

# Trabajo Práctico 4

## Aprendizaje No Supervisado

- Britos, Nicolás Ignacio - 59.529
- Griggio, Juan Gabriel - 59.092
- Roca, Agustín - 59.160

# Objetivos

- Implementar los algoritmos de Hopfield, Kohonen y Oja para problemas de aprendizaje no supervisado
- Comprender el propósito, ventajas y usos de estos algoritmos



# Desarrollo del trabajo

# Tecnología utilizada



**PYTHON**

+



# Resultados

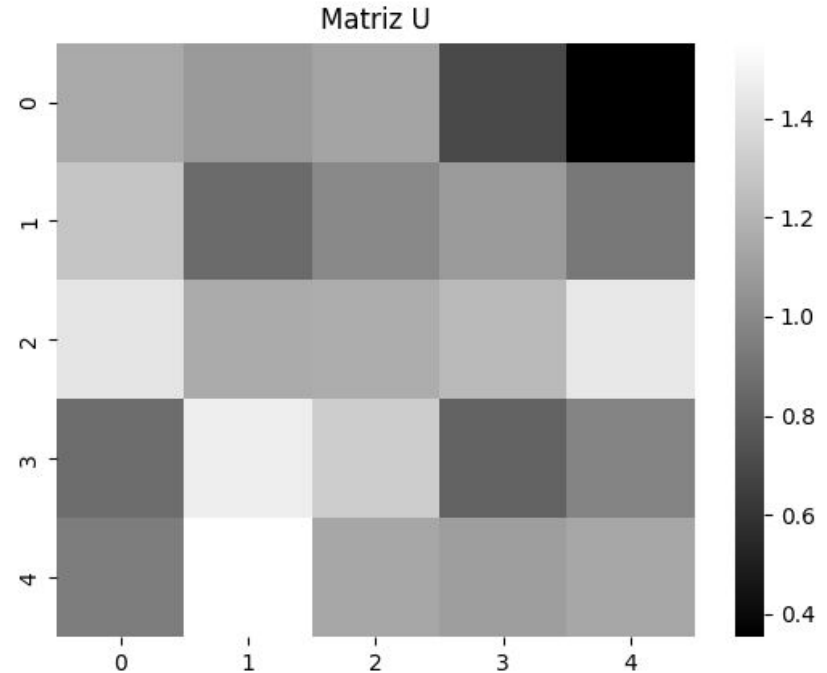
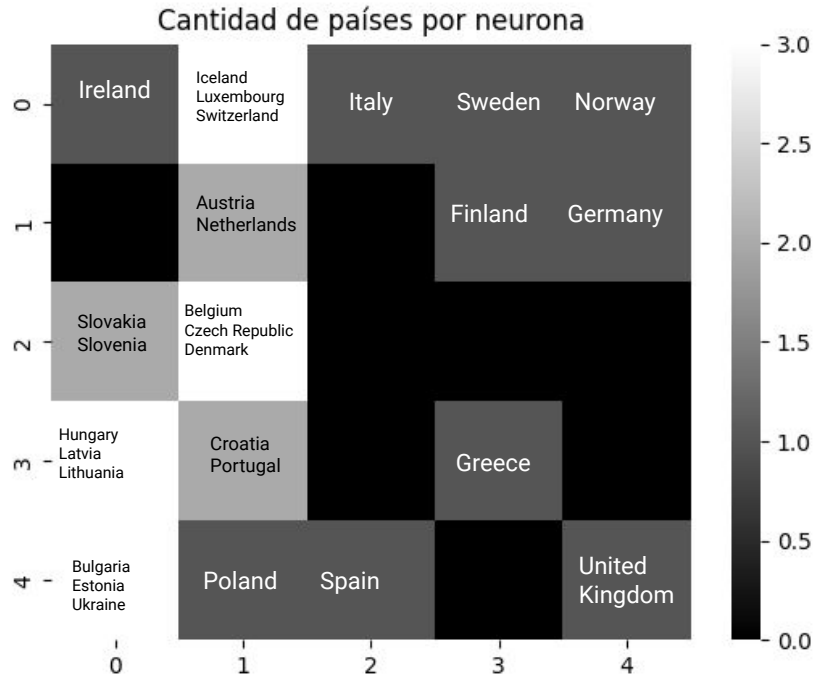
# Kohonen - Ejercicio 1a

# Parámetros utilizados - Kohonen

- $k = 5$
- $500 * n$  iteraciones
- $\eta(t) = 1/t$
- $R = 1$

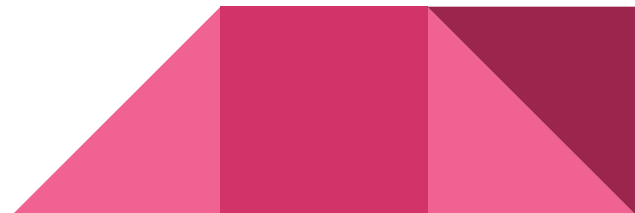
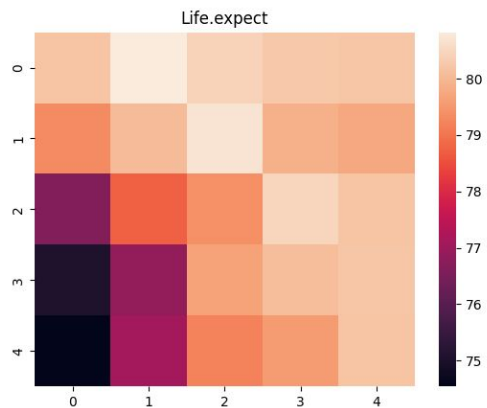
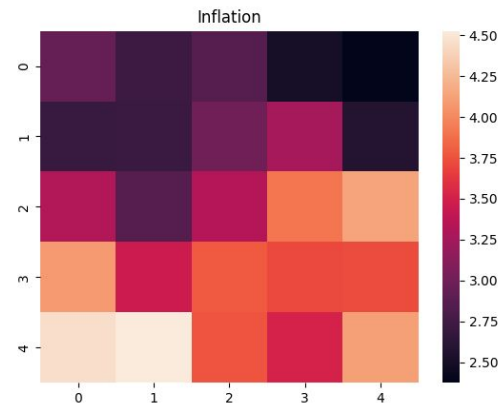
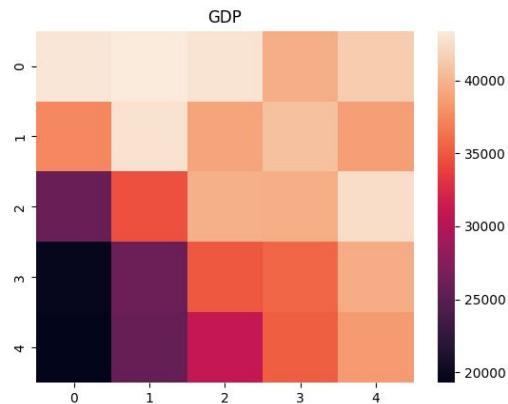
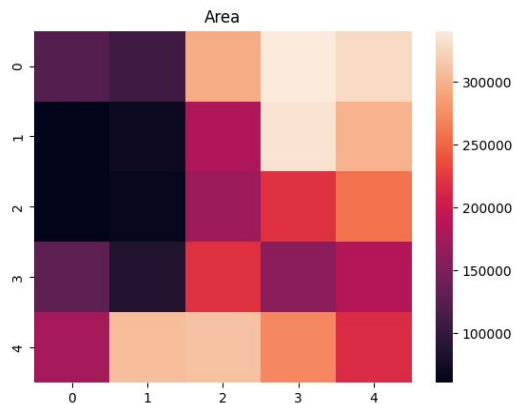


# Resultados - Kohonen

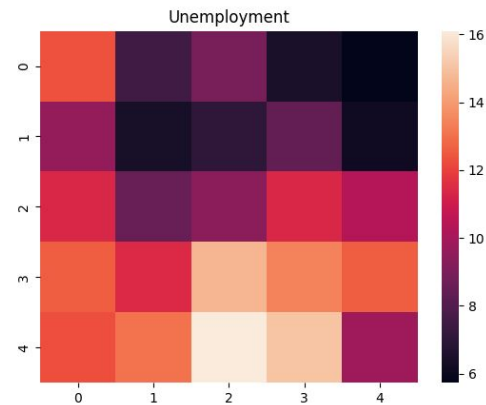
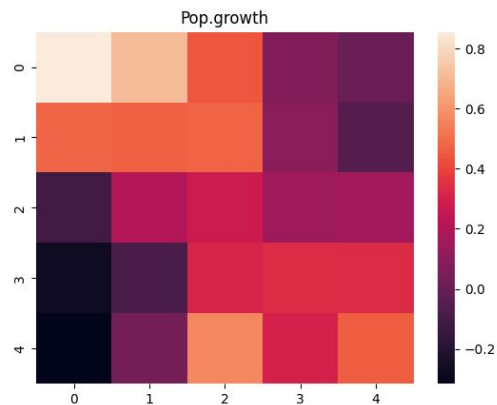
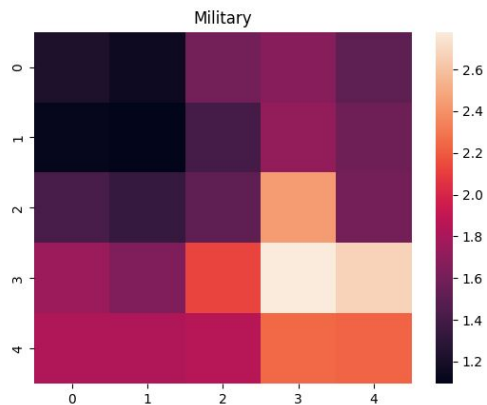




# Resultados - Kohonen



# Resultados - Kohonen



# Oja - Ejercicio 1b

# Parámetros utilizados - Oja

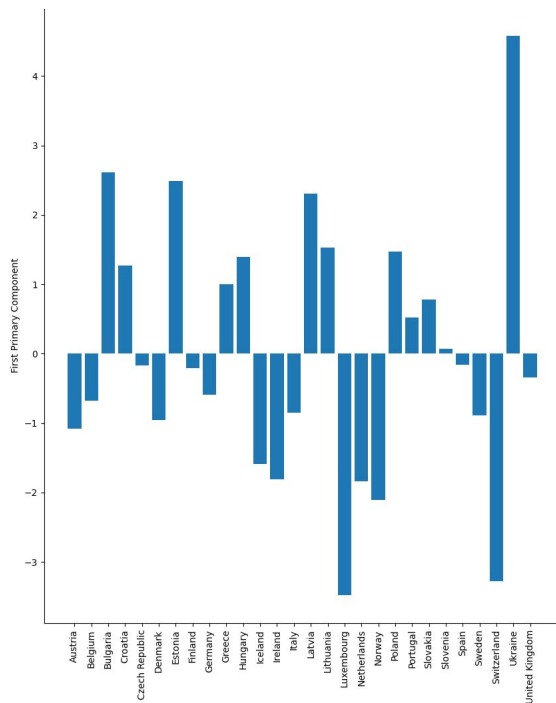
- Épocas = 10000
- $\eta = 0.0001$



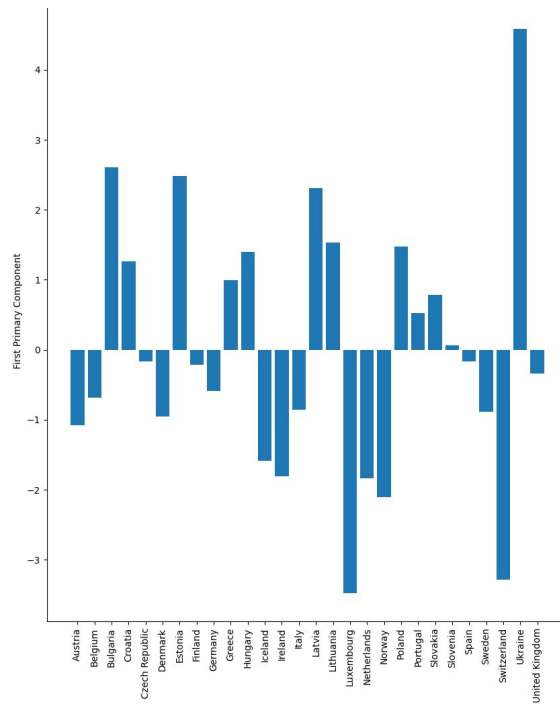
# Primera componente principal

	Area	GDP	Inflación	Expectativa de vida	Militarización	Crecimiento de la población	Desempleo
Sklearn	0.1248739	-0.50050586	0.40651815	-0.48287333	0.18811162	-0.47570355	0.27127044
Oja	0.12557215	-0.50037441	0.40716648	-0.48295445	0.18748873	-0.47548698	0.27127044
Diferencia ( $10^{-4}$ )	6.9825	1.3145	6.4833	0.8112	6.2289	2.1657	0

# Primera componente principal de cada país



Sklearn

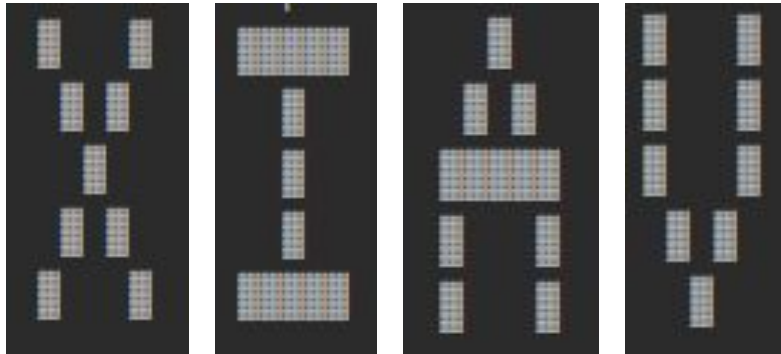


Oja

# Hopfield - Ejercicio 2

# Resultados - Hopfield, conjunto 1

- Elegido porque los productos internos entre estos patrones son cercanos a cero





# Resultados - Hopfield, conjunto 1, letra I

```
*Epoch 0 - H=-4.48000
[noisy pattern]

*Epoch 1 - H=-13.12000
[noisy pattern]

*Epoch 2 - H=-13.12000
[noisy pattern]
```

**Ruido 0.5**

```
*Epoch 0 - H=-3.84000
[noisy pattern]

*Epoch 1 - H=-8.80000
[noisy pattern]

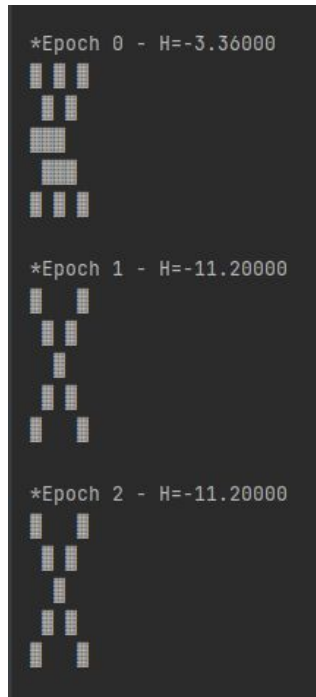
*Epoch 2 - H=-13.12000
[noisy pattern]

*Epoch 3 - H=-13.12000
[noisy pattern]
```

**Ruido 0.6**



# Resultados - Hopfield, conjunto 1, letra X



**Ruido 0.5**

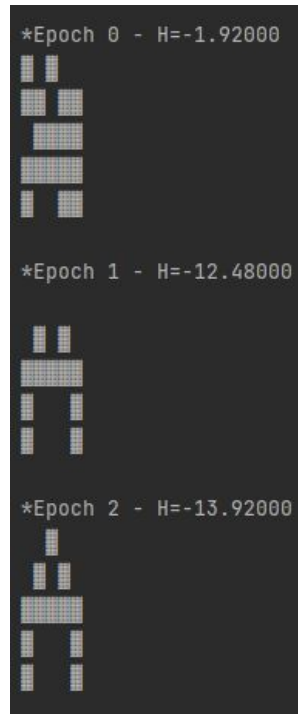


**Ruido 0.6.  
Estado cíclico  
(espúreos)**

# Resultados - Hopfield, conjunto 1, letra A



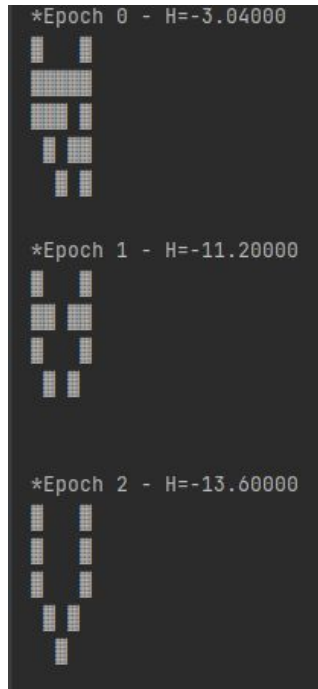
**Ruido 0.5**



**Ruido 0.6**



# Resultados - Hopfield, conjunto 1, letra V

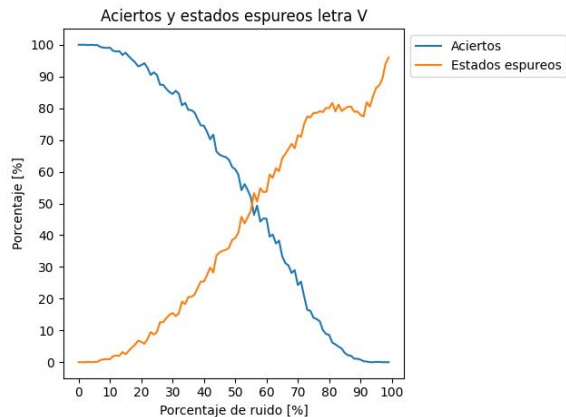
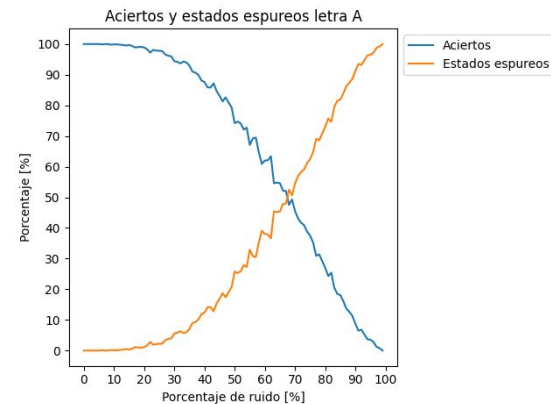
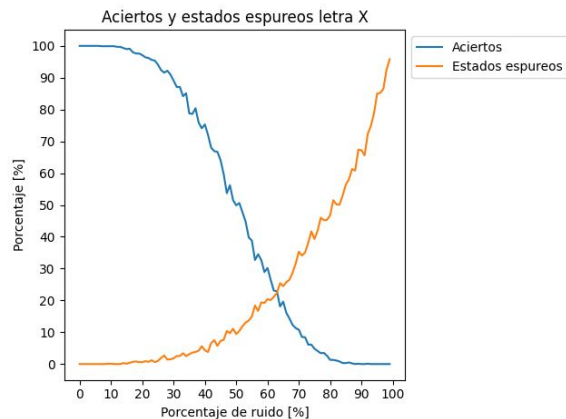
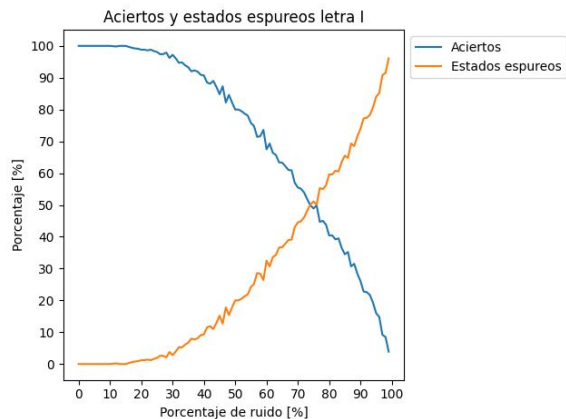


**Ruido 0.5**



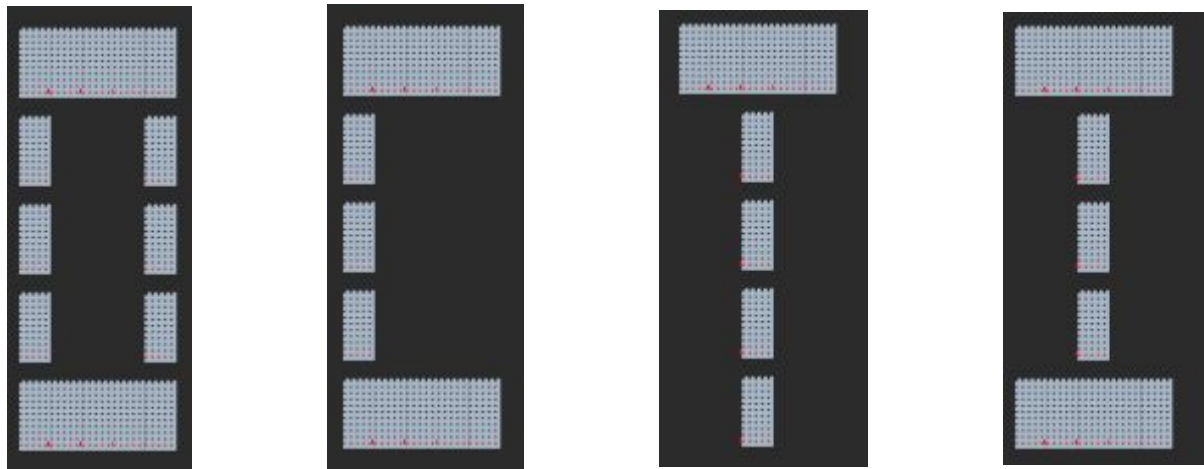
**Ruido 0.6**  
**Estado espúreo**

# Resultados - Hopfield, conjunto 1

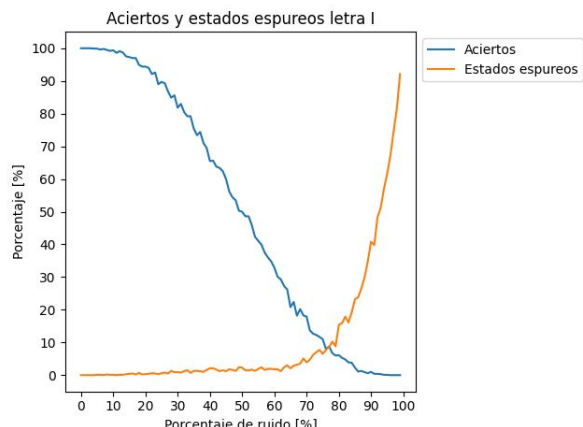
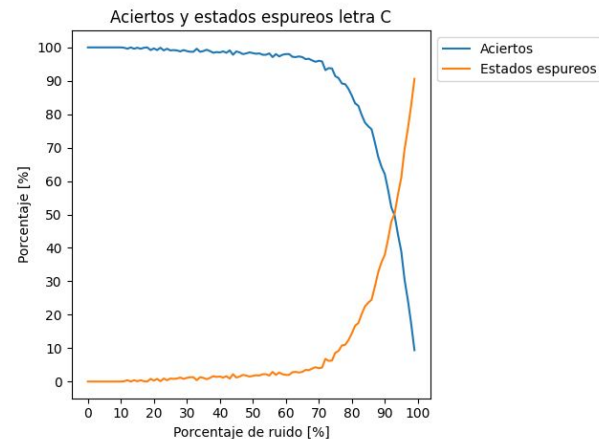
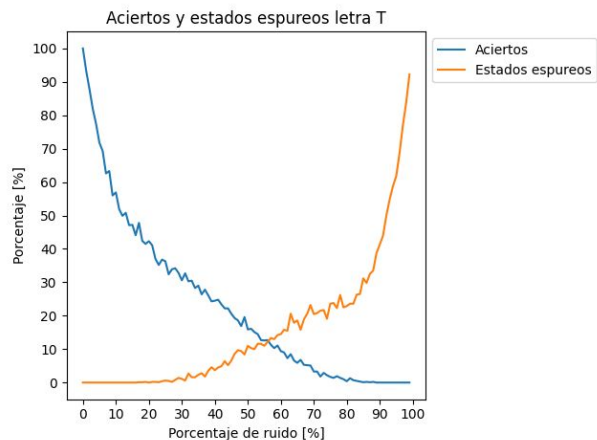
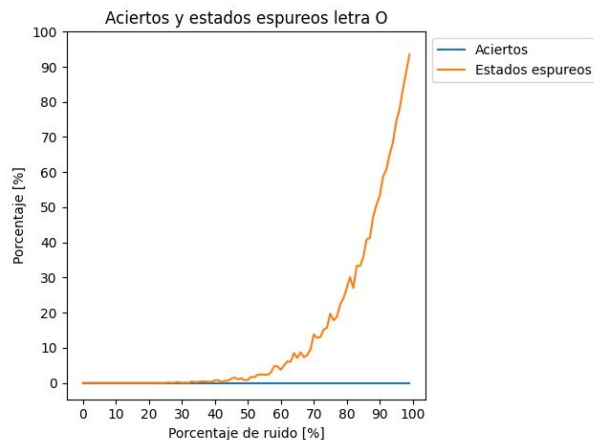


## Resultados - Hopfield, conjunto 2

- ¿Qué pasa si elegimos un conjunto lejos de ser ortogonal?



# Resultados - Hopfield 2



# Conclusiones



# Conclusiones - Kohonen

- Agrupa los elementos de entrada según su similitud
- Analogía: las neuronas se “acercan” y “agrupan” según los países que las activan y su similitud
- Para las variables que aportan mucha información, al aislarlas tienen a agruparse



# Conclusiones - Oja

- Útil para encontrar las Primeras Componentes Principales dentro de un conjunto de datos



# Conclusiones - Hopfield

- Puede detectar con certeza con buenos niveles de ruido, aunque no extremos
- Mientras más cercano a 0 sea el producto interno entre los patrones almacenados, más preciso será el resultado
- A mayor ruido se requieren más iteraciones





Muchas gracias!