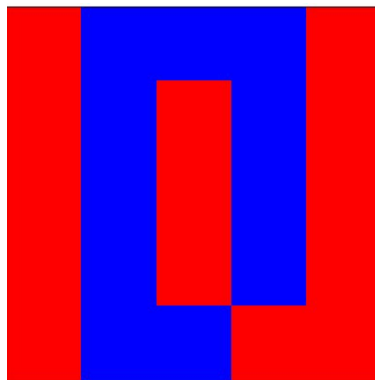
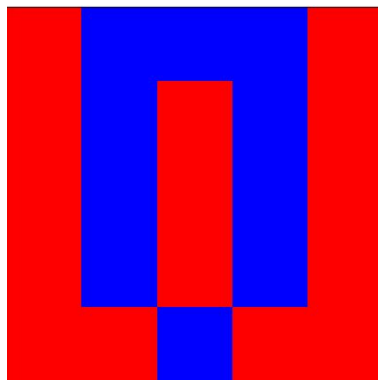
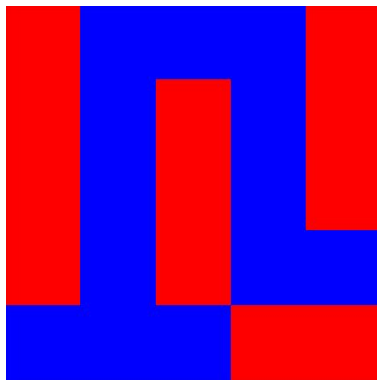


# Estado espúreo

Cuando llega un mínimo que no es un patrón de la lista, notamos que se repite un patrón

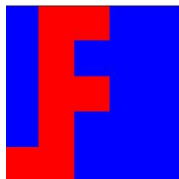
Cortamos cuando se repite un patrón

Estado espúreo llegado en 3 pasos a partir de W con 0.15 de mutación

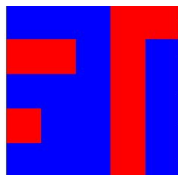


-1	1	1	-1	-1
-1	1	-1	1	-1
-1	1	1	-1	-1
-1	1	-1	-1	-1
-1	1	-1	-1	-1

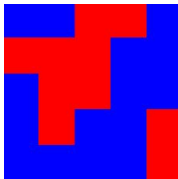
0.2



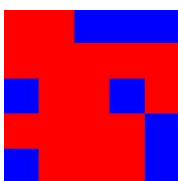
0.25



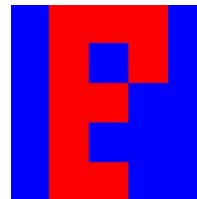
0.3



0.4



ESPÚREO



10%	<ul style="list-style-type: none"> <li>• 55% de aciertos</li> </ul>
15%	<ul style="list-style-type: none"> <li>• 70% de aciertos</li> </ul>
20%	<ul style="list-style-type: none"> <li>• 40% de aciertos</li> </ul>
30%	<ul style="list-style-type: none"> <li>• Acertó solo 4 veces</li> <li>• 20% de aciertos</li> </ul>
40%	<ul style="list-style-type: none"> <li>• Acertó solo 1 vez</li> <li>• 5% de aciertos</li> </ul>



# Conjunto de letras más ortogonal



$$J.T = 3$$

$$J.O = 1$$

$$J.F = 1$$



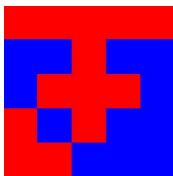
$$O.T = 1$$

$$O.F = 3$$

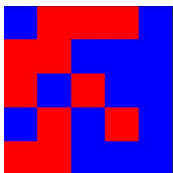
$$T.F = 3$$



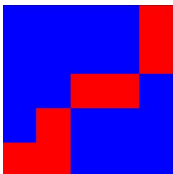
0.2



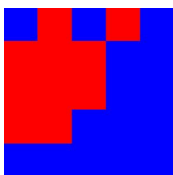
0.25



0.3

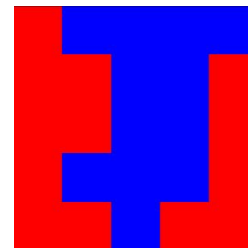


0.4



20%	<ul style="list-style-type: none"><li>100% de aciertos (acertó los 20)</li></ul>
25%	<ul style="list-style-type: none"><li>85% de aciertos (erró 3)</li><li>1 vez encontró una T</li><li>2 veces por estado espúreo</li></ul>
30%	<ul style="list-style-type: none"><li>75% de aciertos (erró 4)</li><li>4 veces a estados espurios</li></ul>
40%	<ul style="list-style-type: none"><li>25% aciertos (erró 15)</li><li>encontró 4T's y 1J</li></ul>

ESPÚREO



# Conclusiones

- Mientras más cercano sea a 0 los productos vectoriales de los patrones, más preciso será el método
- Si existe un estado espúreo parecido al patrón, puede bajar la precisión teniendo patrones con poco ruido
- A mayor ruido se suele requerir mas iteraciones
- La regla de Oja permite hacer el cálculo de las componentes principales en forma iterativa (ventajas computacionales).
- Red Kohonen : Al final del entrenamiento de la red es posible descubrir y exhibir una estructura subyacente de los datos, que mismo que sean observaciones de alta dimensionalidad, pueden ser representados en 2D.