TP4: Métodos de Aprendizaje NO supervisado

- De Simone, Franco - 61100
- Dizenhaus, Manuel - 61101
- Cornidez, Milagros - 61432

¿Qué es el aprendizaje no supervisado?

- Concepto **clave ➔ Conocimiento de la variable respuesta (o solución)**
  - Hasta ahora, los perceptrones utilizados trabajaron siempre con la comparación entre el valor estimado y el valor esperado, haciendo cálculos de error basados en la diferencia entre ambos, y recalculando los pesos sinápticos.

- Pero, si **no** contamos con esta variable, *¿cómo hacemos?*
  - Kohonen, Oja, Hopfield propusieron distintos modelos al respecto.
    - Resuelven problemas de **agrupamiento**, **asociaciones** y **reducción de la dimensionalidad**

Ejercicio 1: Análisis de datos sobre países Europeos

Datos del problema

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

Vemos que, para estos datos, las variables “Area” y “GDP” tienen una escala mucho mayor que las demás. Cuando se realice el producto interno con vectores peso, éstas variables tendrán una incidencia demasiado grande, prácticamente invalidando al resto. ¿Qué se hace? $\longrightarrow$ **Estandarizar** los datos

$$\tilde{X}_i = \frac{X_i - \bar{X}_i}{s_i}$$

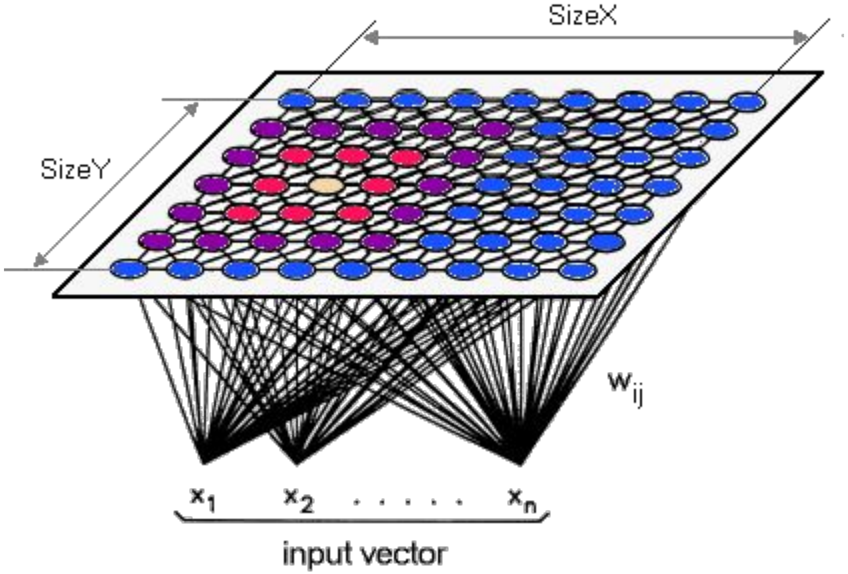
Datos del problema

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

Country	Area	GDP	Inflation	Life expect	Military	Pop growth	Unemployment
Austria	-0.5078	0.6839	0.1144	0.5707	-1.0243	-0.1767	-1.2455
Belgium	-0.8359	0.4170	0.1144	0.4877	-0.3889	-0.1159	-0.5924
Bulgaria	-0.3416	-1.2682	0.6242	-1.3674	1.2630	-1.8606	-0.0699

Ejercicio 1.1: Aprendizaje no supervisado con redes de Kohonen

Redes de Kohonen



Aprendizaje competitivo

Teuvo Kohonen, 2000

Análisis previo de los datos

Calculamos las parejas de países con **distancia euclídea mínima** entre ellos.

1. Bélgica - Dinamarca = 0,73
2. Lituania - Eslovaquia = 1,01
3. Austria - Bélgica = 1,01
4. República Checa - Eslovenia = 1,08
5. Dinamarca - Holanda = 1,09
6. Alemania - Suecia = 1,12
7. Alemania - Finlandia = 1,14
8. Austria - Dinamarca = 1,15
9. Finlandia - Italia = 1,17
10. Finlandia - Suecia = 1,18

A su vez, las peores parejas fueron:

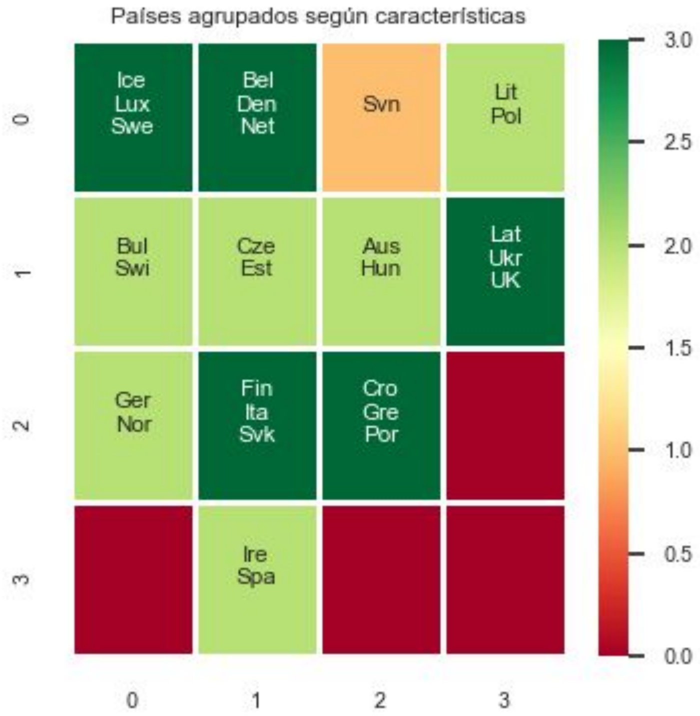
1. Ucrania - Suiza = 8,84
2. Ucrania - Luxemburgo = 8,79
3. Ucrania - Irlanda = 7,77
4. Ucrania - Noruega = 7,49
5. Ucrania - Holanda = 7,46

¿Qué predicciones podemos hacer?

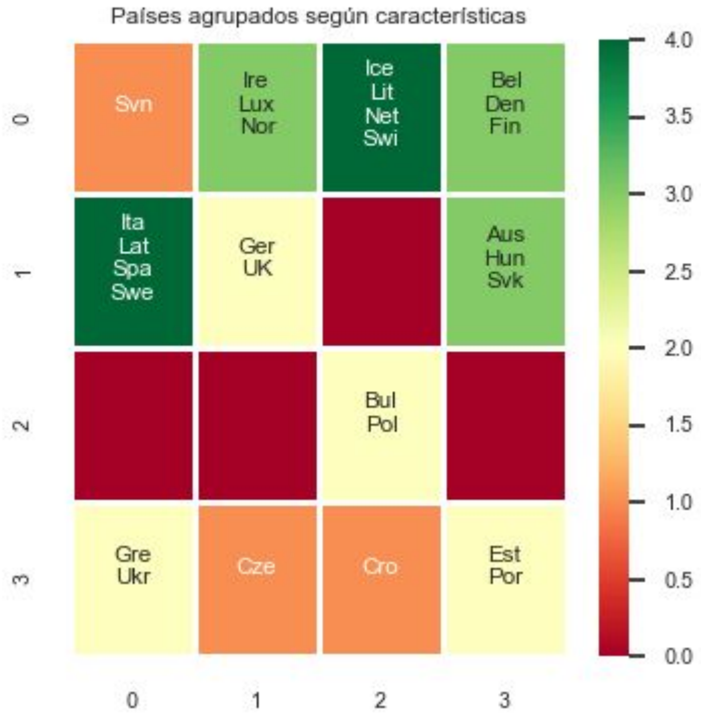
Condiciones del experimento

- Grilla de 4x4
- $R(0) = 4$
- $\eta(0) = 0,5$
- Los pesos iniciales de las neuronas fueron tomados eligiendo entradas al azar del conjunto de entrenamiento.

Primeras dos corridas



Mapa 1



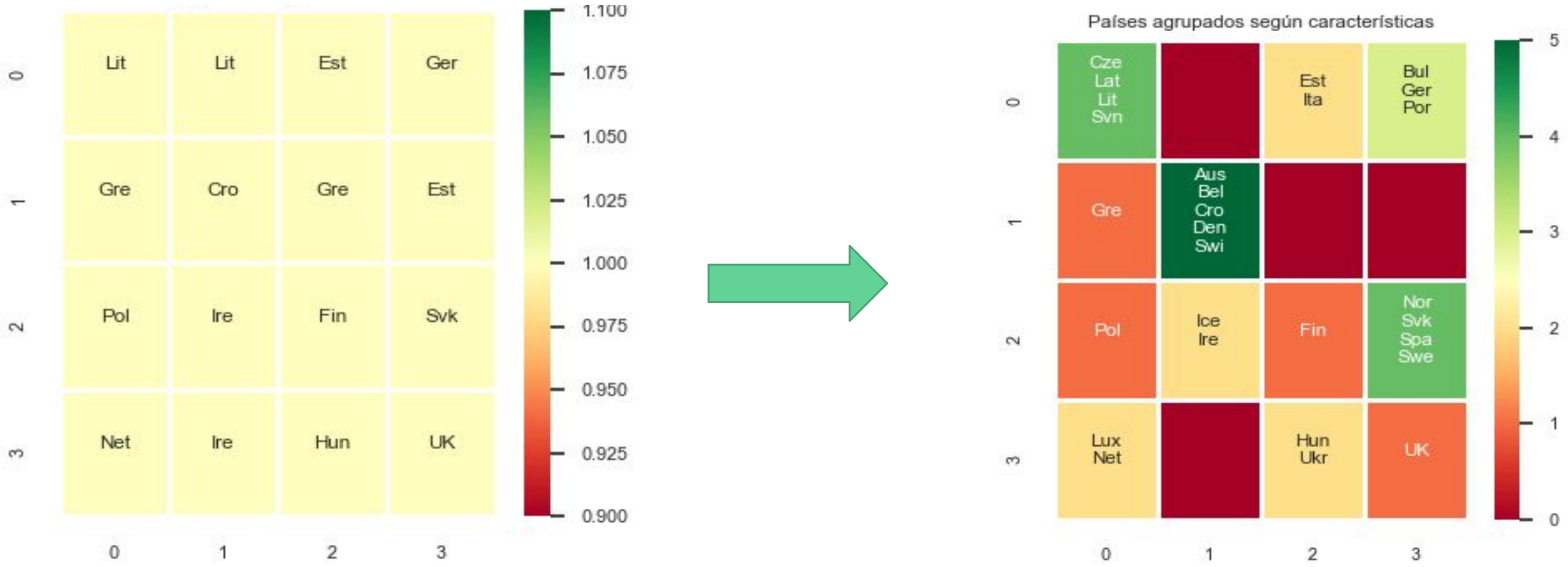
Mapa 2

Primeras impresiones

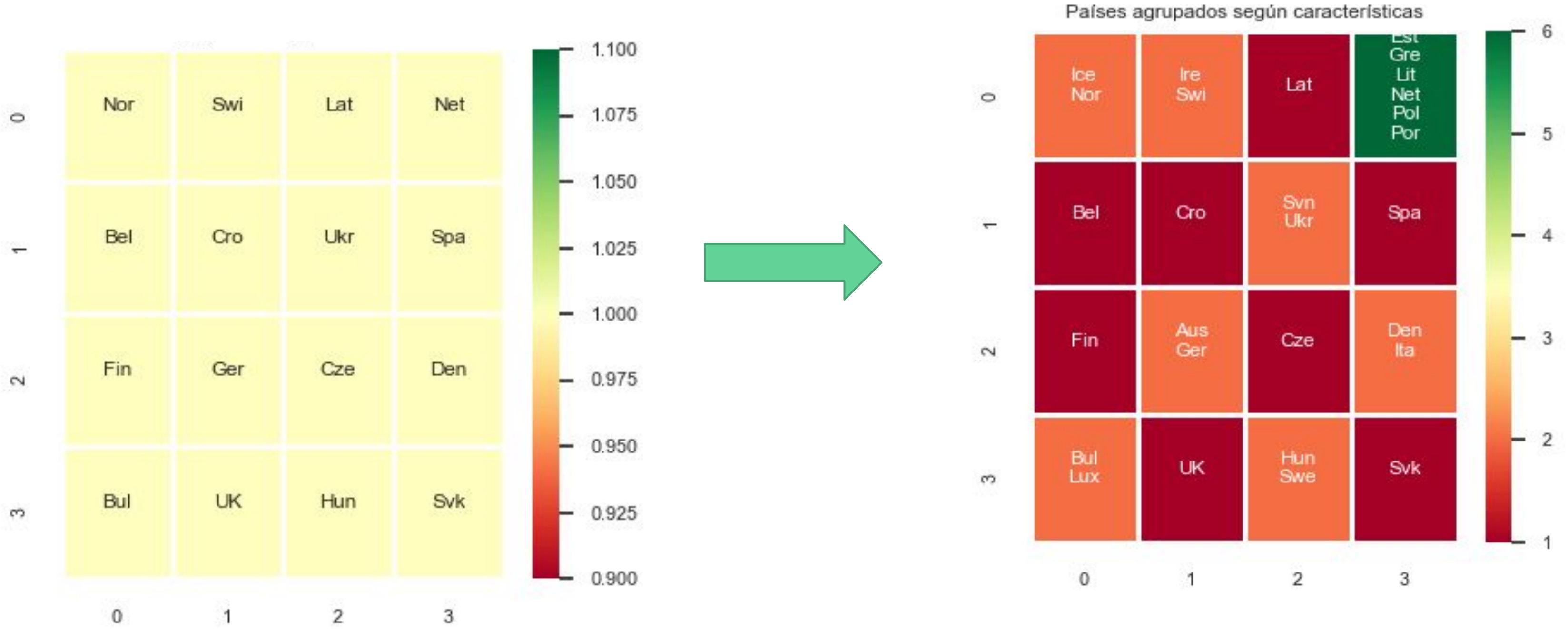
- Los países con características geo y sociopolíticas similares, en líneas generales, quedan cercanas en el “mapa de neuronas”.
- Sin embargo, hay países con características similares que quedan lejos (Lituania y Eslovaquia en el mapa 1, por ejemplo) → ¿Por qué?
- Dos mapas con las mismas condiciones iniciales resultan muy distintos el uno del otro → Gran incidencia de la aleatoriedad
- Muchas neuronas “vacías” → ¿Serán demasiadas?

¿Cómo se genera la grilla inicial?

Se eligen elementos del conjunto de entrada **al azar**.



¿Sin reemplazo?

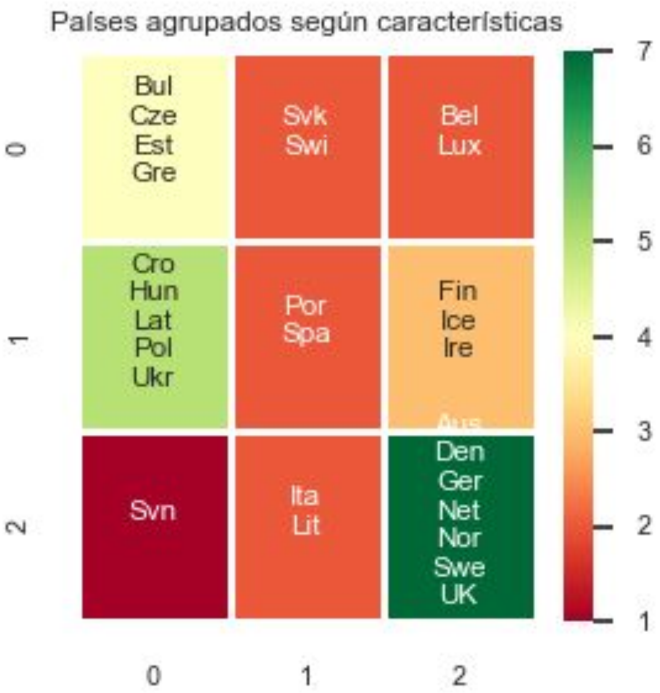


Entonces...

- Los países que formen parte de la grilla inicial tienden a quedar en la neurona que se les asignó inicialmente.
- Con una matriz de 4x4, estamos fijando prácticamente la mitad (16, en el caso de sin reemplazo) de los países antes de siquiera comenzar el algoritmo.

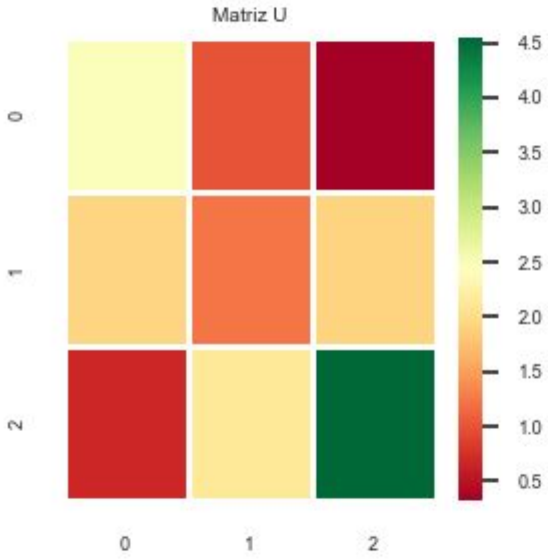
¿No será mejor probar con una grilla más chica?

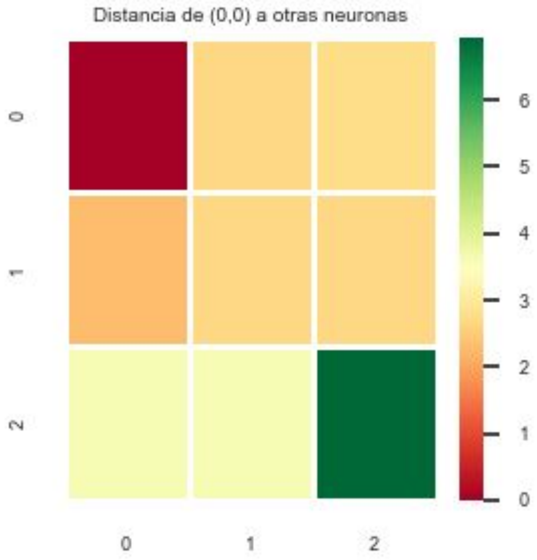
Grilla 3x3



- Austria, Dinamarca y Holanda juntos
- Noruega, Suecia y Alemania juntos
- Finlandia cercano al resto de los nórdicos
- Países de Europa del Este se encuentran cercanos
- Países vecinos compartiendo neurona (España y Portugal, Bélgica y Luxemburgo)
- Ucrania con Polonia y Latvia, los dos países más similares

Matriz U


Matriz U
0
1
2
0
1
2
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5


Distancia de (0,0) a otras neuronas
0
1
2
0
1
2
6
5
4
3
2
1
0

Conclusiones

- La red de Kohonen sirve para **agrupar** datos de entrada similares cuando no se cuenta con una **salida deseada** para ellos.
- Se debe tener en cuenta la incidencia de los pesos iniciales, más aún cuando se cuenta con un conjunto pequeño de registros (como en este caso).
- Si bien no hay un claro criterio para la elección del tamaño de la grilla, si ésta es demasiado grande la red será poco efectiva.
- Para este problema, la red agrupa a los países de acuerdo a la similitud de sus datos que, en este caso, son sus características geopolíticas, económicas y sociales.

Ejercicio 1.2: Regla de Oja

Regla de Oja

- **Erkki Oja** ➔ Demuestra que, tomando un aprendizaje determinado, el vector de pesos sinápticos *w* **converge** hacia la dirección de máxima variación de los datos, es decir, las **cargas** de la primer componente principal.
  - Es necesario que el vector *w* tenga **norma 1.** Si no se cumple esto, el método no convergerá, haciendo que el algoritmo sea inestable.

input: N datos con media 0; η, w_{inic}

1. for epoch in 1:epoca
   1. for i in 1:N
      1. $s = \sum_j (datos[i,j] * w_j)$
      2. $w = w + \eta * s * (datos[i,] - s * w)$
2. return(w)

$datos[i,]$ significa toda la entrada i

Erkki Oja, 2016

Resultados

- Utilizando **‘Regla de *Oja*’** ➔ [0.12558938, -0.50044306, 0.40722235, -0.48302071, 0.18751446, -0.47555222, 0.27130766]

- Con librería **‘SciKit Learn’** ➔ [0.1248739, -0.50050586, 0.40651815, -0.48287333, 0.18811162, -0.47570355, 0.27165582]

(η = 0.0001, epochs = 5000)

Influencia del η

- Con **η = 0.1:** [0.97544815, -0.43102733, 0.59303758, -1.04028203, -0.52411874, 0.01054178, -0.97439045]
- Con **η = 0.01:** [0.20275538, -0.48993905, 0.47946847, -0.50095825, 0.13074739, -0.45684623, 0.232006]
- Con **η = 0.001:** [0.1320901 , -0.49983558, 0.41358595, -0.48437194, 0.18215693, -0.47415024, 0.26813229]
- Con **η = 0.0001:** [0.12558938, -0.50044306, 0.40722235, -0.48302071, 0.18751446, -0.47555222, 0.27130766]

Influencia del η

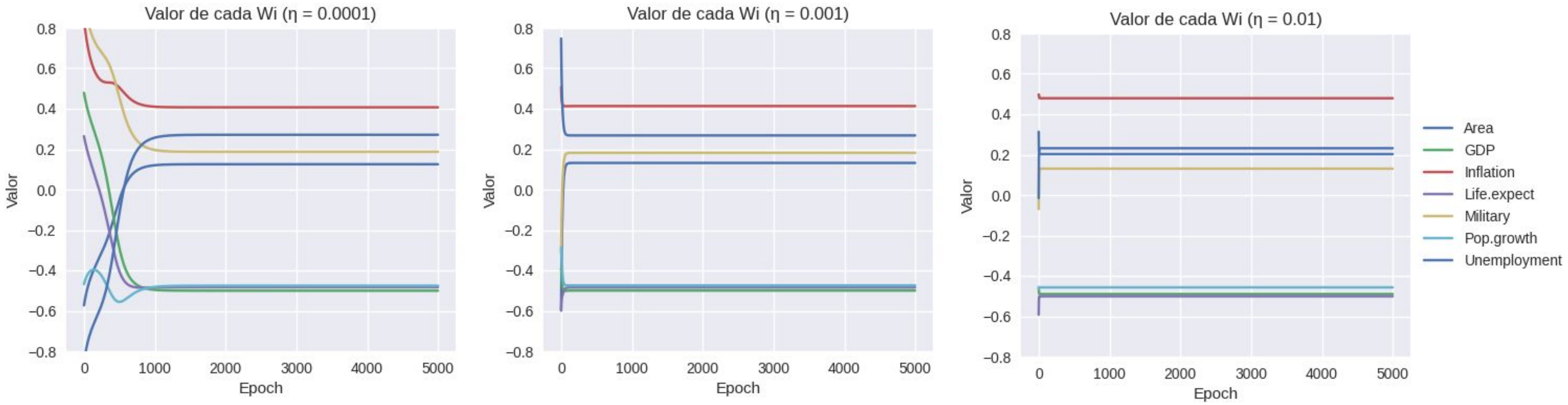
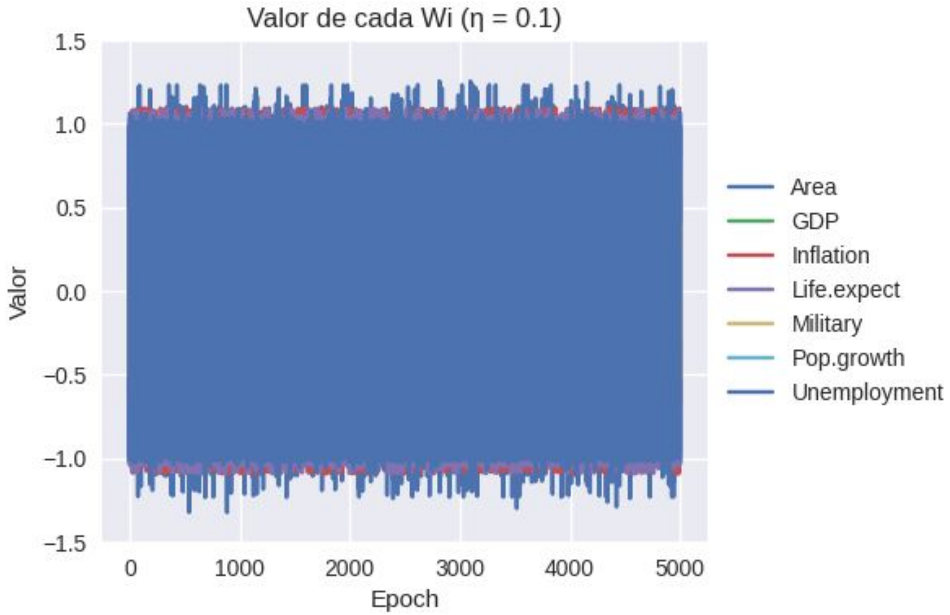


Gráfico resultante con η = 0.1

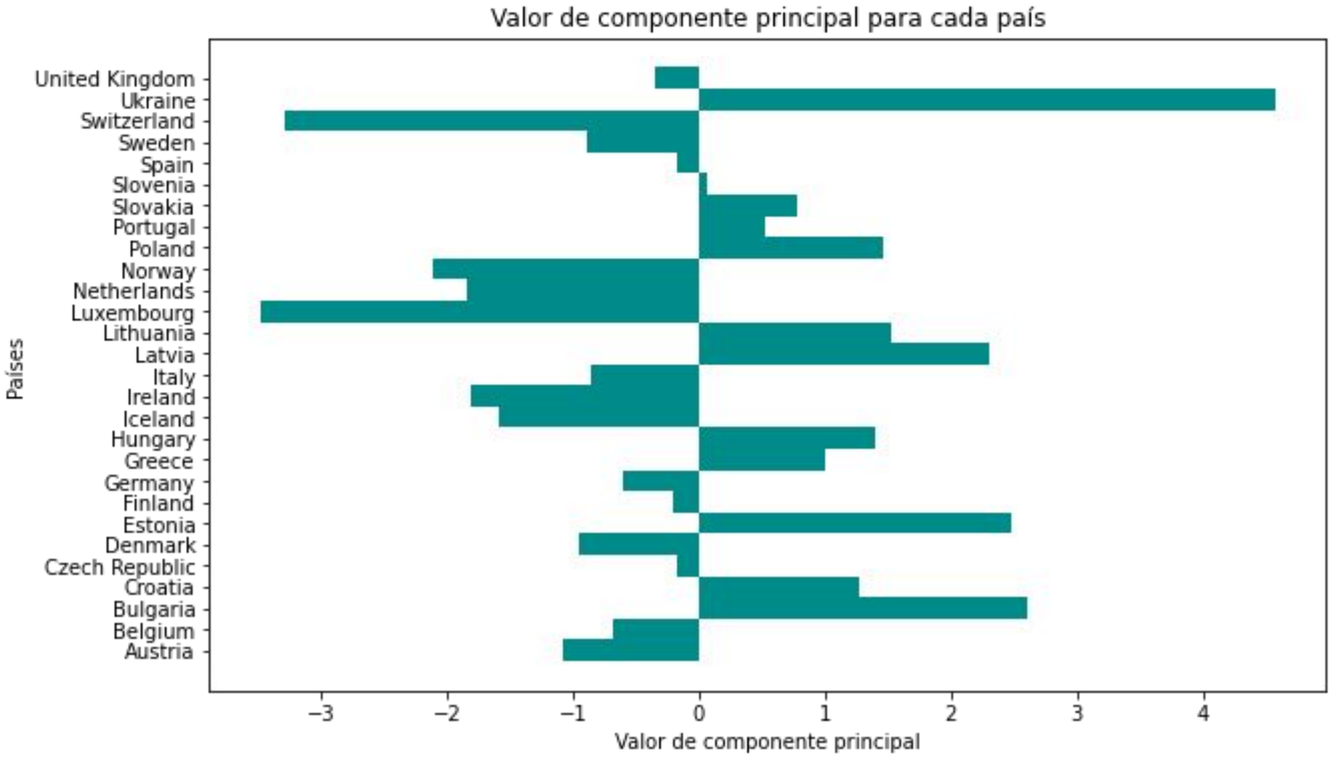


Observación

- El **vector** de pesos sinápticos resultante utilizando la regla de Oja algunas veces converge a los mismos valores absolutos pero con los signos de *todas* las componentes invertidas ➜ ¿Como lo explicamos?
- Autovector ➜ Un vector con norma 1. El único que existe y que mantiene la norma, es el vector que apunta en la misma **dirección** pero en sentido **opuesto.**

Interpretación de la 1ra componente principal

- Retomando el análisis realizado para el ejercicio N°2, el análisis de la primer componente principal está basado en las **cargas** del autovector principal.
  - 0.1255893778340837 X1 (Area)
  - -0.5004430641979144 X2 (GDP)
  - 0.40722234509501504 X3 (Inflation)
  - -0.48302071436583177 X4 (Life.expect)
  - 0.18751445616395815 X5 (Military)
  - -0.4755522215649601 X6 (Pop.growth)
  - 0.2713076579358877 X7 (Unemployment)
- Un país con un valor alto de CP1 muestra que tiene valores altos para las variables *Inflación, Gasto militar, y desempleo*
- Por otro lado, un valor bajo indicaría valores altos de *GDP, Expectativa de Vida, y Crecimiento poblacional*

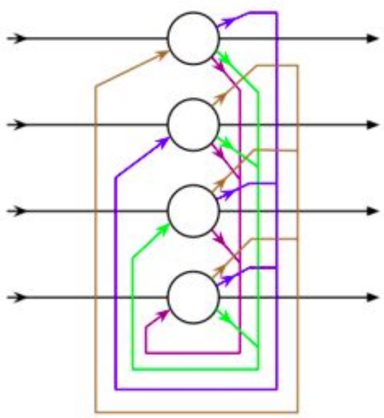


Valor de componente principal para cada país
United Kingdom
Ukraine
Switzerland
Sweden
Spain
Slovenia
Slovakia
Portugal
Poland
Norway
Netherlands
Luxembourg
Lithuania
Latvia
Italy
Ireland
Iceland
Hungary
Greece
Germany
Finland
Estonia
Denmark
Czech Republic
Croatia
Bulgaria
Belgium
Austria
Países
−3
−2
−1
0
1
2
3
4
Valor de componente principal

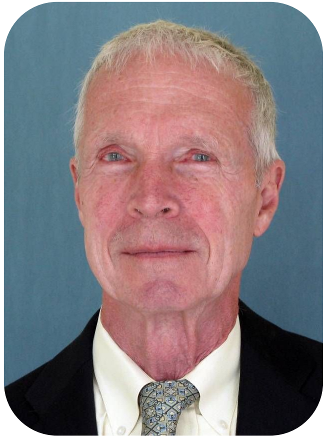
Ejercicio 2: Redes de Hopfield

Redes de Hopfield

- Teniendo almacenados *p* patrones, al presentar un **nuevo patrón A,** la red responde con el patrón **B** más parecido entre los almacenados.
    - Llamamos a A **estímulo,** y a B **respuesta.**
- Redes **recurrentes** ➔ Todos conectados con todos (menos sí mismos)
- Entrada y salida binaria {-1, 1}



Visualización de una red recurrente



John Hopfield, 2018

Redes de Hopfield

- Limitaciones
  - Vector de pesos sinápticos **no** se modifica. Es decir, la red no aprende, no se actualiza ante una nueva entrada.
  - El número **máximo** de patrones que se pueden almacenar son p ≤ 0,15 ∗ N, con N el tamaño de los vectores de entrada.
  - Los patrones almacenados deben ser medianamente ortogonales.
- Cálculo de Ortogonalidad

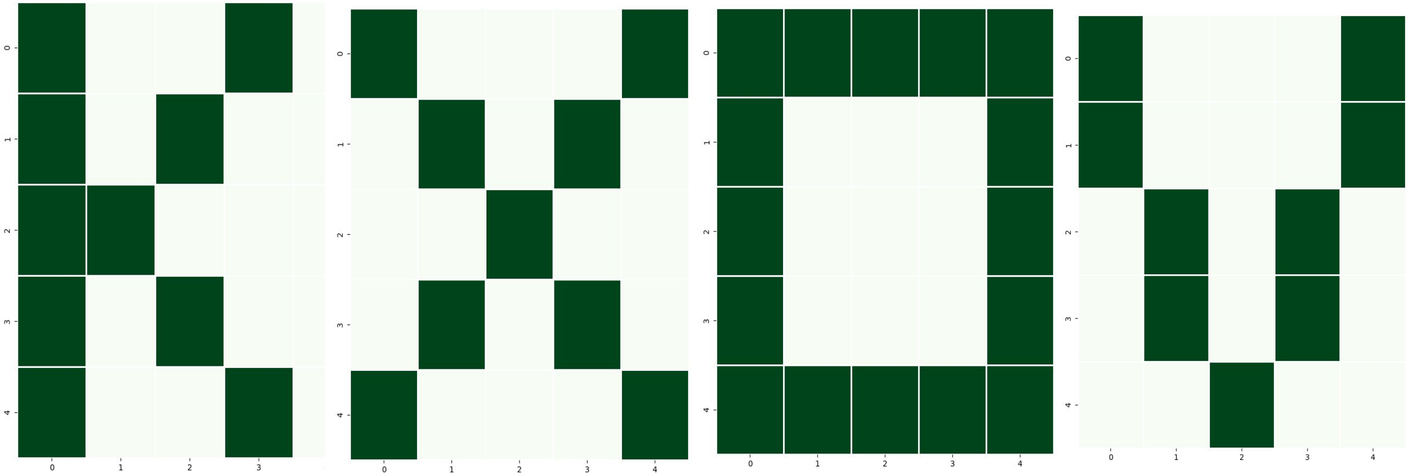
$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

"Sean *x,y* vectores en ℝ^n, son ortogonales si el producto escalar de x con y **es 0.** Esta situación se denota con *x* ⊥ *y*." (Definición adaptada de la siguiente fuente)

$$Orth = \frac{\sum_{i=0}^{n-1} \sum_{j=i+1}^{n} Dot(letter_i, letter_j)}{(n-1)^2}$$

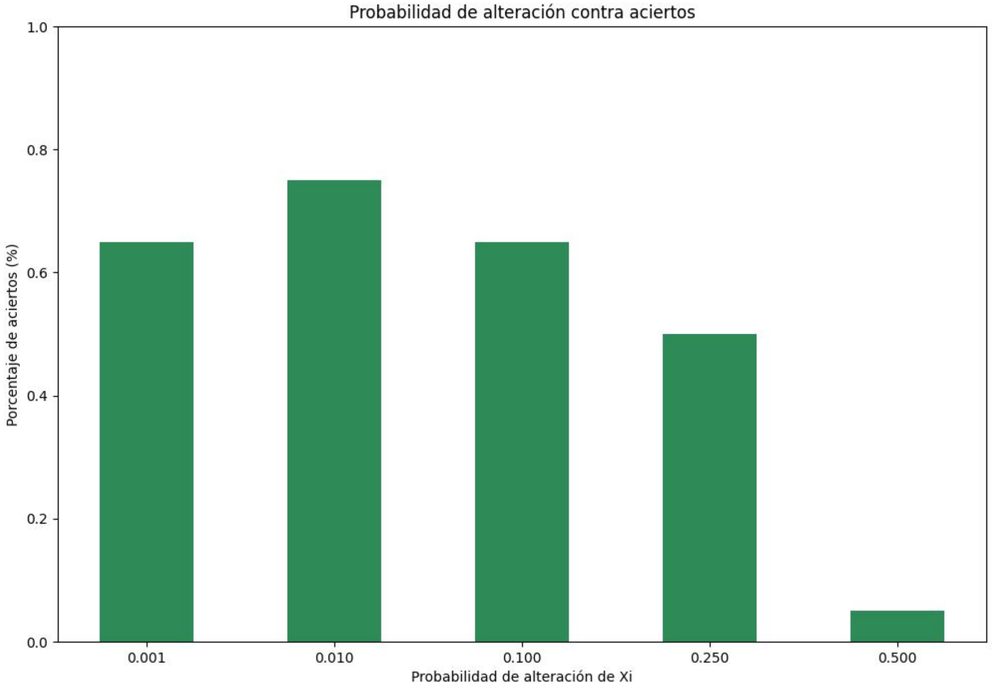
Resultados

- Primer experimento: Letras con **alta ortogonalidad.**



Ortogonalidad promedio = -1.55

Resultados

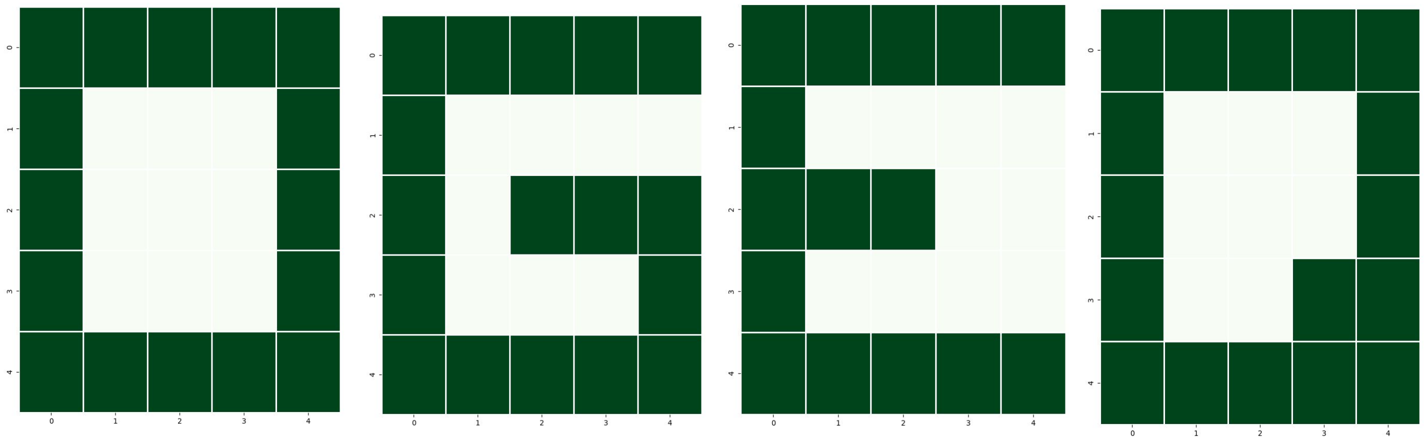



```
Hits/Totales con ruido 0.001: 84.0%

Hits/Totales con ruido 0.01: 76.0%

Hits/Totales con ruido 0.1: 74.0%

Hits/Totales con ruido 0.25: 66.0%

Hits/Totales con ruido 0.5: 18.0%
```

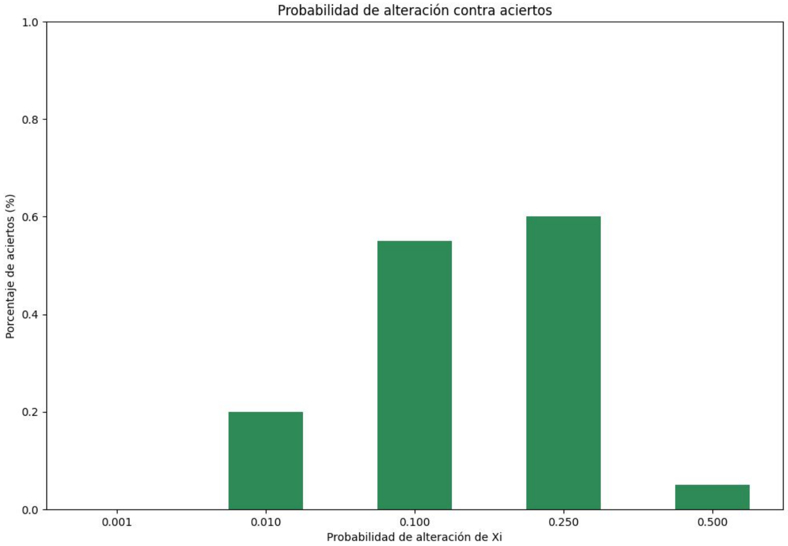
Resultados

- Segundo experimento: Utilizando letras **muy poco ortogonales.**

Ortogonalidad promedio = 11.55



Resultados




```
Hits/Totales con ruido 0.001: 0.0%

Hits/Totales con ruido 0.01: 20.0%

Hits/Totales con ruido 0.1: 55.00000000000001%

Hits/Totales con ruido 0.25: 60.0%

Hits/Totales con ruido 0.5: 5.0%
```

Función de energía

- Hopfield demostró que la red diseñada estaba asociada a una **función de energía,** cuyos mínimos locales eran los patrones**:**

$$H = -\frac{1}{2}\sum_{i,j} w_{ij} S_i S_j$$

- Lo más interesante de esta función es que siempre **decrece o permanece constante.**
- Dicho esto, podemos decir que converge hacia los mínimos locales de la función. Pero, ¿cuáles son estos mínimos locales?

Estados espúreos

- Con ***p = 0.5*** de ruido individual para cada posición, encontramos que la convergencia se producía rápidamente, pero hacia patrones que no estaban almacenados. ➜ **Función de energía:** Así como se demuestra la convergencia de la red de Hopfield hacia los mínimos locales que son los patrones, también existen otros mínimos locales conocidos como estados espúreos.