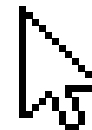


(72.27) Sistemas de Inteligencia Artificial

Trabajo Práctico N°4

Aprendizaje No Supervisado



Alejo Flores Lucey	62622
Andrés Carro Wetzel	61655
Ian Franco Tognetti	61215
Matías Daniel Della Torre	61016

01 Introducción

Presentación de los problemas a resolver.

02 Red de Kohonen

Agrupación de países con características similares.

03 PCA

Estudio de componentes principales. Comparación con Kohonen.

04 Modelo de Oja

Cálculo iterativo de 1ra componente principal. Comparación con PCA.

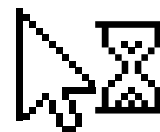
05 Modelo de Hopfield

Asociación de un patrón de consulta con patrones almacenados.

Introducción

Hemos implementado una serie de algoritmos para resolver problemas utilizando el método de **aprendizaje no supervisado**.

El objetivo es encontrar **patrones de similitud** entre los datos.



Red de Kohonen

Durante el proceso de aprendizaje descubre por sí misma regularidades (patrones) en los datos de entrada

Modelo de Oja

Modelo de una red neuronal que permite calcular las componentes principales de forma iterativa

Modelo de Hopfield

Asociar un patrón de consulta binario (con perturbaciones) con alguno de los patrones almacenados

Archivo de configuración

- Toda la configuración se maneja desde un archivo JSON.
- Se configura una semilla para obtener reproducibilidad en el caso de usar valores pseudo-aleatorios.
- Se elige el problema a resolver de las opciones disponibles.
- Los hiperparámetros son completamente parametrizables.

```
config.json

{
  "problem": "KOHONEN | OJA | HOPFIELD | HOPFIELD - ORTOGONALITY",
  "options": {
    "matrixSquareSize": 3, // Only for KOHONEN
    "initialLearningRate": 0.3, // Only for KOHONEN or OJA
    "initialRadius": 1, // Only for KOHONEN
    "changeLearningRate": false, // Only for KOHONEN or OJA
    "changeRadius": false, // Only for KOHONEN
    "distanceCalculator": "EUCLIDEAN", // Only for KOHONEN
    "maxEpochs": 10000, // Only for OJA or HOPFIELD
    "noiseLevel": 0.1, // Only for HOPFIELD
    "letters": ["F", "I", "O", "V"] // Only for HOPFIELD
  },
  "seed": 732
}
```

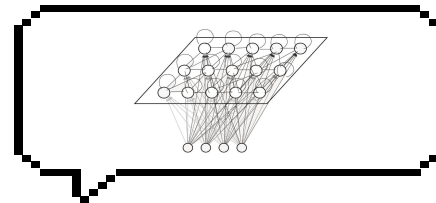
Detalles de implementación

Se hace uso del patrón **Singleton** para servir las constantes definidas en el archivo de configuración.

Se hace uso de **Numpy** para realizar operaciones matriciales y vectoriales con facilidad.

Se hace uso de **SKLearn** para PCA.

Las letras con ruido se generan utilizando un **filtro Salt & Pepper**, para el estudio de la red de Hopfield.



02

Red de Kohonen

El modelo descubre por sí misma regularidades o patrones en los datos de entrada.

¿Cómo se agrupan los países de Europa?

Ajuste de hiperparámetros

La red de neuronas consta de una sola capa de dimensión $N \times N$.

El tamaño de la grilla debe ser lo suficientemente grande para que las agrupaciones aporten información pero lo suficientemente pequeño para que no haya demasiadas neuronas muertas.

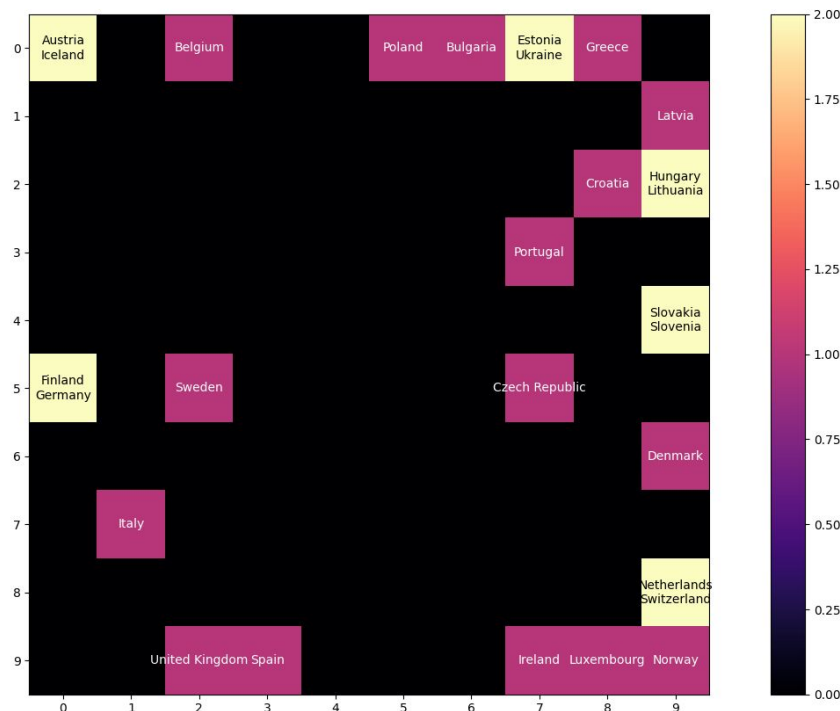
Comenzamos con los siguientes hiperparámetros:

- matrixSquareSize: **10**
- initialLearningRate = **0.5**
- initialRadius: **5**
- changeLearningRate: **true**
- changeRadius: **true**
- distanceCalculator: **EUCLIDEAN**

Ajuste de hiperparámetros

¿Cuál es el mejor valor de N ? Queremos obtener buena información y tener pocas neuronas muertas.

- matrixSquareSize: **10**
- initialLearningRate = **0.1**
- initialRadius: **5**
- changeLearningRate: **true**
- changeRadius: **true**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

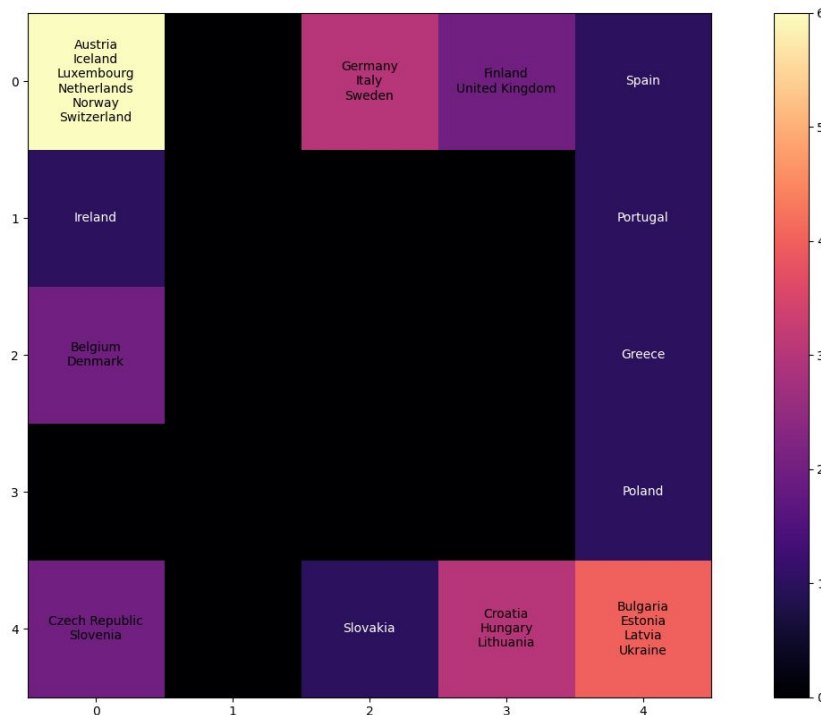
Conclusión:

- ✗ Encontramos muchas neuronas muertas
- ✗ Los clusters tienen pocos países

Ajuste de hiperparámetros

¿Cuál es el mejor valor de N ? Queremos obtener buena información y tener pocas neuronas muertas.

- matrixSquareSize: **5**
- initialLearningRate = **0.1**
- initialRadius: **5**
- changeLearningRate: **true**
- changeRadius: **true**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

Conclusión:

- ✗ Encontramos muchas neuronas muertas
- ✗ Los clusters tienen pocos países

Ajuste de hiperparámetros

¿Usando un único learning rate constante? Queremos obtener buena información y tener pocas neuronas muertas.

- matrixSquareSize: **5**
- initialLearningRate = **0.1**
- initialRadius: **5**
- changeLearningRate: **false**
- changeRadius: **true**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

Conclusión:



Encontramos pocas neuronas muertas



Los clusters tienen pocos países

Ajuste de hiperparámetros

¿Qué ocurre si seguimos disminuyendo N? También se utiliza un learning rate mayor y el menor radio posible.

- matrixSquareSize: **4**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

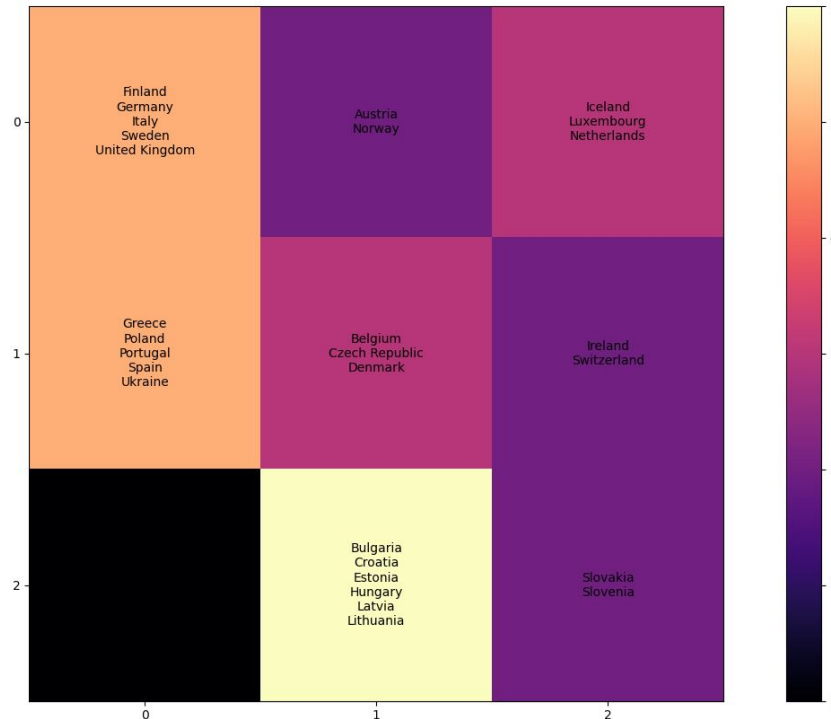
Conclusión:

- ✓ Encontramos pocas neuronas muertas
- ✗ Los clusters tienen pocos países

Ajuste de hiperparámetros

¿Qué ocurre si seguimos disminuyendo N? También se utiliza un learning rate mayor y el menor radio posible.

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

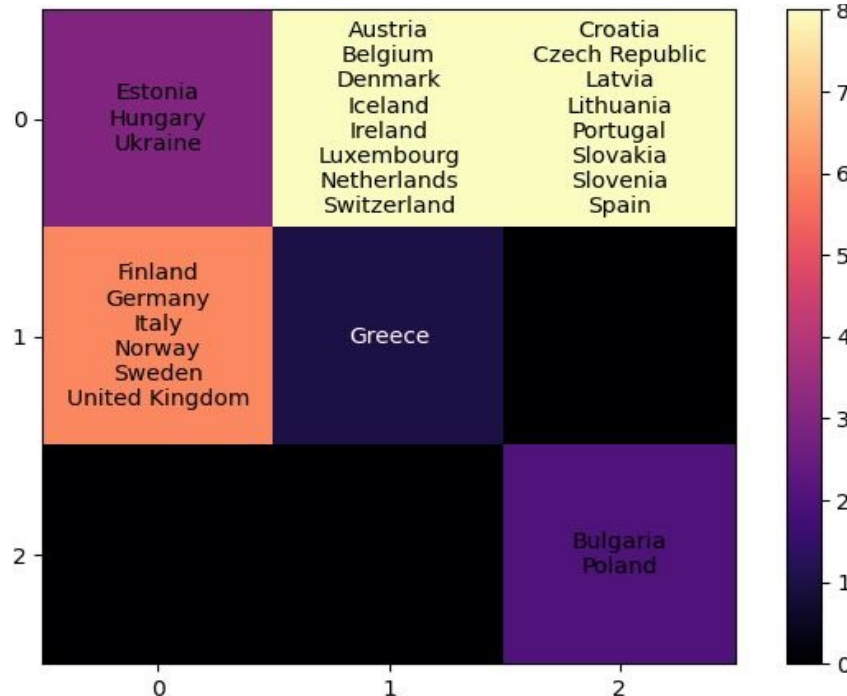
Conclusión:

- ✓ Encontramos una única neurona muerta
- ✓ Los clusters tienen dos o más países
- ✓ Los clusters aportan información

Ajuste de hiperparámetros

¿Afecta la función de cálculo de las distancias? Se busca obtener un resultado igual o mejor al anterior.

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EXPONENTIAL**



KOHONEN

Conclusión:



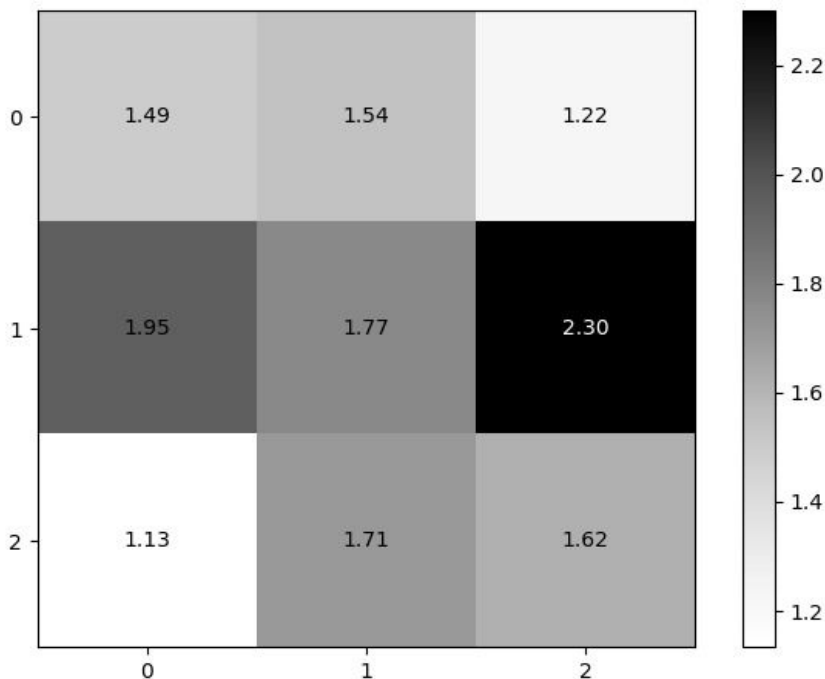
Encontramos muchas neuronas muertas

¿Tienen sentido los datos?

- **Finlandia, Alemania, Italia, Suecia, Reino Unido**: países con GDP, crecimiento poblacional, esperanza de vida similares.
- **Noruega, Austria**: superficie y economías similares.
- **Islandia, Luxemburgo, Países Bajos**: superficie pequeña con alta economía
- **Grecia, Polonia, Portugal, España, Ucrania**: economías con mayores dificultades de Europa, inflación alta.
- **Bulgaria, Croacia, Estonia, Hungría, Latvia, Lituania**: países de Europa del Este, comparten características históricas.
- **Bélgica, Dinamarca, República Checa**: economía estable, baja inflación y crecimiento poblacional moderado.
- **Irlanda, Suiza**: mayor GDP de Europa.
- **Eslovaquia, Eslovenia**: países pequeños, crecimiento poblacional moderado.

Distancia promedio entre vecinos

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

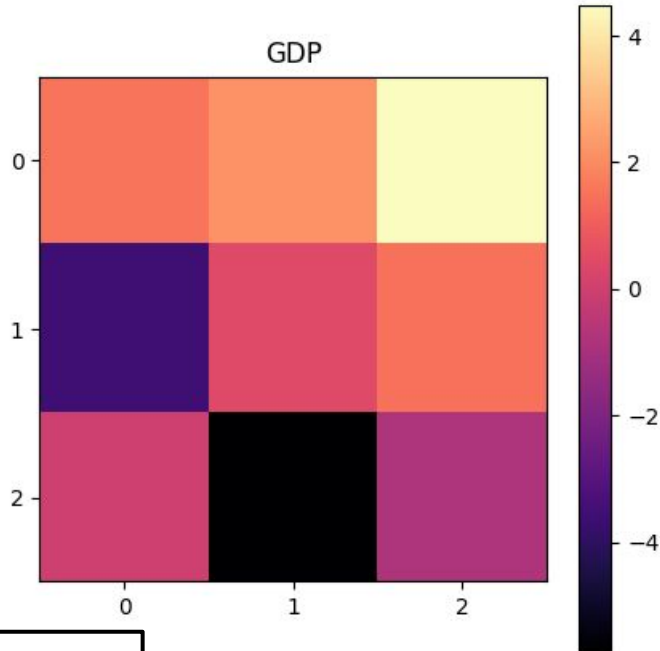
Conclusión:

- Se pueden observar distancias pequeñas
- Se pueden agrupar neuronas

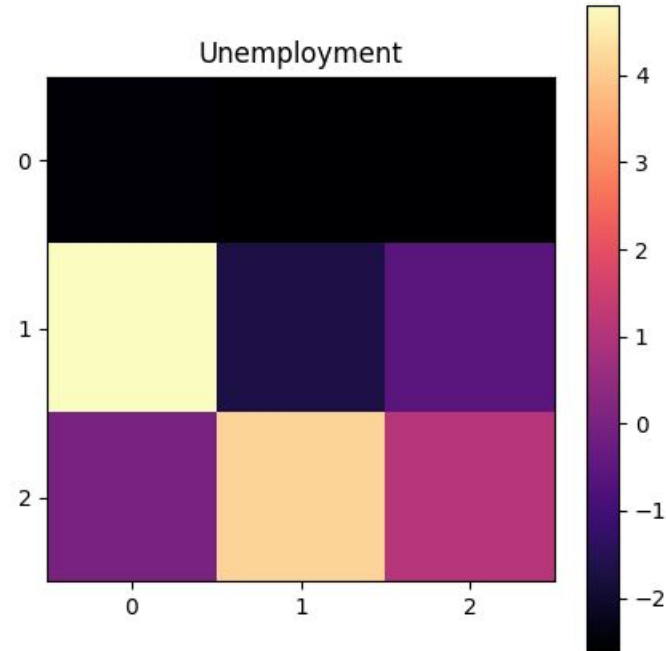
PBI vs Desempleo

¿Tienen alguna relación estas variables?

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

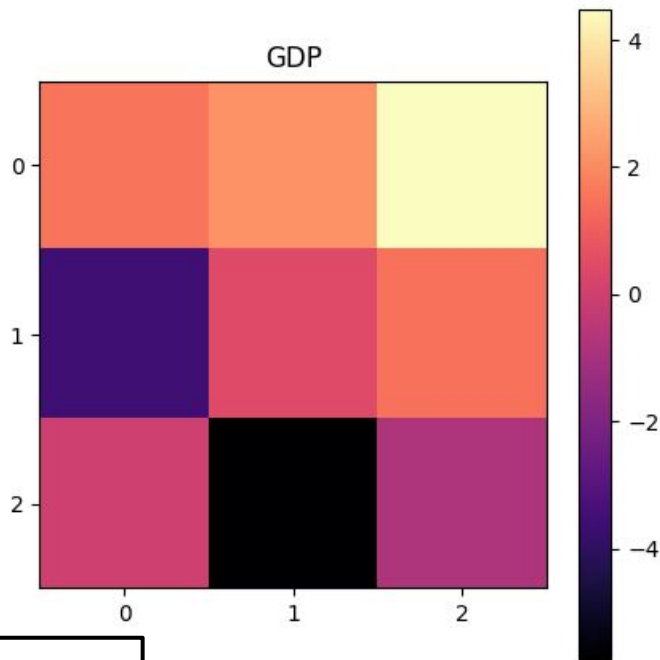


Conclusión:
**inversamente
proporcionales**

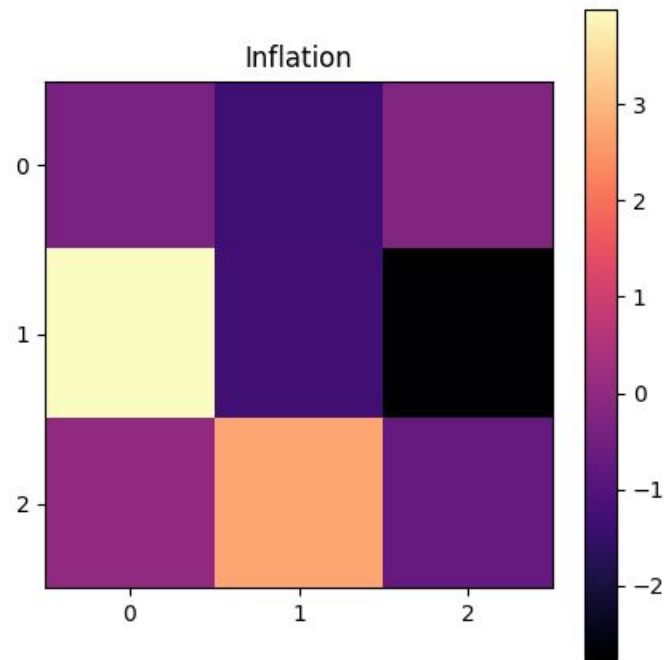
PBI vs Inflación

¿Tienen alguna relación estas variables?

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

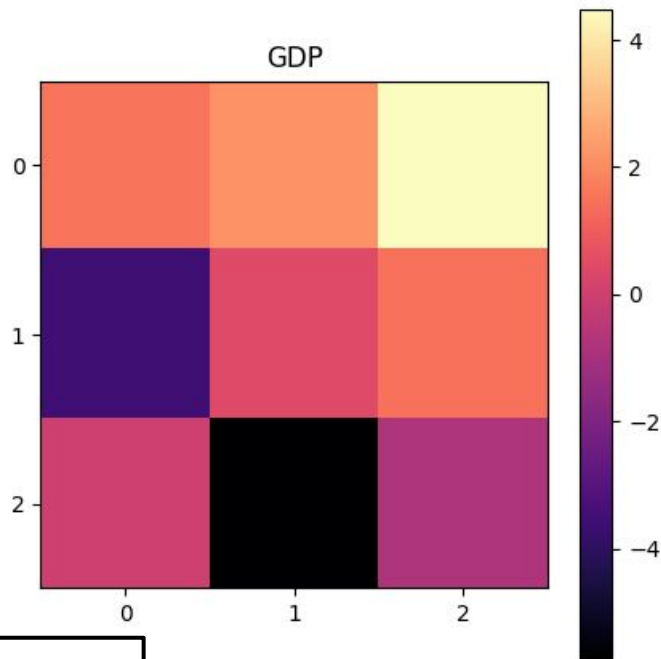


Conclusión:
**inversamente
proporcionales**

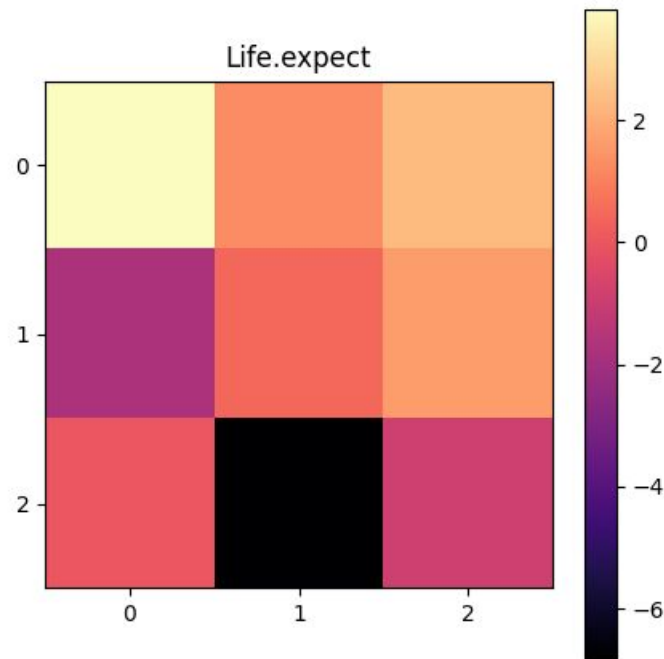
PBI vs Exp. de Vida

¿Tienen alguna relación estas variables?

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

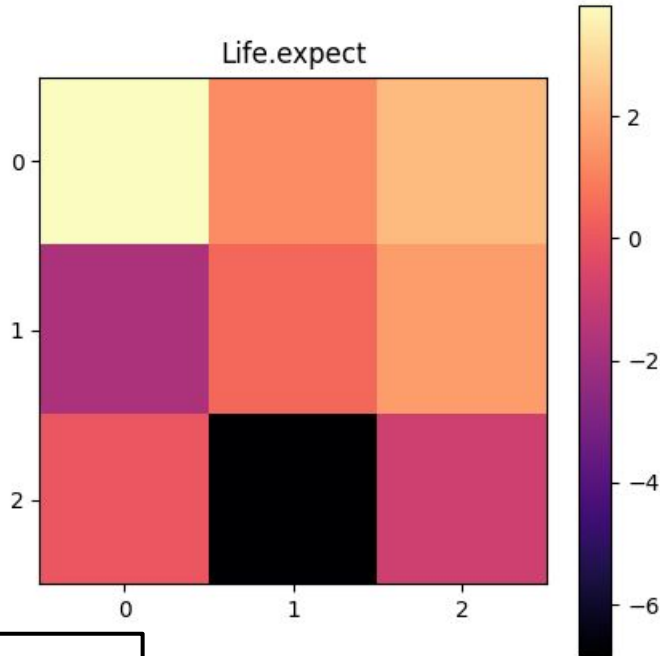


Conclusión:
**directamente
proporcionales**

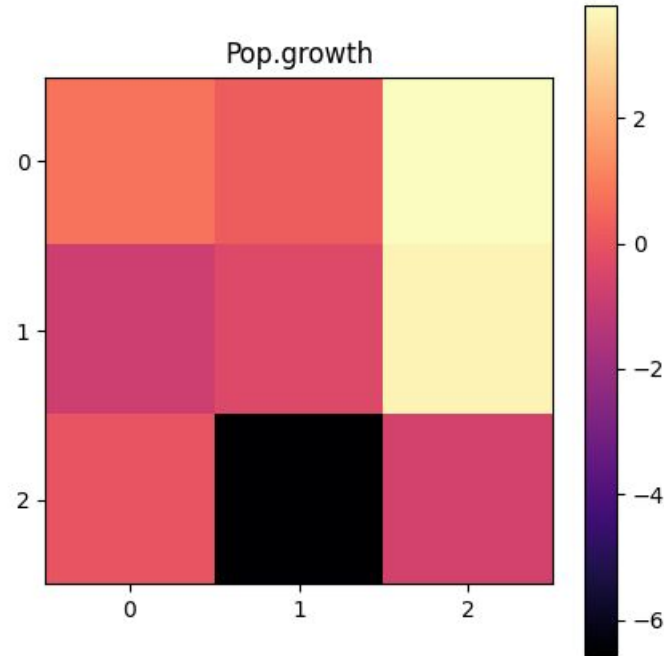
Exp. de Vida vs Crec. Pobl.

¿Tienen alguna relación estas variables?

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



KOHONEN

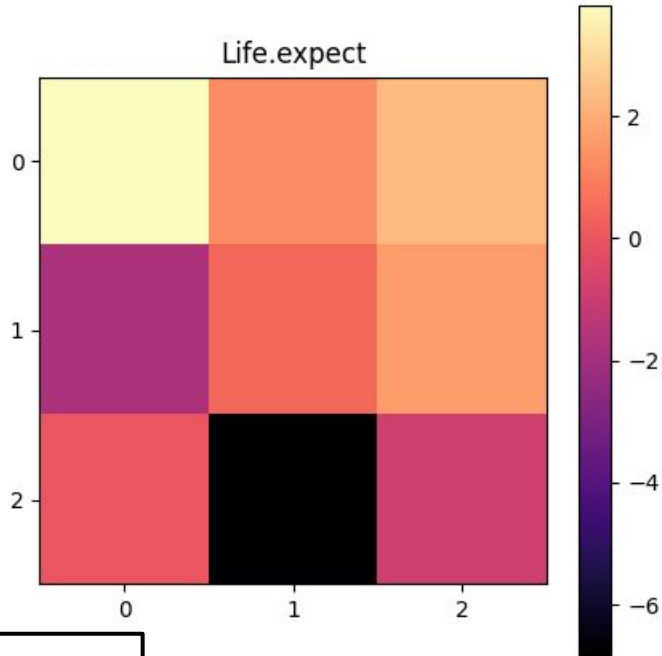


Conclusión:
**directamente
proporcionales**

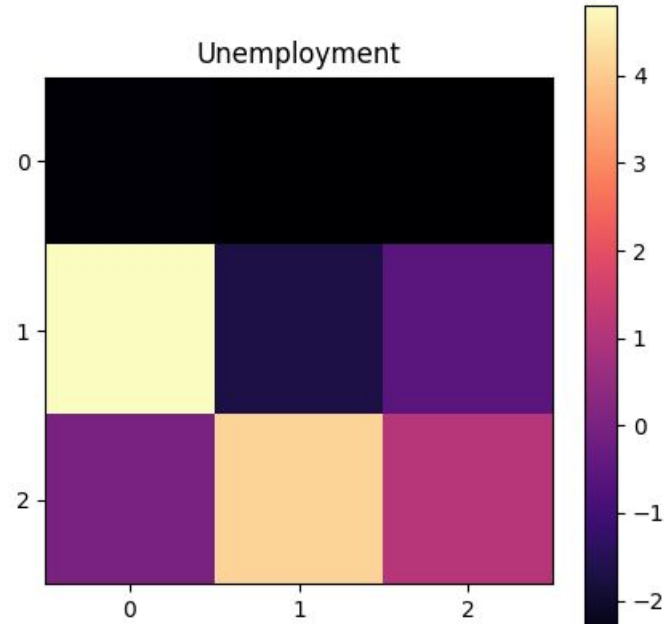
Exp. de Vida vs Desempleo

¿Tienen alguna relación estas variables?

- matrixSquareSize: **3**
- initialLearningRate = **0.3**
- initialRadius: **1**
- changeLearningRate: **false**
- changeRadius: **false**
- distanceCalculator: **EUCLIDEAN**



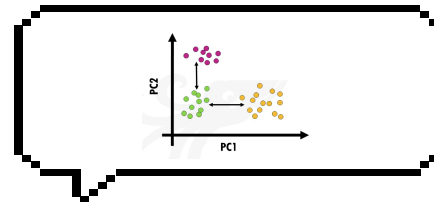
KOHONEN



Conclusión:
**inversamente
proporcionales**

Conclusiones

- Se puede encontrar relaciones entre las variables estudiadas.
- Hay una relación inversamente proporcional entre el PBI y el desempleo, así también con el PBI y la inflación.
- Hay una relación proporcional entre el PBI y la expectativa de vida, la expectativa de vida y el crecimiento poblacional y la expectativa de vida y el Desempleo.
- Las variables que más se pudieron relacionar fueron el PBI y la expectativa de vida.
- La variable que menos se pudo relacionar es el crecimiento poblacional



03

Analisis de PCA

Se utiliza para extraer características destacadas de un conjunto de datos reduciendo la dimensión de estos.

¿Cuáles son las componentes principales del dataset de Europa?

Cargas PC1

El componente principal PC1 será combinación lineal de estos valores:

-0.1248739	0.50050586	-0.40651815	0.48287333	-0.18811162	0.47570355	-0.27165582
------------	------------	-------------	------------	-------------	------------	-------------

$$\begin{aligned} PC1 = & -Area \cdot 0.1248 + GDP \cdot 0.5005 - Inflation \cdot 0.4065 \\ & + Life.Expect \cdot 0.4828 - Military \cdot 0.1881 \\ & + Pop.Growth \cdot 0.4757 - Unemployment \cdot 0.2716 \end{aligned}$$

Interpretación de PC1

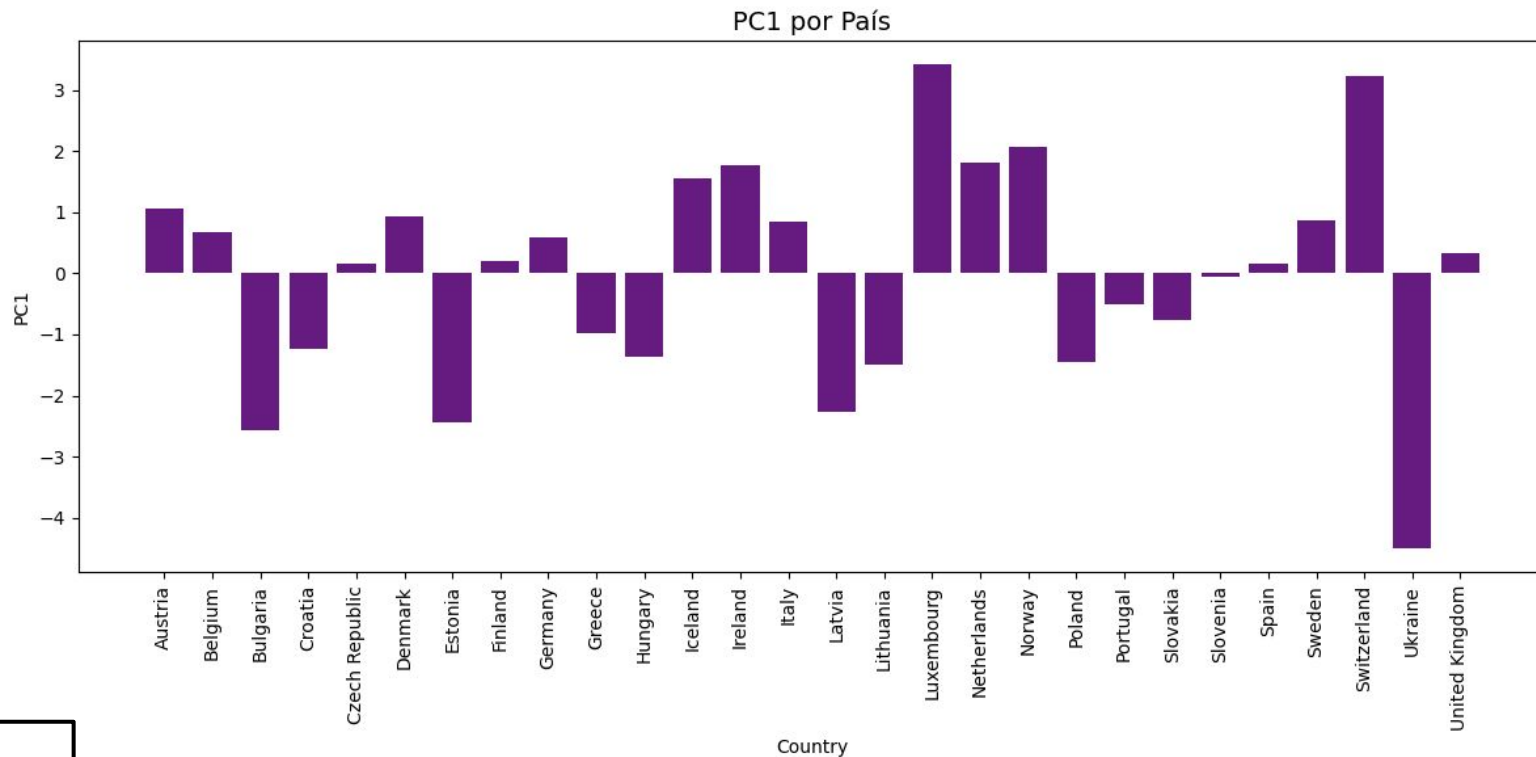
$$\begin{aligned} PC1 = & - Area \cdot 0.1248 + GDP \cdot 0.5005 - Inflation \cdot 0.4065 \\ & + Life.Expect \cdot 0.4828 - Military \cdot 0.1881 \\ & + Pop.Growth \cdot 0.4757 - Unemployment \cdot 0.2716 \end{aligned}$$

La variable que **más influye en PC1 es el producto bruto interno (GDP)**, seguida por la expectativa de vida, el crecimiento de la población y la inflación de manera inversa.

Un **alto PBI hace aumentar PC1** mientras que una alta inflación lo hace bajar. En menor medida, también influyen el área del país, el gasto militar y la tasa de desempleo.

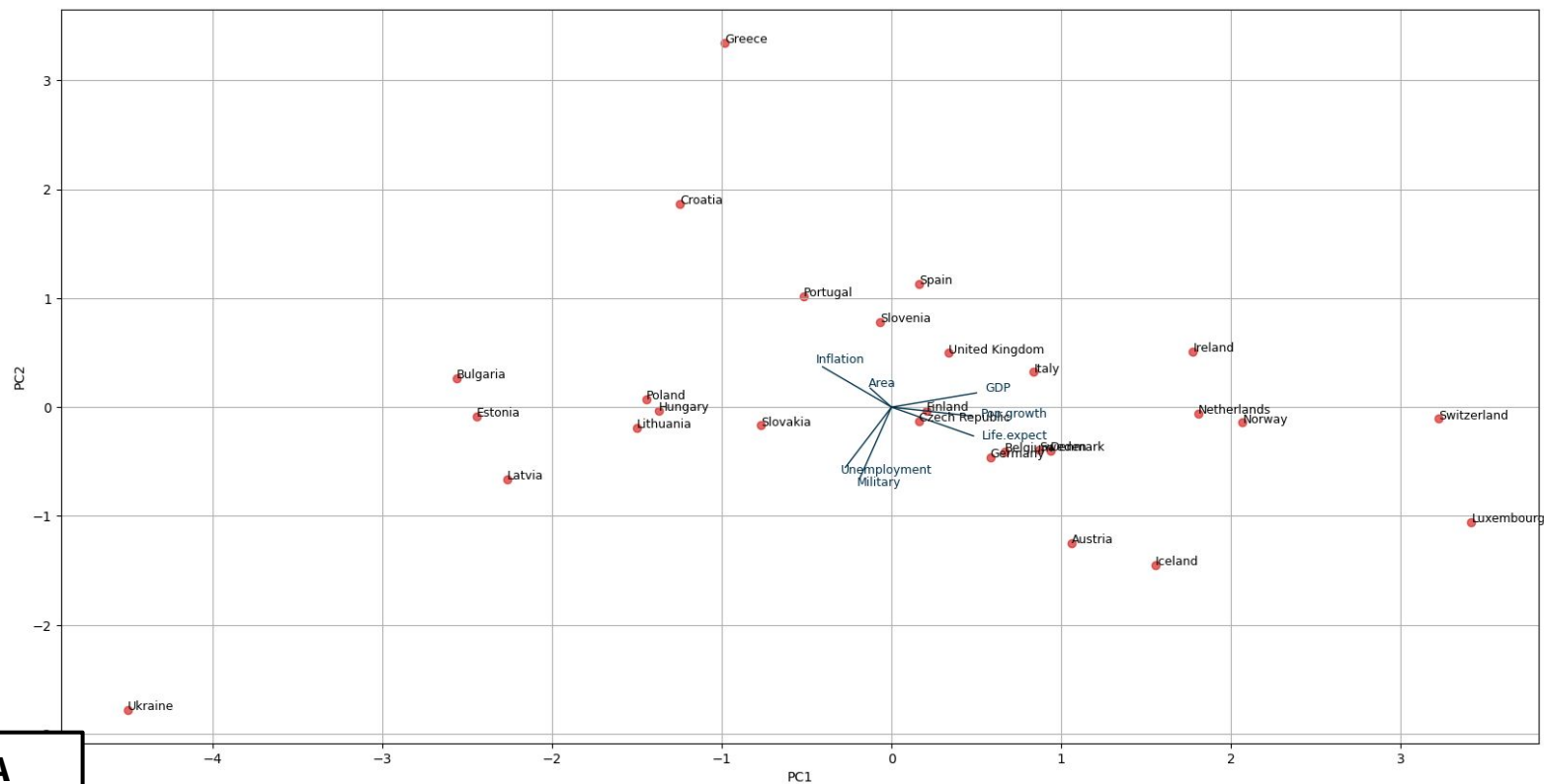
Es decir, es una variable que mide el nivel de vida en el país.

PC1 por país



PCA

Biplot

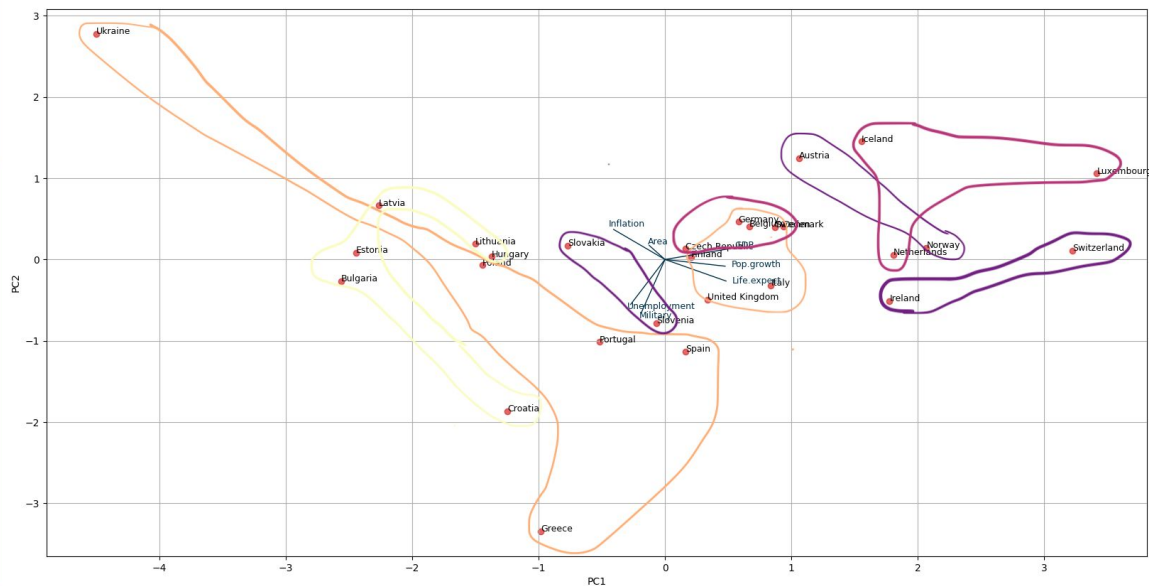
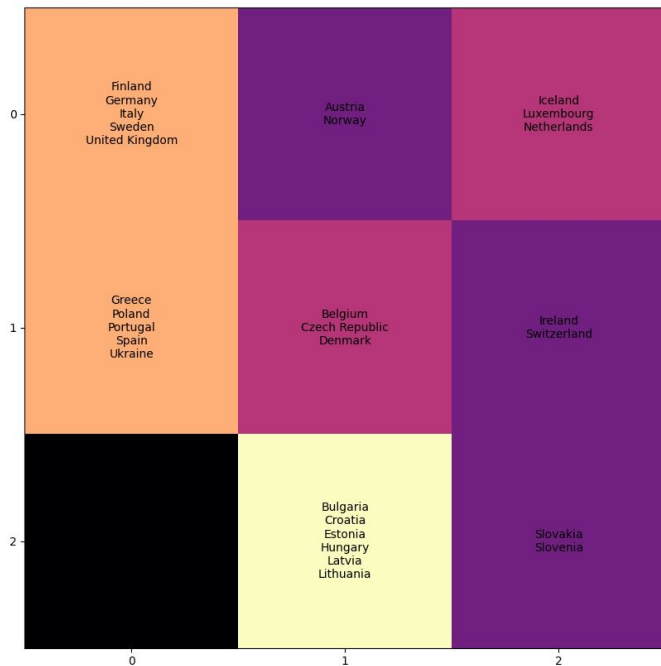


PCA


Conclusiones

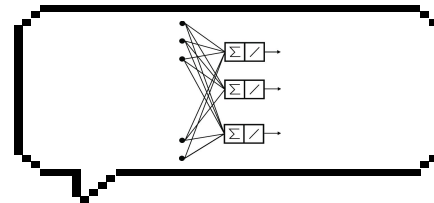
- Irlanda, Noruega, Países Bajos: condiciones económicas y sociales similares.
- Grecia, Croacia, Ucrania: aislados y con diferencias significativas con otros países.
- Bulgaria, Letonia, Estonia: países de Europa del Este características similares, distintas de Europa Occidental.
- Ucrania: outlier, condiciones económicas únicas.
- Alemania, Reino Unido, Francia: alineados en cuanto a PBI y expectativa de vida.
- Hungría, Polonia, Lituania: inflación predominante.
- República Checa, Eslovaquia, Finlandia: equilibrados en las variables medidas.

Comparativa Kohonen vs PCA



Kohonen vs PCA

Conclusión:  Los clusters, en su mayoría, coinciden



04

Modelo de Oja

¿Cómo se calcula la primera componente principal?
Usamos redes neuronales para obtenerla de manera iterativa

Cargas PC1

La red va a converger al **autovector asociado al mayor autovalor** de la matriz de correlaciones.

Cuando la red converge, el ω final va a ser igual al vector de cargas de PC1

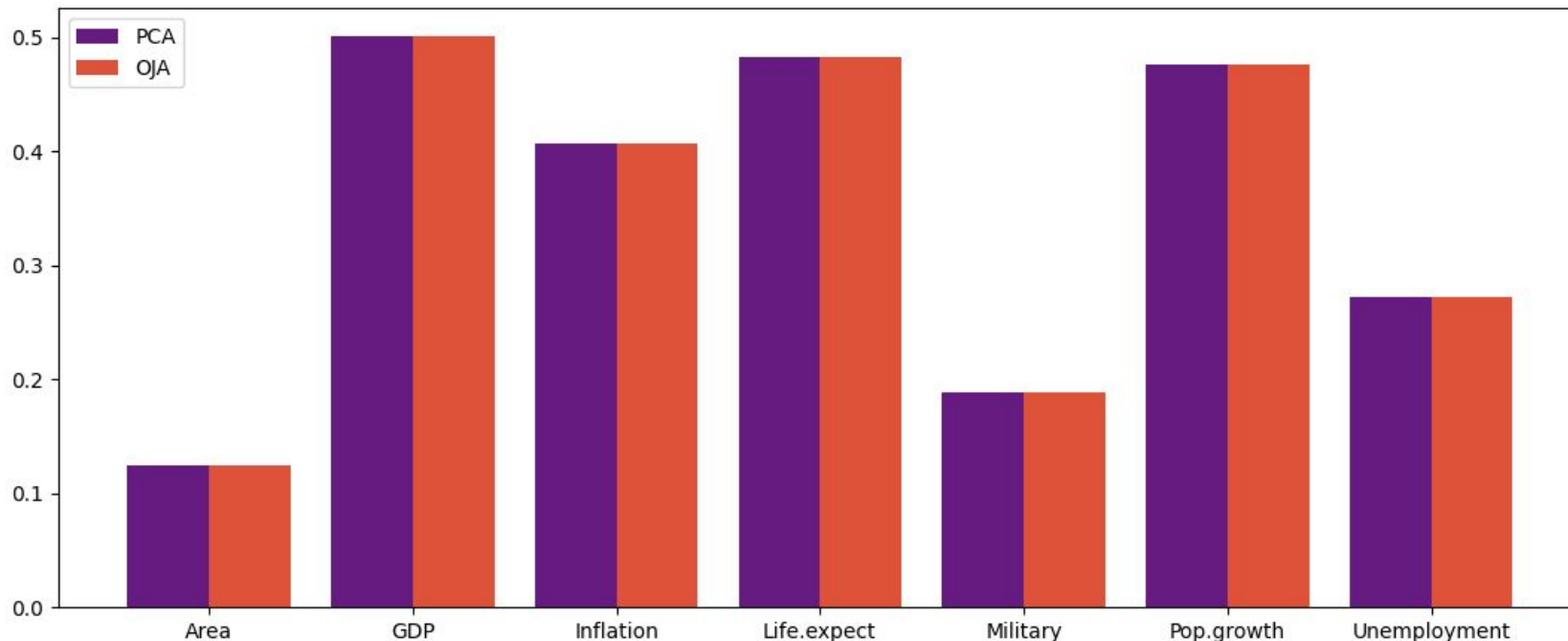
Se obtiene el siguiente vector de valores:

0.12495734	-0.50049711	0.40659913	-0.4828901	0.188041	-0.475686	0.27161368
------------	-------------	------------	------------	----------	-----------	------------

Hiperparámetros utilizados

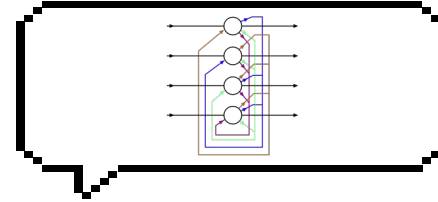
- maxEpochs: **10000**
- initialLearningRate = **0.1**
- changeLearningRate: **true**

Cargas PC1: PCA vs Oja



PCA vs OJA

Conclusión: ✓ Las cargas de PC1 coinciden



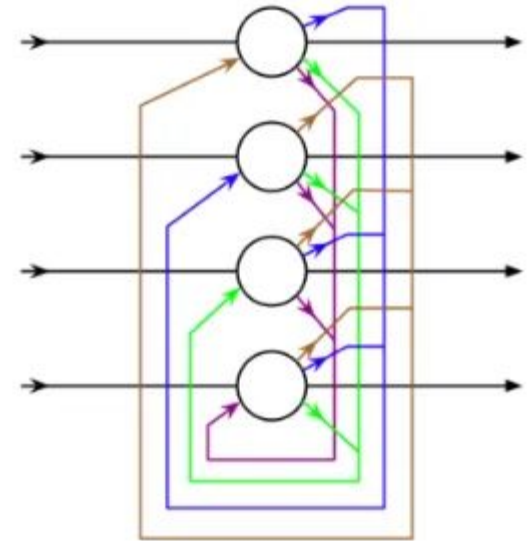
05

Modelo de Hopfield

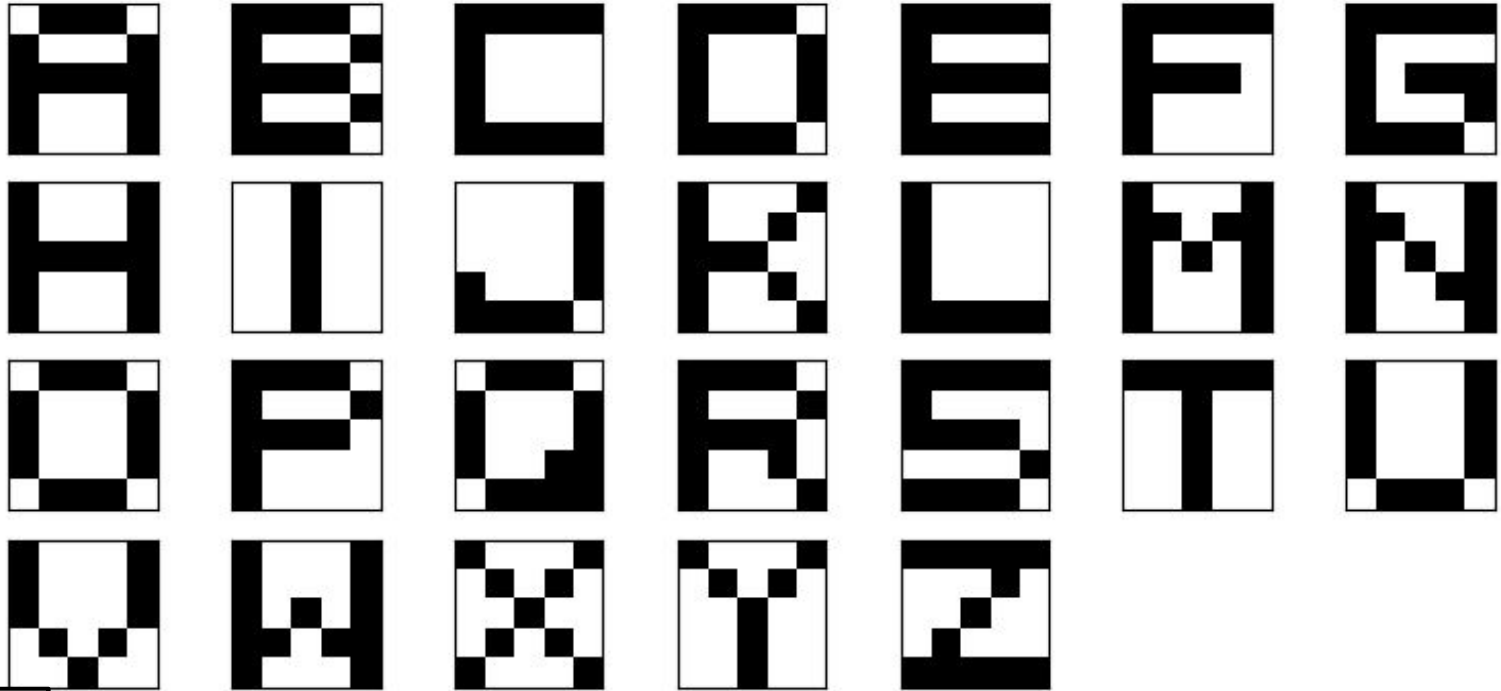
Se busca asociar un patrón de consulta perturbado
con alguno de los patrones almacenados.
¿Se logra identificar letras del alfabeto?

Particularidades del modelo

- Todas las neuronas están conectadas entre sí, pero no consigo mismas.
- El cálculo de los pesos sinápticos se realiza “offline”, cuando se definen los patrones a reconocer.
- El número máximo de patrones que puede almacenar es igual al 15% del número de neuronas de la red.
- Los patrones almacenados deben ser aproximadamente ortogonales.

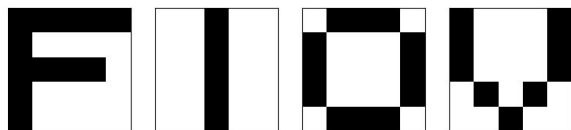


Definición de nuestro alfabeto

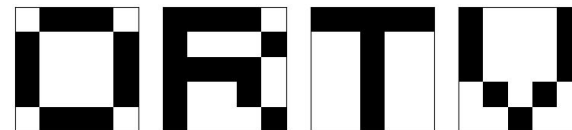


HOPFIELD

Análisis de ortogonalidad



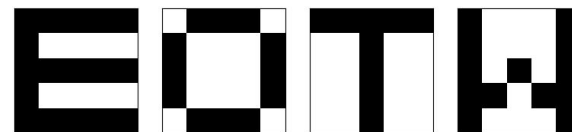
Avg. Ortogonalidad: **1.33**
 Distribution: **{1: 5, 3: 1}**



Avg. Ortogonalidad: **1.33**
 Distribution: **{1: 4, 3: 2}**



Avg. Ortogonalidad: **3.0**
 Distribution: **{1: 3, 5: 3}**



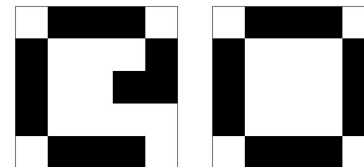
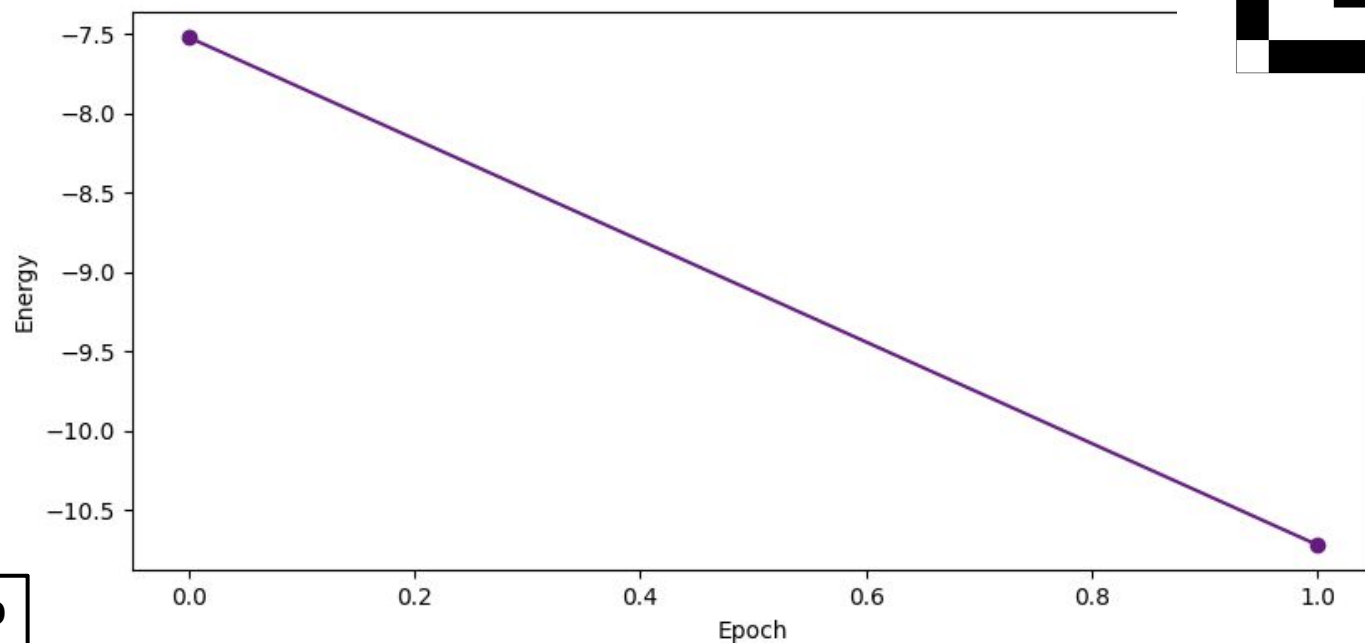
Avg. Ortogonalidad: **3.0**
 Distribution: **{1: 4, 7: 2}**



Avg. Ortogonalidad: **18.33**
 Distribution: **{17: 4, 21: 2}**

Identificación de patrones

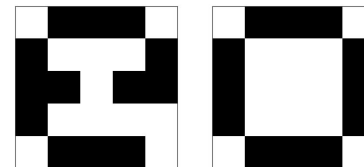
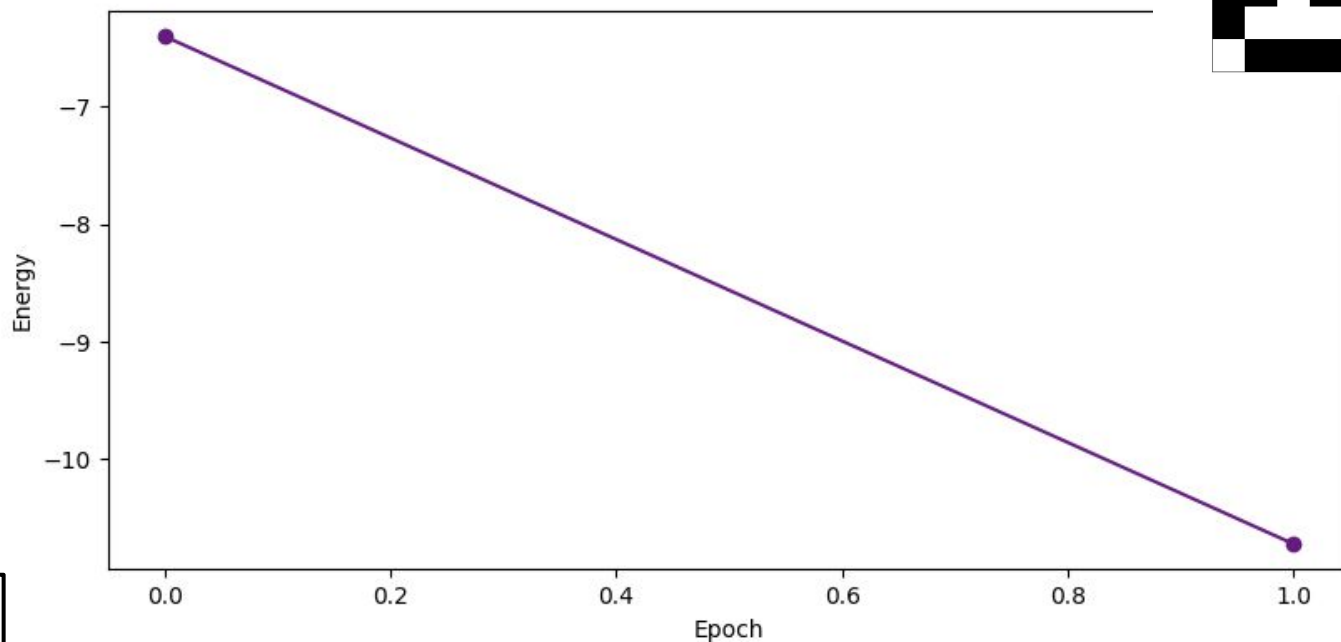
- maxEpochs: 100
- noiseLevel = 0.05
- letters: ['F', 'I', 'O', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

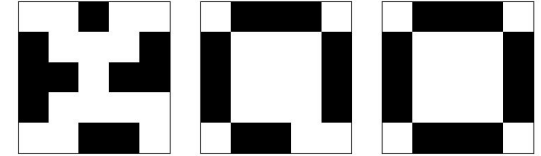
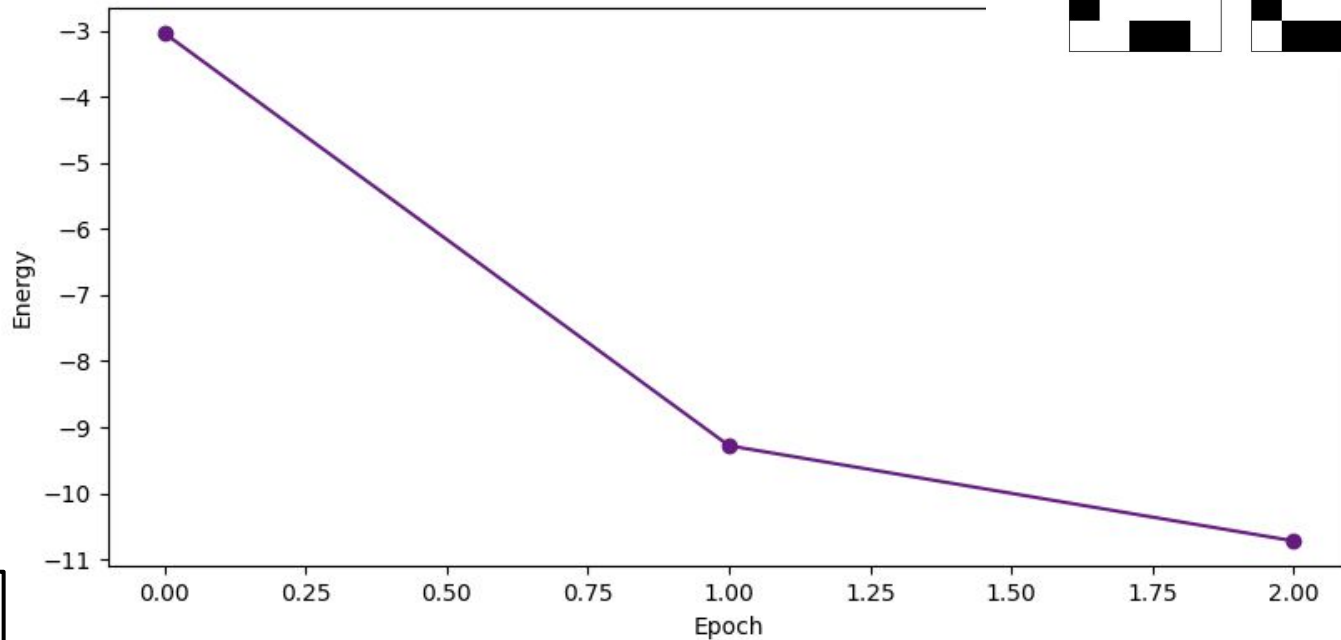
- maxEpochs: 100
- noiseLevel = 0.1
- letters: ['F', 'I', 'O', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

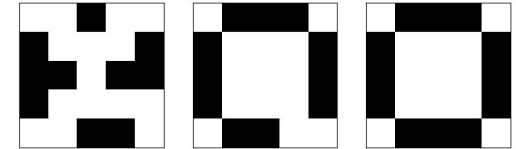
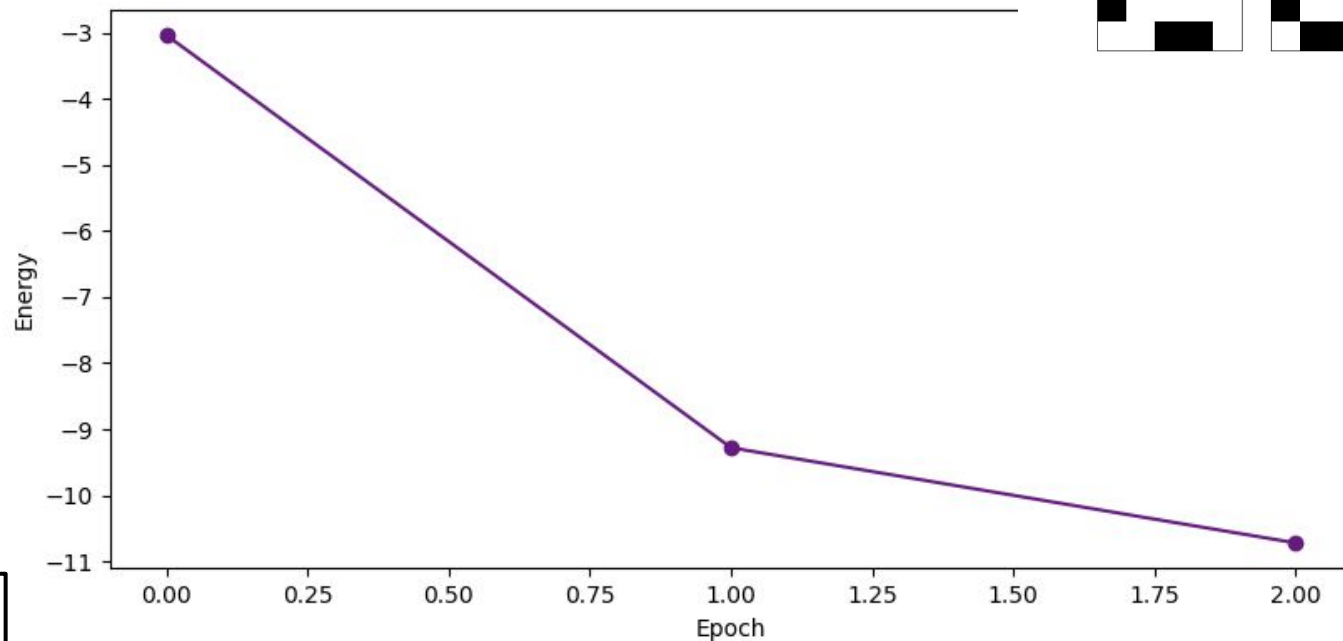
- maxEpochs: 100
- noiseLevel = 0.15
- letters: ['F', 'I', 'O', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

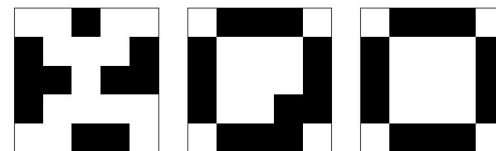
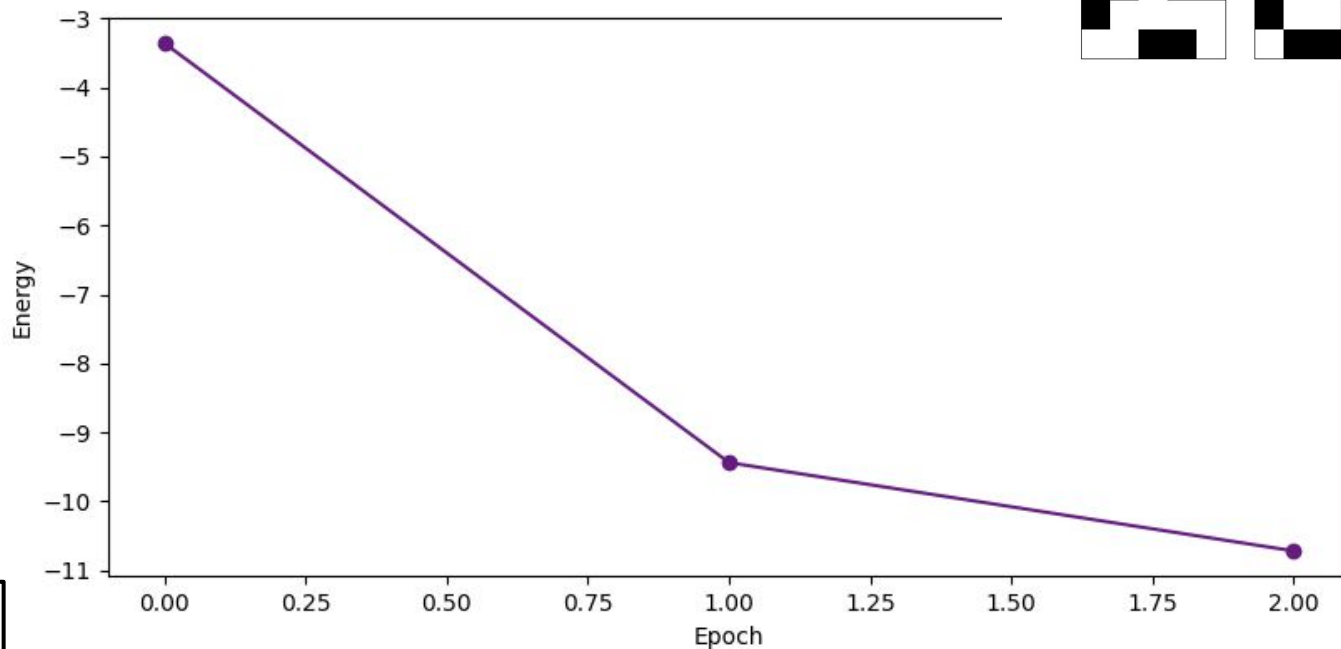
- maxEpochs: 100
- noiseLevel = 0.19
- letters: ['F', 'I', 'O', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

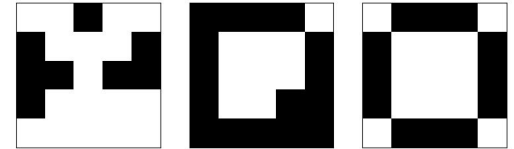
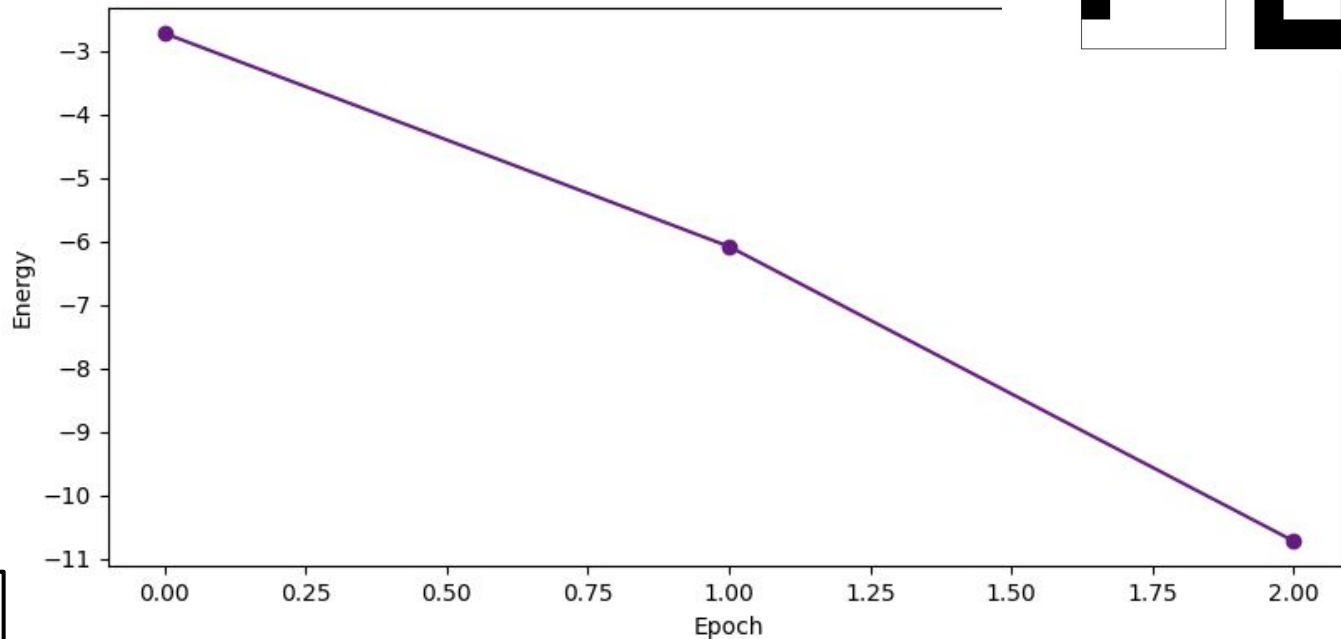
- maxEpochs: 100
- noiseLevel = 0.15
- letters: ['O', 'R', 'T', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

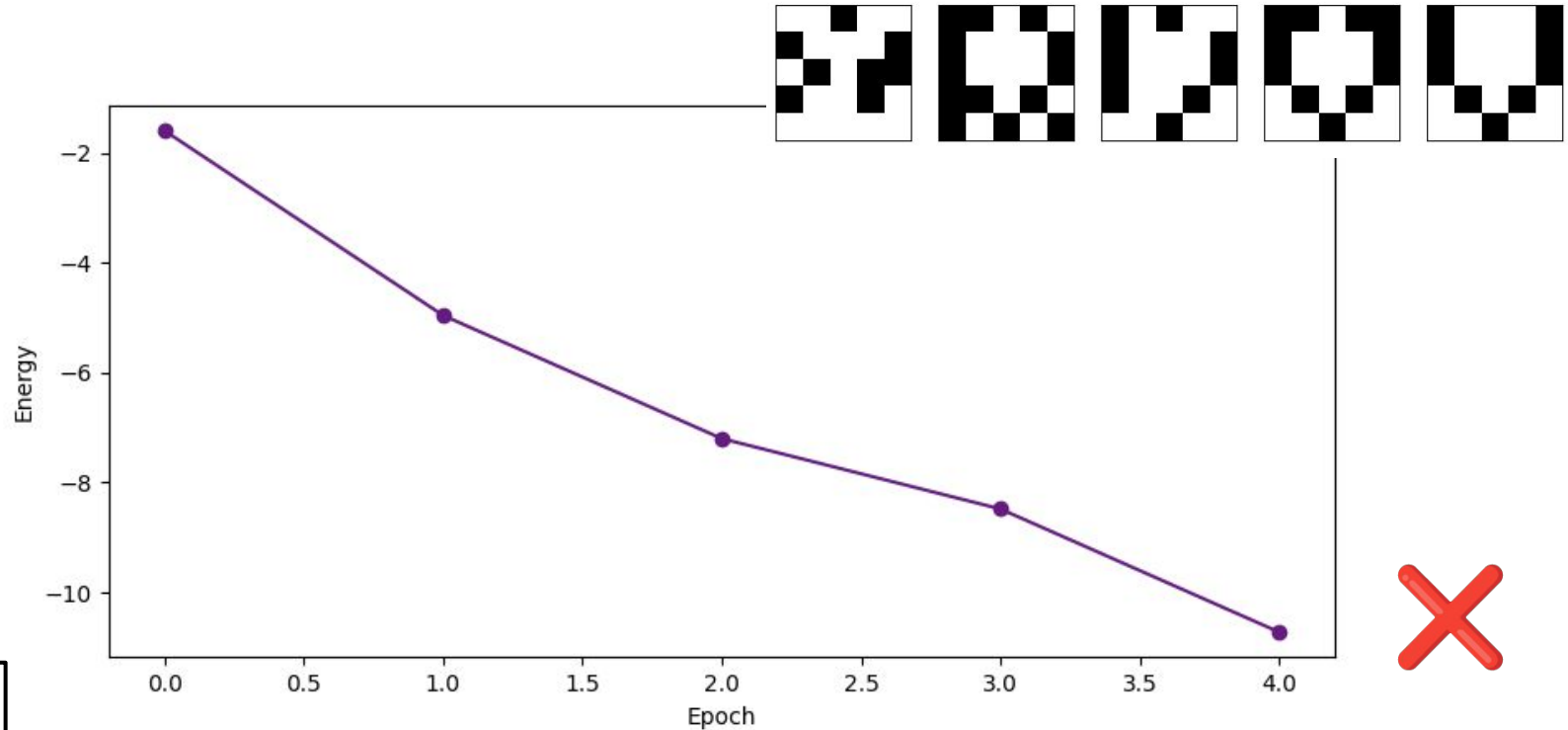
- maxEpochs: 100
- noiseLevel = 0.25
- letters: ['O', 'R', 'T', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

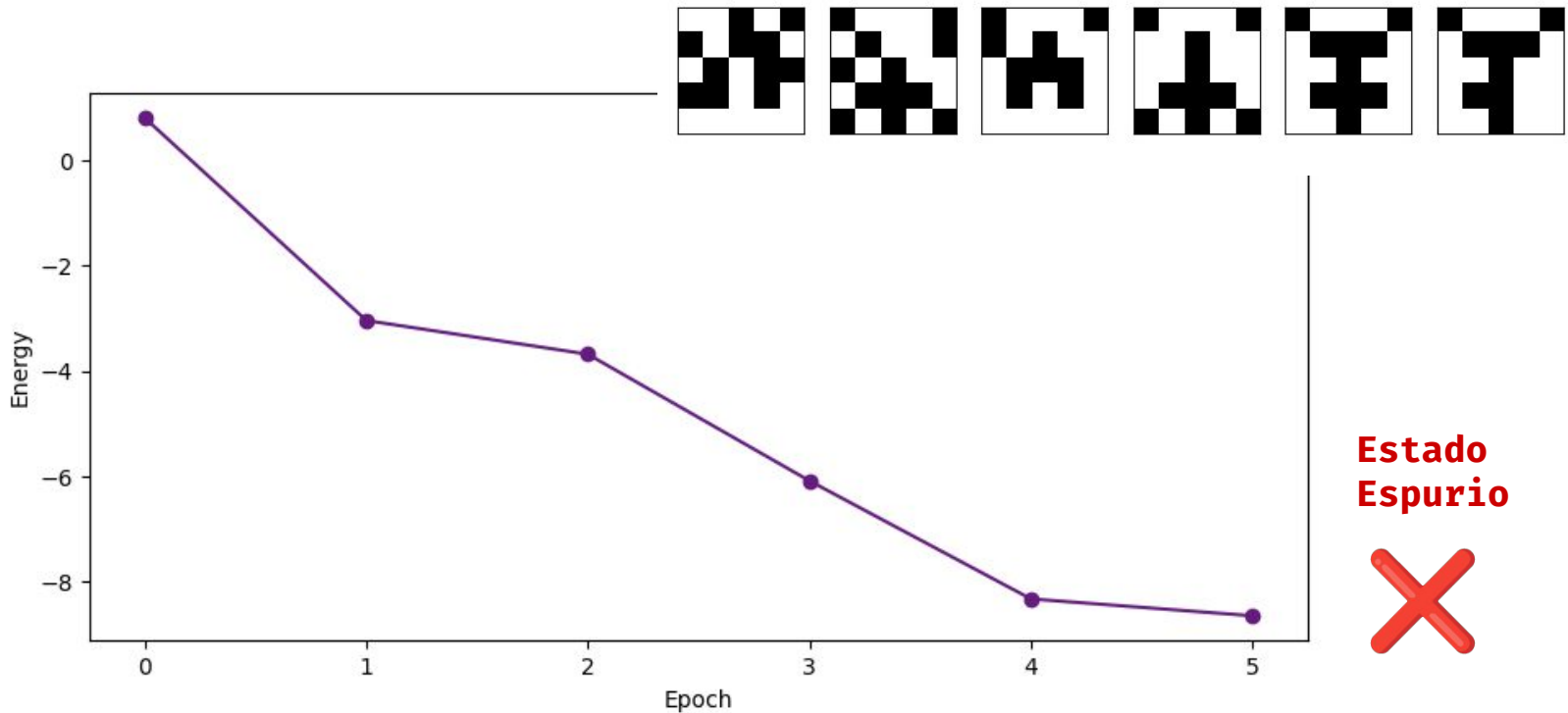
- maxEpochs: 100
- noiseLevel = 0.35
- letters: ['O', 'R', 'T', 'V']
- letterToRecognize: 'O'



HOPFIELD

Identificación de patrones

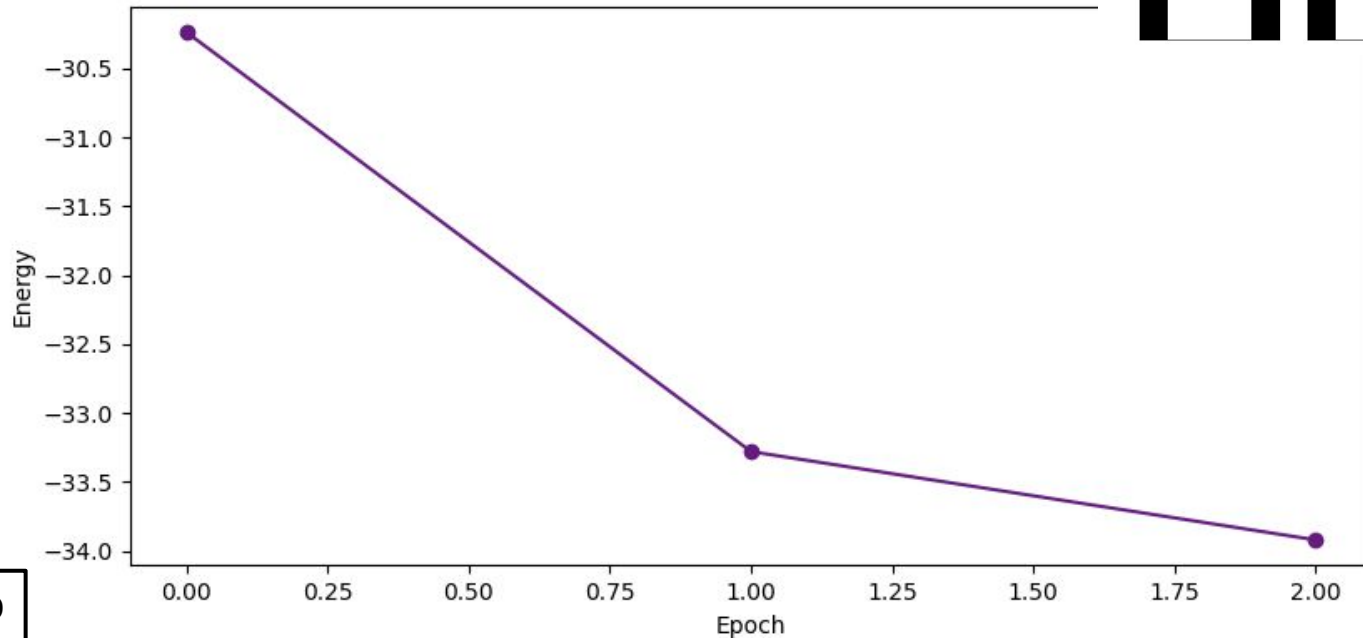
- maxEpochs: 100
- noiseLevel = 0.6
- letters: ['O', 'R', 'T', 'V']
- letterToRecognize: 'O'



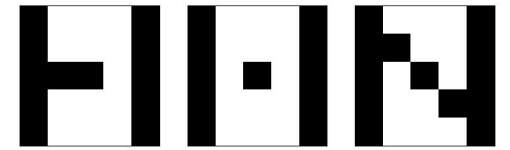
HOPFIELD

Identificación de patrones

- maxEpochs: 100
- noiseLevel = 0.01
- letters: ['H', 'M', 'N', 'W']
- letterToRecognize: 'H'

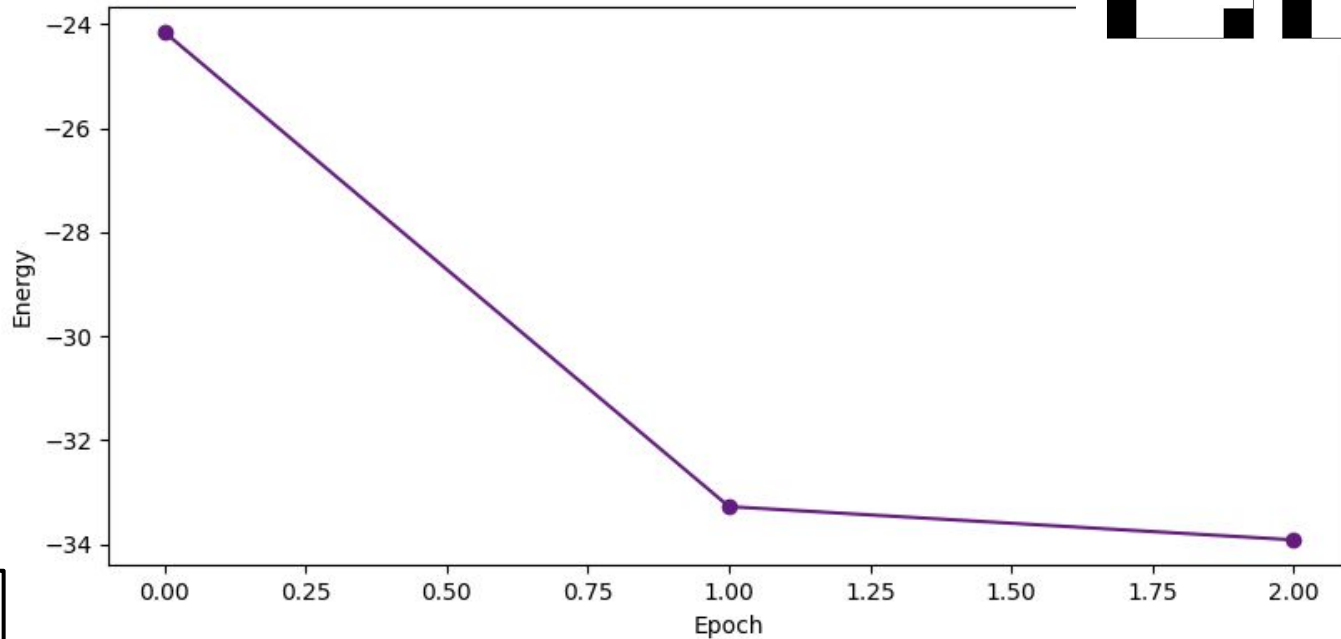


HOPFIELD

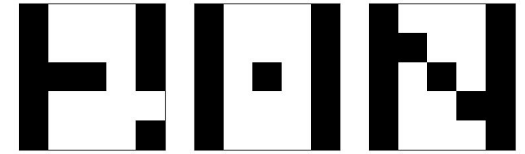


Identificación de patrones

- maxEpochs: 100
- noiseLevel = 0.05
- letters: ['H', 'M', 'N', 'W']
- letterToRecognize: 'H'



HOPFIELD



Conclusiones

- La cantidad de ruido afecta en la asociación de un patrón almacenado.
- Dado el ejercicio a realizar, no requiere de muchas épocas para converger o no converger.
- Si los patrones almacenados son muy parecidos entre sí, más le cuesta distinguir una letra (a pesar de que tenga poco ruido). Esto se asocia a la ortogonalidad de los vectores.
- La distribución de ortogonalidad de los patrones almacenados afecta la capacidad de distinción de la red.

¡Muchas gracias por su atención!

CREDITS: This presentation template was created by
Slidesgo, and includes icons by **Flaticon**, and infographics
& images by **Freepik**

Enlaces Útiles

- <https://github.com/alejofl/sia>
Repositorio del proyecto.