

Inteligencia Artificial

Unidad 2: Redes Neuronales

TEMA 3: Algoritmos de IA Moderna-I

Módulo 4: Mapas Auto-Organizados (SOM) – Redes de Kohonen

Unidad 3

Redes Neuronales

TEMA 3: Algoritmos de IA Moderna-I

Sesión 18

MÓDULO 4: Mapas Auto-Organizados (SOM) – Redes de Kohonen



Contenido

1. ¿Qué son los Mapas Auto-Organizados?
2. SOM: Estructura y funcionamiento
3. ¿Cómo aprenden los SOM?
4. Ejemplo: Topología de Red SOM
5. Aplicaciones de los SOM – Redes de Kohonen



Preguntas

1. ¿Qué son los Mapas Auto-Organizados?

Historia - Mapas Auto-Organizados

- Conocidos en inglés como **Self Organizing Maps (SOM)**, estos mapas fueron desarrollados por el profesor emérito de la Academia de Finlandia **Teuvo Kohonen** en 1982 por el cual también son conocidos como **Redes de Kohonen**.
- La idea de Kohonen SOM sólo se hizo famosa mucho más tarde, en 1988, cuando presentó un artículo sobre "La máquina de escribir fonética neuronal" y su trabajo se hizo ampliamente conocido.
- Desde entonces se han realizado muchos artículos y libros excelentes sobre SOM.



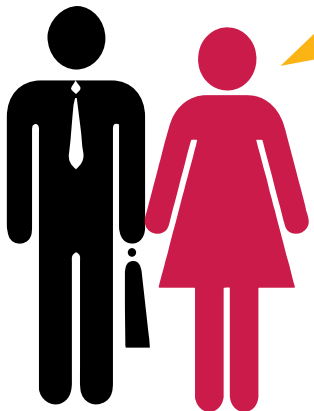
Profesor Teuvo Kohonen

1. ¿Qué son los Mapas Auto-Organizados?

Utilidad - Mapas Auto-Organizados

- Los Mapas Auto-Organizados (SOM), son un tipo de red neural artificial (RNA) que se entrena utilizando el aprendizaje no supervisado.
- En el aprendizaje no supervisado, recordemos que los datos no se encuentran etiquetados, por tanto, no conocemos a priori como se clasifican.

Pero, ¿Qué son y para qué se utilizan estos mapas?



Se utilizan para producir una representación discreta y de baja dimensión (típicamente bidimensional) del espacio de entrada de las muestras de entrenamiento, llamado mapa, y por lo tanto es un método para hacer reducción de la dimensionalidad.

1. ¿Qué son los Mapas Auto-Organizados?

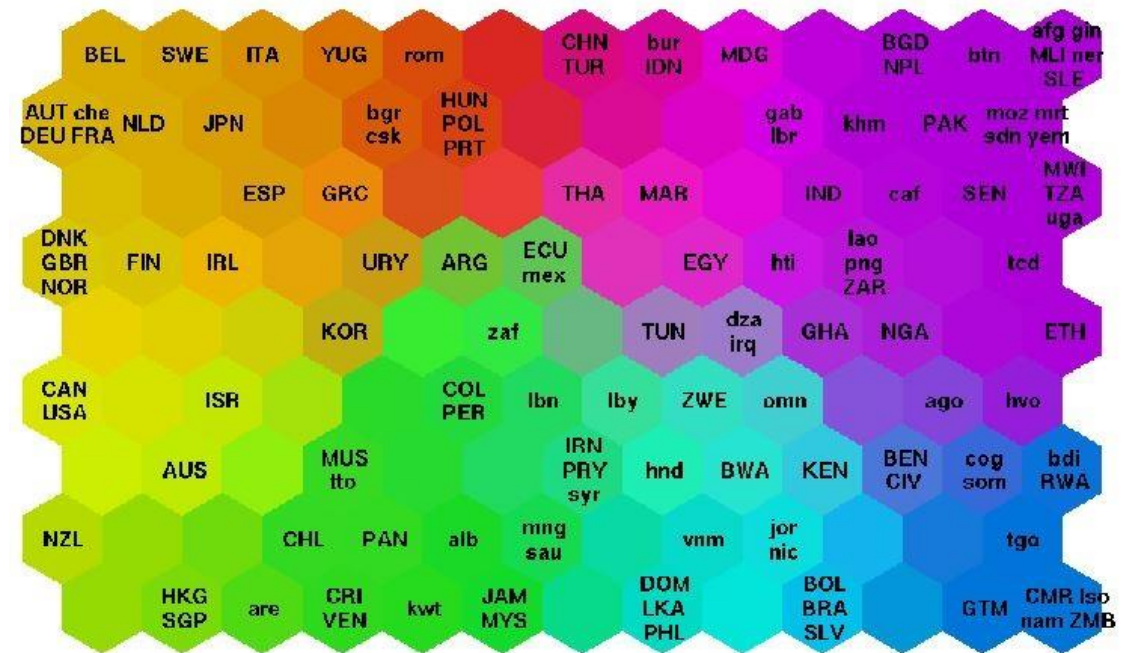
SOM: Reducción de dimensionalidad

- Consideremos que tenemos un conjunto de datos con muchas características (variables) y que deseamos reducir esa cantidad de columnas para el análisis.

Mediante SOM:

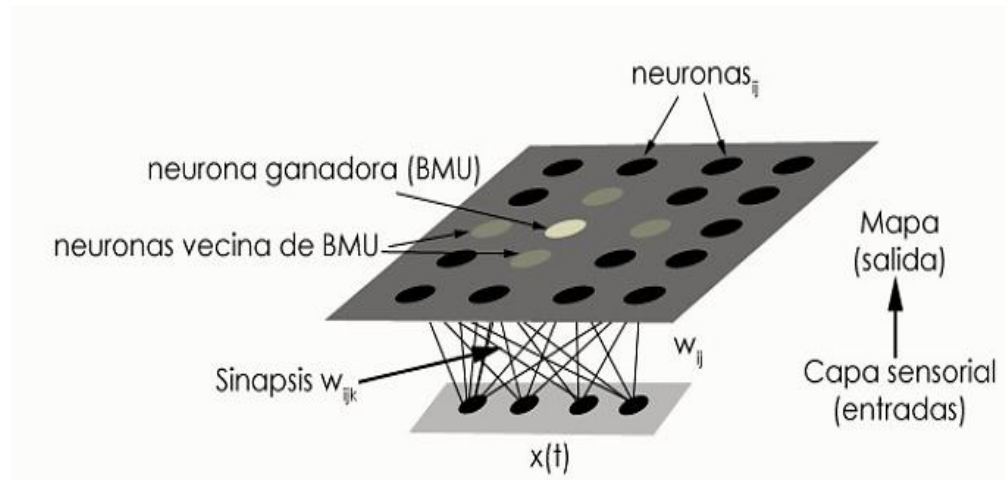
- Podemos reducir este conjunto de datos a dos dimensiones y así poder analizar mas fácilmente los datos y llegar a conclusiones (obtener diferentes patrones).
- Al reducir la dimensionalidad, obtenemos una mapa 2D, donde se agrupan diferentes clusters o agrupaciones de los datos de características similares.
- Trabajamos con conjunto de datos no etiquetados (no se tiene la salida objetivo), por ello, el aprendizaje es **NO-SUPERVISADO**.
- Podemos explorar espacios vectoriales en los que se desconoce la estructura de clasificación de los vectores.

Mapa Auto-Organizado de países según su estado de prosperidad y pobreza



2. SOM: Estructura y funcionamiento

Estructura de un mapa auto-organizado (organización unidireccional de dos capas)



BMU: El nodo ganador es comúnmente conocido como **Best Matching Unit**

- ❖ La idea básica del modelo es crear una imagen de un espacio multidimensional de entrada en un espacio de salida de menor dimensionalidad.
- ❖ Se trata de un modelo con dos capas de neuronas, una de entrada y otra de salida (procesamiento).
- ❖ Las neuronas de la primera capa se limitan a recoger y canalizar la información.
- ❖ La segunda capa está conectada a la primera a través de los pesos sinápticos y realiza la tarea importante: una proyección no lineal del espacio multidimensional de entrada, preservando las características esenciales de estos datos en forma de relaciones de vecindad.
- ❖ El resultado final es la creación del llamado **mapa auto-organizado** donde se representan los rasgos más sobresalientes del espacio de entrada.

2. SOM: Estructura y funcionamiento

SOM: Seudocódigo

1. Inicializar todas las neuronas de la capa de salida.
2. Repetir mientras las neuronas no cambien mucho o se exceda un umbral.
 3. Escoger la siguiente entrada.
 4. Determinar la neurona más cercana a la entrada (neurona ganadora, mejor coincidencia o Best Matching Unit – BMU)
 5. Actualizar los pesos de esta neurona y las neuronas cercanas (en una vecindad especificada por un radio)
6. Finalmente, formar los grupos asignando cada entrada a su neurona más cercana.

2. SOM: Estructura y funcionamiento

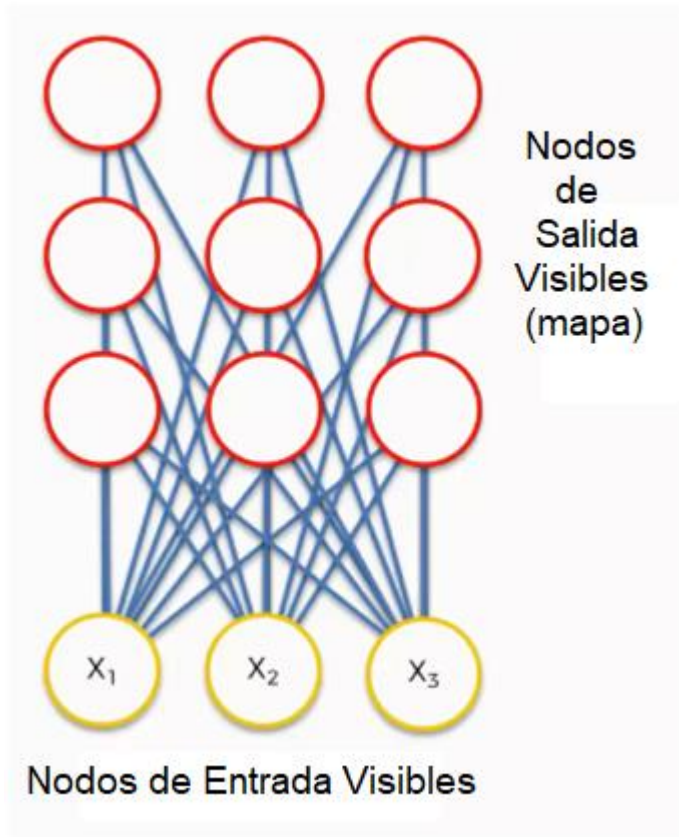
- Los SOM son redes neuronales que emplean métodos de aprendizaje no supervisados, mapeando sus pesos para ajustarse a los datos de entrada dados con el objetivo de representar datos multidimensionales de una forma más fácil y comprensible para el ojo humano (valor pragmático de representar datos complejos).
- Entrenar un SOM no requiere un vector objetivo. Un SOM aprende a clasificar los datos de entrenamiento sin ninguna supervisión externa.
- Compuesto por nodos de entrada y nodos computacionales (de procesamiento o salida), donde cada nodo computacional está conectado a cada nodo de entrada para formar una red.
- No hay interconexiones entre los nodos computacionales.
- El número de nodos de entrada está determinado por las dimensiones del vector de entrada.
- Entendamos mejor su estructura y entrenamiento a través de un ejemplo!

Pero, ¿Cómo se entrena un SOM?



3. ¿Cómo aprenden los SOM?

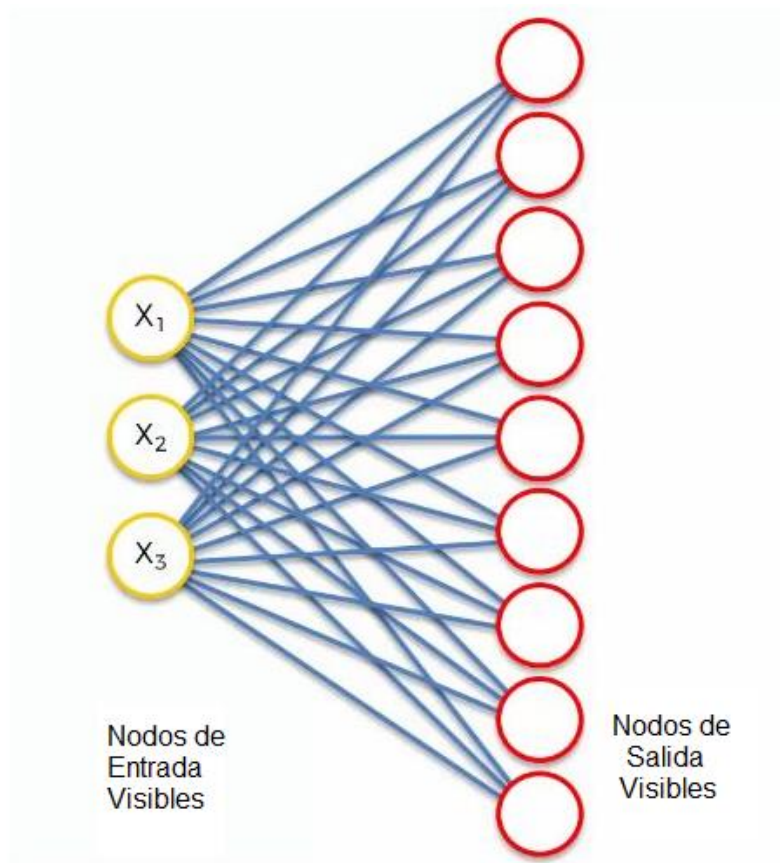
Dada la siguiente topología del SOM:



- Nuestros vectores de entrada tienen tres características x_1 , x_2 , x_3 y tenemos nueve nodos de salida.
- Los tres nodos de entrada representan tres columnas (dimensiones) en el conjunto de datos, pero cada una de estas columnas puede contener miles de filas.
- Los nodos de salida en un SOM son siempre bidimensionales.
- Ahora, lo que haremos será convertir este SOM en un conjunto de entrada que nos resulte más familiar como cuando vimos los métodos de aprendizaje automático supervisados.

3. ¿Cómo aprenden los SOM?

- Consideremos la estructura de autoorganización que tiene **3 nodos de entrada visibles** y **9 salidas** que están conectadas directamente a la entrada como se muestra a continuación:



- Nuestros valores de nodos de entrada son:

$$x_1 = 0.7$$

$$x_2 = 0.6$$

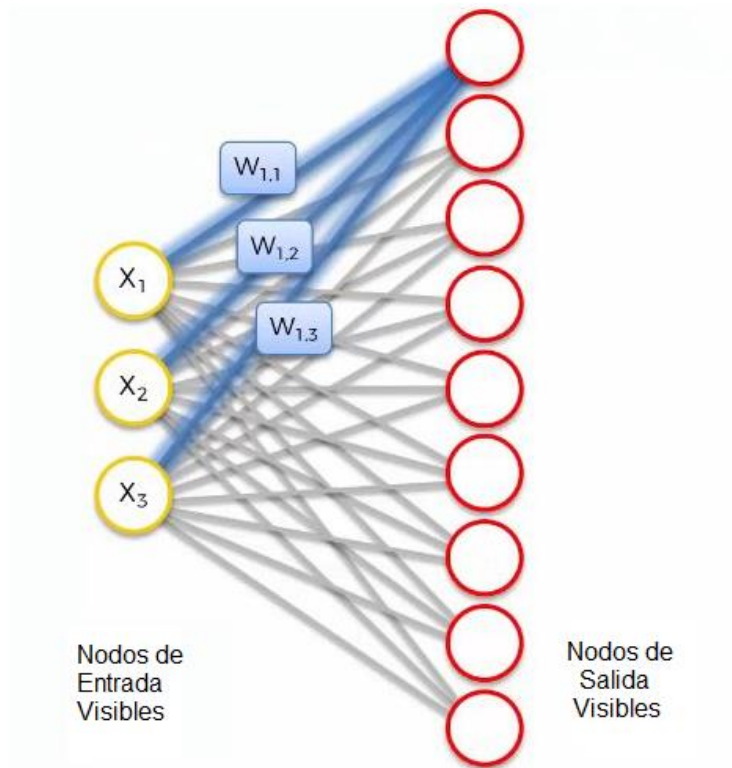
$$x_3 = 0.9$$

- Tasa de aprendizaje = $h_{ck}(t) = \begin{cases} 0.5 & \text{Si la neurona es el BMU.} \\ 0.25 & \text{Si la neurona es el vecino inmediato al BMU.} \\ 0 & \text{En otros casos.} \end{cases}$
- Ahora echemos un vistazo a cada paso en detalle.

3. ¿Cómo aprenden los SOM?

Paso #1: Inicializar los pesos aleatoriamente

- Tomemos el nodo de salida superior y centrémonos en sus conexiones con los nodos de entrada. Como se puede ver, hay un peso asignado a cada una de estas conexiones.



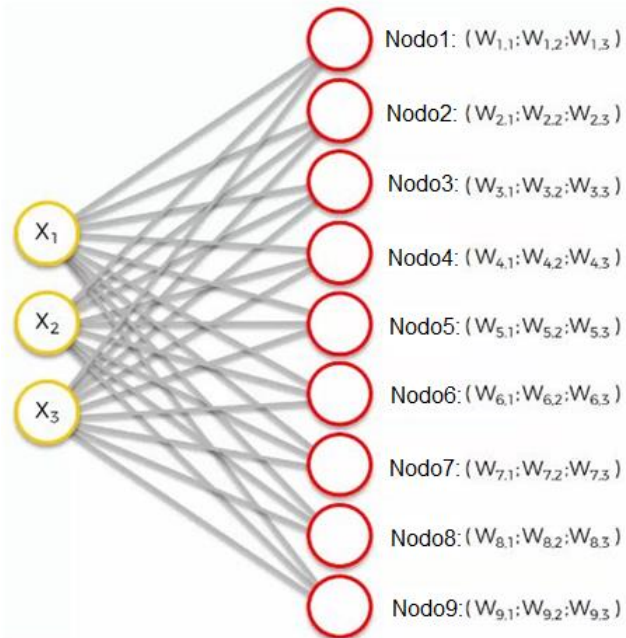
Importante:

- La palabra "**peso**" aquí tiene un significado completamente diferente al que tenía con las redes neuronales artificiales (RNA), donde multiplicábamos el valor del nodo de entrada por el peso y, finalmente, aplicamos una función de activación.
- En los SOM:
 - ✓ NO HAY función de activación.
 - ✓ Los pesos no están separados de los nodos, en un SOM, los pesos pertenecen al propio nodo de salida.
- El nodo de salida contiene los pesos como sus coordenadas.
- Llevando estos pesos, intenta furtivamente encontrar su camino hacia el espacio de entrada.

3. ¿Cómo aprenden los SOM?

Paso #1: Inicializar los pesos aleatoriamente (continuación)

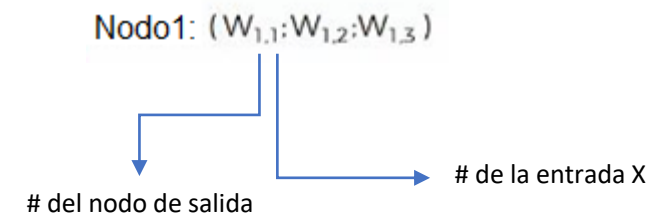
- En el siguiente ejemplo, tenemos un **conjunto de datos 3D** y cada uno de los nodos de entrada representa una coordenada x.
- El SOM los comprimiría en un solo nodo de salida que lleva tres pesos (si tratáramos con un conjunto de datos de 20 dimensiones o características, el nodo de salida, en este caso, llevaría 20 coordenadas de peso).



- Cada uno de estos nodos de salida no se convierte exactamente en parte del espacio de entrada, pero intentan integrarse en él, desarrollando lugares imaginarios para sí mismos.

$W_{1,1} = 0.31$	$W_{1,2} = 0.22$	$W_{1,3} = 0.10$
$W_{2,1} = 0.21$	$W_{2,2} = 0.34$	$W_{2,3} = 0.19$
$W_{3,1} = 0.39$	$W_{3,2} = 0.42$	$W_{3,3} = 0.45$
$W_{4,1} = 0.25$	$W_{4,2} = 0.32$	$W_{4,3} = 0.62$
$W_{5,1} = 0.24$	$W_{5,2} = 0.31$	$W_{5,3} = 0.16$
$W_{6,1} = 0.52$	$W_{6,2} = 0.33$	$W_{6,3} = 0.42$
$W_{7,1} = 0.31$	$W_{7,2} = 0.22$	$W_{7,3} = 0.10$
$W_{8,1} = 0.12$	$W_{8,2} = 0.41$	$W_{8,3} = 0.19$
$W_{9,1} = 0.34$	$W_{9,2} = 0.40$	$W_{9,3} = 0.51$

- Los pesos se inicializan aleatoriamente con valores entre 0 y 1
- Los índices de las variables peso **W** indican:



3. ¿Cómo aprenden los SOM?

Paso #2: Calcular la mejor unidad de coincidencia (Best Matching Unit – BMU ó neurona ganadora)

- El siguiente paso es revisar nuestro conjunto de datos. Para cada una de las filas de nuestro conjunto de datos, intentaremos encontrar el nodo más cercano al vector de entrada.
- Para determinar la mejor unidad de coincidencia, un método es iterar a través de todos los nodos y calcular la distancia euclidiana entre el vector de peso de cada nodo de salida y el vector de entrada actual. El nodo con un vector de peso más cercano al vector de entrada se etiqueta como **BMU (el nodo con la menor distancia hacia el vector de entrada)**.

- La distancia euclidiana se da como:
$$\text{Distancia} = \sqrt{\sum_{i=0}^n (X_i - W_i)^2}$$
Donde X es el vector de entrada actual y W es el vector de peso del nodo.

- A continuación, calcularemos la Unidad de Mejor Coincidencia (BTU) usando la fórmula de Distancia Euclidiana.

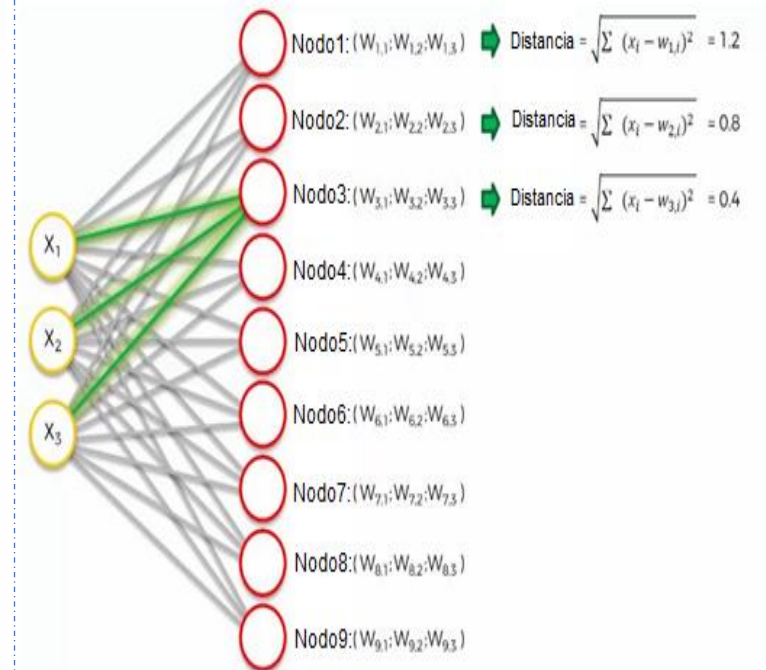
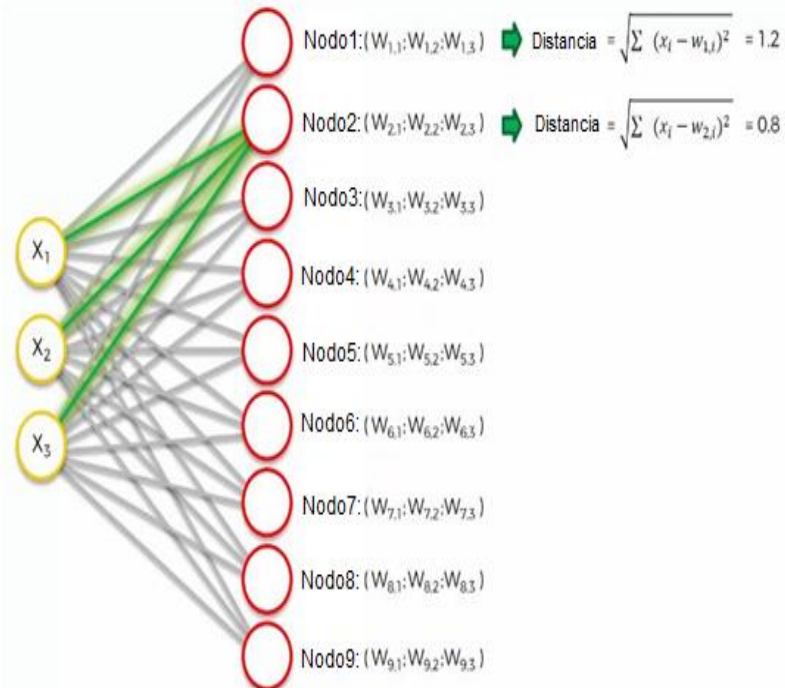
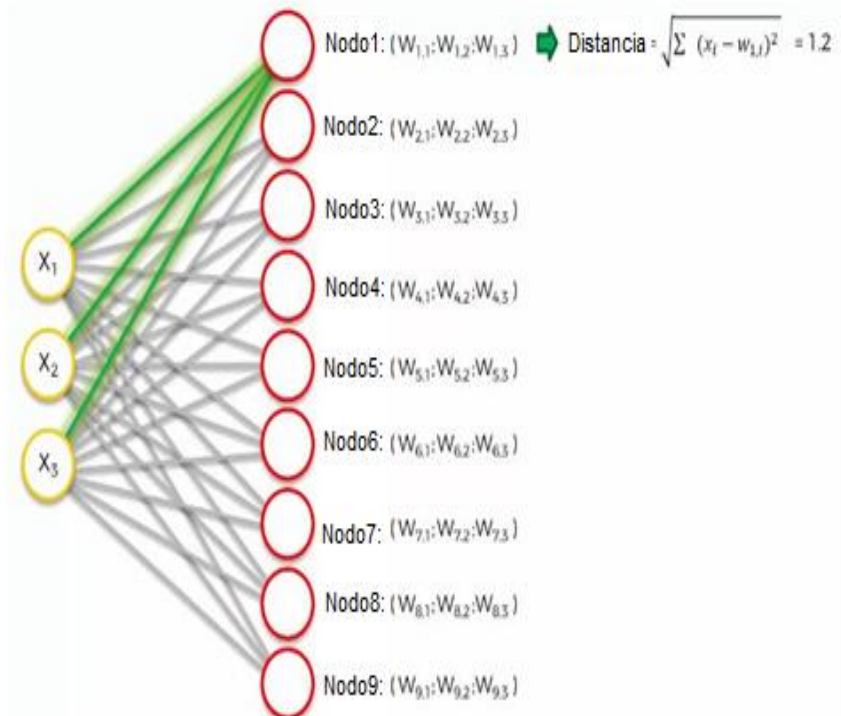
3. ¿Cómo aprenden los SOM?

Paso #2: Calcular la mejor unidad de coincidencia (Best Matching Unit o BMU) - Continuación

- La distancia euclidiana se da como:

$$\text{Distancia} = \sqrt{\sum_{i=0}^n (X_i - W_i)^2}$$

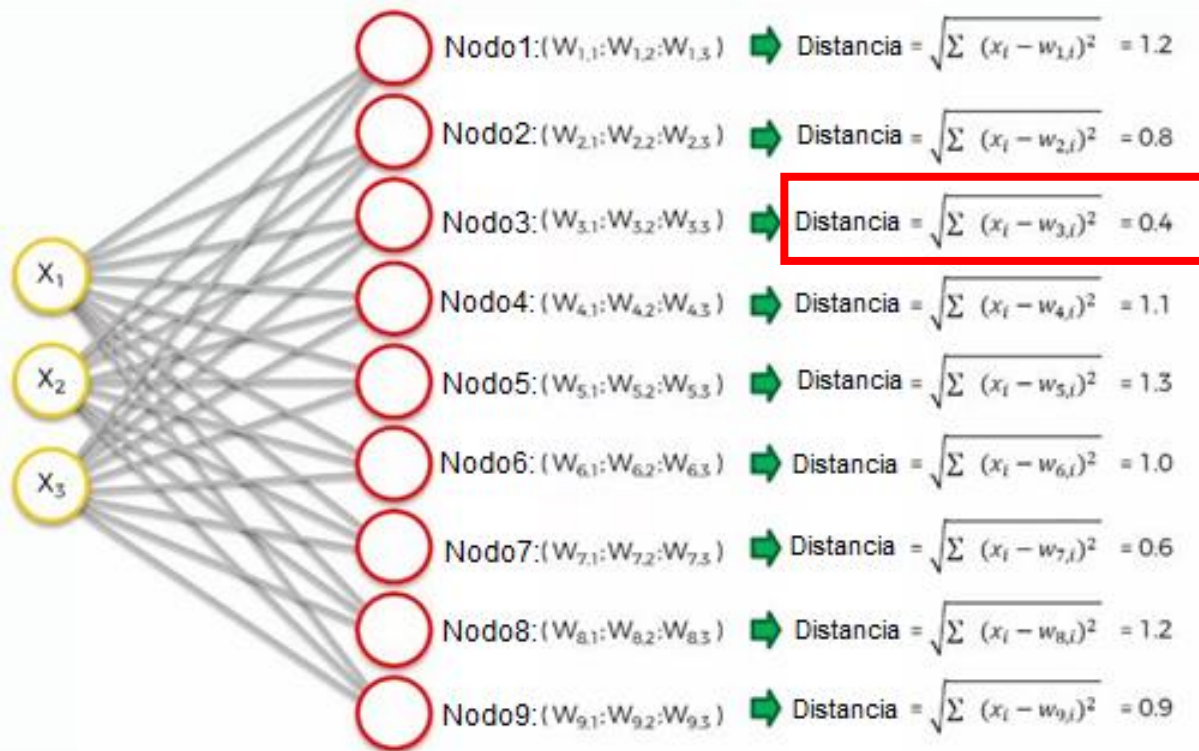
Donde X es el vector de entrada actual y W es el vector de peso del nodo.



3. ¿Cómo aprenden los SOM?

Paso #2: Calcular la mejor unidad de coincidencia (Best Matching Unit o BMU) - Continuación

- Calculamos todos los nodos restantes de la misma manera que puede ver a continuación:



¿Cuál es la mejor unidad de coincidencia?

- Como podemos ver, el **Nodo 3** es el más cercano al vector de la capa de entrada con una distancia de 0.4.
- Llamaremos a este nodo nuestra **BMU** (unidad de mejor coincidencia).

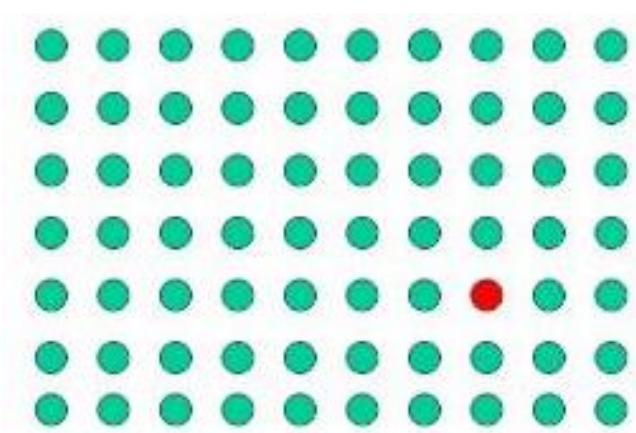
¿Qué pasa después?



3. ¿Cómo aprenden los SOM?

Paso #2: Calcular la mejor unidad de coincidencia (Best Matching Unit o BMU) - Continuación

- El círculo rojo en la figura siguiente representa la mejor unidad de coincidencia (BMU) de este mapa.

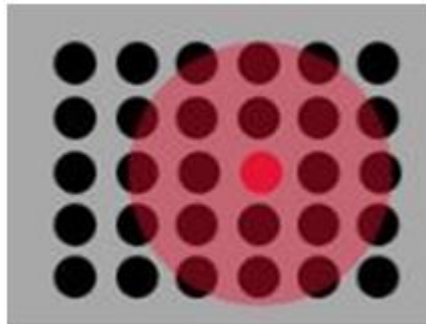


- Ahora, el nuevo SOM tendrá que actualizar sus pesos para que esté aún más cerca de la primera fila de nuestro conjunto de datos.
- La razón por la que necesitamos esto es que nuestros nodos de entrada no se pueden actualizar, mientras que tenemos control sobre nuestros nodos de salida.
- En términos simples, nuestro SOM se está acercando al punto de datos al estirar la BMU hacia él.
- El objetivo final es tener nuestro mapa alineado con el conjunto de datos.
- Vamos al **Paso #3**, donde se calculará el vecindario alrededor del BMU.

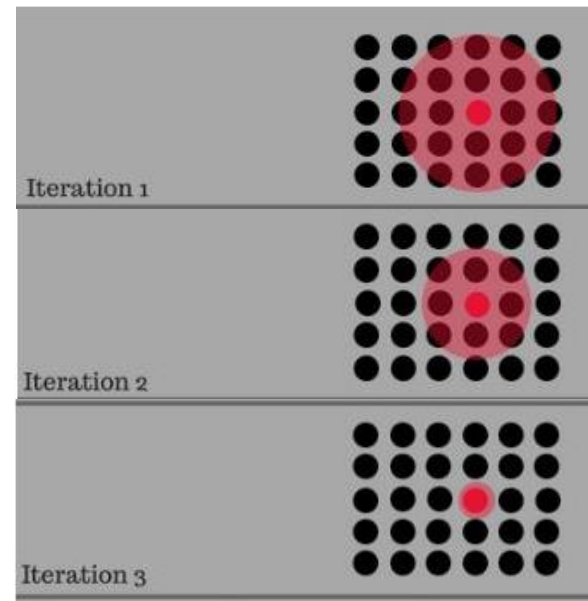
3. ¿Cómo aprenden los SOM?

Paso #3: Calcular el tamaño del vecindario alrededor de la BMU

- En cada iteración, después de que se haya determinado la BMU, el siguiente paso es calcular cuáles de los otros nodos están dentro de la vecindad de la BMU.
- Todos estos nodos tendrán sus vectores de peso alterados en el siguiente paso.
- Primero, calcularemos cuál debería ser el radio de la vecindad y luego se determina si cada nodo está dentro de la distancia radial o no.



- Esta figura muestra un ejemplo del tamaño de un vecindario típico cerca del comienzo del entrenamiento.
- La vecindad está centrada alrededor de la BMU (punto rojo) y abarca la mayoría de los otros nodos y el radio del círculo.



- El tamaño de la vecindad alrededor de la BMU disminuye con una función de disminución exponencial.

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\lambda}\right)$$

σ_0 = El ancho del enrejado en el momento cero

t = Actual iteración

λ = Tiempo constante

Donde $t = 0, 1, 2, 3, \dots$

- Con cada iteración, el vecindario se reducirá al tamaño de un solo nodo ... la BMU.

3. ¿Cómo aprenden los SOM?

Paso #3: Calcular el tamaño del vecindario alrededor de la BMU - Continuación

¿Cómo establecer el valor del radio en el mapa autoorganizado?

El valor del radio en el SOM, dependerá del rango y la escala de los datos de entrada:

- Si se tiene una media cero, estandarizando los valores de sus características, intentaremos un radio $\sigma = 4$.
- Si se está normalizando los valores de las características a un rango de $[0, 1]$, aún se puede intentar $\sigma = 4$, pero un valor de $\sigma = 1$ podría ser mejor.
- Recordemos, que debemos disminuir la **tasa de aprendizaje α** y el tamaño de la **función de vecindad** con iteraciones crecientes, ya que ninguna de las métricas permanece constante a lo largo de las iteraciones en SOM.

El valor del radio también depende del tamaño del SOM:

- Si es un mapa de 10 por 10, entonces usar, por ejemplo, $\sigma = 5$.
- De lo contrario, si es un mapa de 100 por 100, usar $\sigma = 50$.

Vamos al **Paso #4**, donde ajustaremos los pesos del BMU y sus vecinos.

3. ¿Cómo aprenden los SOM?

Paso #4: Ajuste de los pesos de la BMU y su vecindario

Cada nodo dentro de la vecindad de la BMU (incluida la BMU) tiene su vector de peso ajustado de acuerdo con la siguiente ecuación:

Nuevos pesos = pesos antiguos + tasa de aprendizaje (vector de entrada – pesos antiguos)

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

Donde:

t representa el paso de tiempo (iteración)

L es una pequeña variable llamada **tasa de aprendizaje**, que disminuye con el tiempo.

¿Qué nos dice esta ecuación?

- Que el peso recién ajustado para el nodo es igual al peso anterior (W), más una fracción de la tasa de aprendizaje (L) de la diferencia entre el peso anterior (W) y el vector de entrada (V).
- Entonces, de acuerdo con nuestro ejemplo, el **Nodo 3 es la mejor unidad de coincidencia** (como se pudo ver en el paso# 2) Le corresponde ajustar sus pesos:



$W_{3,1} = 0.39$	$W_{3,2} = 0.42$	$W_{3,3} = 0.45$
Input Vector: $X_1 = 0.7$	$X_2 = 0.6$	$X_3 = 0.9$
Tasa de aprendizaje = 0,5		

Ajustemos los pesos del Nodo 3 de acuerdo con la ecuación anterior

3. ¿Cómo aprenden los SOM?

Paso #4: Ajuste de los pesos del Nodo 3 o BMU - Continuación

Nuevos pesos = pesos antiguos + tasa de aprendizaje (vector de entrada – pesos antiguos)

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

Donde:

t representa el paso de tiempo

L es una pequeña variable llamada tasa de aprendizaje, que disminuye con el tiempo.

DATOS para el Nodo 3 (BMU)

$$W_{3,1} = 0.39 \quad W_{3,2} = 0.42 \quad W_{3,3} = 0.45$$

$$\text{Input Vector: } X_1 = 0.7 \quad X_2 = 0.6 \quad X_3 = 0.9$$

$$\text{Tasa de aprendizaje} = L(t) = 0.5$$

Ajustemos los pesos del Nodo 3 de acuerdo con la ecuación anterior

Para: $W_{3,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,1} = 0.39 + 0.5(0.7 - 0.39)$$

$$W_{3,1} = 0.545$$

Para: $W_{3,2}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,2} = 0.42 + 0.5(0.6 - 0.42)$$

$$W_{3,2} = 0.51$$

Para: $W_{3,3}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,3} = 0.45 + 0.5(0.9 - 0.45)$$

$$W_{3,3} = 0.675$$

3. ¿Cómo aprenden los SOM?

Paso #4: Ajuste de los pesos de los nodos del vecindario Nodo 2 y Nodo 4 - Continuación

Nuevos pesos = pesos antiguos + tasa de aprendizaje (vector de entrada – pesos antiguos)

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

Donde:

t representa el paso de tiempo

L es una pequeña variable llamada tasa de aprendizaje, que disminuye con el tiempo.

DATOS para el Nodo 2

$$W_{2,1} = 0.21 \quad W_{2,2} = 0.34 \quad W_{2,3} = 0.19$$

$$x_1 = 0.7 \quad x_2 = 0.6 \quad x_3 = 0.9$$

$$\text{Tasa de aprendizaje} = L(t) = 0.25$$

Ajustemos los pesos del Nodo 2 de acuerdo con la ecuación anterior

Para: $W_{2,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{2,1} = 0.21 + 0.25(0.7 - 0.21)$$

$$W_{2,1} = 0.332$$

Para: $W_{2,2}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{2,2} = 0.34 + 0.25(0.6 - 0.34)$$

$$W_{2,2} = 0.405$$

Para: $W_{2,3}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{2,3} = 0.19 + 0.25(0.9 - 0.19)$$

$$W_{2,3} = 0.367$$

3. ¿Cómo aprenden los SOM?

Paso #4: Ajuste de los pesos de los nodos del vecindario Nodo 2 y Nodo 4 - Continuación

Nuevos pesos = pesos antiguos + tasa de aprendizaje (vector de entrada – pesos antiguos)

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

Donde:

t representa el paso de tiempo

L es una pequeña variable llamada tasa de aprendizaje, que disminuye con el tiempo.

DATOS para el Nodo 4

$$W_{4,1} = 0.25 \quad W_{4,2} = 0.32 \quad W_{4,3} = 0.62$$

$$x_1 = 0.7 \quad x_2 = 0.6 \quad x_3 = 0.9$$

$$\text{Tasa de aprendizaje} = L(t) = 0.25$$

Ajustemos los pesos del Nodo 4 de acuerdo con la ecuación anterior

Para: $W_{4,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{4,1} = 0.25 + 0.25(0.7 - 0.25)$$

$$W_{4,1} = 0.362$$

Para: $W_{4,2}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{4,2} = 0.32 + 0.25(0.6 - 0.32)$$

$$W_{4,2} = 0.39$$

Para: $W_{4,3}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

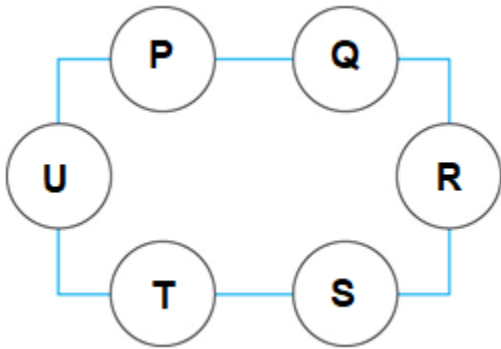
$$W_{4,3} = 0.62 + 0.25(0.9 - 0.62)$$

$$W_{4,3} = 0.69$$

Fin de la primera iteración. Repetir mientras las neuronas no cambien mucho o se exceda un umbral.

4. Ejemplo: Topología de Red SOM

Dada la siguiente topología del SOM:



Entradas $x = \{x_1 = 2, x_2 = -4\}$

Y los pesos para cada neurona:

	P	Q	R	S	T	U
w_1	-1	0	3	-2	3	4
w_2	2	4	-2	-3	2	-1

Tasa de Aprendizaje:

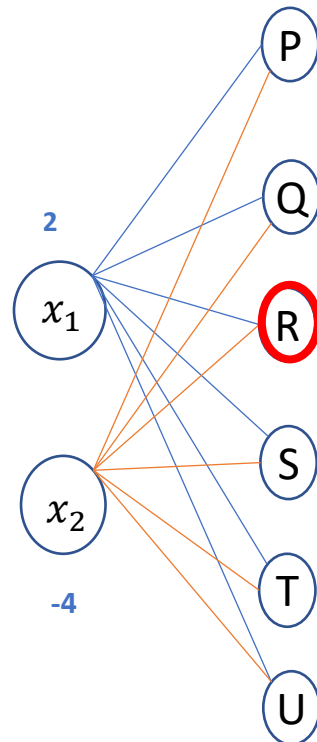
$$h_{ck}(t) = \begin{cases} 0.5 & \text{Si la neurona es el BMU.} \\ 0.25 & \text{Si la neurona es el vecino inmediato al BMU.} \\ 0 & \text{En otros casos.} \end{cases}$$

Solución:

Paso #1: Como tenemos los pesos iniciales ya no los calculamos aleatoriamente.

Paso #2: Usando la distancia Euclidiana, encontramos la neurona ganadora o mejor coincidencia (BMU, la que tiene la distancia mas corta hacia x).

Diagrama de la red



Distancia (hacia la entrada)

Nodo1: ($w_{1,1} = -1, w_{1,2} = 2$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{1,i})^2} = \sqrt{(2 - (-1))^2 + (-4 - 2)^2} = \sqrt{25}$

Nodo2: ($w_{2,1} = 0, w_{2,2} = 4$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{2,i})^2} = \sqrt{(2 - 0)^2 + (-4 - 4)^2} = \sqrt{68}$

Nodo3: ($w_{3,1} = 3, w_{3,2} = -2$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{3,i})^2} = \sqrt{(2 - 3)^2 + (-4 - (-2))^2} = \sqrt{5}$ **BMU**

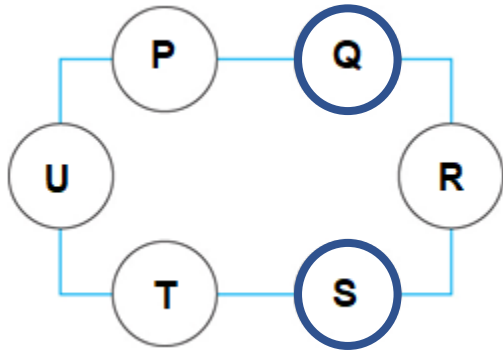
Nodo4: ($w_{4,1} = -2, w_{4,2} = -3$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{4,i})^2} = \sqrt{(2 - (-2))^2 + (-4 - (-3))^2} = \sqrt{17}$

Nodo5: ($w_{5,1} = 3, w_{5,2} = 2$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{5,i})^2} = \sqrt{(2 - 3)^2 + (-4 - 2)^2} = \sqrt{37}$

Nodo6: ($w_{6,1} = 4, w_{6,2} = -1$) \Rightarrow Distancia = $\sqrt{\sum (x_i - w_{6,i})^2} = \sqrt{(2 - 4)^2 + (-4 - (-1))^2} = \sqrt{13}$

4. Ejemplo: Topología de Red SOM

Dada la siguiente topología del SOM:



Entradas $x = \{x_1 = 2, x_2 = -4\}$

Y los pesos para cada neurona:

	P	Q	R	S	T	U
w_1	-1	0	3	-2	3	4
w_2	2	4	-2	-3	2	-1

Tasa de Aprendizaje:

$$h_{ck}(t) = \begin{cases} 0.5 & \text{Si la neurona es el BMU.} \\ 0.25 & \text{Si la neurona es el vecino inmediato al BMU.} \\ 0 & \text{En otros casos.} \end{cases}$$

Solución:

Paso #3: Calcular la vecindad al nodo R (BMU).

Los vecinos inmediatos a la neurona R son Q y S).

Paso #4: Ajustar pesos del **nodo R (BMU)** y de su **vecindad Q y S** (aplicamos la tasa de aprendizaje adecuada, según el caso). Aplicamos la siguiente ecuación:

Nuevos pesos = pesos antiguos + tasa de aprendizaje (vector de entrada – pesos antiguos)

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

Ajustemos los pesos del Nodo 3 o R (BMU) de acuerdo con la ecuación anterior.

Para: $W_{3,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,1} = 3 + 0.5(2 - 3)$$

$$W_{3,1} = 2.5$$

Para: $W_{3,2}$

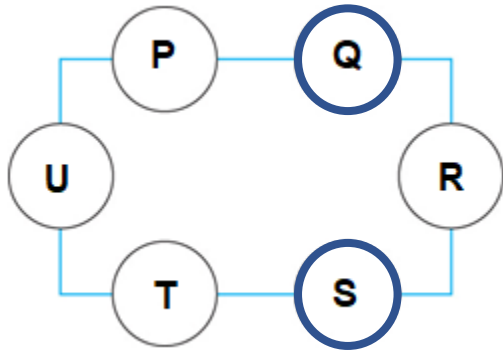
$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,2} = -2 + 0.5(-4 - (-2))$$

$$W_{3,2} = -3$$

4. Ejemplo: Topología de Red SOM

Dada la siguiente topología del SOM:



Entradas $x = \{x_1 = 2, x_2 = -4\}$

Y los pesos para cada neurona:

	P	Q	R	S	T	U
w_1	-1	0	3	-2	3	4
w_2	2	4	-2	-3	2	-1

Tasa de Aprendizaje:

$$h_{ck}(t) = \begin{cases} 0.5 & \text{Si la neurona es el BMU.} \\ 0.25 & \text{Si la neurona es el vecino inmediato al BMU.} \\ 0 & \text{En otros casos.} \end{cases}$$

Solución:

Paso #4: Ajustar pesos del **nodo R (BMU)** y de su **vecindad Q y S** - Continuación

Ajustemos los pesos del Nodo Q ó 2

Para: $W_{2,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,1} = 0 + 0.25(2 - 0)$$

$$W_{3,1} = 0.5$$

$$W_{3,1} = 0.5$$

Ajustemos los pesos del Nodo S ó 4

Para: $W_{4,1}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,1} = -2 + 0.25(2 - (-2))$$

$$W_{3,1} = -2$$

$$W_{3,1} = -2$$

Para: $W_{2,2}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,2} = 4 + 0.25(-4 - 4)$$

$$W_{3,2} = 4 + 0.25(-4 - 4)$$

$$W_{3,2} = 2$$

Para: $W_{4,2}$

$$W(t+1) = W(t) + L(t)(V(t) - W(t))$$

$$W_{3,2} = -3 + 0.25(-4 - (-3))$$

$$W_{3,2} = -3 + 0.25(-4 - (-3))$$

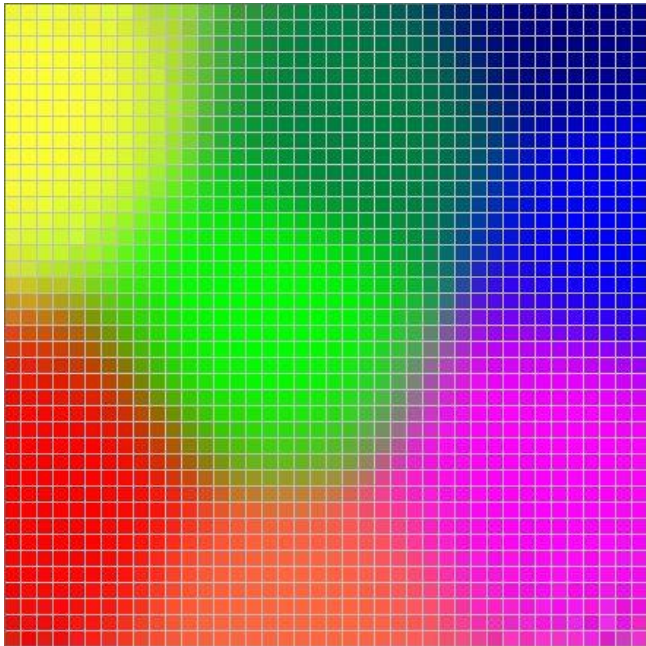
$$W_{3,2} = -3.25$$

Fin de la primera iteración

Si la entrada x tuviera mas filas, a partir de la 2da. Iteración reemplazamos los nuevos pesos ajustados y evaluamos la siguiente entrada de x .

5. Aplicaciones de los SOM – Redes de Kohonen

Matriz de neuronas: para cada neurona visualizamos su vector de pesos W_i (datos en 3D, colores, espacios continuos)



- Entonces, para un mapa de color con 3 entradas, si los pesos de la neurona son (0.7, 0.2, 0.3), mostraríamos un color rojizo con 0.7 de rojo, 0.2 de verdes y 0.3 de azul.

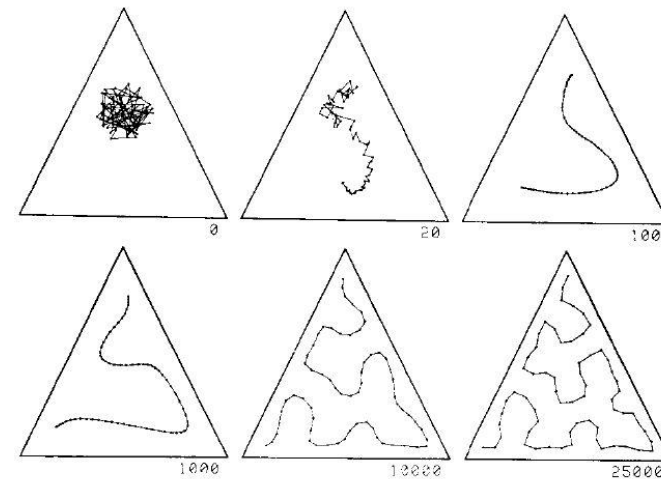


Fig. 4. Weight vectors during the ordering process, one-dimensional array.

- Para un mapa de puntos en el plano con dos entradas, dibujaríamos un punto para cada neurona en posición (W_x , W_y).

5. Aplicaciones de los SOM – Redes de Kohonen

Análisis de datos y monitorización de la crisis

- Estudio sobre la crisis bancaria española de 1977 - 1985
- Haciendo uso de 9 ratios pertenecientes a 66 entidades financieras de la época.

R1	Activo circulante/Activo total
R2	(Activo circulante-Caja)/Activo total
R3	Activo circulante/Deudas
R4	Reservas/Deudas
R5	Beneficio neto/Activo total
R6	Beneficio neto/Fondos propios
R7	Beneficio neto/Deudas
R8	Coste de ventas/Ventas
R9	Cash Flow/Deudas

1 B Union	15 B Valladolid	29 B Garriga N.	43 B Guipuzc.	57 Sind. Banq.
2 B Mas Sardá	16 B Cred. Com.	30 B de Progreso	44 B de Galicia	58 B de Europa
3 B Levante	17 B Prést. y Ahor.	31 B Ind. Bilbao	45 B Hisp. Ind.	59 B de Vasconia
4 B Catalana	18 B Descuento	32 B Int. Español	46 B March	60 B Pop. Español
5 B Ind. Catal.	19 B Com. Occid.	33 B Com. Tran.	47 B Depósitos	61 B Hisp. Amer.
6 B Barcelona	20 B Occidental	34 B Comercio	48 B Herrero	62 B Españ. Créd.
7 B Gerona	21 B Ind. Medit.	35 B Jover	49 B Sabadell	63 B Santander
8 B Alicante	22 B Catal. Desarr.	36 B de Vitoria	50 Bankpyme	64 B Central
9 B Créd. e Inv.	23 B Prom. Neg.	37 B Pueyo	51 B Int. de Com.	65 B Bilbao
10 B Pirineos	24 B López Ques.	38 B Créd Balear	52 B Zaragozano	66 B Vizcaya
11 B Madrid	25 B Asturias	39 B Huesca	53 B Com. Espa.	
12 B de Navarra	26 B Granada	40 B Fomento	54 B Merc. Tarr.	
13 B Cantábrico	27 B Simeón	41 B Pastor	55 B Abel Matutes	
14 B Meridional	28 B. Exp. Indust.	42 B de Castilla	56 B Fin. Indust.	

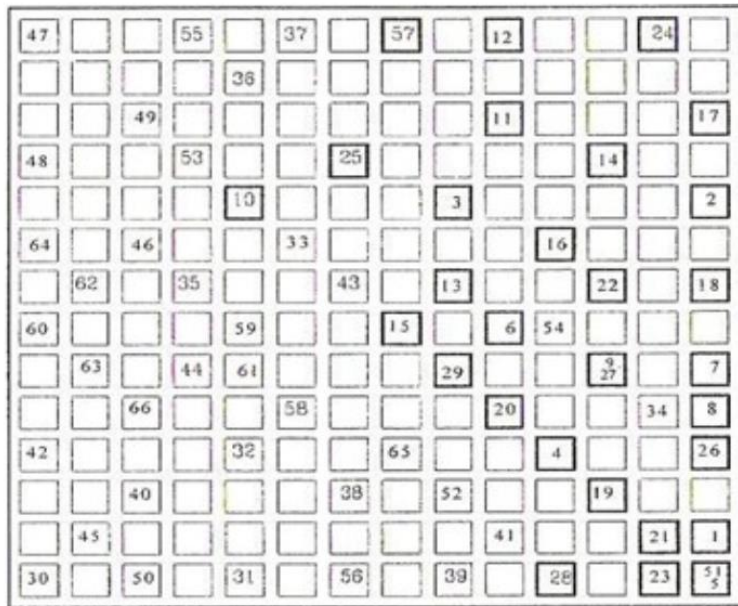


- 37 entidades financieras eran solventes 29 quebradas
- Se entrenó un mapa autoorganizado de 14 * 14 neuronas.
- El resultado del aprendizaje debe ser la especialización de cada neurona hacia algún tipo de banco.

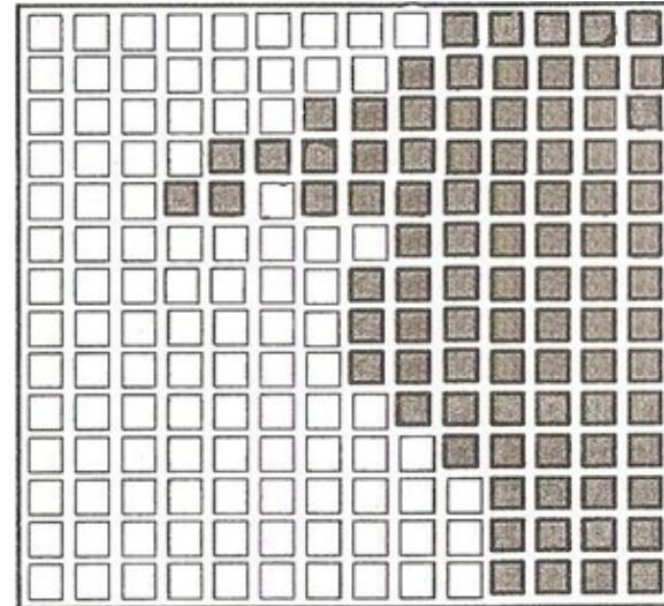
5. Aplicaciones de los SOM – Redes de Kohonen

Análisis de datos y monitorización de la crisis

- Estudio sobre la crisis bancaria española de 1977 - 1985
- Haciendo uso de 9 ratios pertenecientes a 66 entidades financieras de la época.



- El resultado final muestra que neurona responde con mas intensidad ante cada patrón.
- Con trazo ancho los bancos quebrados.



- Suele resultar muy útil delimitar regiones. Por ejemplo, se han etiquetado como neurona de quiebra (oscura) o de solvencia (clara)

PREGUNTAS

Dudas y opiniones