

# Mapeo autoorganizado

El algoritmo de Kohonen

Verónica E. Arriola-Rios

Facultad de Ciencias, UNAM

21 de noviembre de 2020





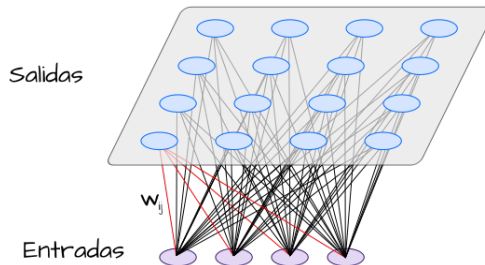
# Temas

- 1 Antecedentes
  - Mapa autoorganizado
  - Aprendizaje competitivo

# Mapa autoorganizado

## Definición

Un *mapa autoorganizado* se caracteriza por la formación de un **mapa topográfico** de los patrones de entrada, en el que la localización espacial (*coordenadas*) de las neuronas en la retícula indican las características estadísticas intrínsecas (o dominio particular) contenidas en los patrones de entrada.



# Mapa de características

## Definición (Mapa de características)

Un *mapa de características* mapea el espacio de entrada  $\mathcal{X}$  hacia un espacio de salida (retícula)  $\mathcal{A}$ .

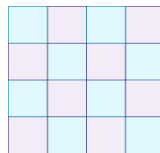
$$\Phi : \mathcal{X} \rightarrow \mathcal{A} \quad (1)$$

- En un contexto neurobiológico el espacio de entradas  $\mathcal{X}$  puede representar el conjunto de coordenadas de los receptores somatosensoriales distribuidos densamente sobre la superficie del cuerpo. El espacio de salidas  $\mathcal{A}$  correspondería al conjunto de neuronas ubicadas en la capa de la corteza cerebral sobre la cual están proyectando los receptores somatosensoriales.

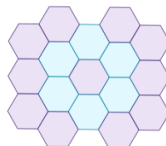
# Retículas

- Para realizar el ordenamiento topológico, la capa de salida de la red suele ser una retícula 2D (también hay versiones 3D) en forma de retícula.
- La retícula puede ser:

## 1 Cuadrada



## 2 Hexagonal



# Temas

- 1 Antecedentes
  - Mapa autoorganizado
  - Aprendizaje competitivo

# Aprendizaje competitivo

- Los mapas autoorganizados utilizan aprendizaje competitivo.
- Las neuronas de salida compiten entre sí para ser activadas (o disparadas), de modo que **sólo una** neurona de salida, o una neurona por grupo, se enciende a la vez.
- Se denomina **la ganadora toma todo**, o simplemente **neurona ganadora**, a la neurona que gana la competencia.
- Una forma de introducir la competencia entre neuronas de salida es utilizar conexiones laterales inhibitorias entre ellas. Esta idea fue propuesta por Rosembatt en 1958.

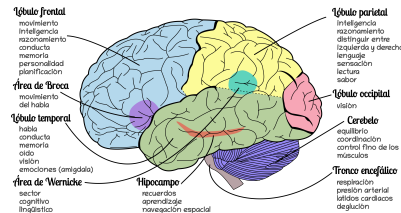


# Ordenamiento topológico

- Los mapeos autoorganizados se inspiran en la siguiente característica del cerebro humano:

*El cerebro se organiza, en varios lugares, de tal modo que distintas entradas sensoriales son representadas por mapas computacionales ordenados topológicamente.*

- Las neuronas transforman la señales de entrada en un *distribución de probabilidad codificada por la posición.*
- Esta distribución representa los valores computados de los parámetros mediante lugares de máxima actividad en el mapa.



Adaptado de:

[https://commons.wikimedia.org/wiki/File:Brain\\_diagram\\_pl.svg](https://commons.wikimedia.org/wiki/File:Brain_diagram_pl.svg)

# Mapeo de Kohonen

- 1 Antecedentes
- 2 Mapeo de Kohonen
- 3 Algoritmo
- 4 Propiedades
- 5 Evaluación del mapeo

# Temas

## 2 Mapeo de Kohonen

- Antecedentes

# Antecedentes

- El modelo de Kohonen fue propuesto en 1982.
- Se desvía ligeramente del modelo biológico con la finalidad de ser computacionalmente tratable, al tiempo que captura las características esenciales de los mapas computacionales en el cerebro.
- Puede realizar compresión de datos, es decir, permite reducir la dimensionalidad de los datos de entrada (número de componentes).
- Pertenece a la familia de algoritmos de *codificación vectorial*.

# Datos de entrenamiento

- Para que el algoritmo de aprendizaje funcione es esencial que haya redundancia en los datos de entrada, de modo que provea conocimiento sobre la estructura subyacente a los patrones de activación de entrada.

# Algoritmo

- 1 Antecedentes
- 2 Mapeo de Kohonen
- 3 Algoritmo**
- 4 Propiedades
- 5 Evaluación del mapeo

# Temas

## 3 Algoritmo

- Datos y parámetros
- Inicialización
- Proceso competitivo
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Datos y parámetros

- Sea  $n$  la dimensión del espacio de los datos de entrada.
- Un vector de entrada se ve entonces como:

$$\vec{x} = [x_1, x_2, \dots, x_n]^T \quad (2)$$

- Los vectores de entrada  $\vec{x}$  se seleccionan al azar de un conjunto de entrenamiento o se generan a partir de la distribución de probabilidad de los datos que se desea codificar.
- Sea  $j$  el índice de una neurona en la capa de salida (retícula). Entonces el vector de pesos sinápticos que conectan a las entradas con la neurona  $j$  es:

$$\vec{w}_j = [w_{1j}, w_{2j}, \dots, w_{nj}]^T, \quad j = 1, 2, \dots, l \quad (3)$$

donde  $l$  es el número total de neuronas en la retícula  $\mathcal{A}$ .



# Temas

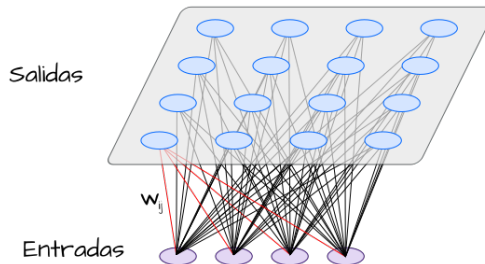
## 3 Algoritmo

- Datos y parámetros
- **Inicialización**
- Proceso competitivo
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Inicialización

Se proponen dos métodos para elegir los valores iniciales:

- Elegir valores aleatorios para los valores iniciales de los vectores de pesos. Los valores para cada peso deben ser diferentes y sus magnitudes, pequeñas.
- Seleccionar aleatoriamente vectores de pesos  $\{\vec{w}_j(0)\}_{j=1}^l$  de entre los posibles vectores de entrada  $\{\vec{x}_i\}_{i=1}^m$



# Temas

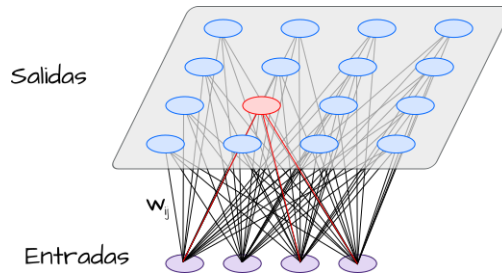
## 3 Algoritmo

- Datos y parámetros
- Inicialización
- **Proceso competitivo**
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Proceso competitivo

## Proceso competitivo

Un espacio continuo de patrones de activación de entradas se mapea a un espacio de neuronas de salida discreto mediante un proceso de competencia entre las neuronas de la red.



# Selección de la neurona ganadora

- Dada una entrada  $\vec{x}$  encontrar:
  - 1 La neurona para la cual el producto vectorial  $\vec{w}_j^T \vec{x}$  es máximo, o
  - 2 Sean los pesos dados por **vectores unitarios**  $\vec{w}_j$ , encontrar la neurona ganadora  $i$  para la cual se minimiza la distancia Euclidiana entre el vector de entrada y el peso:

$$i(\vec{x}) = \underset{j}{\operatorname{argmin}} \|\vec{x} - \vec{w}_j\|, \quad j \in \mathcal{A} \quad (4)$$

- Dependiendo de la aplicación, la respuesta de la red podría ser el índice de la neurona ganadora  $i(\vec{x})$  o el vector de pesos sinápticos más cercano al vector de entrada en el sentido Euclidiano  $\vec{w}_i$ .

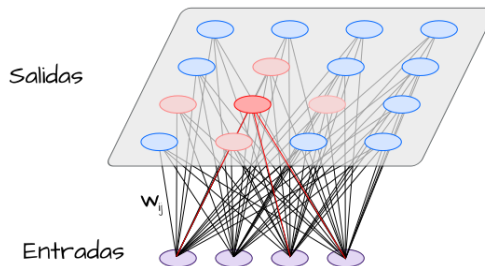
# Temas

## 3 Algoritmo

- Datos y parámetros
- Inicialización
- Proceso competitivo
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Proceso colaborativo

- Existe evidencia neurobiológica de interacción lateral entre un conjunto de neuronas excitadas en el cerebro humano.



# Vecindad

- Sea  $h_{j,i}$  la *vecindad topológica* centrada en la neurona ganadora  $i(\vec{x})$  con vecinos  $j$ .

$$h_{j,i(\vec{x})}(t) = e^{\left(-\frac{d_{j,i}^2}{2\sigma(t)^2}\right)}, \quad j \in \mathcal{A}, t = 0, 1, 2, \dots, \quad (5)$$

$$d_{j,i}^2 = \|\vec{r}_j - \vec{r}_i\|^2 \quad (6)$$

donde  $\vec{r}$  está dado por las coordenadas de las neuronas involucradas pero en el grid.

- El tamaño de vecindad se puede ir reduciendo con el tiempo  $t$  de entrenamiento:

$$\sigma(t) = \sigma_0 e^{\left(-\frac{t}{\tau_1}\right)} \quad t = 0, 1, 2, \dots, \quad (7)$$

donde  $\sigma_0$  es el valor inicial de  $\sigma$  y  $\tau_1$  es una constante temporal a elegir.



# Temas

## 3 Algoritmo

- Datos y parámetros
- Inicialización
- Proceso competitivo
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Proceso de adaptación

- El proceso de aprendizaje se basa en una versión modificada del *aprendizaje Hebbiano*.
- El postulado de Hebb dice que el peso de las sinapsis se incrementa con la ocurrencia simultánea de actividades presinápticas y postsinápticas.
- Para evitar que los pesos se saturen se agrega un *término de olvido*  $g(y_i)\vec{w}_j$  con  $y_i$  la respuesta de la neurona,  $g(y_j) = 0$  para  $y_j = 0$ . Por ejemplo:

$$\begin{aligned}
 (y_j = h_{j,i}(\vec{x})) \\
 \downarrow \\
 g(y_i) = \eta y_j = \eta h_{j,i}(\vec{x})
 \end{aligned}
 \tag{8}$$

pues la activación depende de la cercanía a la neurona ganadora.

- Como heurística, se hace decaer la tasa de aprendizaje, con el tiempo, desde un valor inicial  $\eta_0$ .

$$\eta(t) = \eta_0 e^{\left(-\frac{t}{\tau_2}\right)}, \quad t = 0, 1, 2 \quad (9)$$

- Se actualiza a los vecinos de la neurona ganadora de acuerdo con:

$$\vec{w}_j(t+1) = \vec{w}_j(t) + \eta(t) h_{j,i}(t) (\vec{x}(t) - \vec{w}_j(t)) \quad (10)$$

donde  $\eta(t)$  es la *taza de aprendizaje*.

# Resumen

---

## Algoritmo 1 Kohonen

---

- 1:  $\text{Dim}(\mathcal{A}) \leftarrow l$
  - 2:  $\text{Forma}(\mathcal{A}) \leftarrow \{\text{cuadrada}|\text{hexagonal}\}$
  - 3:  $\vec{w}_j \leftarrow$  muestreo aleatorio
  - 4: **function** KOHONEN( $\mathcal{X}, \eta_0, \tau_2, \tau_1$ )
  - 5:     **loop**
  - 6:         Selecciona  $\vec{x}$  al azar de  $\mathcal{X}$
  - 7:         Encontrar neurona ganadora  $i(\vec{x})$  usando la distancia Euclidiana.
  - 8:         Determinar a los vecinos  $h_{j,i}(t)$
  - 9:         Ajustar los pesos de los nodos  $\{\vec{w}_j\}$
-

# Temas

## 3 Algoritmo

- Datos y parámetros
- Inicialización
- Proceso competitivo
- Proceso colaborativo
- Proceso de adaptación
- Fases del proceso de aprendizaje

# Fases del proceso de aprendizaje

Mientras se realiza el entrenamiento de la red, la adaptación pasa por dos fases:

① *Fase de ordenamiento* o *autoorganización*.

- Se pasa de un estado caótico a un ordenamiento topológico de los vectores de pesos.
- Toma  $\sim 1000$  iteraciones o más.
- Se sugiere que la tasa de aprendizaje  $\eta(t)$  inicie con valores cerca de 0.1 y no descienda a menos de 0.01. Para ello se puede usar en (9):

$$\eta_0 = 0.1$$

$$\tau_2 = 1000$$

- La función de vecindad debe comenzar cubriendo prácticamente todas las neuronas en la retícula y reducirse hasta dos vecinos o sólo la neurona ganadora. Para ello usar en (5):

$$\tau_1 = \frac{1000}{\log \sigma_0}$$

## ② Fase de convergencia

- Durante esta etapa se realizan los ajustes finos al mapa de características.
- El número de iteraciones depende fuertemente de la dimensionalidad del espacio de entrada.

iteraciones  $\sim 500l$

- La tasa de aprendizaje debe ser pequeña pero no cero.

$\eta(t) \sim 0.01$

- Evitar *estados metaestables*, es decir, configuraciones del mapa topológico con defectos.
- La vecindad  $h_{j,i}(\vec{x})$  sólo debe contener a los vecinos más cercanos o a ninguno, actualizándose ya sólo los pesos de la neurona ganadora.

# Propiedades

- 1 Antecedentes
- 2 Mapeo de Kohonen
- 3 Algoritmo
- 4 Propiedades**
- 5 Evaluación del mapeo



# Temas

## 4 Propiedades

- Aproximación del espacio de entradas
- Ordenamiento topológico
- Correspondencia en las densidades
- Selección de características

# Aproximación del espacio de entradas

*El mapa de características  $\Phi$ , representado por el conjunto de vectores de pesos sinápticos  $\{\vec{w}_j\}$  en el espacio de salidas  $\mathcal{A}$ , provee una buena aproximación al espacio de entrada  $\mathcal{X}$ .*

# Temas

## 4 Propiedades

- Aproximación del espacio de entradas
- Ordenamiento topológico
- Correspondencia en las densidades
- Selección de características

# Ordenamiento topológico

*El mapa de características  $\Phi$  se encuentra topológicamente ordenado en el sentido de que la ubicación espacial de una neurona en la retícula corresponde a un dominio particular o característica de los patrones de entrada.*

- La ecuación (10) para actualizar los pesos fuerza al vector de pesos sinápticos  $\vec{w}_i$  de la neurona ganadora  $i(\vec{x})$  a modificarse en dirección al vector de entrada  $\vec{x}$ .

# Temas

## 4 Propiedades

- Aproximación del espacio de entradas
- Ordenamiento topológico
- Correspondencia en las densidades
- Selección de características

# Correspondencia en las densidades

*El mapa de características  $\Phi$  refleja variaciones en las estadísticas de la distribución de entradas: Regiones del espacio de entradas  $\mathcal{X}$  de la cuales se muestrean los vectores  $\vec{x}$  con probabilidad alta, son mapeados a dominios grandes en el espacio de salida y con mejor resolución que regiones de  $\mathcal{X}$  de donde es poco probable obtener muestras.*

# Temas

## 4 Propiedades

- Aproximación del espacio de entradas
- Ordenamiento topológico
- Correspondencia en las densidades
- Selección de características

# Selección de características

*Dados datos del espacio de entradas, el mapa autoorganizado puede seleccionar un conjunto de las mejores características para aproximar la distribución subyacente.*



# Evaluación del mapeo

- 1 Antecedentes
- 2 Mapeo de Kohonen
- 3 Algoritmo
- 4 Propiedades
- 5 Evaluación del mapeo

# Temas

- 5 Evaluación del mapeo
  - Medidas de la calidad del mapa

# Distorsión

- Se evalúa la precisión de la proyección utilizando la *distorsión* promedio:

$$\varepsilon_t = \frac{1}{N} \sum_{k=1}^N \|\vec{x}_k - \vec{x}'_k(c)\|^2 \quad (11)$$

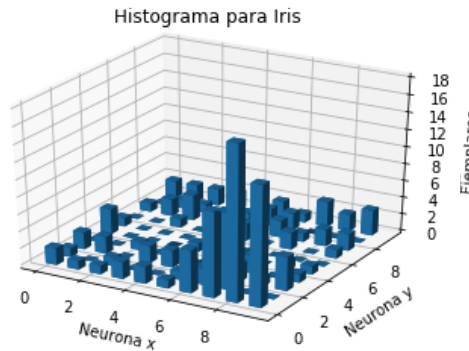
donde  $\vec{x}_k$  es una entrada a la red, el código  $\vec{c}(x)$  es la neurona ganadora  $i(\vec{x})$  y  $\vec{x}'(c)$  es la reconstrucción del vector de entrada, dado por el vector de pesos asociados a la neurona ganadora  $\vec{w}_{i(\vec{x})}$ , es decir:

$$\vec{c}(x) = i(\vec{x}) \quad (12)$$

$$\vec{x}'(c) = \vec{w}_{i(\vec{x})} \quad (13)$$

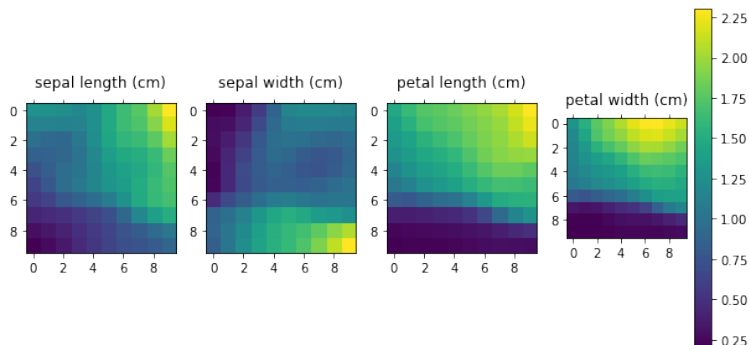
# Histograma de datos

- Se muestra cuántos vectores pertenecen a cada *cúmulo* definido por una neurona dado un conjunto de datos.



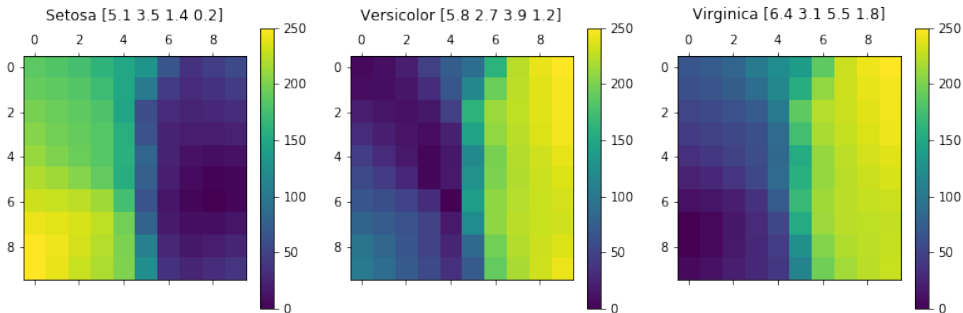
# Mapas de calor

- Se puede visualizar un mapa de calor por cada característica en el vector de entrada, esto mostrará la formación del mapa topológico.



## Mapas de calor de distancias

- Se utilizan mapas de calor para identificar la(s) neurona(s) más activas dado un ejemplar. Lo que se mide es la distancia entre el ejemplar y los vectores  $\{\vec{w}_j\}_{j=1}^l$



**Figura:** Claramente fue posible separar a Setosa de las demás especies, aunque Versicolor y Virginica quedaron en regiones más semejantes.

# Referencias I



Haykin, Simon (2009). *Neural Networks and Learning Machines*. 3rd. Prentice Hall, Pearson.



Self-Organising Maps for Customer Segmentation using R

<https://www.shanelynn.ie/self-organising-maps-for-customer-segmentation-using-r/>

# Licencia

Creative Commons  
Atribución-No Comercial-Compartir Igual

