




# Mapas Auto-Organizados

## Redes Neuronales, DC-FCEyN-UBA

Rosana Matuk

Primer Cuatrimestre 2017

- Tongue
- Lips
- Squint
- Fingers
- Wrist (A)
- Wrist (B)
- Forearm
- Elbow
- Foot
- Saccade

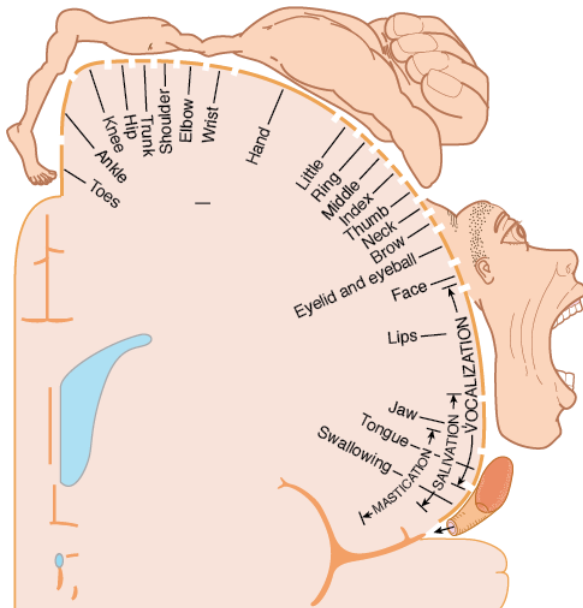
- 
- ☐ *Feature Mapping*
  - ☐ *Self Organizing Maps (SOM)*

## Feature mapping

Si  $x_1$  y  $x_2$  son dos patrones de entrada, y  $r_1$  y  $r_2$  son las localizaciones espaciales de las correspondientes neuronas ganadoras en aprendizaje competitivo, entonces  $r_1$  y  $r_2$  estarán cada vez más cerca, hasta eventualmente coincidir, a medida que  $x_1$  y  $x_2$  se hagan más similares. Adicionalmente, no debería suceder que  $r_1 = r_2$  salvo que  $x_1$  y  $x_2$  sean similares. Una red que hace este mapping es denominada **feature map**.

Técnicamente, lo que pedimos es un mapa que preserve la topología del espacio de los posibles inputs, al espacio de las unidades de salida. Un mapa que preserva la topología, o **mapa topográfico**, es esencialmente un mapeo que preserva relaciones de vecindad.

# Ejemplo: Mapa topográfico del cuerpo en el cerebro



## Ejemplo: Mapa topográfico del cuerpo en el cerebro

Partes cercanas del cuerpo, se mapean en neuronas cercanas en la corteza cerebral.

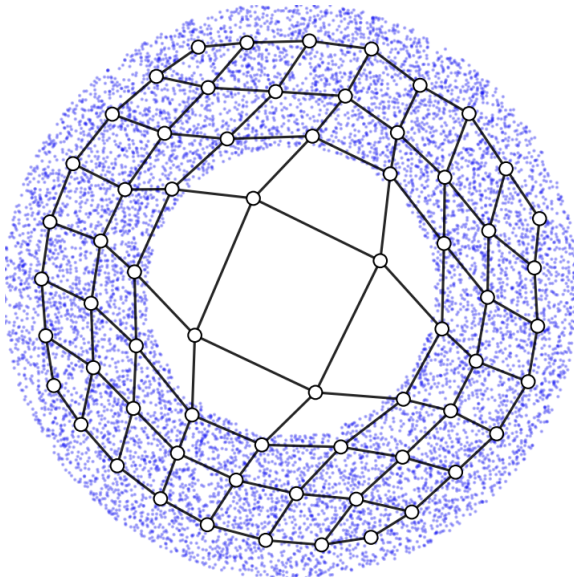
Por otro lado, se observa también un fenómeno de cuantización vectorial, ya que a mayor cantidad de sensores del cuerpo (por ejemplo, las palmas de las manos tienen mayor densidad de elementos sensoriales que otras partes del cuerpo), hay una mayor cantidad de neuronas en la corteza cerebral.

- 
- ☒ *Feature Mapping*
  - ☐ *Self Organizing Maps (SOM)*

# ¿Qué buscamos?

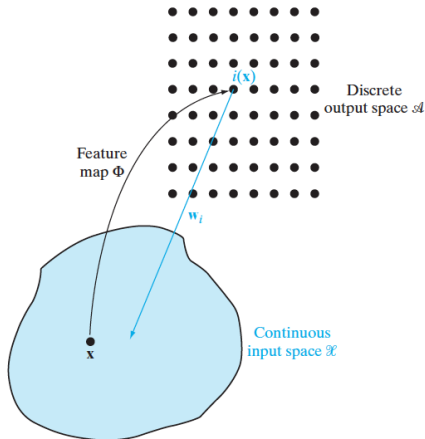
Nuestro objetivo en SOM es encontrar una arquitectura de neuronas en 1 o 2 dimensiones, que se **auto-organicen**, para **mapear el espacio de los inputs a un espacio de menor dimensión, preservando las relaciones topológicas** del espacio de las entradas. Una propiedad fundamental de esta red, es que proveerá una **cuantización vectorial** del espacio de los inputs.

## Gráficamente, buscamos una red así:





# El SOM aproxima el espacio de los inputs, preservando la topología y la densidad de los inputs



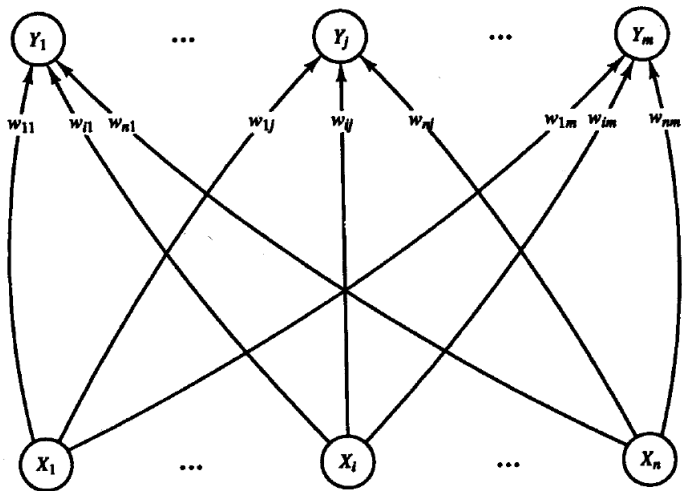
El feature map  $\Phi : \mathcal{H} \rightarrow \mathcal{A}$ , representado por el conjunto de vectores de pesos  $w_j$  en el espacio de los outputs  $\mathcal{A}$ , provee una buena aproximación del espacio de los inputs  $\mathcal{H}$ .

# ¿Cómo logramos nuestro objetivo?

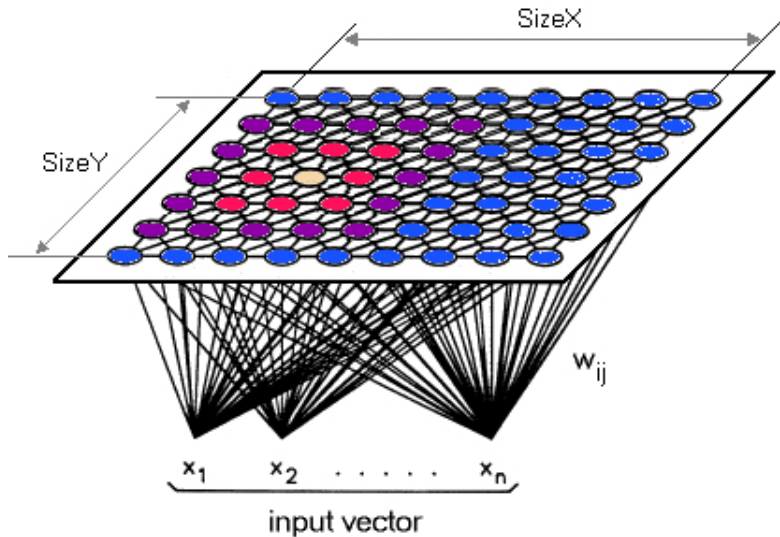
Vamos a usar **aprendizaje competitivo** con los siguientes agregados:

1. La **arquitectura** de la red neuronal es generalmente en 1 o 2 dimensiones.
2. Definimos una **función de vecindad**, para que no sólo se actualice la neurona ganadora, sino también las neuronas en su campo de vecindad.
3. Decrementamos en el tiempo las áreas de vecindad y el factor de aprendizaje (**enfriamiento**).

# Arquitectura SOM lineal



# Arquitectura SOM planar

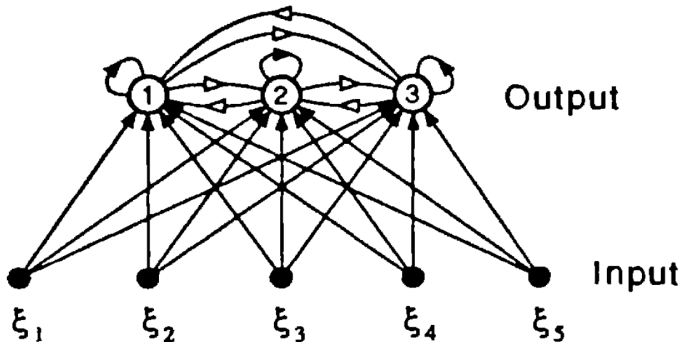


## Arquitectura SOM: recomendaciones de diseño

En general, hacer arquitecturas basadas en arreglos regulares. Algunas veces se usan arreglos rectangulares de los nodos por simplicidad. Sin embargo, los arreglos **hexagonales** son visualmente mucho más ilustrativos y exactos, y son recomendados. Cualquiera sea la arquitectura regular usada, es recomendable **seleccionar las longitudes de las dimensiones horizontales y verticales del arreglo para que se correspondan con las longitudes de las dos primeras componentes principales** (i.e., aquellas con los más altos autovalores de la matriz de correlación de los inputs). Los arreglos regulares elongados tienen la ventaja de garantizar una convergencia más rápida y segura.

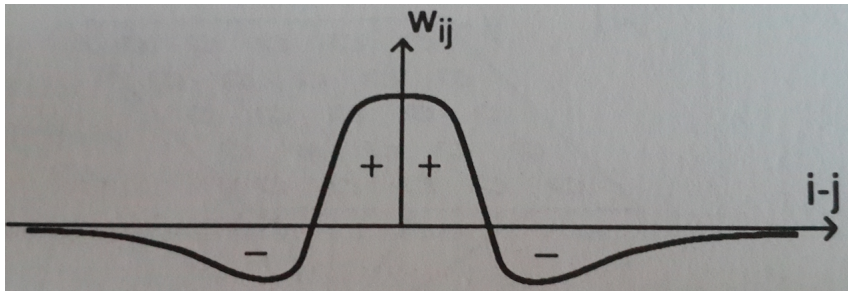
Otras consideraciones: arreglos cíclicos para evitar el efecto borde, arreglos que crezcan con la densidad de los inputs, etc.

## Aprendizaje Competitivo *winner-takes-all*



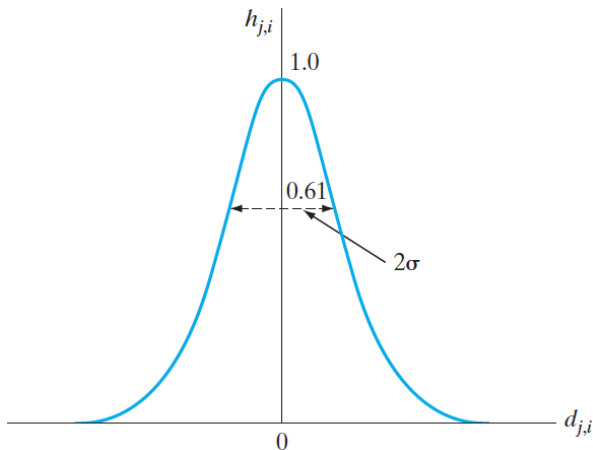
La neurona más similar al patrón de entrada será la neurona ganadora. Las flechas blancas son inhibitorias y las flechas negras con excitatorias. La neurona ganadora inhibe a todas las otras. La neurona ganadora actualiza sus pesos para hacerse más similar al patrón de entrada.

## Vecindad en competitivas: Sombrero mexicano



Hay evidencia neurobiológica de que hay interacción lateral en las neuronas del cerebro humano. La fuerza de las conexiones excitatorias (+) e inhibitorias (-) de las neuronas en las redes competitivas tienen forma de "sombrero mexicano". **La fuerza de la conexión excitatoria decrece con la distancia a la ganadora.**

# Función de vecindad en SOM



El *sombrero mexicano* en su parte excitatoria, se asemeja a una función gaussiana. La función de vecindad debe ser 1 para la neurona ganadora, y decrecer en proporción a la distancia a la neurona ganadora, en la grilla de neuronas.



# Función de vecindad en SOM

El PDF de la función gaussiana es

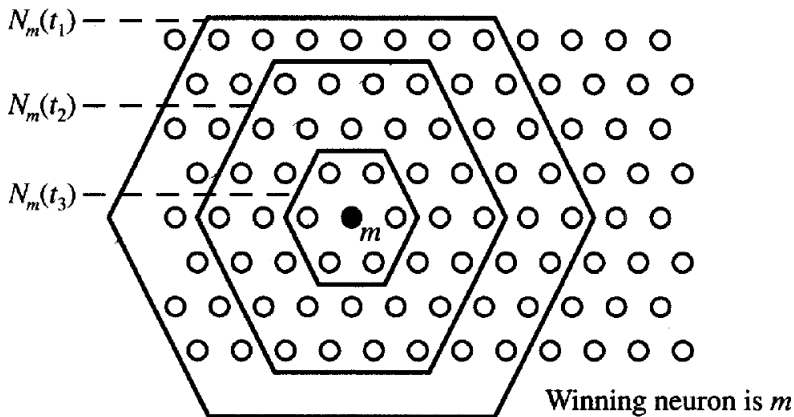
$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

Si  $i^*$  es la neurona ganadora, una elección típica de la función de vecindad en SOM es:

$$h(j, i^*) = \exp\left(-\frac{d_{j,i^*}^2}{2\sigma^2}\right)$$

donde  $d_{j,i^*}$  es la distancia geométrica entre los nodos  $j$  e  $i^*$  en la grilla. Típicamente, será  $d_{j,i^*}^2 = \|r_j - r_{i^*}\|^2$ , donde  $r_i$  es la posición de la neurona  $i$  en el espacio de la grilla de la red neuronal.

## Enfriamiento



Al inicio del entrenamiento, el área de vecindad y el factor de aprendizaje son grandes, para que la red neuronal se **ordene**. El área de vecindad y el factor de aprendizaje van decreciendo con el tiempo, a medida que avanza el entrenamiento, para que el sistema se **estabilice**.

## Enfriamiento

Para decrementar el área de la vecindad en el tiempo, se decrementa el parámetro  $\sigma$  de la función de vecindad  $h$  (ya que  $\sigma$  determina el ancho de la gaussiana), habitualmente según:

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\tau_1}\right)$$

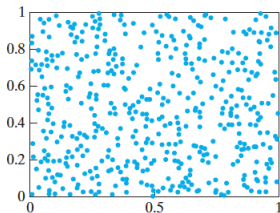
dónde  $t = 0, 1, 2, \dots$  es el número de iteración,  $\sigma_0$  es el valor de  $\sigma$  al inicio del entrenamiento, y  $\tau_1$  es una constante de tiempo elegida por el diseñador.

Para decrementar el factor de aprendizaje  $\eta$  con el tiempo, se suele usar la siguiente fórmula:

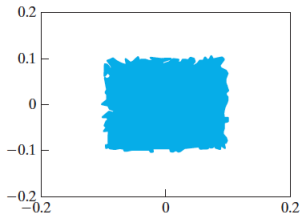
$$\eta(t) = \eta_0 \exp\left(-\frac{t}{\tau_2}\right)$$

dónde  $\eta_0$  es el valor de  $\eta$  al inicio del entrenamiento, y  $\tau_2$  es una constante de tiempo elegida por el diseñador.

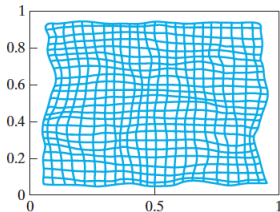
# Fases en el entrenamiento SOM



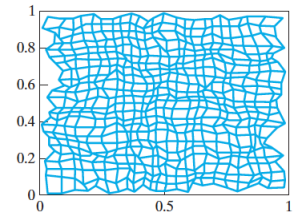
(a) Input distribution



Time = 0  
(b) Initial weights



Time = 160 K  
(c) Ordering phase



Time = 800 K  
(d) Convergence phase

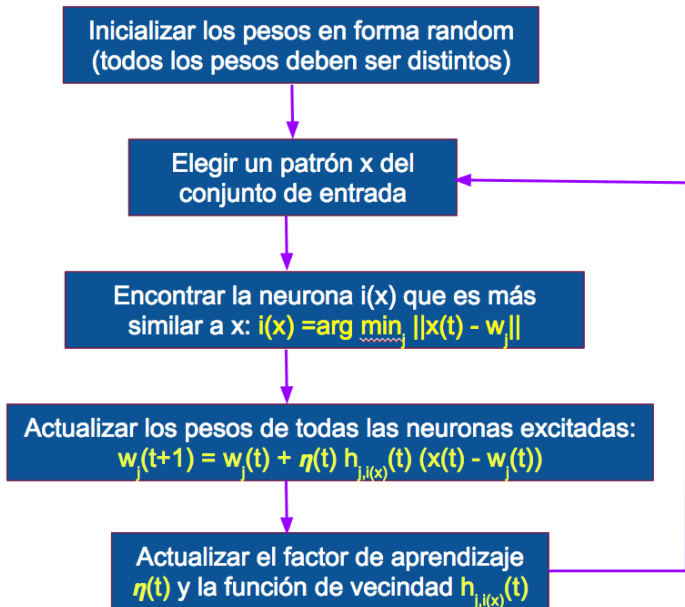
## Fases en el entrenamiento SOM

1. **Fase de Ordenamiento:** En esta fase se produce el ordenamiento topológico de los pesos. Esta fase puede tomar 1000 iteraciones o más. Algunos valores típicos de los parámetros para 1000 iteraciones son  $\eta_0 = 0.1$  y  $\tau_2 = 1000$ . El  $\eta(t)$  debe decrecer, pero siempre debe mantenerse mayor a 0.01. La función de vecindad podría incluir inicialmente todas las neuronas de la red, y debe reducirse al final del ordenamiento para abarcar sólo a las neuronas más vecinas, o sólo a la neurona ganadora. Un valor típico para  $\tau_1 = \frac{1000}{\log \sigma_0}$ .
2. **Fase de Convergencia:** En esta fase se hace el ajuste fino de los pesos. Se recomienda que el número de iteraciones sea al menos 500 veces el número de neuronas en la red. El factor de aprendizaje debe mantenerse mayor a 0, típicamente se usa 0.01. La función de vecindad debe incluir sólo a las neuronas más cercanas, o eventualmente ser sólo la neurona ganadora.

# Inicialización en SOM

La red de pesos se inicializa en valores random de magnitud pequeña. Otra forma de inicialización, es elegir en forma random algunos patrones del conjunto de entrada, e inicializar los pesos con ellos.

# Diagrama del Algoritmo SOM (Kohonen)



## Referencias

Haykin, *Neural Networks and Learning Machines*, 2009.

Hertz, *Introduction to the theory of neural computation*, 1991.

Kohonen, *Essentials of the self-organizing map*, Neural Networks 37, pp 52–65, 2013.

Zurada, *Introduction to artificial neural systems*, 1992.